# CHAPITRE 4 : ANALYSE ET CONCEPTION

## 1. Introduction

Dans cette section, nous présentons les différentes étapes de l’analyse et de la conception de notre application mobile de recherche de stages et d’emplois.  
L’objectif est de relever les besoins fonctionnels et non fonctionnels identifiés lors de l’étude préalable, puis de les traduire sous forme de modèles conceptuels, logiques et physiques de données. Ces modèles constituent la base pour la phase de développement.

## 2. Analyse des besoins

#### 2.1 Besoins fonctionnels

Les principales fonctionnalités attendues de l’application sont :

* Gestion des utilisateurs (candidats et recruteurs).
* Publication et gestion des offres de stages/emplois.
* Soumission et suivi des candidatures.
* Notifications (push, SMS, email) pour informer les candidats.
* Forum de discussion structuré en fils et messages.
* Interactions via un bot (Telegram) pour consulter rapidement les offres.
* Recherche centraliser

#### 2.2 Besoins non fonctionnels

* **Performance** : application fluide et réactive.
* **Sécurité** : protection des données personnelles (mots de passe, CV, lettres).
* **Fiabilité** : disponibilité des services, minimisation des pannes.
* **Ergonomie** : interface intuitive, adaptée aux jeunes et professionnels.
* **Extensibilité** : possibilité d’ajouter de nouvelles fonctionnalités ultérieurement.

## 3. Modélisation conceptuelle

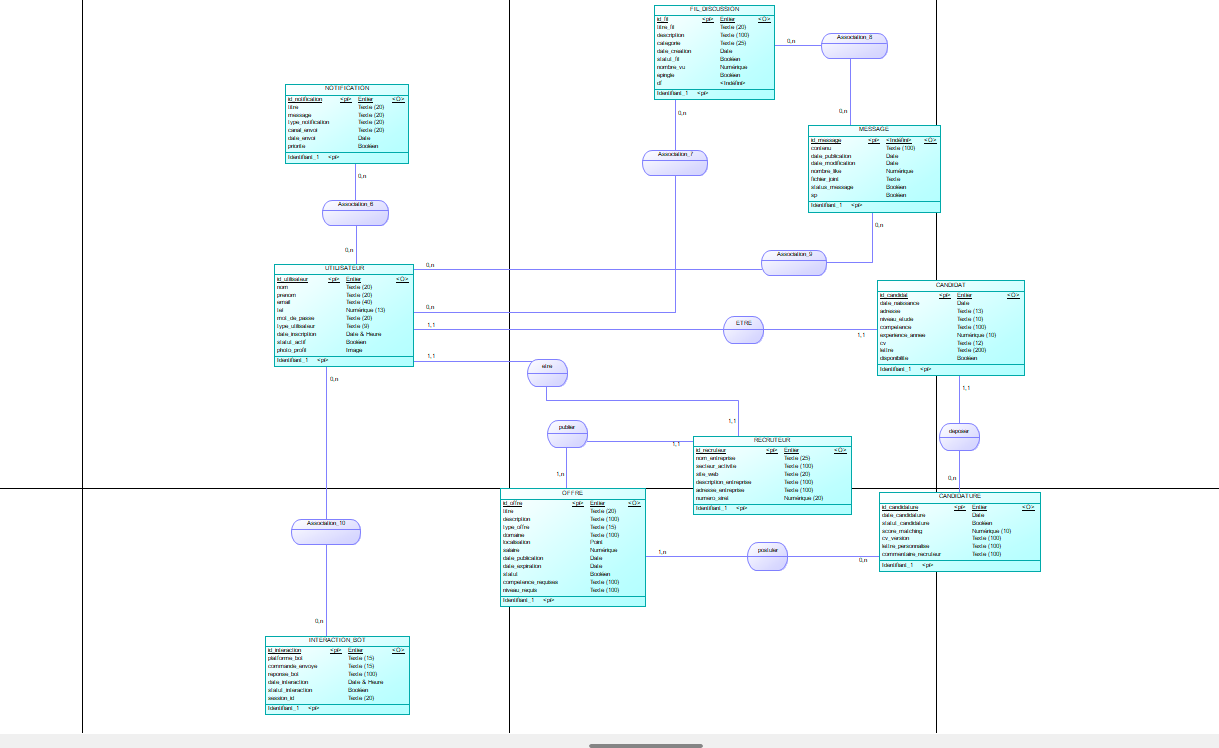
#### 3.1 Présentation du MCD

Le **Modèle Conceptuel de Données (MCD)** représente les entités principales du système, leurs attributs et les associations entre elles.

Les entités principales sont :

