

**CYCLE INGENIEUR DE L'ECOLE POLYTECHNIQUE D'AGADIR  
FILIERE GENIE INFORMATIQUE**

## **STAGE TECHNICIEN**

# **Exploration des Concepts Fondamentaux de l'Intelligence Artificielle**

**Réalisé par :  
Amine AICHANE**

**Superviseur :  
M. Ismail WITAM  
Tuteur :  
M. Abdelaaziz EL  
AISSAOUI**

## REMERCIEMENTS

Je tiens à exprimer ma plus profonde gratitude à M. Ismail Witam, PDG de Wire IT et mon encadrant de stage, pour m'avoir offert l'opportunité de réaliser ce stage au sein de son entreprise.

Son leadership et sa vision inspirante ont été des sources de motivation constantes pour moi. Grâce à ses conseils avisés et à son accompagnement attentif, j'ai pu acquérir des compétences précieuses et réaliser mes objectifs de manière efficace. Sa disponibilité et sa patience m'ont permis de surmonter les défis rencontrés tout au long de cette expérience, et d'en tirer des enseignements importants pour ma carrière future.

Je souhaite également souligner l'importance des échanges réguliers avec M. Ismail Witam, qui ont été d'une grande richesse. Ses retours constructifs et ses suggestions pertinentes ont constamment orienté mes efforts et m'ont permis d'améliorer la qualité de mon travail. Son approche pédagogique et son attention aux détails ont été déterminants pour le développement de mes compétences techniques et professionnelles.

Je remercie chaleureusement M. Ismail pour son soutien infaillible, sa confiance, et son investissement personnel dans ma formation. Ses conseils judicieux et son expertise ont été déterminants pour le bon déroulement de mon stage et la réalisation de mes missions. Son encouragement à penser de manière critique et à aborder les problèmes sous différents angles a été particulièrement précieux pour ma croissance professionnelle.

Enfin, je souhaite remercier M. Ismail pour avoir créé un environnement de travail positif et collaboratif, où l'apprentissage et le partage de connaissances sont encouragés. Cette expérience enrichissante et formatrice restera un moment clé de mon parcours professionnel, et je suis profondément reconnaissant pour l'opportunité qui m'a été offerte de travailler sous sa direction.

# TABLE DES MATIERES

INTRODUCTION GENERALE.....	4
<b>CHAPITRE 1 : Présentation de l'entreprise .....</b>	<b>7</b>
I- INTRODUCTION.....	8
II- RAISON SOCIALE & STATUT JURIDIQUE .....	8
2-1- Activités.....	8
2-2- Organisation .....	8
2-3- Systèmes de production/service .....	8
III- LES METIERS DU TECHNICIEN AU SEIN DE L'ENTREPRISE D'ACCUEIL .....	8
IV- CONCLUSION .....	8
<b>CHAPITRE 2 : Recherche documentaire .....</b>	<b>9</b>
I- INTRODUCTION.....	10
II- OUTILS ET METHODES UTILISEES .....	12
III- CONCLUSION .....	18
<b>CHAPITRE 3 : Missions et travaux réalisés.....</b>	<b>20</b>
I- INTRODUCTION.....	21
II- LES MISSIONS EFFECTUEES .....	22
2-1- La mission principale : Comprendre et appliquer les notions de bases de l'IA générative	22
2-2- Les missions secondaires .....	31
CONCLUSION GENERALE .....	33
BIBLIOGRAPHIE .....	34
ANNEXES .....	35

## TABLE DES ILLUSTRATIONS

### Liste des figures

Figure1 : Légende de la figure 1.....	page
Figure2 : Légende de la figure 2.....	page

### Liste des tableaux

Tableau1 : Légende du tableau 1.....	page
Tableau2 : Légende du tableau 2.....	page

## INTRODUCTION GENERALE

J'ai effectué mon stage au sein de l'entreprise Wire IT, un cabinet d'ingénierie et de conseil engagé à offrir des services de haut niveau en ingénierie et management pour ses clients. Mon stage a duré deux mois, de juin à août 2024, et s'est déroulé à Marrakech, Maroc. Ce stage m'a permis de découvrir et de me familiariser avec les dynamiques du secteur de l'ingénierie et du conseil, ainsi que de développer des compétences spécifiques en intelligence artificielle (IA) et en développement de solutions basées sur l'IA. J'ai pu observer de près les processus métiers de cette industrie et comprendre les enjeux liés à l'intégration de technologies avancées dans divers projets d'ingénierie et de management.

Wire IT est une entreprise dynamique qui opère dans les domaines de l'ingénierie et du conseil, offrant des services variés allant de la conception technique à la gestion de projets complexes. Durant mon stage, j'ai principalement travaillé sous la supervision de mon maître de stage, M. Ismail, qui est également le PDG de Wire IT. Il m'a guidé tout au long de mon stage, me fournissant des directives claires et des feedbacks constructifs pour me permettre de progresser dans la compréhension et l'application des notions de base de l'IA générative.

L'objectif principal de mon stage était de comprendre et d'appliquer les notions de base de l'IA générative. La mission principale consistait à analyser les besoins de Wire IT en matière d'IA, à collecter et nettoyer les données pertinentes, à entraîner le modèle d'IA avec ces données, et à tester et valider les solutions développées pour s'assurer qu'elles répondaient de manière précise et pertinente aux demandes des utilisateurs. Ce projet m'a permis de mettre en pratique les compétences acquises au cours de ma formation et de contribuer activement à un projet concret au sein de l'entreprise.

Le rapport se divise en trois chapitres principaux. Le Chapitre 1 présente Wire IT, incluant sa raison sociale, son statut juridique, ses activités, son organisation interne, et ses systèmes de production. Il conclut par une description des métiers du technicien au sein de l'entreprise. Le Chapitre 2 est dédié à la recherche documentaire, détaillant la mission principale du stage ainsi que les outils et méthodes utilisés. Enfin, le Chapitre 3 décrit les missions et travaux réalisés durant le stage, en se concentrant sur la compréhension et l'application des notions de base de l'IA générative, ainsi que sur les missions secondaires telles que l'optimisation de la base de données, l'implémentation de la génération augmentée de récupération (RAG), le

développement de l'interface utilisateur, et la gestion de projet. Le rapport se termine par une conclusion générale résumant les réalisations et perspectives.

# CHAPITRE 1 :

## Présentation de l'entreprise

## I- INTRODUCTION

Tout chapitre doit contenir une introduction et conclusion.

Dans l'introduction de ce chapitre, l'étudiant va :

- Argumenter le choix de l'entreprise d'accueil ;
- Introduire les différentes parties de ce chapitre.

**Remarque :** La documentation nécessaire pour élaborer ce premier chapitre, sera obtenue auprès des managers, des ingénieurs ou des techniciens de l'entreprise d'accueil. L'étudiant peut également se baser sur le site internet de l'entreprise.

## II- RAISON SOCIALE & STATUT JURIDIQUE

### 2-1- Activités

Dans cette partie, il faut décrire les activités, services ou produits développés et/ou commercialisés par l'entreprise.

### 2-2- Organisation

Dans cette partie, il faut représenter l'organigramme de l'entreprise et définir brièvement le rôle de chaque fonction y figurant.

### 2-3- Systèmes de production/service

Dans cette partie, il faut décrire le processus de fabrication ou de service (machines, produits, services etc.) du département où le stage a été effectué ;

## III- LES METIERS DU TECHNICIEN AU SEIN DE L'ENTREPRISE D'ACCUEIL

Dans cette dernière partie, il faut décrire les missions et responsabilités des techniciens dans l'entreprise d'accueil.

