

# Introduction générale

Grâce au progrès de la recherche scientifique et technologique, l'informatique n'a cessé d'évoluer et de s'adapter aux besoins de l'homme.

De ce fait, toute entité professionnelle telle que les grandes entreprises, cherche à utiliser divers applications web dans le but de contribuer une meilleure rentabilité et afin de communiquer avec ses clients non seulement en personne mais aussi à distance.

C'est dans ce cadre que se situe notre travail qui consiste à concevoir et implémenter une application de gestion des offres d'emploi et formations. Cette application va traiter les publications des offres ou des formations par les recruteurs et la publication des cv par les demandeurs d'emploi.

Le présent rapport a pour objectif de donner une idée claire sur le travail effectué pendant la réalisation du projet.

Il s'articule autour de quatre chapitres : le premier chapitre présente l'étude préalable qui contient le cadre général du sujet, la présentation de l'existant, le problématique et la solution proposé. Le second chapitre sera dédié à la spécification des besoins de l'application. Ensuite un troisième chapitre qui présente la vue conceptuelle du projet. Finalement un dernier chapitre pour traiter tous les détails de la réalisation ainsi qu'un aperçu sur les importants résultats obtenus. Nous finissons par une conclusion générale qui résumera le travail et contiendra des perspectives d'avenir.

# **Chapitre 1**

## **Etude préalable**

## Introduction

Dans ce présent chapitre, nous exposons le contexte général de l'élaboration de notre projet. Ainsi, nous présentons dans un premier lieu une étude de l'existant afin de dégager par la suite les problèmes que notre application doit pallier et les objectifs auxquels elle doit répondre.

### 1.1. Cadre général du projet

Le présent projet intitulé « développement d'une application web de gestion des offres d'emplois » est réalisé dans le cadre de validation de la 3ème année licence fondamentale parcours sciences de l'informatique au sein de la faculté des sciences de Monastir pour l'année universitaire 2019-2020. Il a été réalisé au sein de laboratoire RLANTIS-FSM.

### 1.2. Présentation de l'existant

#### 1.2.1. Interface de Tunisie Travail

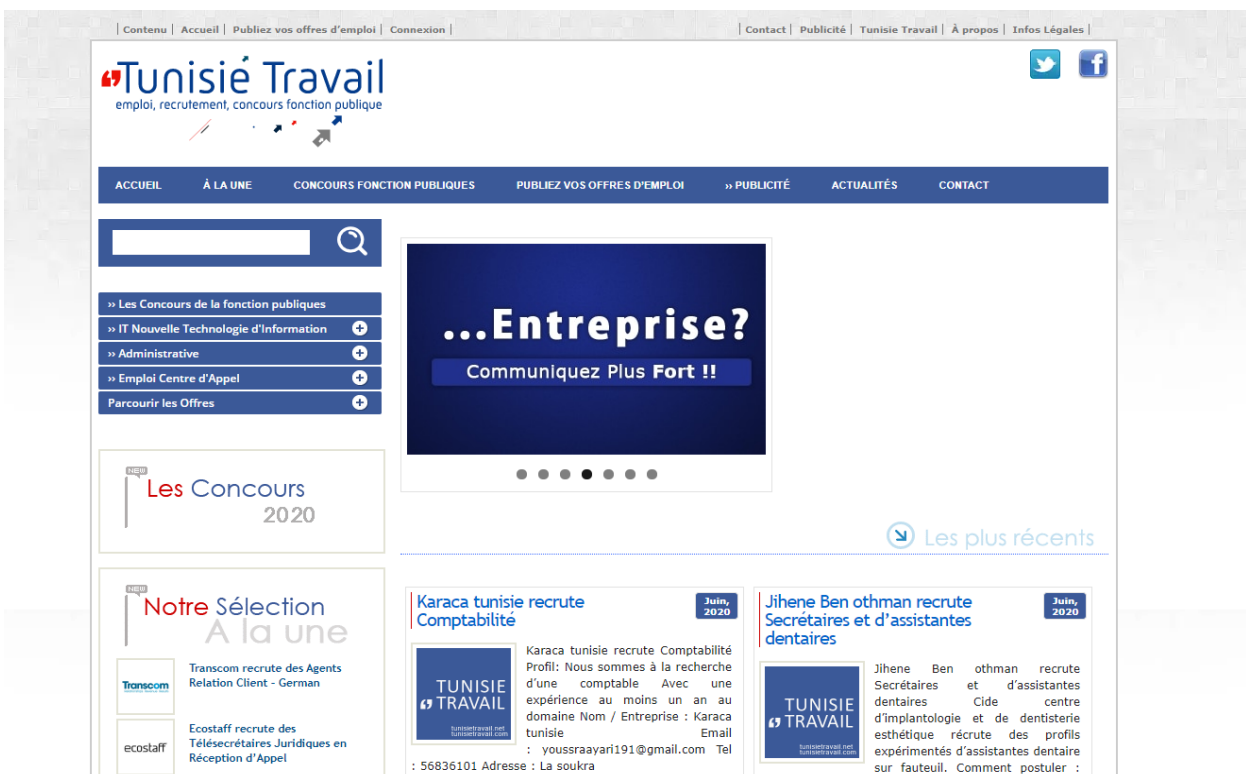


Figure 1.1 : Page d'accueil de Tunisie Travail

## 1.2.2. Interface de FAROJOB

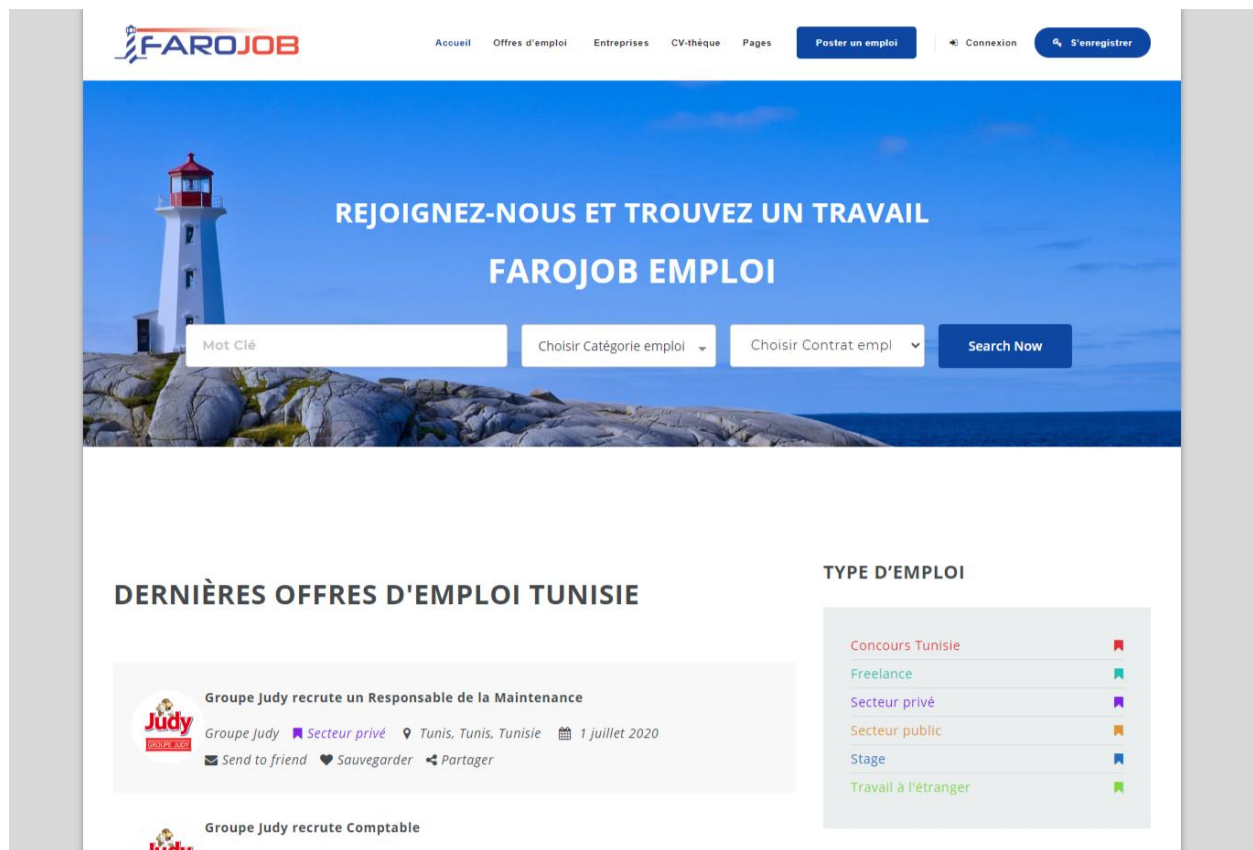


Figure 1.2 : Page d'accueil de FAROJOB

## 1.2.3. Description de l'existant

- **Tunisie Travail :**

Dans ce site web les candidats à la recherche d'emploi, Tunisie Travail permet de retrouver des offres d'emploi actualisées en Tunisie, à l'étranger, les concours et les recrutements freelance. Les entreprises à la recherche de collaborateurs, Tunisie Travail permet de diffuser des annonces d'emploi.

- **FAROJOB :**

Est un site web dédié aux entreprises pour publier des offres d'emplois et aux chercheurs d'emploi de publier pour une telle offre avec aussi la possibilité de postuler un cv pour que les entreprises peuvent trouver des employés grâce à leur cv.

#### **1.2.4. Les critiques de l'existant**

- **Tunisie Travail :**

Dans ce site web, à propos les recruteurs, il n'y a pas d'espace des candidatures ce qui signifie qu'ils ne peuvent pas recruter des employeurs uniquement s'ils disposent de ses propre site web pour postuler à un tel emploi.

- **FAROJOB :**

Dans ce site web si un recruteur veut publier une formation, elle n'a pas l'autorisation parce qu'il n'y a pas d'espace spécifique pour les formations.

#### **1.2.5. Problématique**

Dans nombreux sites web ils existent plusieurs problèmes. En fait, parmi ces derniers, la publication des formations dans plusieurs domaines. Aussi l'absence d'une espace spécifique dédié aux candidatures pour que les recruteurs peuvent consulter et prendre des décisions sans contacter les chercheurs qui ont postulé dans une telle offre.