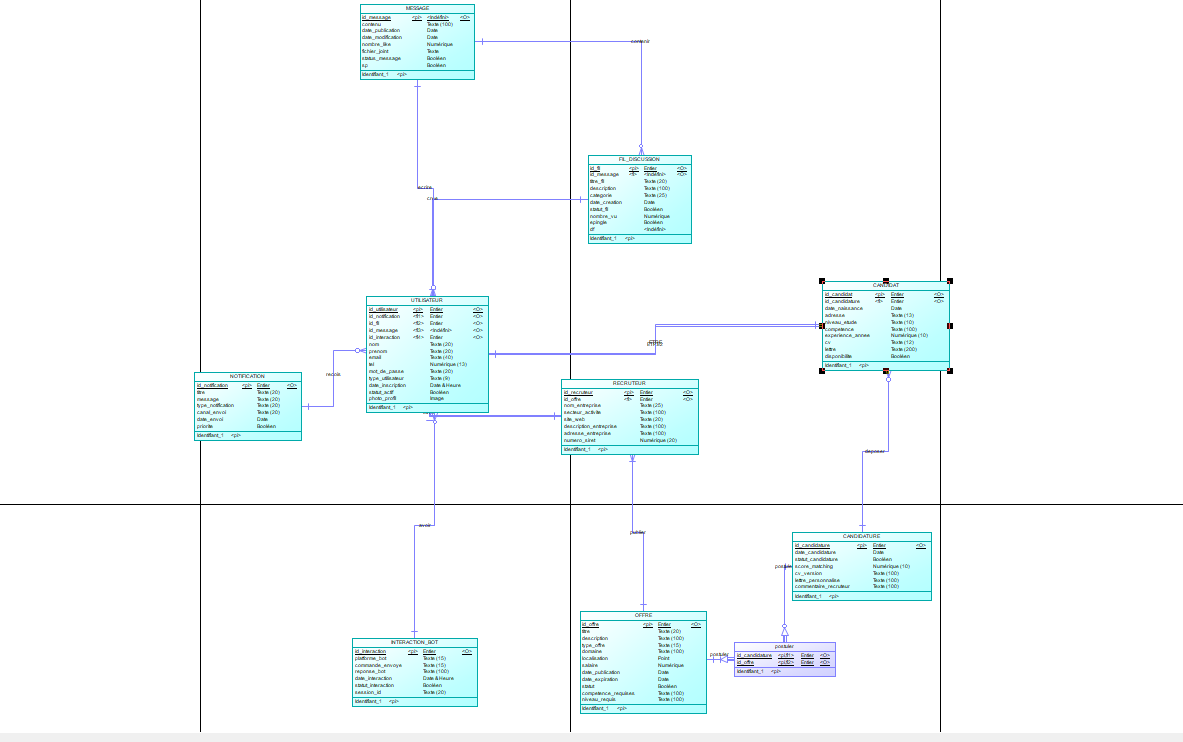
* **Utilisateur** : informations de base de tout utilisateur de la plateforme.
* **Candidat** : sous-type d’utilisateur, avec informations académiques et CV.
* **Recruteur** : sous-type d’utilisateur, avec informations sur l’entreprise.
* **Offre** : stages ou emplois proposés par les recruteurs.
* **Candidature** : relation entre candidat et offre, avec date, statut et score IA.
* **Notification** : messages envoyés aux candidats.
* **Fil de discussion** : thèmes de discussion dans le forum.
* **Message** : contributions des utilisateurs dans un fil.
* **Interaction Bot** : échanges entre un utilisateur et le bot Telegram.

Chaque entité est liée aux autres par des associations avec des **cardinalités précises**.



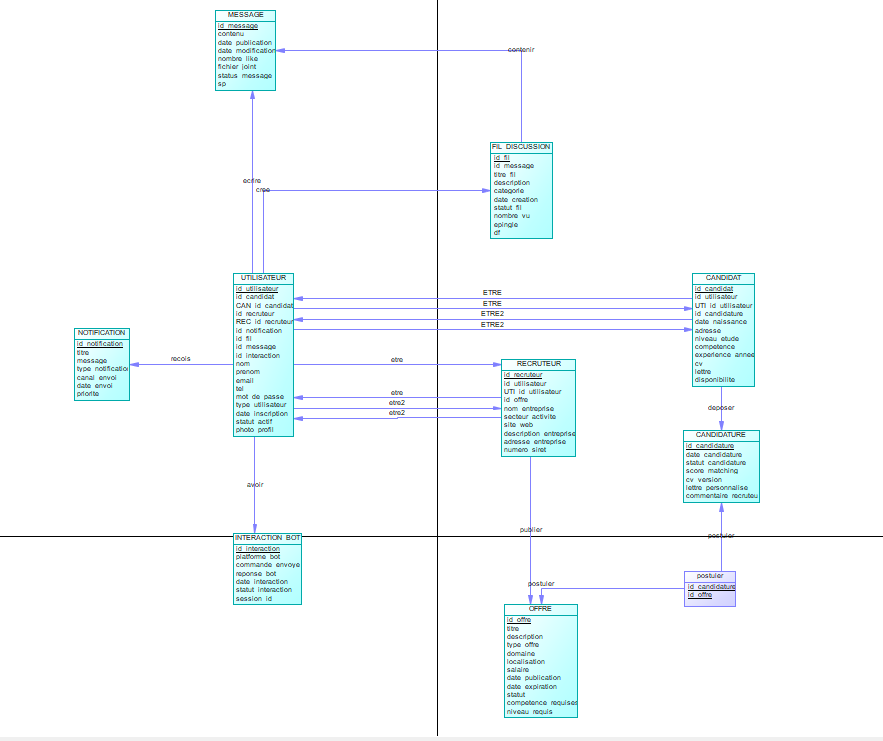
## 4. Modélisation logique

Le **Modèle Logique de Données (MLD)** traduit le MCD en un schéma relationnel adapté aux bases de données relationnelles.  
Les entités deviennent des **tables**, les attributs deviennent des **colonnes**, et les associations sont transformées en **clés étrangères**.



## 5. Modélisation physique

Le **Modèle Physique de Données (MPD)** décrit concrètement la structure de la base de données qui sera implémentée dans **Supabase (PostgreSQL)**.  
Il inclut les types de données précis, les clés primaires, les clés étrangères et les contraintes d’intégrité.



## 6. Conclusion

La phase d’analyse et conception nous a permis de :

* Identifier clairement les besoins fonctionnels et non fonctionnels.
* Construire un **MCD** cohérent représentant le domaine métier.
* Déduire un **MLD** structuré, prêt à être traduit en schéma relationnel.
* Définir un **MPD** complet, adapté à PostgreSQL via Supabase.

Ces modèles constituent une base solide pour la prochaine étape du projet : l’implémentation et le développement de l’application.