## IV- CONCLUSION

Dans la conclusion de ce chapitre, l'étudiant doit mettre en évidence les points qui ont marqués l'étudiant au sein de l'entreprise d'accueil.

**Remarque :** Ce chapitre doit contenir 3 à 6 pages.



# CHAPITRE 2 :

## Recherche documentaire

## I- INTRODUCTION

Dans cette introduction, je vais présenter la mission principale de mon stage chez Wire IT, les outils que j'ai utilisés et les méthodes que j'ai employées pour accomplir les diverses tâches. Ce stage, réalisé sous la supervision de M. Ismail WITAM, a été une expérience enrichissante qui m'a permis d'approfondir mes connaissances en intelligence artificielle et en développement de chatbots.

### 1. Mission principale du stage

La mission principale de mon stage chez Wire IT était de développer un chatbot utilisant Mistral AI, en explorant les concepts fondamentaux de l'intelligence artificielle. Cette mission avait pour objectif de créer un outil interactif capable de répondre aux requêtes des utilisateurs de manière efficace et intuitive. L'outil était conçu pour ne répondre qu'en fonction de la base de connaissances que je lui avais fournie, composée notamment de fichiers PDF. En limitant ses réponses aux informations contenues dans ces documents, le chatbot pouvait garantir des réponses fiables et pertinentes, en s'assurant que les utilisateurs recevaient des informations exactes et vérifiables. Les tâches spécifiques incluaient :

- **L'intégration de l'IA** : Intégrer Mistral AI.
- **L'apprentissage des concepts fondamentaux de l'IA** : Comprendre les bases théoriques et pratiques de l'intelligence artificielle.
- **La formation du chatbot** : Entraîner le modèle d'IA pour qu'il puisse comprendre et répondre aux questions des utilisateurs.

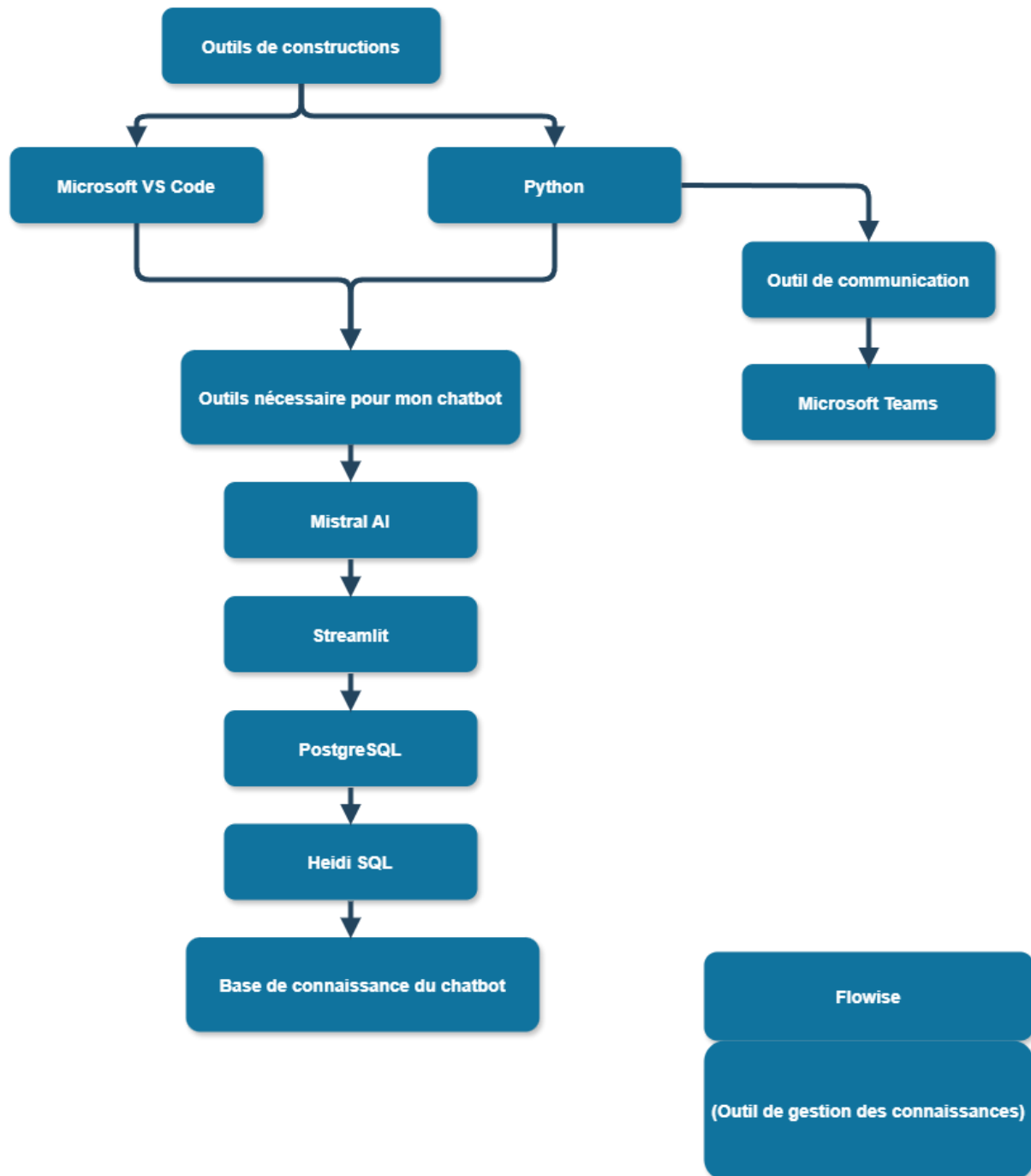
### 2. Outils utilisés

Durant mon stage, j'ai eu l'occasion de travailler avec divers outils qui m'ont aidé à mener à bien les différentes tâches assignées. Voici les principaux outils que j'ai utilisés :

- **Mistral AI**
- **Streamlit**
- **PostgreSQL**
- **Heidi SQL**
- **Flowise**
- **Microsoft Teams**
- **Microsoft VS Code**

- **Python**
- **Base de connaissance du chatbot**

Voici un diagramme illustrant les divers outils et leurs interconnexions :



### 3. Méthodes employées

Pour accomplir les objectifs de mon stage, j'ai utilisé plusieurs méthodes adaptées aux diverses tâches que j'ai effectuées. Voici un aperçu des principales méthodes que j'ai employées :

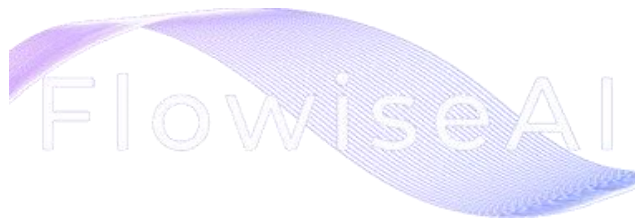
- **Analyse des besoins**
- **Entraînement du modèle**
- **Test et validation**

## II- OUTILS ET METHODES UTILISEES

### 1. Outils utilisés

#### a) Outil de gestion de connaissances

##### i. Flowise



Flowise a été employé pour me familiariser avec les concepts et les outils de l'intelligence artificielle. En réalisant des projets simples avec Flowise, j'ai pu apprendre les principes fondamentaux de l'IA et du traitement du langage naturel, ce qui m'a permis de mieux comprendre et appliquer ces concepts dans le développement du chatbot.