#### **1.2.6. La solution proposée**

En constatant les insuffisances des sites actuel et afin de pallier aux défaillances, ce projet de gestion des offres d'emplois doit répondre à plusieurs objectifs qui sont :

- Assurer la simplicité du l'application tout en gardant la sécurité et la confidentialité des différents postes
- Concevoir des interfaces claires et simples a manipuler

- La rapidité, la fiabilité et la facilité des publications des offres d'emplois ou des formations

La solution que nous proposons est de créer une application web de gestion des offres d'emplois et formations. Cette application va traiter les publications des offres ou des formations par les recruteurs et la publication des cv par les chercheurs, ces derniers ont aussi la possibilité de postuler une candidature pour un tel emploi.

### **1.3. Conclusion**

Dans ce qui précède, nous avons situé le projet dans son cadre général, puis nous avons essayé d'analyser l'existant afin d'éclaircir les objectifs de notre application. En se basant sur cette étude, nous spécifierons dans le chapitre suivant les différentes fonctionnalités de notre projet.

## **Chapitre 2**

# **Analyse et spécification des besoins**

## Introduction

Dans ce chapitre, nous allons nous focaliser sur les fonctionnalités offertes par notre application ; cela en recensant les différents besoins fonctionnels et non fonctionnels qu'elle doit satisfaire et en présentant, à l'aide de la méthodologie UML (Unified Modeling Language), les diagrammes des cas d'utilisation.

### 2.1. Spécifications non formelles des besoins

Dans cette partie et dans le but d'identifier les services que doit fournir notre application, nous allons détailler les exigences spécifiques qui consistent en des besoins fonctionnels et des besoins non fonctionnels.

#### 2.1.1. Spécification des besoins fonctionnels

L'application doit pouvoir gérer trois types d'utilisateurs ; administrateur, le recruteur et le chercheur.

Par rapport à l'administrateur, notre application doit répondre aux exigences suivantes :

- L'administrateur est un utilisateur statique avec un email et un mot de passe pour s'authentifier.
- Il doit exister un espace admin sur l'application permettant l'administrateur de contrôler les différentes postes avec la possibilité de visualiser, ajouter ou supprimer un tel poste.

Par rapport à le recruteur, notre application doit correspondre aux exigences suivantes :

- Le recruteur doit avoir un email et mot de passe pour s'authentifier.
- Il doit exister un espace recruteur sur l'application permettant le recruteur de publier les différentes offres d'emplois ou formations avec la possibilité de modifier, ajouter ou supprimer une offre ou une formation.
- Le recruteur peut identifier facilement les candidatures qui font une demande.
- Le recruteur doit avoir l'accès de visualiser les cv.

Par rapport aux chercheurs, notre application doit répondre aux exigences suivantes :

- Les chercheurs doivent être capable de créer un compte sur l'application et avoir ainsi un email et un mot de passe à saisir pour se s'authentifier.



- Le chercheur peut postuler un Cv sur l'application avec des champs obligatoires.

### 2.1.2. Spécification des besoins non fonctionnels

Afin d'assurer un bon fonctionnement de l'application et de garantir la satisfaction de l'utilisateur, des contraintes doivent être prises en compte tout au long du développement du projet :

- **Ergonomie et convivialité** : L'application doit fournir une interface simple et élégante pour l'utilisateur afin de faciliter l'exploitation des services de l'application.
- **Portabilité, maintenance et évolution** : L'application doit être multiplateforme c'est à-dire fonctionnelle sur n'importe quel système d'exploitation. Le code doit être bien lisible, compréhensible et modulaire pour garantir la souplesse, l'évolution et la maintenance de la solution et ainsi répondre aux changements du marché.
- **Rapidité et robustesse** : L'application doit assurer la rapidité de réponse et la fiabilité du résultat procuré : les chercheurs doivent être capable de postuler.

Facilement aux offres et le recruteur doit être capable de sélectionner facilement

Efficacement et d'une manière fiable les chercheurs les plus pertinentes.

## 2.2. Spécification semi-formelle des besoins

Afin d'obtenir de manière plus formelle une vue globale sur les besoins de l'application et aussi réussir une bonne spécification des besoins, ces derniers doivent être modélisés. Cette partie consiste à modéliser les besoins en ayant recours aux concepts d'UML.

## 2.3. Les acteurs

Dans le cadre d'entreprendre la présentation des diagrammes, il faut identifier les acteurs qui sont en interaction avec l'application.

Notre application nécessite l'intervention de trois acteurs principaux : L'administrateur qui fait le contrôle des postes, Le recruteur qui fait la gestion des offres d'emplois et formations et la recherche des employées, le chercheur qui est à la recherche d'un emploi.

## 2.4. Présentation de l'UML

### 2.4.1. Définition

UML, c'est l'acronyme anglais pour « Unified Modeling Language ». On le traduit par « Langage de modélisation unifié ». La notation UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter.

UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc...

### 2.4.2. Caractéristique

UML est un moyen d'exprimer des modèles objet en faisant abstraction de leur implémentation, c'est-à-dire que le modèle fourni par UML est valable pour n'importe quel langage de programmation. UML est un langage qui s'appuie sur un métamodèle, un modèle de plus haut niveau qui définit les éléments d'UML (les concepts utilisables) et leur sémantique (leur signification et leur mode d'utilisation).

Le métamodèle permet de se placer à un niveau d'abstraction supérieur car il est étudié pour être plus générique que le modèle qu'il permet de construire.

Le méta-modèle d'UML en fait un langage formel possédant les caractéristiques suivantes :

- Un langage sans ambiguïtés.
- Un langage universel pouvant servir de support pour tout langage orienté objet.
- Un moyen de définir la structure d'un programme.
- Une représentation visuelle permettant la communication entre les acteurs d'un même projet.
- Une notation graphique simple, compréhensible même par des non informaticiens.

Le métamodèle permet de donner des bases solides et rigoureuses à ce langage graphique, dont les représentations graphiques ne sont là que pour véhiculer des concepts de réalisation.

## 2.5. Diagramme des cas d'utilisations

### 2.5.1. Définition

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel.

### 2.5.2. Diagrammes des cas d'utilisation associés à l'application

- **Diagramme du cas d'utilisation général de l'application**

Le cas d'utilisation, qui présente l'ensemble des fonctionnalités offerte par l'application pour nos utilisateurs (Recruteur et Chercheur d'emploi), est donné par la figure 2.1.

Ce Diagramme met en évidence les fonctionnalités générales attendues de notre application.

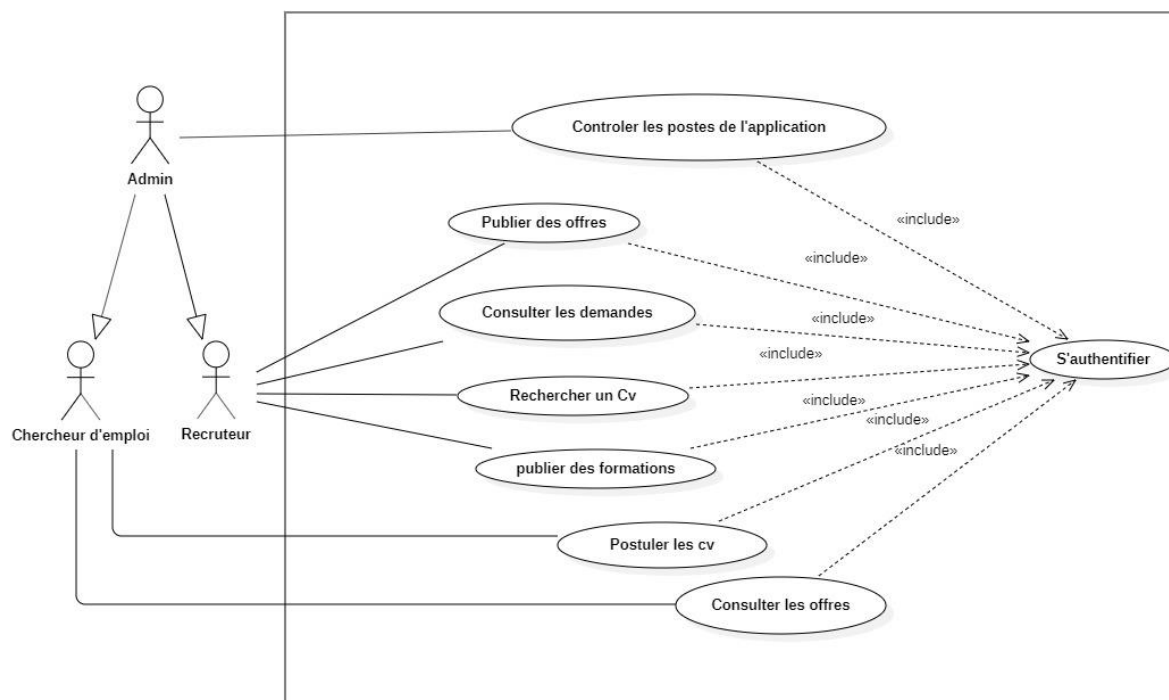


Figure 2.1 Diagramme du cas d'utilisation général de l'application

- **Diagramme du cas d'utilisation :S'authentifier**

La figure 2.2 montre le diagramme relatif au cas d'utilisation : s'authentifier. Selon son rôle, lors de l'authentification, chaque utilisateur pourra accéder à certains volets de l'application. Chacun des utilisateurs a un email et un mot de passe qu'il doit retenir et saisir correctement à chaque fois qu'il désire se connecter. En cas d'erreur de saisie de l'email et/ou mot de passe, un message d'erreur sera affiché

