

NOM	SAINT-HUBERT
Prénom	Emma
Date de naissance	02/02/1998

## Bloc 3 Développement d'une solution Data/IA



**Projet : Création d'un site web d'offres de stage pour les étudiants de l'Université de Bordeaux**

## Dossier : Concevoir, spécifier et sécuriser une application

### 1. En se basant sur l'annexe 1, établissez la partie spécification fonctionnelle du cahier des charges. (30 points)

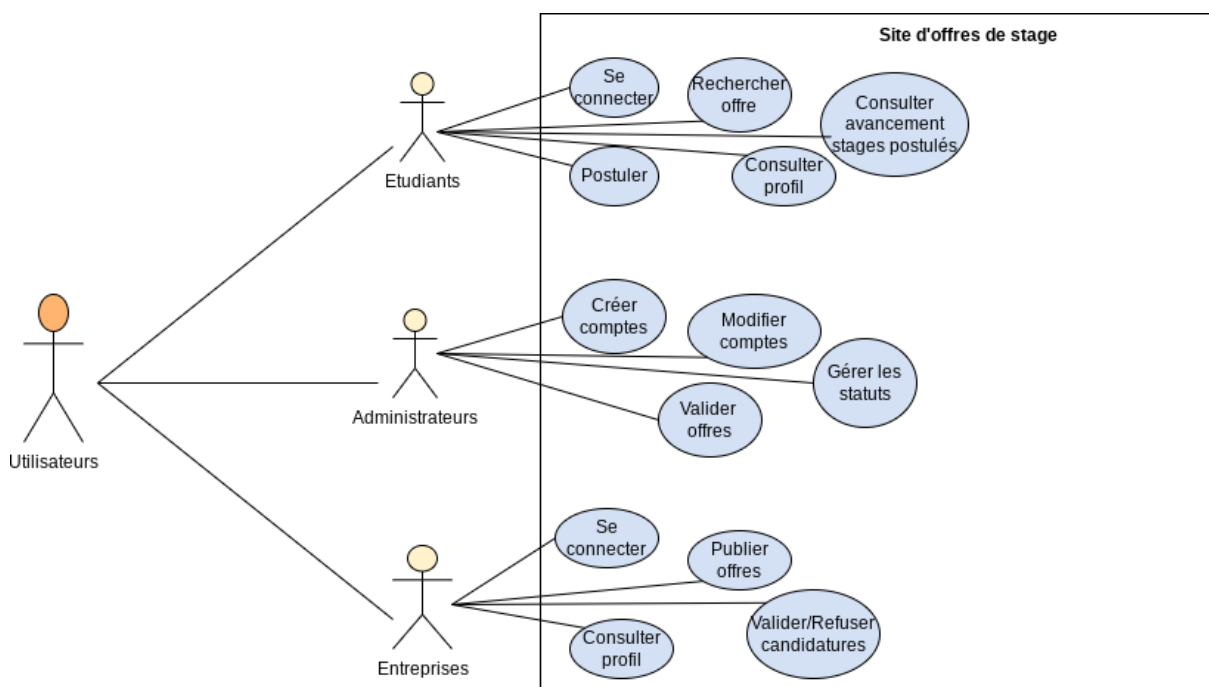
L'objectif de l'application est de fournir aux étudiants une plateforme leur permettant de trouver des offres de stage correspondant à leur domaine d'études.

Mais également de permettre aux entreprises de publier des offres de stage facilement et rapidement.

Pour cela j'ai identifié trois profils d'utilisateurs pouvant utiliser le site :

- Etudiants
- Entreprises
- Administrateurs

Ci-après le diagramme des cas d'utilisation UML :



Fonctionnalités pour les administrateurs
Créer des comptes étudiants, entreprises et administrateurs
Modifier des informations sur les comptes déjà créés
Gérer les statuts des utilisateurs, y compris l'activation et la désactivation des comptes.
Valider les offres de stage publiées par les entreprises

Fonctionnalités pour les étudiants
Se connecter à l'application grâce à son login et mot de passe créés par l'administrateur
Consulter son profil avec la possibilité de mettre à jour son CV et sa lettre de motivation.
Rechercher des offres de stage disponibles, en fonction du domaine, secteur et métier,
Postuler aux offres de stage disponibles avec envoi de candidature
Consulter les stages postulés ainsi que leur statut d'avancement (en attente (EA), refusé (RE), accepté (AC))