#### b) Outils nécessaires pour mon chatbot

##### i. Mistral AI



Mistral AI a été le cœur technologique du projet de développement du chatbot. Il a permis de créer un système capable de comprendre et de générer du langage naturel de manière sophistiquée. Cette technologie a été essentielle pour entraîner le modèle à répondre aux

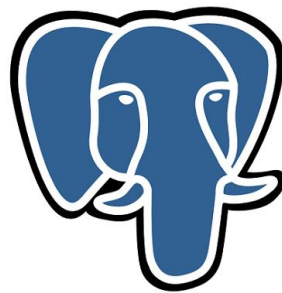
questions des utilisateurs avec précision et pertinence, en utilisant des algorithmes avancés de traitement du langage naturel (NLP).

### ii. Streamlit



Streamlit a été utilisé pour développer l'interface utilisateur du chatbot. Ce framework de création d'applications web interactives a facilité la mise en place d'une interface intuitive et réactive. Grâce à Streamlit, j'ai pu créer des visualisations et des contrôles interactifs qui ont permis aux utilisateurs d'interagir avec le chatbot de manière simple et efficace.

### iii. PostgreSQL



PostgreSQL

PostgreSQL a servi de base de données relationnelle pour stocker les données du chatbot. PostgreSQL a été choisi pour sa robustesse, sa capacité à gérer efficacement les opérations de lecture et d'écriture, ainsi que pour sa conformité aux standards SQL, garantissant ainsi l'intégrité et la sécurité des données stockées. Cette démarche a permis de créer une infrastructure solide et fiable pour soutenir les fonctionnalités avancées de mon chatbot, contribuant ainsi à offrir une expérience utilisateur fluide et intuitive.

#### **iv. Heidi SQL**



Heidi SQL a été utilisé pour la gestion et la visualisation de la base de données PostgreSQL. Cet outil m'a permis de manipuler les données de manière efficace, de visualiser les tables et d'exécuter des requêtes SQL, facilitant ainsi la gestion quotidienne et le suivi des informations stockées.

#### **v. Base de connaissance du chatbot**



La base de connaissances du chatbot, alimentée par des fichiers PDF issus des modules de formation et stockée dans la base de données PostgreSQL, a été conçue pour enrichir ses capacités. Cela permettait au chatbot de rechercher et de répondre à des questions basées sur le contenu de ces documents.

### **c) Outil de communication**

#### **i. Microsoft Teams**



Microsoft Teams a été l'outil principal pour la communication avec l'équipe de Wire IT. Il a permis de coordonner les réunions, d'échanger des informations et de collaborer efficacement à distance, en assurant une communication fluide et continue tout au long du projet.

#### d) Outils de construction

##### i. Microsoft VS Code



Microsoft VS Code a été utilisé comme éditeur de texte pour coder le chatbot. Ses fonctionnalités telles que la coloration syntaxique et son interface intuitive ont simplifié l'écriture et l'édition du code, rendant le processus de développement plus fluide et efficace.

##### ii. Python



Python a été le langage principal utilisé pour le développement du chatbot. Sa flexibilité et ses bibliothèques spécialisées en traitement du langage naturel ont été cruciales pour intégrer Mistral AI et développer les fonctionnalités du chatbot.

## 2. Méthodes utilisées

### a) Analyse des besoins

Pour débiter le projet, j'ai mené une analyse approfondie des besoins spécifiques de Wire IT en matière de chatbot en suivant cette méthodologie :

#### 1. Consultation avec l'encadrant de stage :

- **Objectif** : Comprendre les attentes et les exigences pour le chatbot.
- **Méthode** : Discussions approfondies avec mon encadrant de stage pour recueillir les besoins et les fonctionnalités souhaitées. Cela a permis de définir les scénarios d'utilisation courants et les fonctionnalités essentielles.

#### 2. Analyse des processus existants :

- **Objectif** : Identifier les points de douleur actuels.

- **Méthode** : Étude des flux de travail et analyse des tickets de support pour repérer les questions fréquentes et les problèmes récurrents. Cette analyse a révélé les domaines où un chatbot pourrait améliorer l'efficacité et réduire la charge de travail.

### 3. Définition des fonctionnalités clés :

- **Objectif** : Établir les fonctionnalités essentielles du chatbot.
- **Méthode** : Élaboration d'une matrice des fonctionnalités prioritaires, notamment :
  - Réponses automatisées
  - Base de connaissances intégrée
  - Interface utilisateur intuitive
  - Apprentissage et amélioration continue

### 4. Évaluation des technologies disponibles :

- **Objectif** : Sélectionner les outils et technologies adaptés.
- **Méthode** : Évaluation de différentes plateformes et frameworks de chatbot, et choix de Mistral AI pour sa robustesse et ses fonctionnalités avancées en traitement du langage naturel.

L'analyse des besoins a permis de définir un cadre clair pour le développement d'un chatbot intuitif et efficace, capable de répondre aux requêtes des utilisateurs grâce à une base de connaissances bien structurée et une interface utilisateur optimisée.

### b) Entraînement du modèle

L'objectif de cette étape était de former le chatbot à comprendre et répondre à un large éventail de requêtes de manière précise et efficace.

Pour atteindre cet objectif, j'ai suivi plusieurs étapes essentielles en utilisant Mistral AI :

#### 1. Collecte des données :

- **Objectif** : Rassembler un ensemble de données pertinent pour constituer la base de connaissance du chatbot.
- **Méthode** : J'ai collecté les données à partir des PDF de mes modules académiques. Ces documents contenaient des informations cruciales qui ont été extraites pour alimenter la base de connaissances du chatbot.

#### 2. Nettoyage des données :



- **Objectif** : Préparer les données collectées pour l'entraînement du modèle en éliminant les informations inutiles, les erreurs et les incohérences.
- **Méthode** : Les données ont été nettoyées pour supprimer les doublons, corriger les fautes de frappe, et normaliser les formats. Ce processus était crucial pour s'assurer que les données utilisées pour l'entraînement du modèle étaient de haute qualité et pertinentes.

### 3. Réalisation de sessions d'entraînement intensives :

- J'ai effectué plusieurs sessions d'entraînement intensives pour affiner la capacité du chatbot à traiter divers types de demandes. Chaque session consistait à entraîner le modèle sur le jeu de données nettoyé et à évaluer ses performances en termes de précision et de pertinence des réponses.
- Pendant ces sessions, j'ai utilisé des techniques d'évaluation continue pour mesurer les performances du modèle et ajuster les paramètres en conséquence. Cela impliquait des cycles itératifs de formation et d'évaluation, permettant d'améliorer progressivement la capacité du modèle à comprendre et à répondre aux requêtes des utilisateurs.

Grâce à ces étapes méthodologiques, j'ai pu entraîner le chatbot à utiliser Mistral AI pour répondre de manière efficace à un large éventail de requêtes. Le modèle a été optimisé pour offrir des réponses précises et pertinentes, améliorant ainsi l'expérience utilisateur et l'efficacité des interactions avec le chatbot.

### c) Test et validation

Pour évaluer les performances du chatbot, j'ai réalisé une série de tests rigoureux basés sur des scénarios d'utilisation réels. Ces tests visaient à vérifier la précision et la pertinence des réponses du chatbot, tout en identifiant les points d'amélioration nécessaires avant sa mise en service.

#### 1. Tests rigoureux :

- **Objectif** : Évaluer la capacité du chatbot à répondre correctement et de manière pertinente aux requêtes des utilisateurs.
- **Méthode** : J'ai utilisé des scénarios d'utilisation réels, qui incluaient des questions fréquemment posées et des requêtes complexes. Chaque interaction a été testée pour vérifier si les réponses étaient précises et utiles.