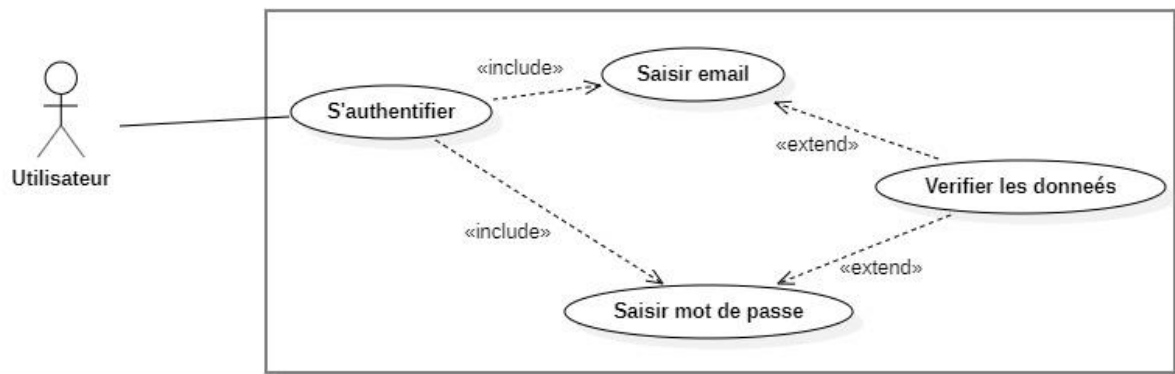


Figure 2.2 Diagramme du cas d'utilisation : S'authentifier

- **Diagramme du cas d'utilisation : postuler pour un emploi raffiné**

La figure 2.3 montre le diagramme relatif au cas d'utilisation : Pour la postulation, Le chercheur doit commencer par la création d'un compte et Une fois connecté, il peut consulter des offres d'emploi ajoutées par le recruteur et les formations qu'il désire y postuler. Pour pouvoir postuler, un formulaire doit être rempli

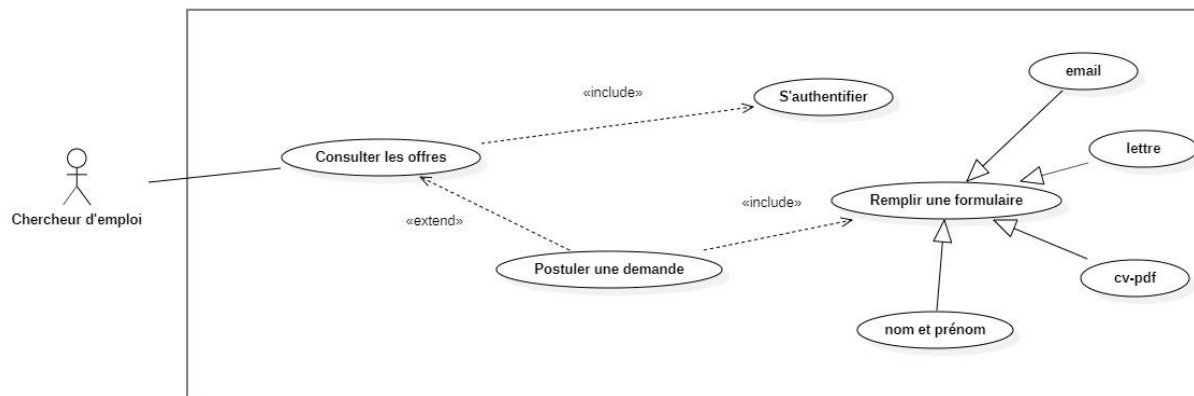


Figure 2.3 Diagramme du cas d'utilisation : Postuler pour un emploi raffiné

### • Diagramme du cas d'utilisation : Raffinement de publications formations et emplois

La figure 2.4 montre le diagramme relatif au cas d'utilisation : gestions des formations. Le recruteur, après son authentification, il sera capable de publier des offres d'emplois et formations, de les modifier ou bien les supprimer.

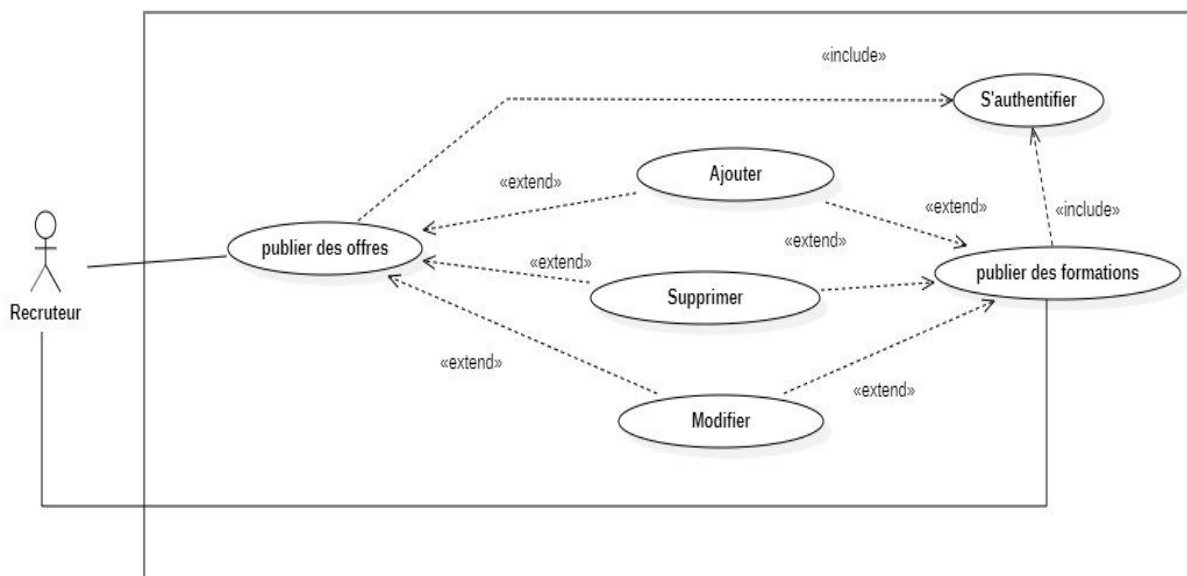


Figure 2.4 Diagramme du cas d'utilisation : Raffinement de publications formations et emploi

## • Diagramme du cas d'utilisation : Raffinement de control des postes

La figure 2.5 montre le diagramme relatif au cas d'utilisation : control des postes raffinés. L'administrateur, après son authentification, il sera capable de contrôler des offres d'emplois, des formations, les demandes, et de les ajoutés ou bien les supprimer.

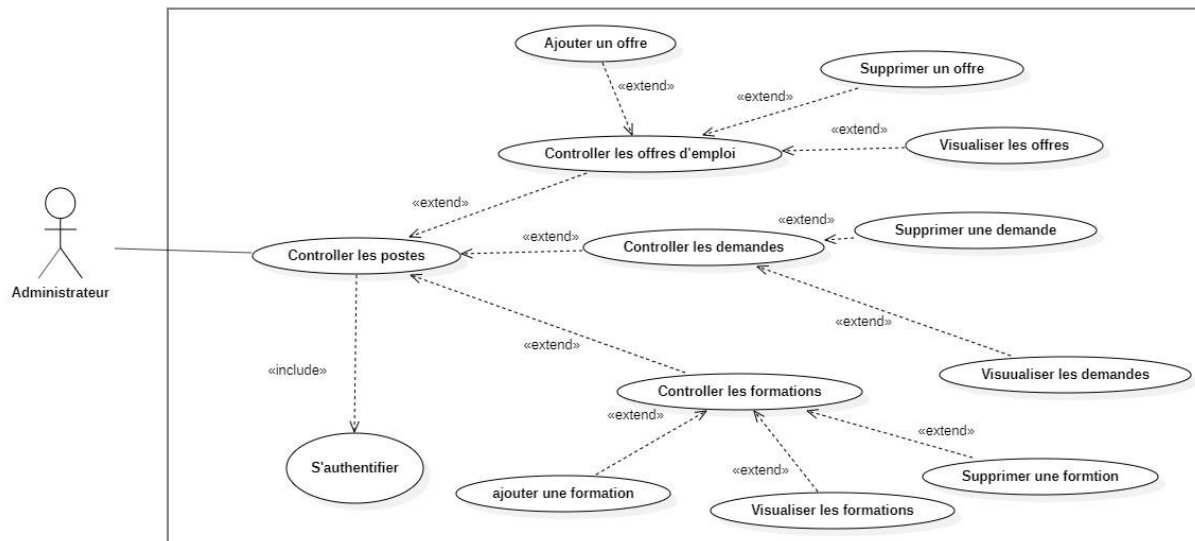


Figure 2.5 : Diagramme du cas d'utilisation : Control des postes raffinés

## 2.6. Conclusion

Dans ce chapitre, nous avons spécifié les besoins de notre application pour avoir une vision plus claire sur le sujet et une compréhension plus profonde des tâches à réaliser. Cette étude nous a permis de bien comprendre les fonctionnalités de notre système afin de faciliter la mise en œuvre de l'application. Nous sommes maintenant prêts à passer à la description détaillée de l'aspect architectural de l'application dans le chapitre suivant

# **Chapitre 3**

## **Conception**

## Introduction

Après avoir fixé les besoins et les objectifs dans le chapitre précédent, nous nous focalisons sur l'aspect architectural de l'application. Cette phase a pour but de concevoir les schémas généraux qui permettent la modélisation et la description d'une manière non ambiguë du fonctionnement désiré de l'application. Dans ce chapitre deux vues conceptuelles seront décrites. La première donne une vue globale de l'architecture tandis que la deuxième offre une vue détaillée en se basant sur le langage UML (Unified Modeling Language).

### 3.2. Architecture globale

Afin de réussir l'étape de conception, il est nécessaire de suivre un contexte conceptuel qui répond aux spécificités et aux besoins fondamentaux de l'application, menant ainsi à la satisfaction de ses utilisateurs. L'architecture client/serveur est l'architecture qui paraît le mieux adapté à notre application. Nous allons entamer cette partie par la définition de cette architecture.

#### 3.2.1. Présentation de l'architecture client/serveur :

Cette architecture est basée sur l'utilisation de deux types de logiciels, à savoir un logiciel serveur et un logiciel client s'exécutant normalement sur deux machines différentes. On appelle logiciel serveur un programme qui offre un service sur le réseau. Le serveur accepte des requêtes, les traite et renvoie le résultat au demandeur. On appelle logiciel client un programme qui utilise le service offert par un serveur. Le client envoie une requête et reçoit la réponse. La seule obligation de la coopération entre le client et le serveur est le respect du protocole qui assure la communication inter-processus : HTTP pour le Web, POP/IMAP/SMTP pour le courrier électronique, SNMP pour l'administration de réseau.

