

# Rapport de Projet : Agent IA de Rédaction de Rapport de Stage

## 1. Introduction

Ce projet consiste en la création d'un agent IA pour aider un utilisateur à rédiger un rapport de stage. L'objectif est de transformer des notes et des recherches web en un document Markdown/PDF.

## 2. Architecture

L'agent suit un graphe où chaque action communique avec les autres. L'agent sait à tout moment ce qu'il a déjà fait et ce qu'il lui reste à faire.

étapes :

- **La préparation** : Au lancement, le nœud « initialisation » discute avec l'utilisateur pour comprendre le contexte du stage et génère automatiquement un plan de rapport.
- **Le choix du chemin** : L'agent décide de sa stratégie. Selon le titre du chapitre, il va soit chercher des infos sur le Web, soit chercher dans les notes de stage de l'utilisateur stockées localement. Si il a déjà tout en mémoire, il passe directement à l'écriture.
- **La rédaction** : Une fois les informations rassemblées, le nœud « redaction » génère le texte au format Markdown.
- **Interaction utilisateur** : l'agent n'écrit pas tout le rapport d'un coup dans son coin. Après chaque chapitre terminé, le programme s'arrête et pose une question à l'utilisateur : "Voulez-vous rédiger le chapitre suivant ?".
  - Si tu dis « oui », il repart dans la boucle pour le chapitre suivant.
  - Si tu dis « non », il s'arrête.
- **Rapport Final** : une fois tout les chapitre rédigés l'agent renvoie un rapport de stage au format pdf.

A chaque lancement l'agent demande si il faut continuer avec les informations qu'il a déjà ou réinitialisé les informations.

## 3. Stratégie de Gestion du Contexte

Le projet utilise deux méthodes pour enrichir la rédaction :

- **Persistance (State.json)** : L'état de l'avancement est sauvegardé en temps réel dans un fichier JSON, permettant d'arrêter et de reprendre le travail sans perte de données.

- **RAG (Retrieval-Augmented Generation) :**
  - **Externe** : Recherche d'informations sur l'entreprise via des outils de recherche web.
  - **Interne** : Extraction de contextes spécifiques depuis une base de données vectorielle ChromaDB contenant les notes de stage.

## 4. Choix des Outils et Observabilité

- **LangGraph** : Choisi pour sa capacité à gérer des cycles (boucles) et des conditions logiques complexes.
- **FPDF2** : Utilisé pour l'export final en PDF avec gestion des polices Unicode pour éviter les erreurs d'encodage.

## 5. Observabilité

Le terminal affiche chaque étape de réflexion ("Analyse de la progression", "Recherche locale"), permettant à l'utilisateur de comprendre les décisions de l'IA.

## 6. Roadmap du Développement

Le projet a suivi une progression logique :

1. **Phase 1** : Création du workflow de base et de la persistance JSON.
2. **Phase 2** : Construction du cerveau (initialisation, logique de décision, boucle de validation).
3. **Phase 3** : Rédaction et intelligence (moteur d'écriture, auto-analyse).
4. **Phase 4** : Finalisation de l'export PDF.