2. **Évaluation de la précision et de la pertinence :**

- **Objectif** : Mesurer l'exactitude et la pertinence des réponses fournies par le chatbot.
- **Méthode** : Chaque réponse du chatbot a été comparée aux réponses attendues pour évaluer sa précision et sa pertinence. Les critères d'évaluation incluaient la justesse des informations, la clarté des réponses, et leur adéquation par rapport à la question posée.

3. **Analyse des résultats :**

- **Objectif** : Identifier les forces et les faiblesses du chatbot.
- **Méthode** : J'ai analysé les résultats des tests pour repérer les erreurs récurrentes et les types de requêtes mal gérées. Cela a permis de déterminer les domaines nécessitant des améliorations.

4. **Identification des points d'amélioration :**

- **Objectif** : Trouver les aspects du chatbot qui peuvent être optimisés pour améliorer ses performances.
- **Méthode** : En se basant sur l'analyse des résultats, j'ai identifié des ajustements spécifiques à apporter pour améliorer la précision et la pertinence des réponses du chatbot.

5. **Ajustement et optimisation :**

- **Objectif** : Apporter des modifications au modèle pour améliorer ses performances.
- **Méthode** : J'ai effectué des ajustements sur le modèle en utilisant les données supplémentaires collectées pendant les tests. Des sessions d'entraînement supplémentaires ont été réalisées pour affiner les réponses du chatbot.

Grâce à ces tests rigoureux et à l'analyse approfondie des résultats, j'ai pu identifier et implémenter les améliorations nécessaires. Le chatbot a été optimisé pour offrir des réponses précises et pertinentes, améliorant ainsi l'expérience utilisateur et l'efficacité des interactions avant sa mise en service.

### III- CONCLUSION

Ce deuxième chapitre a permis de détailler les méthodes et outils utilisés durant le stage chez Wire IT, offrant ainsi une vue d'ensemble sur le développement et l'optimisation du chatbot. L'analyse des besoins a révélé l'importance de comprendre les exigences spécifiques pour

définir les fonctionnalités adaptées. L'entraînement du modèle avec Mistral AI a été un processus clé, nécessitant une collecte rigoureuse des données et des ajustements techniques pour garantir une performance optimale. Les tests et la validation ont assuré que le chatbot répondrait efficacement aux attentes des utilisateurs en identifiant et en corrigeant les éventuelles failles.

Cette étude de la littérature pave le chemin vers le prochain chapitre, où nous approfondirons les concepts fondamentaux de l'intelligence artificielle qui ont soutenu le développement du chatbot. Nous examinerons les principes théoriques et les technologies sous-jacentes, en mettant en lumière leur influence sur la création et l'optimisation des systèmes de traitement du langage naturel. Cette transition nous permettra de mieux saisir les bases théoriques qui ont guidé le projet et de comprendre comment ces concepts se traduisent en pratiques concrètes dans le développement des chatbots.

# CHAPITRE 3 :

## Mission et travaux réalisés

## I- INTRODUCTION

Dans ce chapitre, nous allons détailler les différentes missions que j'ai effectuées au sein de Wire IT pendant mon stage. Ces missions couvrent à la fois des tâches principales et secondaires, mettant en lumière les divers aspects de mon apprentissage et de mes contributions au projet. Voici un aperçu de ces missions :

### 1. Mission principale : Comprendre et appliquer les notions de base de l'IA générative

- **1er cas d'usage : Flowise**
  - Cette section abordera l'utilisation de Flowise pour la création de flux de modèles de langage personnalisés (LLM), en décrivant les avantages et les expériences personnelles tirées de cette plateforme.
- **2ème cas d'usage : Chatbot avec Streamlit et Mistral AI**
  - Nous explorerons le développement d'un chatbot utilisant Streamlit et Mistral AI, détaillant le processus, les défis rencontrés et les solutions mises en œuvre.

### 2. Les missions secondaires

- **Optimisation et maintenance de la base de données**
  - Description des tâches effectuées pour améliorer les performances et la fiabilité de la base de données utilisée dans le projet.
- **Apprentissage et implémentation de la génération augmentée de récupération (RAG)**
  - Explication de l'apprentissage et de l'intégration de la génération augmentée de récupération pour améliorer les capacités de recherche et de récupération d'informations.
- **Développement de l'interface utilisateur**
  - Développement et amélioration de l'interface utilisateur pour rendre l'application plus intuitive et user-friendly.
- **Gestion de Projet**
  - Gestion des différentes phases du projet, coordination avec les membres de l'équipe et suivi de l'avancement des tâches pour assurer la réalisation des objectifs dans les délais impartis.

Cette structure permettra de mettre en évidence les différentes compétences acquises et les contributions apportées tout au long de mon stage. Chacune des sections suivantes fournira une description détaillée des missions réalisées, illustrant ainsi le processus d'apprentissage et les résultats obtenus.

### II- LES MISSIONS EFFECTUEES

#### 2-1- La mission principale : Comprendre et appliquer les notions de bases de l'IA générative

##### a) 1<sup>er</sup> cas d'usage : Flowise

###### – Définition :



En cherchant comment comprendre les notions de base de l'IA générative, j'ai découvert Flowise AI, un outil visuel d'interface utilisateur open-source qui a retenu mon attention. Flowise AI simplifie considérablement le processus de création de flux de modèles de langage personnalisés (LLM) en utilisant LangchainJS. Cet outil, écrit en Node Typescript/Javascript, offre une plateforme intuitive pour les utilisateurs souhaitant créer leurs propres applications LLM.

Flowise AI permet de visualiser et de gérer facilement les flux de travail, rendant le développement de modèles de langage plus accessible même pour ceux qui ont peu d'expérience en programmation. Grâce à ses fonctionnalités avancées, il est possible de voir les applications LLM fonctionner en temps réel, d'intégrer des composants personnalisés et d'explorer divers exemples de chaînes LLM, tels que la récupération de questions-réponses (QnA), la traduction de langues, et les agents conversationnels avec mémoire.

De plus, Flowise AI est disponible gratuitement pour une utilisation commerciale et personnelle, et prend en charge Docker pour un déploiement facile. Pour commencer, il suffit d'installer Flowise AI avec npm, ce qui en fait un outil extrêmement pratique pour quiconque

cherche à se plonger dans l'IA générative et à développer des applications de langage naturel personnalisées.