Cette figure illustre l'architecture client/serveur adapté pour cette application.



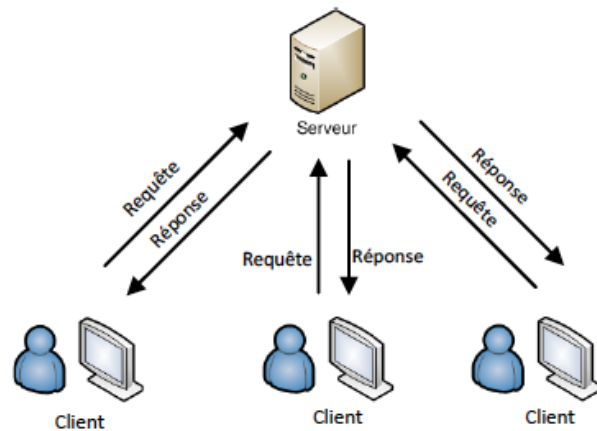


Figure 3.1 : Architecture client/serveur

### 3.2.2. Les avantages de l'architecture client/serveur :

Pour le développement de notre application nous avons opté pour l'architecture client/serveur pour plusieurs raisons à savoir :

- Modèle adéquat pour la distribution des données.
- Prend avantage des fonctionnalités réseaux.
- Facile d'ajouter des serveurs supplémentaires ou de mettre à jour les serveurs existants.
- Intégrité : les données du serveur sont gérées sur le serveur de façon centralisée. Les clients restent individuels et indépendants.
- Partage des ressources : un serveur traite plusieurs clients en même temps et contrôle leurs accès aux ressources.

### 3.3. Conception détaillée

Cette partie illustre les différents diagrammes relatifs à l'application, en utilisant la méthodologie UML.

### 3.3.1. Diagramme de classes

Le diagramme de classes exprime la structure statique du système en termes de classes et de relations entre ces classes. Il fait abstraction des aspects dynamiques et temporels. La figure 3.2 montre le diagramme de classes qui collaborent au fonctionnement de notre application.

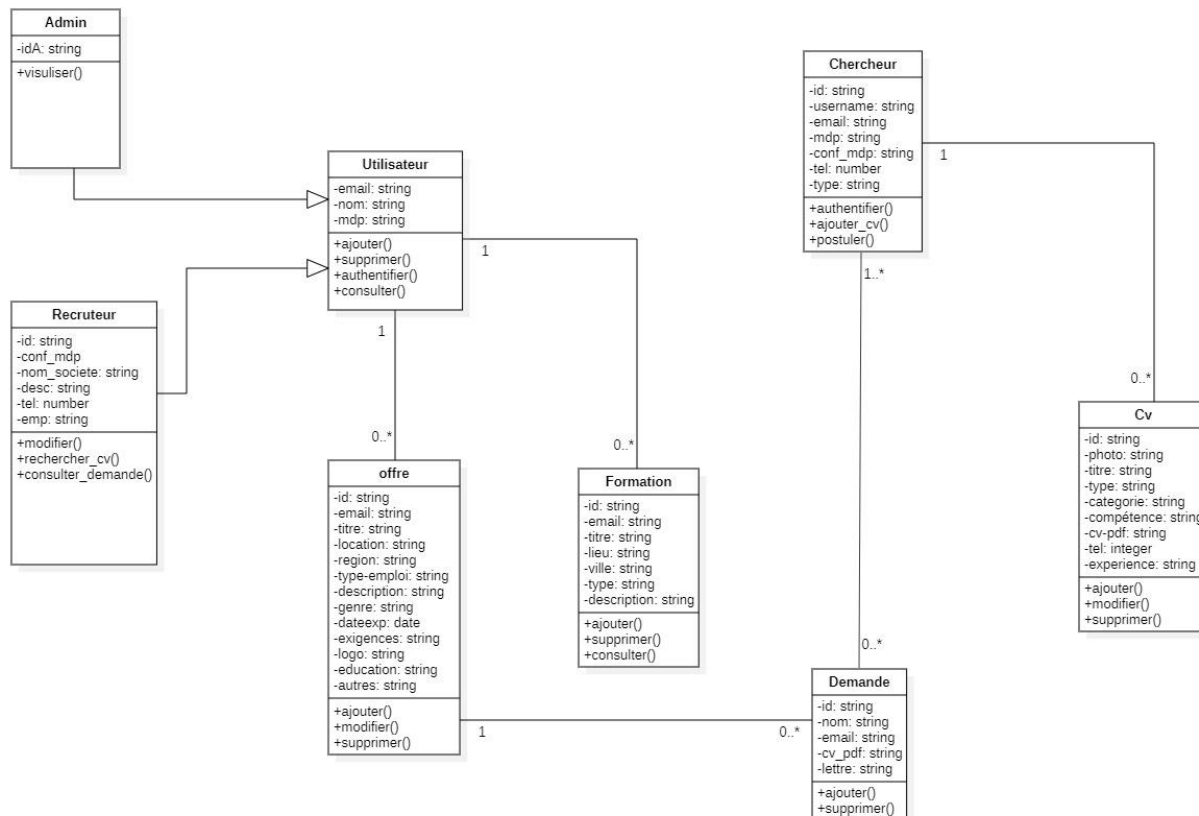


Figure 3.2 : Diagramme de classes

Ces classes sont décrites comme suit :

- **La classe Admin :**

Cette classe a comme attributs la clé primaire qui identifie l'administrateur (idA : Int), le login (email : String) et le mot de passe (mdp : String). L'administrateur doit s'authentifier pour pouvoir contrôler les différents postes

- **La classe Recruteur :**

Cette classe a comme attributs la clé primaire qui identifie le recruteur est (id : Int), le login (email : String), le nom (nom : String), numéro de téléphone (tel : Int) et le mot de passe (mdp : String), un nom du société (nom\_société : titre), un logo pour la société (logo), et une description (desc : String). Le recruteur doit s'authentifier pour pouvoir consulter la liste des demandes d'emploi et la liste des CV.

- **La classe Chercheur :**

Cette classe a comme attributs la clé primaire qui identifie le chercheur est (id : Int), l'email (email : String), le nom d'utilisateur (username : Int), le nom (nom : String), numéro de téléphone (tel : Int), l'adresse (location : String) et le mot de passe (mdp : String). Le chercheur doit s'authentifier pour publier des cv ou postuler pour une offre.

- **La classe Offre :**

Cette classe a comme attributs la clé primaire qui identifie l'offre est (id : Int), l'email (email : String), le titre (titre : String), la localisation (location : String), le région (région : String), le type de contrat (type-emploi : String), la description de l'offre (description : String), le genre des candidats (genre : String), la date d'expiration de candidature (dateexp : date), les exigences de l'offre (exigences : String), le logo (logo : String), l'éducation requis (education : String), les autres bénéfices de l'offre (autres : String).

- **La classe Formation :**

Cette classe a comme attributs la clé primaire qui identifie la formation est (id : Int), l'email (email : String), le titre (titre : String), le lieu (lieu : String), la ville (ville : String), le type de formation (type : String), et la description (description : String).

- **La classe Demande :**

Cette classe a comme attributs la clé primaire qui identifie la demande est (id : Int), le nom (nom : String), l'email (email : String), le cv en PDF (cv-pdf : String) et une lettre de motivation (lettre : String).

- **La classe Cv :**

Cette classe a comme attributs la clé primaire qui identifie le Cv est (id : Int), une image (photo : String), le titre (titre : String), le type (type : String), la catégorie (categorie : String), les compétences (competance : String), le cv en PDF (cv-pdf : String), un numéro de téléphone (tel : Integer), et l'expérience (experience : String).

### 3.3.2. Diagramme de séquences

#### 3.3.2.1. Définition :

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation. Les composants de diagramme de séquence sont :

- ✚ Acteur : Un participant qui est externe au système.
- ✚ Les messages : Plusieurs types de messages (actions) peuvent transiter entre les acteurs et les objets.
- ✚ Ligne de vie : Une ligne verticale qui représente la séquence des événements qui se produisent dans un participant pendant une interaction, alors que le temps progresse en bas de ligne.
- ✚ Les opérateurs d'interaction : Un opérateur d'interaction définit le type d'un fragment composé.

#### 3.3.2.2. Diagrammes des séquences associés à l'application :

Dans cette partie, nous présentons les diagrammes de séquences associés à notre application. Le but étant de décrire comment se déroulent les actions entre les acteurs ou les objets. Dans ce qui suit nous présentons les diagrammes les plus importants de notre application.

### • Diagramme de séquences relatif à l'Administrateur :

La figure 3.3 ci-dessous décrit les principales fonctionnalités offertes par l'application à l'administrateur qui doit s'identifier à travers l'interface d'authentification, la vérification se fait au niveau de la base de données. Une fois redirigé vers son espace, il aura la possibilité de consulter, ajouter et supprimer les différents postes.