Fonctionnalités pour les entreprises
Se connecter à l'application grâce à son login et mot de passe créés par l'administrateur
Consulter les informations sur son profil et mettre à jour la description sur son entreprise
Publier des offres de stage via un formulaire à remplir
Consulter les candidatures reçues pour les offres de stage publiées et les valider ou les refuser

## Modélisation de la base de données

### Dictionnaire de données

Entité : utilisateurs			
Champs	Type	Taille	Description
id_user	Integer	-	Identifiant unique de l'utilisateur
nom	Varchar	25	Nom de l'utilisateur
prenom	Varchar	25	Prénom de l'utilisateur
email	Varchar	30	Adresse e-mail de l'utilisateur
password	Varchar	20	Mot de passe de l'utilisateur
num_tel	Varchar	15	Numéro de téléphone de l'utilisateur
id_profil	Integer	-	Identifiant du profil. Sert de clé étrangère
id_statut	Integer	-	Identifiant du statut associé à utilisateur : clé étrangère
num_voie	Varchar	10	Numéro de voie de l'adresse de l'utilisateur
nom_voie	Varchar	40	Nom de voie de l'adresse de l'utilisateur
code_postal	Varchar	10	Code postal de l'utilisateur
commune	Varchar	25	Nom de la commune de l'adresse de l'utilisateur

Entité : profils			
Champs	Type	Taille	Description
id_profil	Integer	-	Identifiant unique du profil : Clé primaire
lib_profil	Varchar	15	Libellé du profil (Etudiant, Entreprise, Administrateur)

Entité : statuts			
Champs	Type	Taille	Description
id_statut	Integer	-	Identifiant unique du statut : Clé primaire
lib_statut	Varchar	15	Libellé du statut (actif :A, désactivé :D, publié :P, en attente : EA, accepté :AC, refusé :RE)

Entité : entreprises			
Champs	Type	Taille	Description
id_ent	Integer	-	Identifiant unique de l'entreprise : Clé primaire
id_user	Integer	-	Identifiant de l'utilisateur : clé étrangère
desc_ent	Varchar	1500	Description de l'entreprise

Entité : domaines			
Champs	Type	Taille	Description
id_dom	Integer	-	Identifiant unique du domaine : clé primaire
lib_dom	Varchar	50	Libellé du domaine

Entité : secteurs			
Champs	Type	Taille	Description
id_sec	Integer	-	Identifiant unique du secteur : clé primaire
lib_sec	Varchar	50	Libellé du secteur
id_dom	Integer	-	Identifiant du domaine associé au secteur : clé étrangère

Entité : metiers			
Champs	Type	Taille	Description
id_metier	Integer	-	Identifiant unique du métier : clé primaire
lib_metier	Varchar	100	Libellé du métier
id_sec	Integer	-	Identifiant du secteur associé au métier : clé étrangère
desc_metier	Varchar	1500	Description du métier

Entité : etudiants			
Champs	Type	Taille	Description
id_etu	Integer	-	Identifiant unique de l'étudiant : clé primaire
id_user	Integer	-	Identifiant de l'utilisateur associé à l'étudiant : clé étrangère
cv	Bytea		CV de l'étudiant
lm	Bytea		Lettre de motivation de l'étudiant