### – Avantages et bénéfices :

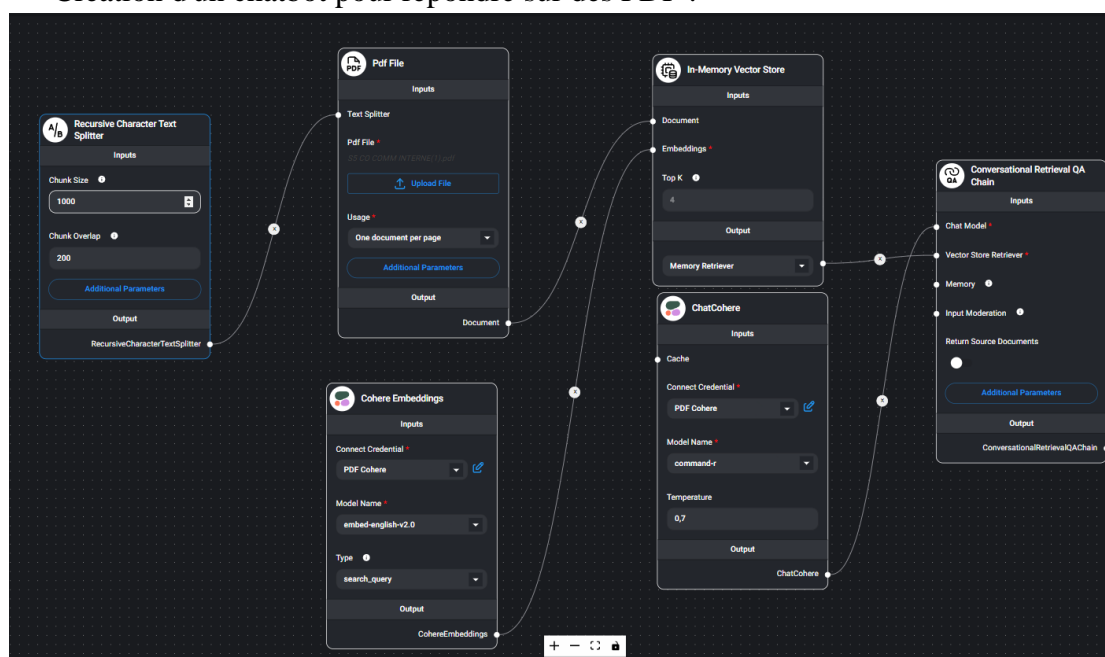
Le développement de tâches de modèles de langage (LLM) peut souvent être complexe, nécessitant des connaissances techniques approfondies et de nombreuses itérations pour parvenir à des résultats satisfaisants. Cependant, Flowise simplifie considérablement ce processus en offrant une approche intuitive et sans code. Grâce à son interface par glisser-déposer, les utilisateurs peuvent rapidement passer de la phase de test à la production.

Flowise propose une panoplie de nœuds fonctionnels tels que les représentations vectorielles, les magasins vectoriels, les extracteurs de données web, et les chaînes LLM. Cette diversité permet aux utilisateurs de connecter facilement ces nœuds pour définir le flux de leur application, transformant ainsi leurs idées en réalité de manière efficace et personnalisée.

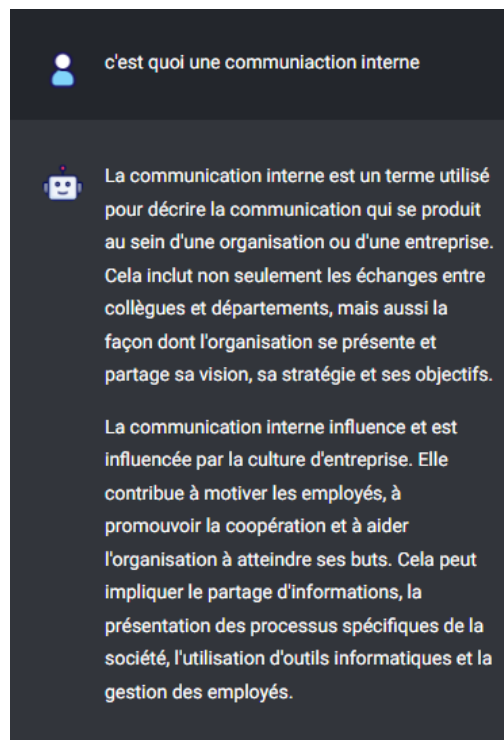
En utilisant Flowise, les développeurs et les professionnels de l'IA peuvent non seulement accélérer le cycle de développement, mais aussi maximiser la flexibilité et la précision de leurs applications LLM. Cela fait de Flowise une solution puissante pour ceux qui cherchent à innover dans le domaine des modèles de langage personnalisés tout en réduisant la complexité technique associée à leur mise en œuvre.

### – Expériences personnelles :

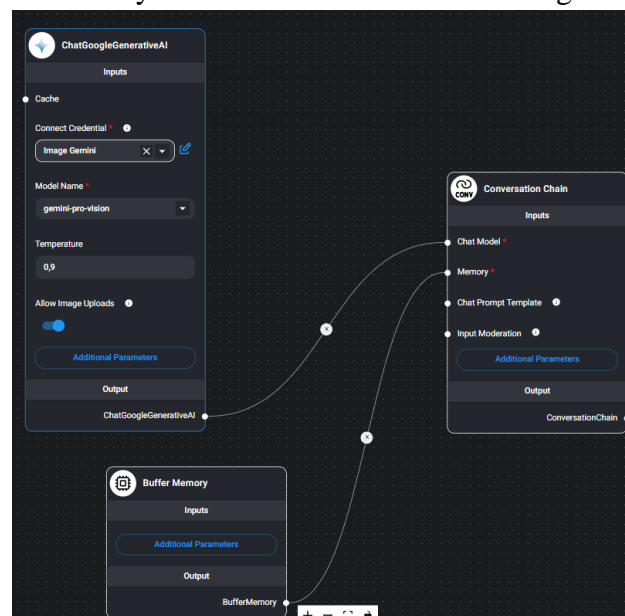
- Création d'un chatbot pour répondre sur des PDF :



Utilisant à la fois Flowise AI et l'API Cohere, j'ai conçu et mis en œuvre un chatbot spécialisé dans l'analyse de documents PDF. Cette initiative a impliqué la création d'une chaîne LLM personnalisée avec Flowise AI, permettant au chatbot d'interpréter de manière intelligente les informations extraites des documents téléchargés. Grâce à l'intégration de l'API Cohere, les capacités du chatbot ont été enrichies par une analyse sémantique avancée. Cette combinaison a permis au système de répondre de manière contextuelle et précise aux requêtes des utilisateurs, en fournissant des réponses adaptées et informatives basées sur le contenu spécifique des PDF analysés.



- Développement d'un Système de Reconnaissance d'Image :





En explorant les fonctionnalités avancées de Flowise AI, j'ai développé un système de reconnaissance d'image sophistiqué. Ce projet m'a permis de concevoir un système robuste capable d'analyser et de classifier des images selon divers critères prédéfinis, en particulier en utilisant le célèbre jeu de données CIFAR-10. En intégrant l'API Google Gemini avec Flowise AI, j'ai configuré des modèles de classification d'image précis et efficaces. Cette approche m'a permis de tester et d'optimiser les capacités du système pour assurer une reconnaissance visuelle de haute qualité, répondant aux exigences spécifiques du traitement d'images dans des contextes variés.



### b) 2<sup>ème</sup> cas d'usage : Chatbot avec Streamlit et Mistral AI

#### – Définition



En faisant mes recherches, j'ai trouvé Streamlit, un framework Python qui facilite grandement le développement rapide d'applications web interactives. Streamlit se distingue par sa capacité à permettre aux développeurs de créer des applications en quelques lignes de code seulement. Il offre une bibliothèque de composants prêts à l'emploi tels que des graphiques, des widgets interactifs et des visualisations de données dynamiques, ce qui simplifie considérablement la création d'interfaces utilisateur conviviales.

J'ai personnellement utilisé Streamlit pour développer mon chatbot, profitant de sa facilité d'utilisation et de sa flexibilité pour intégrer des fonctionnalités avancées. Ce framework permet de transformer facilement des scripts Python en applications web robustes, sans nécessiter une expertise approfondie en développement web.

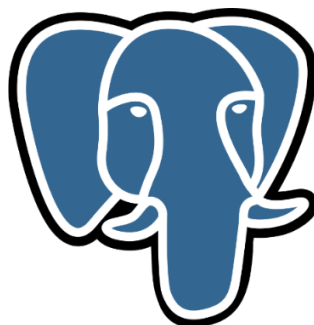


En plus de Streamlit, j'ai opté pour Mistral AI en raison de ses capacités avancées et de ses performances supérieures dans le domaine du traitement du langage naturel. Mistral AI se distingue par sa capacité à gérer efficacement des tâches complexes telles que la génération de texte, la traduction automatique et l'analyse de sentiment. Son architecture optimisée et ses fonctionnalités avancées en font un choix idéal pour développer des applications de pointe nécessitant une précision et une fiabilité accrues dans le traitement des données linguistiques.