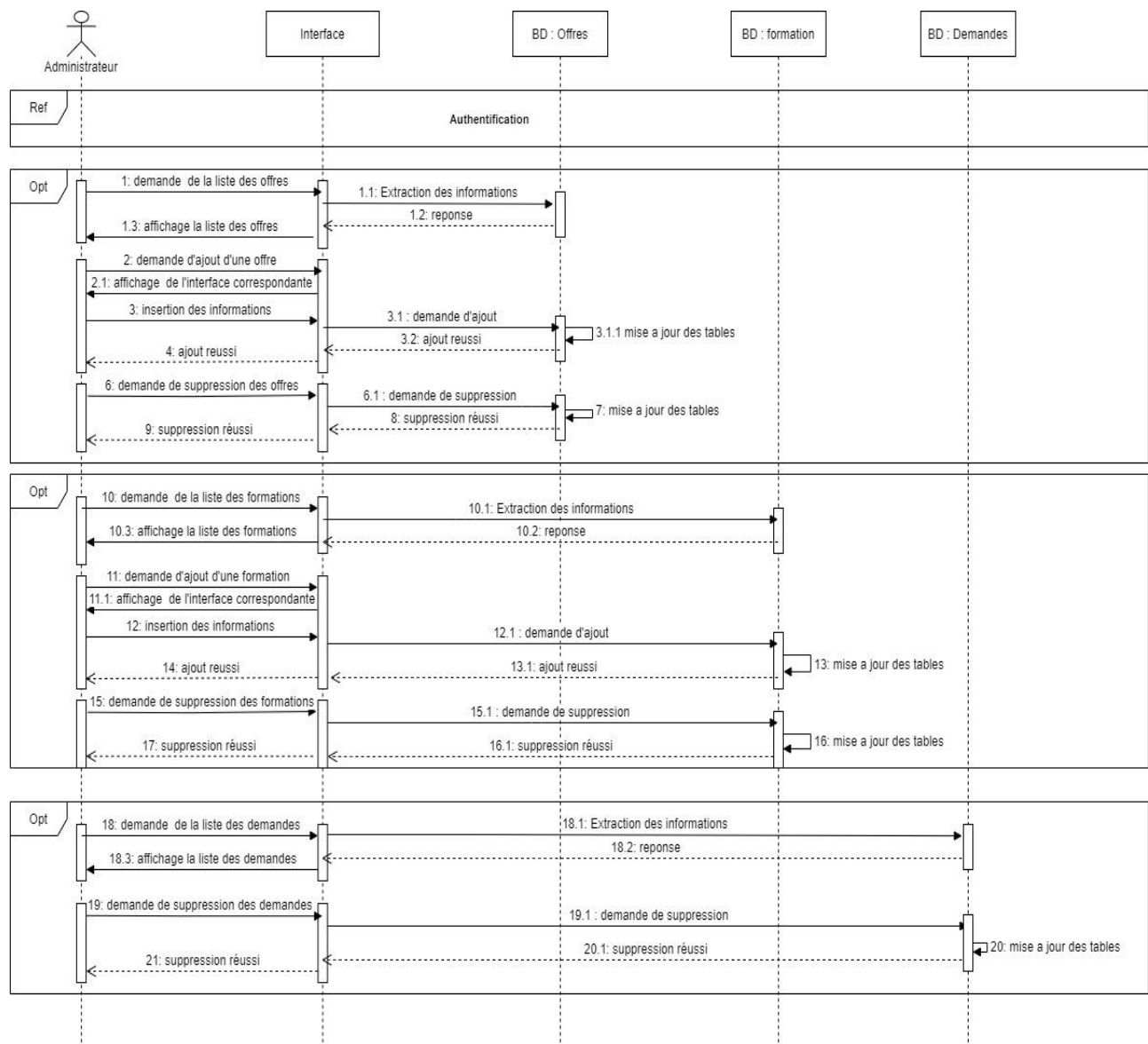


Figure 3.3 : Diagramme de séquences relatif à l'administrateur

- **Diagramme de séquences relatif au recruteur :**

La figure 3.4 de la page 22 décrit le scénario effectué par le recruteur, Après la saisie des coordonnées pour se connecter, et si les données sont bien correctes, le recruteur sera directement redirigé vers son espace.

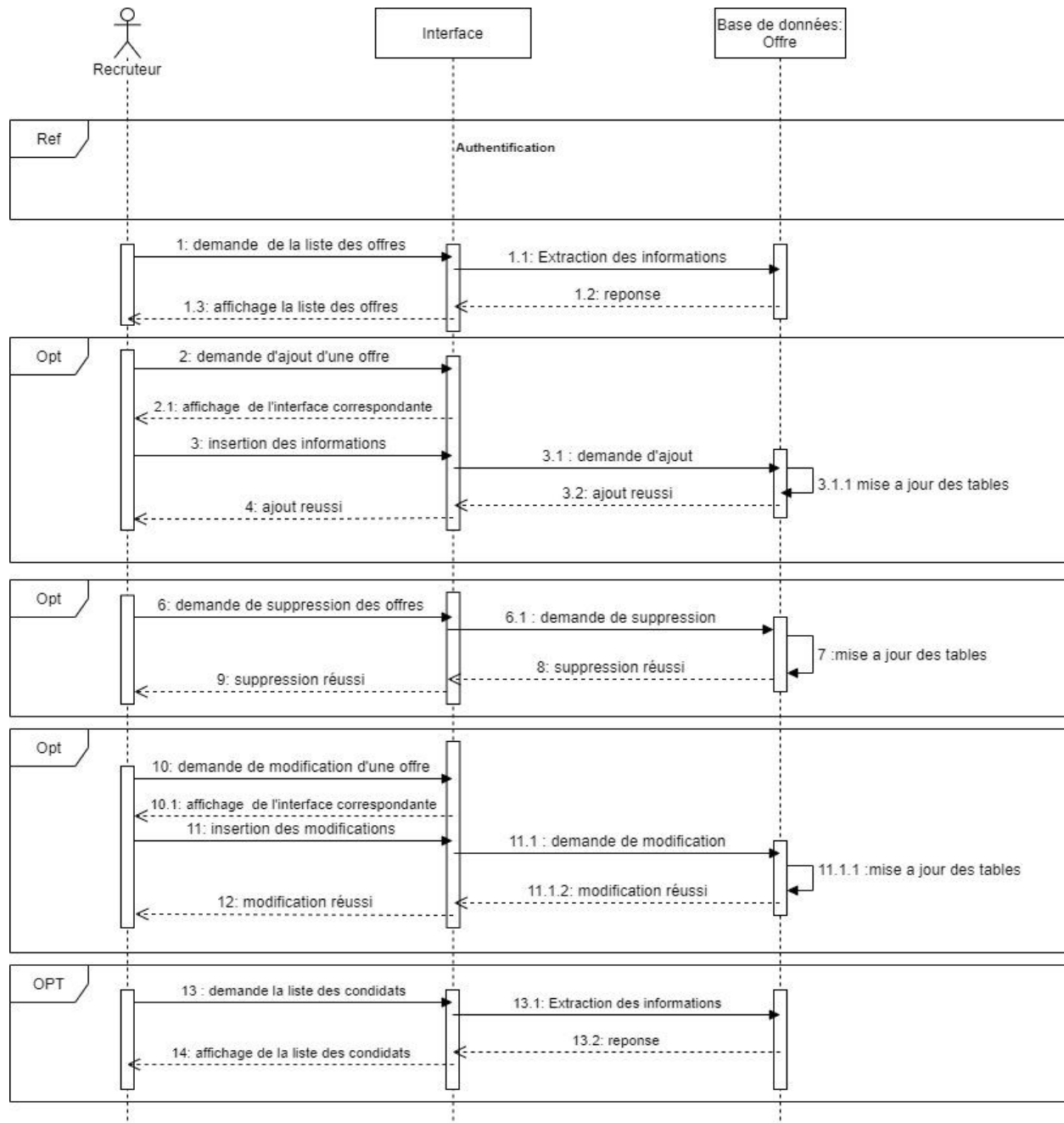


Figure 3.4 Diagramme de séquences relatif au recruteur

Il aura ainsi la possibilité de :

- ✓ Gérer les candidatures : il demande la consultation de la liste des candidatures disponibles, après une étude détaillée des CV il peut ainsi prendre sa décision pour chaque candidature en les contactant.
- ✓ Gérer les offres d'emploi : il aura ainsi la possibilité de voir les informations relatives à chaque offre d'emploi. En effet, il peut ajouter, supprimer ou modifier des offres selon ses besoins. Dans les trois cas, les données sont bien enregistrées et la base de données est mise à jour. De même pour la gestion des formations.

- **Diagramme de séquences relatif au chercheur**

La figure 3.5 de la page 24 décrit les étapes qui permettant le chercheur de postuler à une offre d'emploi, de même pour lui, il doit passer par une étape d'authentification et de vérification des données. S'il désire consulter la liste des offres d'emploi disponibles, grâce au filtre de recherche il aura la possibilité de consulter seulement les offres relatives à sa spécialité. Après le choix d'une offre pour y postuler, un formulaire avec des champs clés obligatoires sera affiché directement sur l'interface, le chercheur sera invité à joindre Un Cv. Une fois que la postulation est validée, un message de confirmation apparaitre.