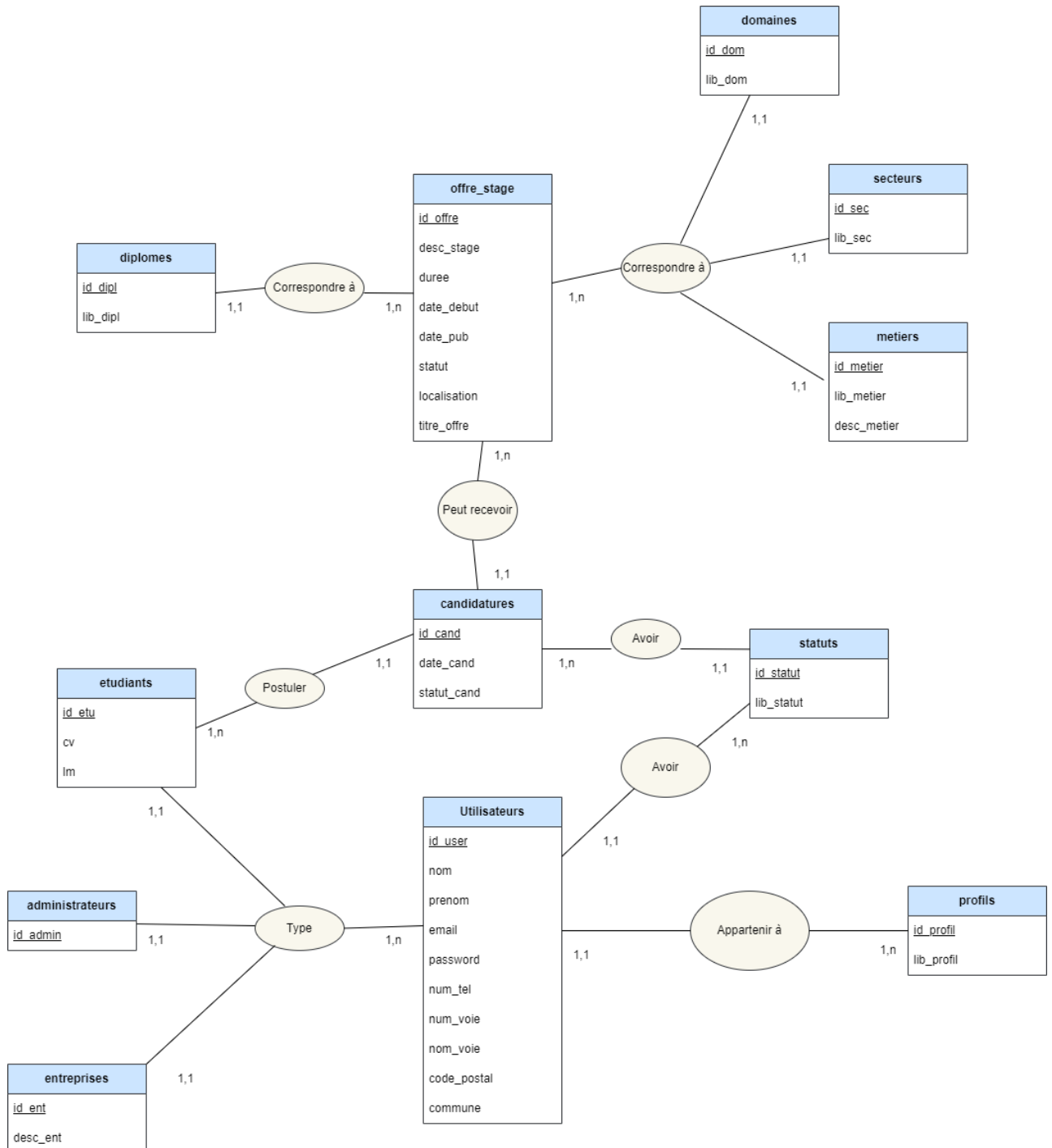
Entité : diplomes			
Champs	Type	Taille	Description
id_dipl	Integer	-	Identifiant unique du diplôme : clé primaire
lib_dipl	Varchar	50	Libellé du diplôme (licence, master)

Entité : administrateurs			
Champs	Type	Taille	Description
id_admin	Integer	-	Identifiant unique de l'administrateur
id_user	Integer	-	Identifiant de l'utilisateur associé à l'administrateur : clé étrangère

Entité : candidatures			
Champs	Type	Taille	Description
id_cand	Integer	-	Identifiant unique de la candidature
id_offre	Integer	-	Identifiant de l'offre de stage associé à la candidature : clé étrangère
date_cand	Date		Date de la candidature
id_etu	Integer	-	Identifiant de l'étudiant associé à la candidature : clé étrangère
id_statut	Integer		Identifiant du secteur associé à l'offre de stage : clé étrangère

Entité : offre_stage			
Champs	Type	Taille	Description
id_offre	Integer	-	Identifiant unique de l'offre de stage : clé primaire
id_dom	Integer	-	Identifiant du domaine associé à l'offre de stage : clé étrangère
id_sec	Integer	-	Identifiant du secteur associé à l'offre de stage : clé étrangère
id_metier	Integer	-	Identifiant du métier associé à l'offre de stage : clé étrangère
desc_stage	Varchar	5000	Description de l'offre de stage
duree	Varchar	25	Durée du stage en mois
date_debut	Date		Date de début du stage
date_pub	Date		Date de publication de l'offre de stage
statut	Varchar	20	Statut de l'offre de stage (publiée : P, en attente : EA)
id_ent	Integer	-	Identifiant du domaine associé à l'offre de stage : clé étrangère
localisation	Varchar	100	Localisation de l'offre de stage
id_dipl	Integer	--	Identifiant du domaine associé à l'offre de stage : clé étrangère
id_statut	Integer		Identifiant du secteur associé à l'offre de stage : clé étrangère
titre_offre	Varchar	50	Titre de l'offre de stage

## Modélisation conceptuelle des données



## 2. Quels langages allez-vous utiliser et à quel moment ? (30 points)

J'ai utilisé le langage Python avec Django pour le développement du backend du site d'offres de stage de l'université de Bordeaux. Django est un Framework web Python qui facilite la création d'applications web robustes et évolutives. Python avec Django a été utilisé pour gérer la logique métier, l'authentification des utilisateurs, les interactions avec la base de données et les flux de données. Django fournit également des fonctionnalités de sécurité intégrées, ce qui est essentiel pour protéger les données des utilisateurs et prévenir les attaques.