J'ai choisi Mistral AI plutôt qu'OpenAI en raison de considérations spécifiques liées aux besoins de mon projet. Mistral AI offre des fonctionnalités plus adaptées aux exigences particulières que j'avais, notamment en matière de personnalisation et de traitement des données spécifiques. De plus, les coûts et les conditions d'utilisation de Mistral AI étaient plus favorables pour mon application, ce qui m'a permis de maximiser l'efficacité et la flexibilité du développement tout en respectant le budget et les contraintes du projet.

### – **Expériences personnelles :**

- Création d'une base de données PostgreSQL :

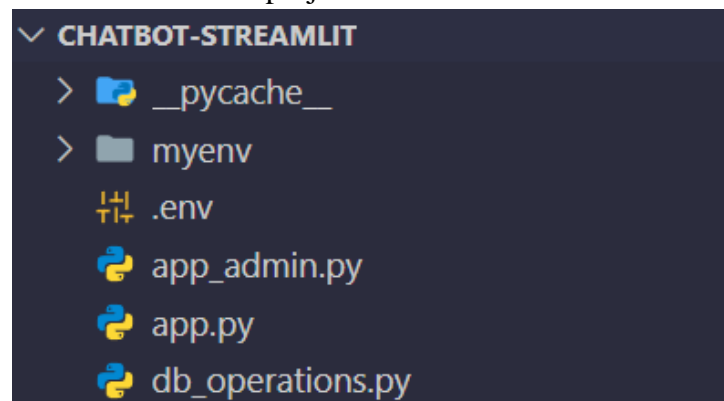


Pour ce projet, j'ai entrepris tout d'abord la création d'une base de données PostgreSQL avec une structure minimale comprenant les champs id, title et content. Cette base de données a été spécifiquement conçue pour stocker les données essentielles nécessaires au bon fonctionnement de mon chatbot. La conception du schéma s'est concentrée sur la définition

précise de ces trois champs afin de faciliter la gestion efficace des informations requises pour les interactions avec les utilisateurs.

id	🔑	title	content
14		Sous-requêtes	Module : Bases de donnéesrelationnelles3ème ...
15		Triggers	1 TP9 INF3081 BASE DE DONNEES RE...
18		Ordonnancement	Dr Oussama SMIMITEoussama.smimite@e-p...
19		Guide de stage technicien	Dr. Mifdal LahcenTout homme reçoit deux sorte...
21		Communication interne	COMMUNICATIONORGANISATIONNELLEPr.Y...
22		Jointures	1 TP7 INF3081 BASE DE DONNEES RE...

- Conception de la structure de mon projet :



La structure de mon projet Streamlit est organisée de manière à faciliter la gestion et le développement de l'application. Voici une description détaillée de chaque composant :

- **1. Fichier : .env**

- **Description :** Ce fichier contient toutes les variables d'environnement sensibles nécessaires au fonctionnement de l'application. Il inclut des données critiques telles que les clés API et les informations de connexion à la base de données.
- **Contenu Typique :**
  - **Clé API de Mistral AI :** Cette clé est utilisée pour authentifier les requêtes vers l'API de Mistral AI.
  - **Informations de Connexion à PostgreSQL :** Détails incluant l'hôte, le port, le nom d'utilisateur, le mot de passe et le nom de la base de données PostgreSQL.
- **Importance :** La sécurisation de ce fichier est essentielle pour protéger les informations sensibles et empêcher tout accès non autorisé.

- **2. Fichier : app\_admin.py**

- **Description :** Ce fichier gère l'interface administrateur de l'application. Il permet aux administrateurs d'ajouter et de supprimer différents fichiers PDF dans la base de données.
- **Fonctionnalités :**
  - **Ajout de PDFs :** Permet l'upload et le stockage de nouveaux fichiers PDF dans la base de données.
  - **Suppression de PDFs :** Permet la gestion et la suppression des fichiers PDF existants dans la base de données.
  - Interface pour gérer les fichiers PDF de manière sécurisée et organisée.

- **3. Fichier : app.py**

- **Description :** Ce fichier est responsable de l'interface utilisateur de l'application. Il permet aux utilisateurs de poser des questions directement au chatbot, qui répondra en utilisant les informations stockées dans les fichiers PDF de la base de données.
- **Fonctionnalités :**
  - Interface conviviale permettant aux utilisateurs de poser des questions au chatbot.
  - Intégration avec la base de données pour accéder aux fichiers PDF et extraire les informations pertinentes pour répondre aux questions.

- **4. Fichier : db\_operations.py**

- **Description :** Ce fichier regroupe toutes les opérations en relation avec la base de données. Il sert d'interface entre l'application et la base de données PostgreSQL pour gérer les fichiers PDF.
- **Fonctionnalités :**
  - **Ajout de PDFs :** Fonction pour ajouter de nouveaux fichiers PDF à la base de données.
  - **Suppression de PDFs :** Fonction pour supprimer des fichiers PDF de la base de données.
  - **Récupération de Contenu de PDFs :** Fonction pour recevoir et traiter le contenu des fichiers PDF stockés dans la base de données.

- Création de l'interface admin de mon application web :

**Chat with PDF**

Upload your PDF

Drag and drop file here  
Limit 200MB per file • PDF

Browse files

**Ask Questions about Stored PDFs**

Ask a question about the PDFs:

**Delete a PDF**

Select a PDF to delete

14: Sous-requêtes

Delete PDF

L'interface administrateur de mon projet Streamlit est conçue pour être simple et intuitive, permettant une gestion efficace des fichiers PDF et une interaction directe avec le chatbot. Voici les principales fonctionnalités :

### Fonctionnalités Principales :

#### 1. Upload de PDFs :

- **Description :** L'interface permet aux administrateurs d'ajouter de nouveaux fichiers PDF dans la base de données. Cette fonctionnalité est essentielle pour enrichir la base de données avec de nouveaux contenus.
- **Processus :**
  - L'administrateur sélectionne un fichier PDF à partir de son système local.
  - L'administrateur entre un nom pour le fichier, qui sera utilisé comme 'Title' dans la base de données.
  - Le fichier est ensuite téléchargé et stocké dans la base de données PostgreSQL.

#### 2. Interaction avec le Chatbot :

- **Description :** Les administrateurs ont la possibilité de poser des questions directement au chatbot, de la même manière que les utilisateurs finaux. Cela permet de tester et de vérifier le contenu des fichiers PDF stockés dans la base de données.

- **Utilisation :**

- L'administrateur saisit une question dans l'interface.
- Le chatbot répond en utilisant les informations extraites des fichiers PDF disponibles dans la base de données.

### 3. Suppression de PDFs :

- **Description :** Cette fonctionnalité permet aux administrateurs de supprimer des fichiers PDF de la base de données lorsqu'ils ne sont plus nécessaires ou souhaitables.
- **Processus :**
  - L'administrateur sélectionne un fichier PDF à supprimer.
  - L'interface affiche l'ID et le nom (Title) du fichier pour confirmation.
  - Le fichier est alors retiré de la base de données, assurant ainsi une gestion efficace et à jour des contenus.