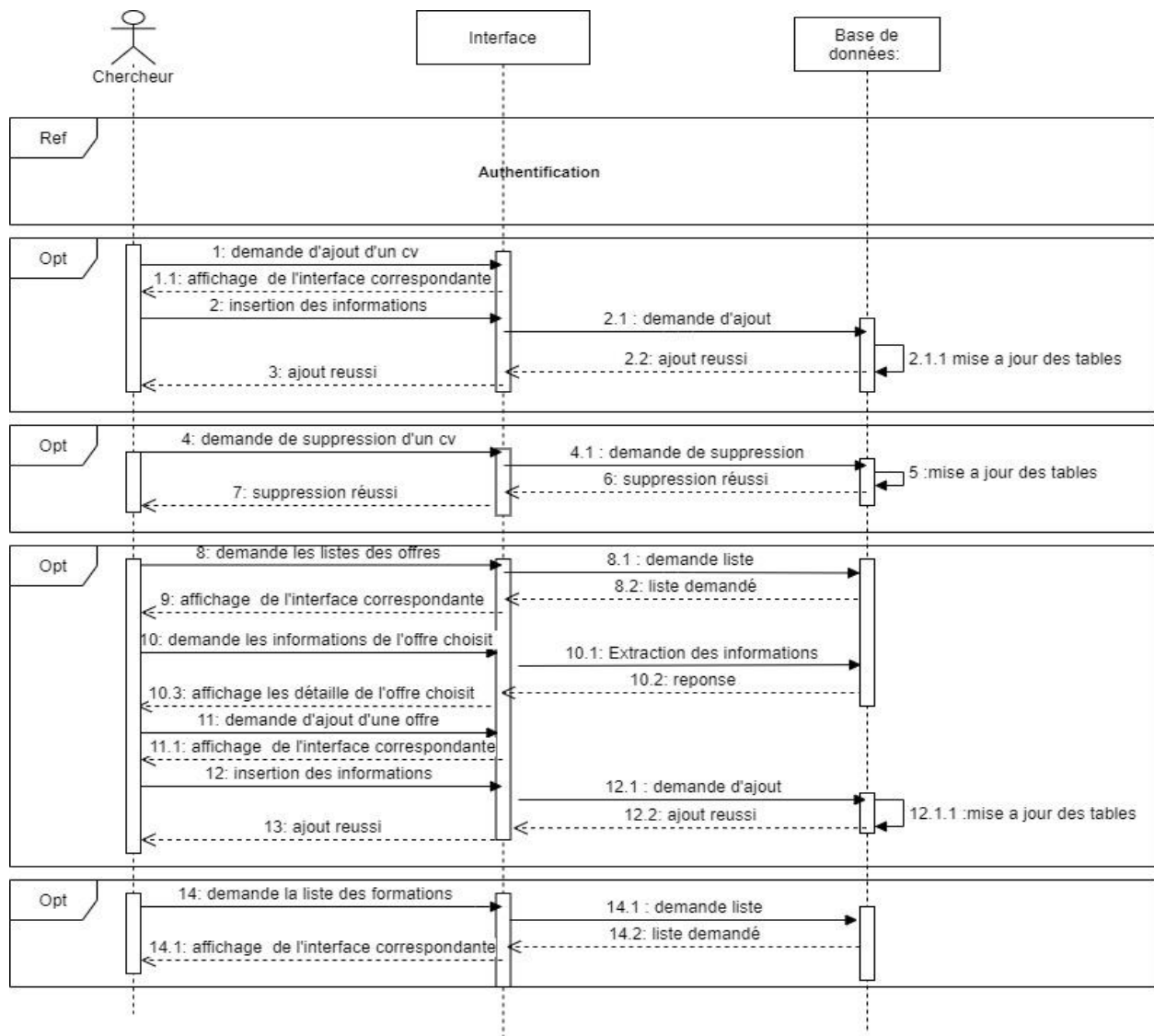


Figure 3.5 : Diagramme de séquences relatif au chercheur

- Diagramme de séquences de l'authentification**

Le diagramme de séquence présenté par la figure 3.6 de la page 25 est relatif au scénario de l'authentification qui est détaillé comme suit : l'utilisateur de l'application peut être : le recruteur, l'administrateur ou le chercheur. En effet, l'authentification a pour but de profiter



des différentes fonctionnalités offertes par l'application. Certainement, l'utilisateur doit avoir un compte déjà créé. Pour l'enchaînement il se déroule comme suit : Premièrement, l'utilisateur demande l'accès au système en saisissant ses informations : son identifiant et son mot de passe. Par rapport au système il vérifie l'existence de l'utilisateur dans la base de données. S'il existe, le système affiche l'interface qui lui est relative. Une fois connecté, l'utilisateur doit avoir l'accès aux fonctionnalités. Pour l'enchaînement alternatif : Dans le cas où les informations saisies sont incorrectes, le système affiche à l'utilisateur un message d'erreur.

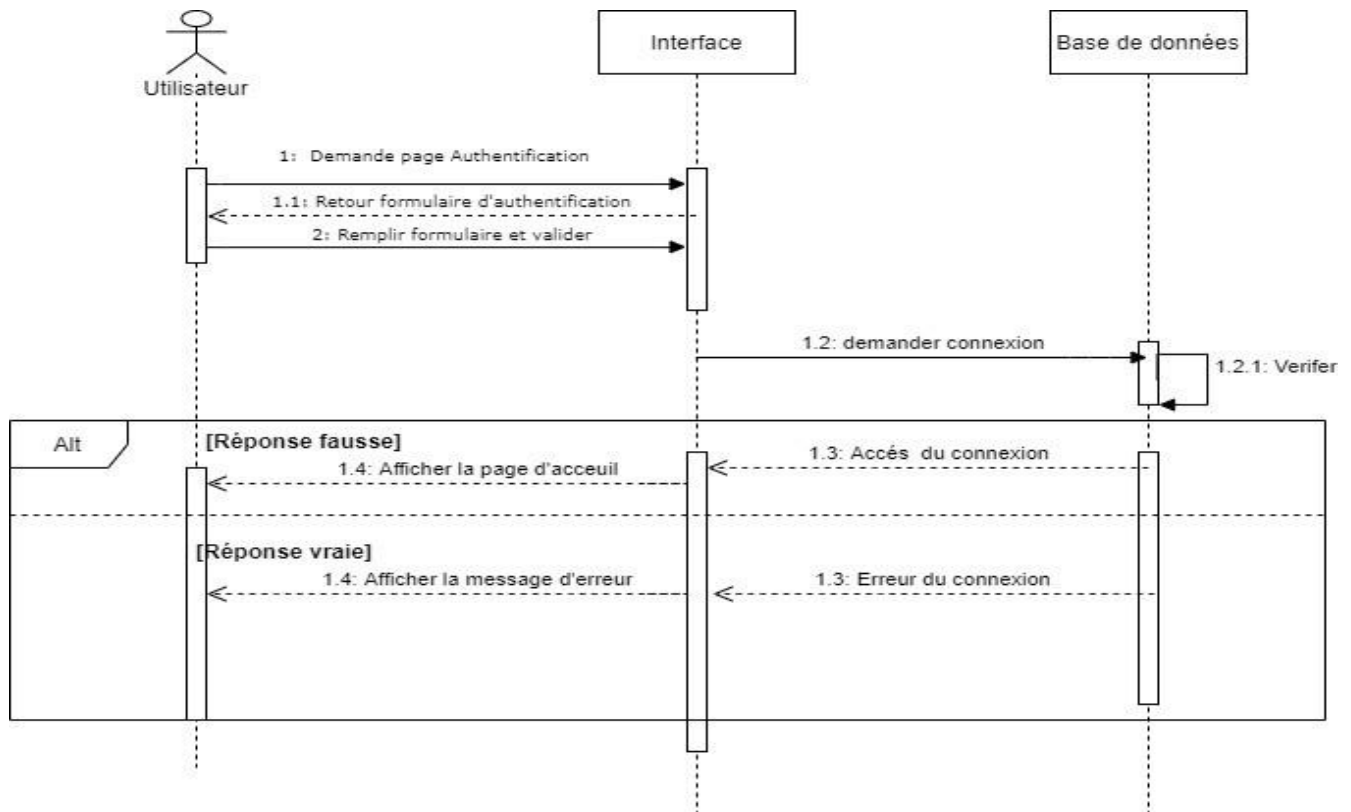


Figure 3.6 : Diagramme de séquences de l'authentification

### 3.4. Conclusion

A travers ce chapitre, nous avons présenté notre conception proposée pour l'application. Nous avons fourni, dans un premier lieu, une conception globale. Par la suite, nous avons détaillé la conception à travers les diagrammes UML qui nous ont permis de dégager la vue statique ainsi que la vue dynamique de l'application. Afin d'illustrer la concrétisation de cette conception, nous présenterons dans le chapitre suivant une description de l'état de la réalisation de notre projet.

# **Chapitre 4**

## **Réalisation**

## Introduction

Cette partie contient le dernier volet de ce rapport. Elle a pour objectif d'exposer le travail achevé. Dans un premier temps, nous présentons l'environnement matériel, logiciel et les différents outils de développement utilisés. Dans un second temps, nous illustrons la réalisation de notre travail par des imprimes écran des interfaces les plus importantes de notre application.

### 4.1. Environnement de travail

Nous présentons dans cette section l'environnement matériel mis à la disposition du présent projet, ainsi que l'environnement logiciel utilisé pour le développement et la mise en place de notre application

#### 4.1.1. Environnement matériel

Pour la réalisation de ce projet, nous avons disposé d'un ordinateur HP caractérisé par :

- Processeur : Intel® Intel(R) Core (TM) I5 -5200U CPU@ 2.20 GHz
- Mémoire : 8 Go de RAM
- Disque dur : 1 To
- Carte graphique : Intel® HD Graphics 5500
- Système d'exploitation : Windows 10

#### 4.1.2. Environnement logiciel

Dans ce qui suit, nous présentons l'environnement logiciel utilisé pour mener à terme ce sujet.

##### A. Framework utilisé

– **ReactJS** (aussi appelé React.js ou ReactJS) : est une bibliothèque JavaScript libre développée par Facebook depuis 2003. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page (ou portion) HTML à chaque changement d'état.

- **Node JS** : est une plateforme logicielle libre en JavaScript orientée vers les applications réseau événementielles hautement concurrentes qui doivent pouvoir monter en charge.
- **Mongo DB** : est un système de gestion de base de données orienté documents, répartissable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données. Il est écrit en C++. Il fait partie de la mouvance NoSQL.
- **Express JS** : est un Framework pour construire des applications web basées sur Node.js. C'est de fait le Framework standard pour le développement de serveur en Node.js.
- **Bootstrap** : est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option.

#### **B. Langages de programmation :**

- **JavaScript** : est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs avec l'utilisation (par exemple) de Node.js. Le langage JavaScript permet des pages web interactives, et à ce titre est une partie essentielle des applications web. Une grande majorité des sites web l'utilisent, et la majorité des navigateurs web disposent d'un moteur JavaScript dédié pour l'interpréter, indépendamment des considérations de sécurité qui peuvent se poser le cas échéant.

#### **C. Logiciels de développement :**

- **Visual Studio Code** : Visual Studio Code est un éditeur de code multiplateforme édité par Microsoft. Cet outil destiné aux développeurs supporte plusieurs dizaines de langages de programmation comme le HTML, C++, PHP, Javascript, Markdown, CSS, etc... Visual Studio Code intègre plusieurs outils facilitant la saisie de code par les développeurs comme la coloration syntaxique ou encore le système d'auto-complétions IntelliSense. En outre, l'outil permet aux développeurs de corriger leur code et de gérer les différentes versions de leurs fichiers de travail puisqu'un module de débogage est aussi de la partie.

- **HTML (HyperText Markup Language) :** C'est le format de données conçu pour représenter les pages web, c'est un langage de balisage qui permet d'écrire de l'hypertexte, d'où son nom. HTML permet également d'architecturer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie, et des éléments programmables tels que des applets.
- **CSS (Cascading Style Sheets) :** Ce sont les feuilles de styles qui désignent l'ensemble d'attributs de caractères et de formats de paragraphes pouvant être appliqués en une seule opération à un ou plusieurs paragraphes, à un livre etc. De façon extensive, c'est l'ensemble de contraintes typographiques, de maquette, de couleur, de reliure etc.... qui définissent l'unité d'une collection littéraire, d'une revue périodique etc...