Pour le développement du frontend, j'ai utilisé HTML pour la structure et la présentation des pages web du site. HTML a permis de définir les éléments de contenu tels que les formulaires, les listes, les titres, les paragraphes, etc.

J'ai également utilisé Bootstrap, un Framework CSS, pour faciliter la conception et le développement du frontend. Bootstrap offre une bibliothèque de composants prédéfinis, des grilles responsives et des styles prêts à l'emploi, ce qui a permis de créer rapidement une interface utilisateur attrayante et adaptative.

J'ai également utilisé le langage de programmation JavaScript côté client pour ajouter des fonctionnalités interactives au site. JavaScript a été utilisé pour la validation des formulaires, les effets visuels, l'interaction avec les utilisateurs, et éventuellement pour intégrer des fonctionnalités supplémentaires telles que la recherche en temps réel ou la mise à jour dynamique des contenus.

En ce qui concerne la base de données, j'ai utilisé PostgreSQL comme système de gestion de base de données relationnelle. PostgreSQL a été utilisé pour stocker et gérer les données du site en concevant le schéma de la base de données et en tenant compte des entités et des relations identifiées lors de la modélisation de la base de données.

Pour interagir avec PostgreSQL depuis Django, j'ai utilisé des ORM (Object-Relational Mapping) pour faciliter la manipulation des données.

## 3. Lister les logiciels que vous allez utiliser et pourquoi ? (15 points)

J'ai utilisé Visual Studio Code pour le développement. C'est un environnement de développement intégré (IDE) populaire et léger. Il offre de nombreuses fonctionnalités et extensions qui facilitent la programmation.

Pour la partie base de données, j'ai utilisé PostgreSQL et pgAdmin.

- PostgreSQL est un système de gestion de base de données relationnelle (SGBDR) puissant, fiable et open-source. Il permet de stocker et gérer les données du site. Sa robustesse, sa performance et sa capacité à gérer de grands volumes de données en font un choix courant pour les applications web.
- pgAdmin est une interface graphique pour gérer et administrer les bases de données PostgreSQL. Il facilite la création et la gestion des tables, l'exécution de requêtes SQL, la surveillance des performances, etc.

Pour héberger mon code j'ai utilisé GitHub qui est une plateforme de développement collaboratif basée sur Git. Elle facilite la gestion du code source, la collaboration entre développeurs et la gestion des versions.



Enfin, pour le déploiement de mon site, j'ai utilisé Render qui est une plateforme de déploiement qui facilite le déploiement et la gestion d'applications web.

Render prend en charge l'hébergement de sites web développés avec Django, Python et PostgreSQL. Il offre une configuration et un déploiement simplifiés pour les applications Django. C'était donc le choix parfait pour héberger mon site web.

#### 4. **Selon vous, comment peut-on améliorer la sécurité du site web ? (25 points)**

Selon moi, il existe plusieurs solutions pour améliorer la sécurité du site web.

Dans un premier temps, une meilleure gestion des mots de passe serait essentielle :

- Encourager les utilisateurs à choisir des mots de passe forts et complexes en imposant des critères de complexité.
- Utilisez des méthodes de hachage sécurisées pour stocker les mots de passe des utilisateurs.
- Mettre en place une politique de renouvellement régulier des mots de passe et informez les utilisateurs sur les bonnes pratiques de sécurité des mots de passe.

Ensuite, mettre en place une protection contre les attaques de contrefaçon de requêtes intersites (CSRF) :

- Mettre en œuvre des jetons anti-CSRF pour prévenir les attaques de contrefaçon de requêtes intersites.
- Vérifiez l'origine des requêtes pour vous assurer qu'elles proviennent d'un domaine légitime.

Il serait aussi important de mettre en place des formations et sensibilisation à la sécurité :

- Sensibiliser les utilisateurs aux bonnes pratiques de sécurité, notamment en matière de gestion des mots de passe, de protection des données sensibles.