- Création de l'interface utilisateur de mon application web :

**Chat with PDF**  

**Ask Questions about Stored PDFs**

Ask a question about the PDFs:

L'interface utilisateur de mon projet Streamlit est conçue pour offrir une expérience conviviale et intuitive, facilitant une interaction directe avec le chatbot. Voici la principale fonctionnalité :

### Fonctionnalité Principale :

#### Interaction avec le Chatbot :

- **Description :** L'interface utilisateur permet aux utilisateurs finaux de poser des questions directement au chatbot. Le chatbot répond en utilisant les informations stockées dans les fichiers PDF de la base de données.
- **Utilisation :**
  - L'utilisateur saisit une question dans l'interface.
  - Le chatbot traite la question et répond en utilisant les informations extraites des fichiers PDF disponibles dans la base de données.

- La réponse du chatbot est basée sur le contenu pertinent des fichiers PDF, offrant ainsi des informations précises et utiles.

### Chat with PDF

#### Ask Questions about Stored PDFs

Ask a question about the PDFs:

quel sont les types de jointures

Answer: Il existe plusieurs types de jointures en SQL :

1. INNER JOIN (jointure interne) : sélectionne les enregistrements qui ont des valeurs correspondantes dans les deux tables.
2. LEFT JOIN (jointure externe gauche) : sélectionne tous les enregistrements de la table de gauche et les enregistrements correspondants de la table de droite. Si aucune correspondance n'est trouvée dans la table de droite, les résultats auront des valeurs NULL.
3. RIGHT JOIN (jointure externe droite) : sélectionne tous les enregistrements de la table de droite et les enregistrements correspondants de la table de gauche. Si aucune correspondance n'est trouvée dans la table de gauche, les résultats auront des valeurs NULL.
4. FULL OUTER JOIN (jointure externe complète) : sélectionne tous les enregistrements des deux tables, avec des valeurs NULL pour les enregistrements qui n'ont pas de correspondance dans l'autre table.
5. CROSS JOIN (jointure croisée) : renvoie un produit cartésien des deux tables, c'est-à-dire qu'il combine chaque ligne de la première table avec chaque ligne de la deuxième table.
6. SELF JOIN (auto-jointure) : permet de joindre une table à elle-même en utilisant des alias de table différents. Elle est souvent utilisée pour interroger des données hiérarchiques ou pour comparer des lignes au sein d'une même table.

## 2-2- Les missions secondaires

### a) Optimisation et maintenance de la base de données :

- **Analyse et amélioration des performances des requêtes SQL**
- **Mise en place de stratégies efficaces pour organiser et structurer les données afin de faciliter leur accès et leur utilisation par le chatbot.**

### b) Apprentissage et implémentation de la génération augmentée de récupération (RAG) :

- **Étudier les concepts de la génération augmentée de récupération (RAG).**
- **Intégrer et tester les techniques de RAG dans un autre projet pour améliorer la précision et la pertinence des réponses générées.**

Utilisant PG Vector et RAG, j'ai développé une application de base qui :

1. Extrait le texte des documents (par exemple, les PDF).
2. Convertit ce texte en embeddings en utilisant un modèle pré-entraîné de Mistral AI.
3. Stocke ces embeddings dans une base de données PostgreSQL avec PG Vector.

4. Interroge la base de données pour trouver les documents dont les embeddings sont les plus similaires à l'embedding d'un texte de requête.

Ce processus transforme le contenu textuel des documents PDF en formes vectorisées et les interroge pour la similarité, en utilisant PostgreSQL avec PG Vector et SQLAlchemy.

```
(myenv) PS C:\Users\admin\Downloads\Stage> python ragpgvector.py
COMMUNICATION
ORGANISATIONNELLE
Pr.Youssef ELMINOR
Session :Hiver
Lacommunication
interne

DEFINITION
Onappelle communication interne lacommunication quis'effectue entre
une organisation et ceux qui lui fournissent leur force de travail. La
communication interne existe dans touttype d'organisation, publique ou
privée, économique ouassociative.

FORMES
La communication interne peut revêtir quatre formes différentes:
-Elle peut être « descendante »: il s 'agit d'information qui est adressée
parlahiérarchie del'entreprise àessalariés.
-Elle peut être « ascendante » : lorsque les salariés adressent des
informations à leur hiérarchie, ils peuvent l'effectuer via des enquêtes
quileursont adressées, desdispositifs detype «boites àidées ».
FORMES (SUITE)
-La communication peut être« latérale » lorsqu'elle s'effectue entre
collègues.
-Elle peut enfin être« multilatérale » dans le cas de communications
transversales à plusieurs types de salariés. Dans cette dernière
hypothèse, les outils reposent sur des forums de discussion ouverts à
tous, desréunions deservices, desrencontres avec la direction.
OUTILS
On peut classer les outils de communication en trois grandes familles.
```

Cette petite expérience m'a permis de mieux comprendre les concepts de la génération augmentée de récupération (RAG) et de PG Vector. Grâce à cela, j'ai pu réaliser avec succès mon projet Streamlit.

### c) Développement de l'interface utilisateur :

- **Améliorer et peaufiner l'interface utilisateur de l'application pour offrir une expérience utilisateur plus intuitive et agréable.**
- **Ajouter des fonctionnalités interactives pour améliorer l'engagement des utilisateurs.**

### d) Gestion de Projet :

- **Suivi de l'avancement des tâches et des projets.**
- **Coordination avec les différentes parties prenantes pour s'assurer que les objectifs sont atteints dans les délais impartis.**



## CONCLUSION GENERALE

L'objectif de mon stage chez Wire IT était de comprendre et d'appliquer les notions de base de l'intelligence artificielle générative. Ce stage, d'une durée de six mois, s'est déroulé de juin à août 2024 à Marrakech, au Maroc. Wire IT, un cabinet d'ingénierie et de conseil, m'a offert l'opportunité de travailler sur des projets liés à l'IA, sous la supervision directe de M. Ismail, le CEO de l'entreprise.

Durant mon stage, j'ai mené plusieurs missions. La mission principale consistait à appliquer les principes de l'IA générative, en incluant la collecte et le nettoyage des données, l'entraînement du modèle d'IA, et les tests des solutions développées. J'ai également été impliqué dans l'optimisation et la maintenance de la base de données, l'implémentation de la génération augmentée de récupération (RAG), et le développement de l'interface utilisateur. Ces missions ont permis d'atteindre les objectifs fixés et d'apporter des solutions concrètes aux besoins identifiés.

Ce stage m'a permis d'acquérir des compétences techniques avancées en intelligence artificielle, notamment en gestion de données et en entraînement de modèles. J'ai pu développer des points forts dans la conception et la mise en œuvre de solutions IA, tout en identifiant des domaines à améliorer, comme la gestion des paramètres des modèles. Sur le plan personnel, j'ai renforcé mes compétences en communication et en gestion de projet, en apprenant à travailler de manière autonome et à collaborer efficacement avec un encadrant.

Je souhaite exprimer ma plus sincère gratitude à M. Ismail Witam, le CEO de Wire IT, pour son soutien constant et ses conseils précieux tout au long de mon stage. Sa supervision a été déterminante pour ma compréhension et mon application des concepts d'IA générative. Je remercie également la direction de Wire IT pour m'avoir offert cette opportunité unique de contribuer à des projets innovants et d'acquérir une expérience professionnelle précieuse. Enfin, je suis reconnaissant envers la direction de mon école pour leur accompagnement et leur soutien durant cette période, facilitant ainsi le bon déroulement de mon stage et la réussite de cette expérience professionnelle.