#### **D. Langage de Modélisation**

- **UML :** "Unified Modeling Language" est un langage de modélisation graphique, il est aujourd'hui utilisé comme standard international, incontournable, stabilisé et industriel.

#### **E. Logiciel de Modélisation**

- **Star UML :** est un logiciel de modélisation UML, qui a été "cédé comme open source" par son éditeur, à la fin de son exploitation commerciale (qui visiblement continue ...), sous une licence modifiée de GNU GPL.
- **Draw.io :** est une application gratuite en ligne, accessible via son navigateur (protocole https) qui permet de dessiner des diagrammes ou des organigrammes. Cet outil vous propose de concevoir toutes sortes de diagrammes, de dessins vectoriels, de les enregistrer au format XML puis de les exporter.

### **4.2. Travail réalisé**

Dans cette partie, nous présentons notre travail en exposant des captures d'écran des interfaces les plus importantes de notre application.

### 4.2.1. Page d'accueil

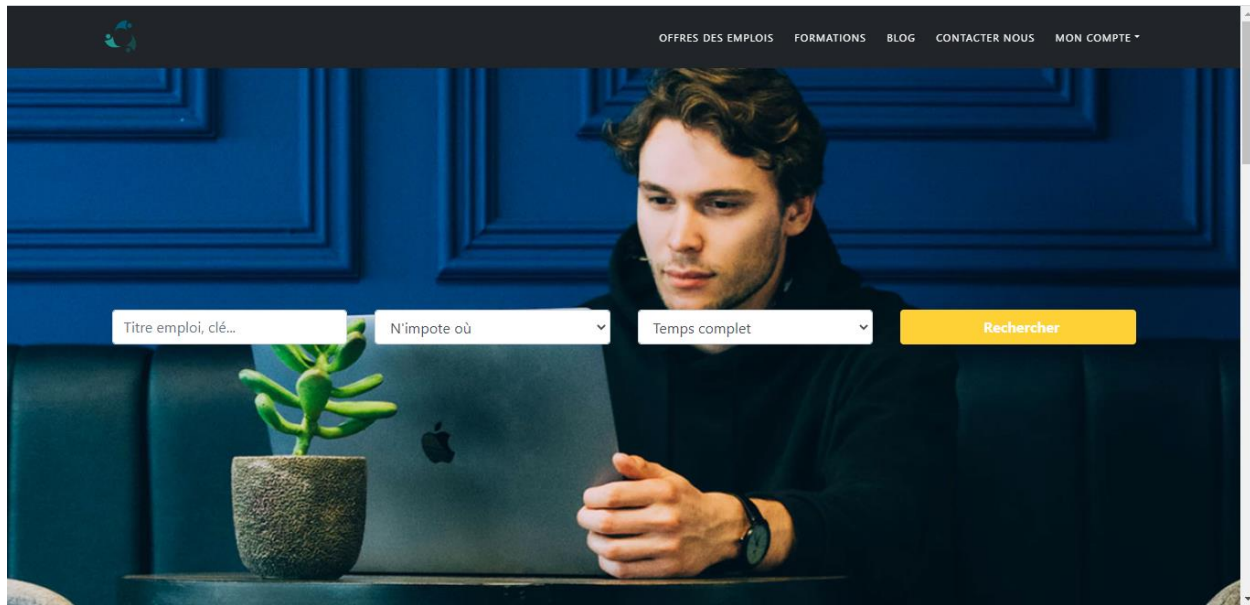


Figure 4.1 : Interface d'accueil

L'interface d'accueil de l'application contient le menu principal du site Web. Des liens hypertextes donnent la possibilité à l'utilisateur d'accéder directement aux rubriques qui l'intéressent. Ces pages constituent également des menus qui peuvent être subdivisés en sous-menus.

Après la Si l'utilisateur clique sur le lien « offres d'emploi » de l'interface d'accueil, il est dirigé vers une interface dont les listes des emplois. Aussi si l'utilisateur clique sur le lien « connexion », il est invité à se connecter s'il a un compte sinon il doit s'inscrire.

### 4.2.2. Interface d'authentification

The screenshot displays a web application's authentication interface. At the top, a dark navigation bar contains a logo on the left and a list of links: OFFRES DES EMPLOIS, FORMATIONS, BLOG, CONTACTER NOUS, and CONNEXION. The main content area is divided into two panels. The left panel, titled 'Connexion :', contains a form with two input fields: 'Email :\*' and 'Mot de passe :\*'. Both fields have 'Default input' text. Below the password field is a note: 'Must be 8-20 characters long.' A yellow 'Se Connecter' button is positioned below the form. At the bottom of this panel are two links: 'Mot de passe oublié ?' and 'Ou Rejoignez-nous'. The right panel, titled 'Se connecter via les réseaux sociaux', features two buttons: a yellow 'Sign in with Facebook' button and a white 'Sign in with Google' button with the Google logo.

Figure 4.2 : Interface d'authentification

L'utilisateur doit s'authentifier pour pouvoir accéder à son espace. Dans cette interface, l'utilisateur saisit son email et son mot de passe. S'ils sont corrects il sera dirigé vers la page d'accueil. S'il veut connecter à l'aide du « Facebook » ou « Google » il doit cliquer sur l'un des deux buttons. S'il a oublié son mot de passe, il peut le récupérer en cliquant sur "Mot de passe oublié". Dans ce cas, un lien de changement de mot de passe sera envoyé sur l'adresse mail saisie lors de l'inscription. S'il ne dispose pas d'un compte, il doit s'inscrire en cliquant sur "Rejoignez-nous".

### 4.2.3. Interface Administrateur :

La figure 4.3 présente l'interface de l'administrateur qui contient 3 menus et un lien de page profile



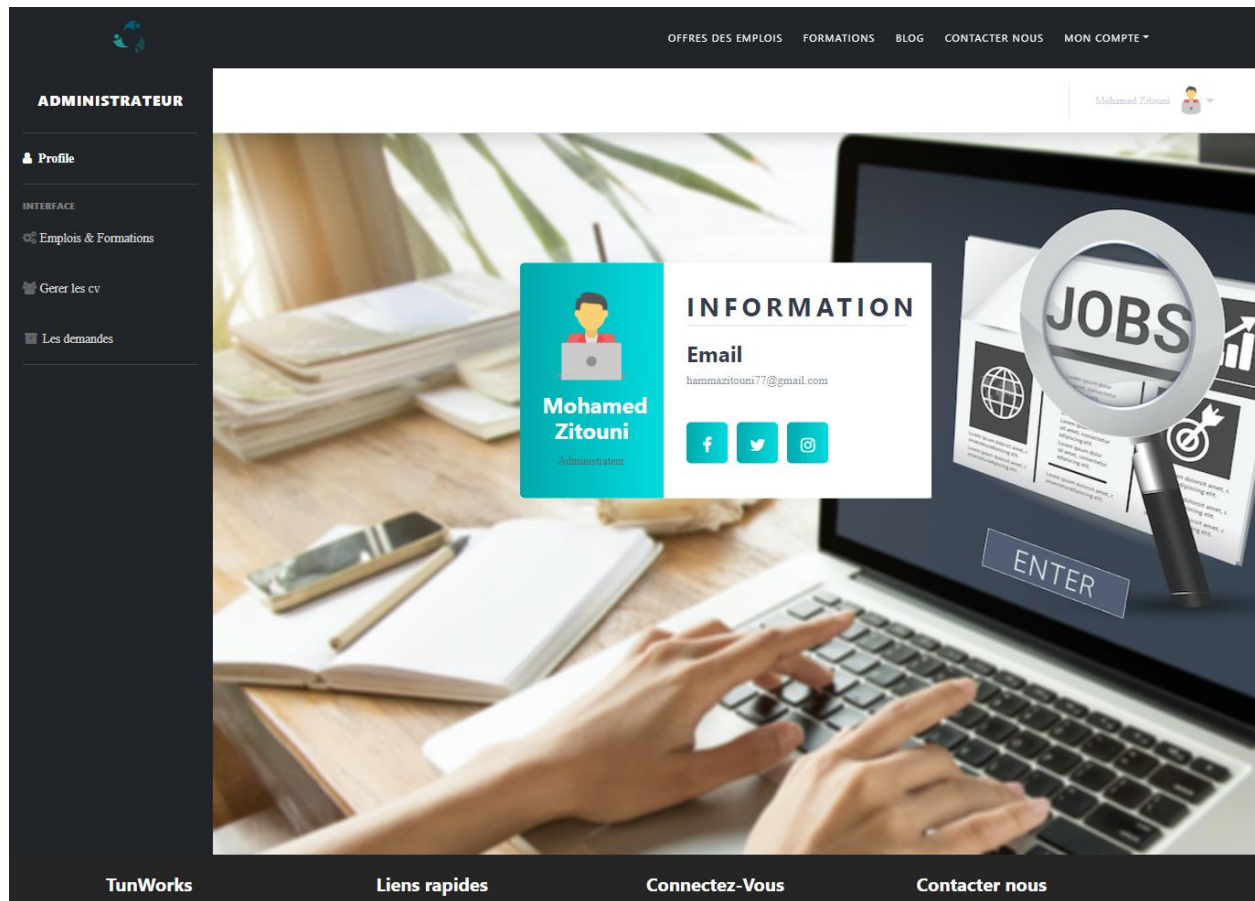


Figure 4.3 : Interface d'espace administrateur

#### 4.2.4. Interface de control des offres d'emplois

L'interface illustre la liste avec les détails de chaque offre d'emploi, l'administrateur trouvera ainsi le titre de l'offre, l'email de recruteur, et localisation de l'offre avec la possibilité d'ajouter ou supprimer une telle offre.