---

# BIBLIOGRAPHIE

Lister le détail (auteur, titre, date et lieu d'édition, etc.) de tous les documents utilisés lors de la recherche bibliographique précédant la phase de conception du projet ou ayant servi de référence.

Numéroter ces références ou indiquer entre crochets les premières lettres de son auteur et l'année de parution de l'article ou du livre.

Deux manières de les classer peuvent être envisagées : par ordre alphabétique ou par ordre d'apparition dans le document.

Si dans votre document, vous faites références à un article ou un ouvrage, indiquer dans le paragraphe concerné la référence entre crochets.

Exemple:

[Ber79] : Berouti M., Schwartz R. and Makhoul J., "Enhancement of Speech Corrupted by Acoustic Noise," Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 208-211, April 1979.

[Bol78] : Boll S.F., "Suppression of Noise in Speech Using the SABER Method," ICASSP 78, Tulsa, Oklahoma, U.S.A., pp. 606-609, 1978.

**A. Documentations des outils utilisés**

Outils	Documentation
<b>Flowise</b>	<a href="https://docs.flowiseai.com/">https://docs.flowiseai.com/</a>
<b>Mistral AI</b>	<a href="https://docs.mistral.ai/">https://docs.mistral.ai/</a>
<b>Streamlit</b>	<a href="https://docs.streamlit.io/">https://docs.streamlit.io/</a>
<b>PostgreSQL</b>	<a href="https://www.postgresql.org/docs/">https://www.postgresql.org/docs/</a>
<b>Heidi SQL</b>	<a href="https://www.heidisql.com/help.php">https://www.heidisql.com/help.php</a>
<b>Microsoft VS Code</b>	<a href="https://code.visualstudio.com/docs">https://code.visualstudio.com/docs</a>
<b>Python</b>	<a href="https://docs.python.org/3/">https://docs.python.org/3/</a>

**B. Codes sources**

#### a) Code de l'interface administrateur

##### # Upload a PDF file

```
pdf = st.file_uploader("Upload your PDF", type='pdf')

if pdf is not None:
    pdf_reader = PdfReader(pdf)
    text = ""
    for page in pdf_reader.pages:
        text += page.extract_text()

    title = st.text_input("Enter a title for this PDF:")
    if st.button("Upload PDF"):
        if title:
            pdf_id = db_operations.upload_pdf(title, text)
            st.success(f"PDF uploaded with ID: {pdf_id}")
        else:
            st.error("Please provide a title for the PDF.")
```

Ce code est une partie de l'interface administrateur qui permet de téléverser des fichiers PDF, d'en extraire le texte, et d'uploader ce texte dans une base de données avec un titre fourni.

#### 1. Téléversement du PDF :

- L'administrateur est invité à téléverser un fichier PDF à l'aide d'un widget de téléversement (`file_uploader`).

#### 2. Lecture et extraction du texte du PDF :

- Une fois le fichier PDF téléversé, il est lu à l'aide de `PdfReader`.
- Le code itère à travers chaque page du PDF et extrait le texte de chaque page, en les combinant en une seule chaîne de texte.

#### 3. Entrée du titre :

- L'administrateur est ensuite invité à entrer un titre pour le PDF dans un champ de texte (`text_input`).

#### 4. Téléversement du texte et du titre dans la base de données :

- Un bouton "Upload PDF" est fourni pour que l'administrateur confirme l'upload.
- Lorsque l'administrateur clique sur ce bouton, le code vérifie que le titre a été fourni.

- Si un titre est fourni, le texte extrait du PDF et le titre sont uploadés dans la base de données à l'aide d'une fonction spécifique (`upload_pdf`) de `db_operations`.
- Un message de succès est affiché avec l'identifiant unique du PDF uploadé.
- Si aucun titre n'est fourni, un message d'erreur demande à l'administrateur de fournir un titre.

### b) Code de l'interface utilisateur

```
query = st.text_input("Ask a question about the PDFs:")
if query:
    client = load_client()
    if client:
        all_pdfs_content = db_operations.get_all_pdfs_content()
        context = "\n".join([pdf[1] for pdf in all_pdfs_content])
        message = ChatMessage(
            role="user",
            content=f"Context: {context}\nQuestion: {query}"
        )
        response = client.chat(
            model="mistral-large-latest",
            messages=[message]
        )
        answer = response.choices[0].message.content
        st.write("Answer:", answer)
```

Ce morceau de code fait partie de l'interface utilisateur qui permet de poser des questions sur le contenu des PDF précédemment téléversés et stockés dans une base de données. Voici ce que fait ce code :

#### 1. Entrée de la question :

- L'utilisateur est invité à entrer une question dans un champ de texte (`text_input`).

#### 2. Vérification de la question :

- Si une question est entrée par l'utilisateur, le code continue.

#### 3. Chargement du client Mistral AI :

- Le code charge un client pour interagir avec l'API de Mistral AI (`load_client`).

#### 4. Récupération du contenu des PDF :

- Si le client est chargé avec succès, le code récupère tout le contenu des PDF stockés dans la base de données à l'aide de `db_operations.get_all_pdfs_content()`.
- Le contenu de tous les PDF est combiné en une seule chaîne de texte (`context`).

#### 5. Création du message de chat :

- Un message de chat est créé, incluant le contexte (tout le texte combiné des PDF) et la question de l'utilisateur.

#### 6. Interaction avec le modèle Mistral AI :

- Le message de chat est envoyé au modèle `mistral-large-latest` via le client Mistral AI.
- Le modèle génère une réponse à la question de l'utilisateur.

#### 7. Affichage de la réponse :

- La réponse générée par le modèle est affichée à l'utilisateur à l'aide de `st.write`.

### c) Code des opérations de la base de données

```
def delete_pdf(pdf_id):
    conn = connect_db()
    cursor = conn.cursor()

    # Fetch the title of the PDF before deleting
    cursor.execute("SELECT title FROM pdfs WHERE id = %s", (pdf_id,))
    pdf_title = cursor.fetchone()[0]

    # Delete the PDF
    cursor.execute("DELETE FROM pdfs WHERE id = %s", (pdf_id,))
    conn.commit()
    cursor.close()
    conn.close()

    return pdf_title # Return the title of the deleted PDF
```

Ce code est un extrait des opérations de la base de données, définissant une fonction `delete_pdf` qui supprime un enregistrement PDF à partir de son identifiant (`pdf_id`). Avant de supprimer le PDF, la fonction récupère son titre afin de le retourner une fois la suppression effectuée.

### 1. Connexion à la base de données :

- La fonction commence par établir une connexion à la base de données à l'aide de `connect_db()`.
- Elle crée ensuite un curseur pour exécuter des commandes SQL.

### 2. Récupération du titre du PDF :

- Avant de supprimer le PDF, la fonction exécute une requête SQL pour récupérer le titre du PDF correspondant à l'identifiant (`pdf_id`).
- La requête `SELECT title FROM pdfs WHERE id = %s` est exécutée avec le `pdf_id` comme paramètre.
- Le titre du PDF est récupéré et stocké dans la variable `pdf_title`.

### 3. Suppression du PDF :

- La fonction exécute une autre requête SQL pour supprimer l'enregistrement PDF de la base de données.
- La requête `DELETE FROM pdfs WHERE id = %s` est exécutée avec le `pdf_id` comme paramètre.
- Les changements sont confirmés dans la base de données avec `conn.commit()`.

### 4. Fermeture de la connexion :

- Le curseur et la connexion à la base de données sont fermés pour libérer les ressources.

### 5. Retour du titre du PDF supprimé :

- La fonction retourne le titre du PDF qui a été supprimé.