- ✓ De même pour les formations et les demandes

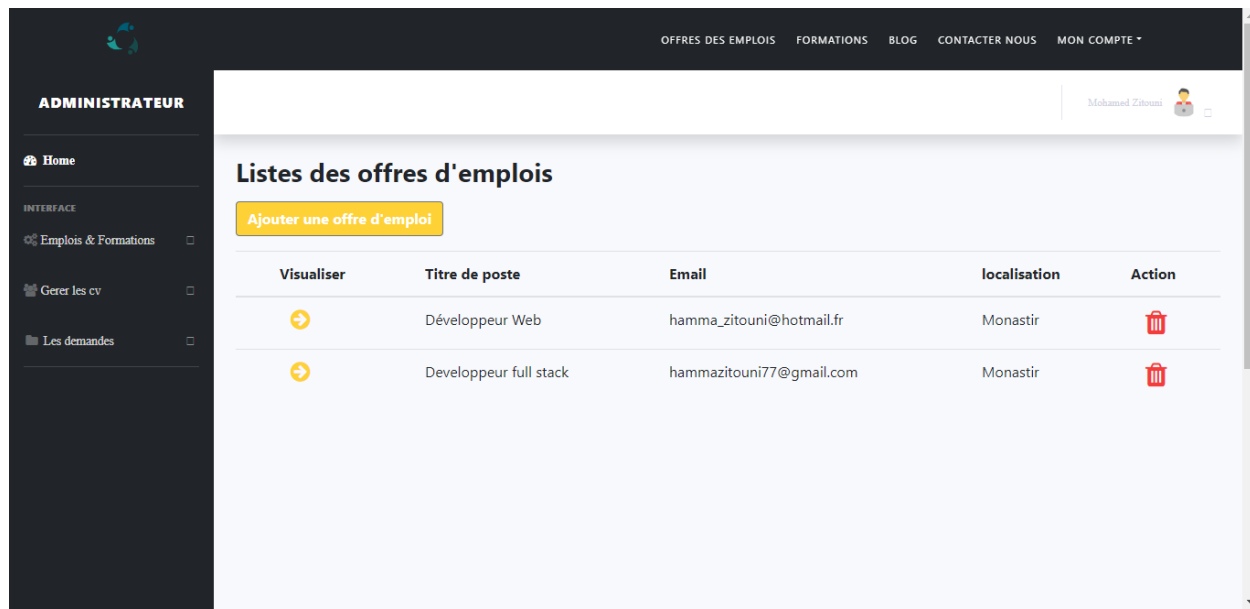


Figure 4.4 : Interface de control des offres d'emplois

#### 4.2.5. Interface des offres d'emploi

L'interface illustre les différentes offres d'emplois, si l'utilisateur choisi une offre pour consulter, il doit cliquer sur le bouton « plus détails » pour voir les différents détails de l'offre

- ✓ De même pour l'interface des formations

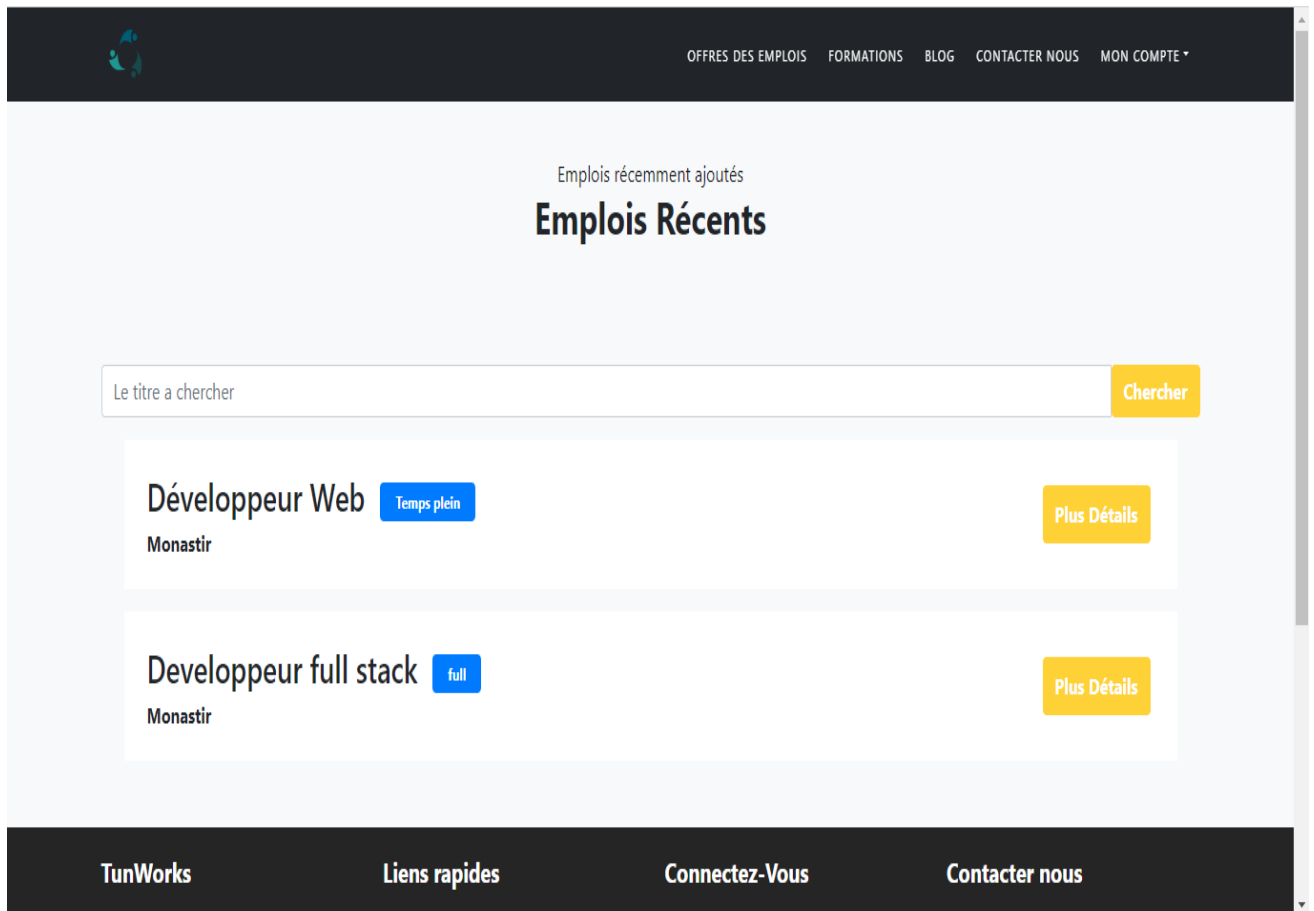


Figure 4.5 : Interface des offres d’emplois

#### 4.2.6. Interface contacter nous

La figure 4.6 ci-dessous montre l'interface se formulaire de contact pour toute information supplémentaire ou réclamation.

**Adresse** Cité basatine C5 Monastir - Tunisie

**Contactez nous** +21631820908

**Aide** :Jobboard12@gmail.com

### Envoyez-nous un message

NOM & PRÉNOM \*

First name Last name

EMAIL \*

Eg. example@email.com

NUMÉRO DE TELEPHONE

Eg. +1 800 000000

MESSAGE \*

Write us a message

**ENVOYER**

**TunWorks**

**Liens Rapides**

Connectez-vous  
Offres Emplois  
Formations

**Connectez-Vous**

Rejoignez-nous pour meilleur offre d'emploi.

**Connectez Vous**

**Contacter Nous**

hammazitouni77@gmail.com

f @

Copyright TunWorks © 2020. All rights reserved.

Figure 4.6 : Interface contacter nous

### 4.3. Conclusion

A ce stade, nous atteignons la fin de l'étude du projet. Dans ce dernier chapitre, nous avons à la fois décrit les environnements matériel et logiciel sur lesquels nous avons construit notre application. Ensuite, nous avons illustré les fonctionnalités importantes de l'application en fournissant quelques interfaces graphiques de l'application. A présent, nous passerons, dans la partie suivante, à la conclusion globale de notre projet.

## Conclusion générale

Dans notre projet nous avons réalisé une application web permettant la gestion des offres d'emplois et formations au sein de laboratoire RLANTIS-FSM. L'objectif est d'éclaircir le marché de l'emploi et offrir plus de visibilité à l'offre d'emploi afin de motiver les candidats à déposer leurs CV.

Durant mon travail sur ce projet j'ai rencontré plusieurs obstacles et des difficultés citons la situation du monde entier, le virus COVID-19 et les dégâts qui l'ont causé comme stress et panique puisque ce virus m'empêche de vivre mon quotidien qui m'a privé d'aller au café culturel où j'aime et je trouve toutes les conditions favorables pour travailler sur mon projet.

Cependant, des améliorations de ce projet peuvent être envisagées. En effet, l'application peut être étendue en y ajoutant une application mobile et d'autres fonctionnalités comme le chatbot pour la communication avec les clients et un espace de feedback

Ce projet nous a permis de s'adapter, de s'améliorer dans le développement des applications web, d'enrichir et pénétrer plus avant dans nos connaissances techniques. En outre, il nous a été bénéfique dans la familiarisation avec la conduite de projet informatique de façon méthodique et organisée.

# Bibliographie

1. URL: <https://openclassrooms.com/fr>
2. Jean-Paul kurtz : Dictionnaire Etymologique des Anglicismes et des Américanismes, Volume 2
3. URL: <https://web.maths.unsw.edu.au>
4. URL : <https://wikipedia.org>
5. URL : Bootstrap. [En ligne] <http://getbootstrap.com/>.
6. React.js. [En ligne] <https://www.c-sharpcorner.com/article/what-and-why-reactjs/>.
7. Express.js. [En ligne] [https://developer.mozilla.org/en-US/docs/Learn/Serverside/Express\\_Nodejs](https://developer.mozilla.org/en-US/docs/Learn/Serverside/Express_Nodejs).
8. HTML5. [En ligne] <http://www.html5-css3.fr/html5/introduction-html5>.
9. CSS3. [En ligne] <http://www.html5-css3.fr/css3/introduction-css3>.
10. Star UML. [En ligne] <https://www.methodsandtools.com/tools/staruml.php>.
11. Draw.io. [En ligne] <https://www.getapp.com/it-management-software/a/draw-dot-io/>.