



المدرسة العليا  
للتكنولوجيا - اسفي  
ÉCOLE SUPÉRIEURE  
DE TECHNOLOGIE -SAFI



Université Cadi Ayyad  
École Supérieure De Technologie-Safi  
Département : Informatique  
Filière :ISIR

## Rapport de mini-Projet XML & UML

---

### Projet de XML & UML

---

Réalisé par :

ZAGHLI Khadija

NAJIM Soufiane

BENHADDOU Ayman

Encadré par :

Mme. ARHID Khadija

M. CHEKRY Abderrahman

ANNÉE UNIVERSITAIRE : 2024/2025

# Remerciement

Nous souhaitons exprimer nos vifs remerciements et notre profonde gratitude à tous ceux qui ont contribué, de près ou de loin, à la réussite de ce projet intitulé Développement d'une application de gestion des employés et des offres d'emploi .

Sans aucun doute, la réalisation de ce projet a été le fruit d'une collaboration exceptionnelle au sein de notre équipe. Ensemble, nous avons su mettre en commun nos compétences, notre créativité et notre engagement, permettant ainsi de mener ce projet à terme avec succès. Chacun de nous a apporté une contribution unique, enrichissant le travail par la diversité de nos idées et de nos approches. Cette synergie a été un atout majeur pour surmonter les défis et exploiter les opportunités qu'offrait ce projet.

Nous tenons à exprimer nos sincères remerciements à Mme.Arhid et Mr.Chekry pour leur encadrement précieux et leur appui essentiel tout au long de ce projet. Leurs orientations éclairées, leur disponibilité et leur expertise ont été des facteurs déterminants dans notre réussite. Leur soutien constant nous a motivés à donner le meilleur de nous-mêmes et à atteindre nos objectifs.

Nous souhaitons également remercier toutes les personnes qui ont contribué à ce projet, que ce soit par leurs conseils, leurs encouragements ou leur aide technique. Chaque échange et chaque interaction ont enrichi notre travail, apportant une valeur inestimable à la réussite de ce projet.

Enfin, notre gratitude va aussi à nos familles, amis et collègues pour leur soutien indéfectible, qui nous a permis de travailler dans les meilleures conditions possibles. Ce projet n'aurait pas été possible sans leur aide précieuse.

# Table des matières

<b>Résumé</b>	<b>5</b>
<b>Introduction</b>	<b>6</b>
1 XML . . . . .	6
2 UML . . . . .	6
3 Design Pattern Prototype . . . . .	6
4 Objectifs du Projet . . . . .	8
5 Méthodologie . . . . .	9
6 Résultats Attendus . . . . .	9
<b>Concept &amp; Prérequis</b>	<b>10</b>
1 Introduction . . . . .	10
2 Besoins Fonctionnels de l'Application . . . . .	10
3 Besoins Non-Fonctionnels de l'Application . . . . .	11
<b>Conception</b>	<b>13</b>
1 Diagramme de Cas d'Utilisation . . . . .	13
2 Diagramme de Classe . . . . .	14
3 Diagramme de Séquence . . . . .	15
4 Diagramme d'activité . . . . .	16
5 Diagramme de Classe avec le Design Pattern Prototype . . . . .	17
<b>Outils et Technologies de Travail</b>	<b>19</b>
1 Visual Studio Code . . . . .	19
2 XML Copy Editor . . . . .	19
3 StarUML . . . . .	20
4 Java . . . . .	20
5 XML . . . . .	21
6 Docker . . . . .	21

---

7	PostgreSQL . . . . .	22
<b>Realisation</b>		<b>23</b>
1	Page Login . . . . .	23
2	Dashboard de l'Administrateur . . . . .	24
3	Page d'Ajout d'un Employé . . . . .	25
4	Page de Modification de l'Employé . . . . .	26
5	Page de Liste des Détails des Employés . . . . .	27
6	Dashboard du Recruteur . . . . .	28
7	Page d'Ajout d'une Offre d'Emploi . . . . .	28
8	Page d'Édition d'une Offre d'Emploi . . . . .	29
9	Page de Liste des Offres d'Emploi . . . . .	29
<b>Conclusion</b>		<b>30</b>
<b>Références</b>		<b>31</b>

# Table des figures

1	Structure du Design Pattern Prototype . . . . .	8
2	Diagramme de Cas d'Utilisation . . . . .	14
3	Diagramme de Classe . . . . .	15
4	Diagramme de Séquence . . . . .	16
5	Diagramme d'activité . . . . .	17
6	Diagramme de Classe avec le Design Pattern Prototype . . . . .	18
7	Logo de Visual Studio Code . . . . .	19
8	Logo de XML Copy Editor . . . . .	20
9	Logo de StarUML . . . . .	20
10	Logo de Java . . . . .	21
11	Logo de XML . . . . .	21
12	Logo de Docker . . . . .	22
13	Logo de PostgreSQL . . . . .	22
14	Login Page . . . . .	23
15	Dashboard de l'Administrateur . . . . .	24
16	Page d'Ajout d'un Employé . . . . .	25
17	Page de Modification de l'Employé . . . . .	26
18	Page de Liste des Détails des Employés . . . . .	27
19	Dashboard du Recruteur . . . . .	28
20	Page d'Ajout d'une Offre d'Emploi . . . . .	28
21	Page d'Édition d'une Offre d'Emploi . . . . .	29
22	Page de Liste des Offres d'Emploi . . . . .	29

# Résumé

Dans le cadre de ce projet, nous avons développé une application innovante et fonctionnelle répondant à des besoins spécifiques en matière de gestion des employés et des offres d'emploi .

Le projet portait sur la création d'une application permettant de centraliser et de simplifier la gestion des employés, des congés, et des offres d'emploi. Nous avons utilisé XML comme technologie principale, en raison de sa souplesse pour structurer et manipuler des données.

Parmi les fonctionnalités développées, nous avons conçu un module pour la gestion des employés, comprenant l'ajout, la modification et la suppression des données, avec stockage dans des fichiers XML validés par des schémas XSD. De plus, un système de gestion des congés permet la soumission et la validation des demandes, en garantissant la conformité des données via des règles définies dans les fichiers XSD.

Un autre aspect clé de notre application est la gestion des offres d'emploi, avec la création et l'affichage de postes disponibles, le tout en exploitant les capacités de transformation des données grâce à XSLT. Les fichiers XSLT ont également été utilisés pour générer des rapports dynamiques et des vues interactives, comme des tableaux ou des formulaires web, offrant une interface utilisateur intuitive et ergonomique.

Ce projet nous a permis de mettre en œuvre des solutions numériques modernes et adaptées, tout en exploitant pleinement nos compétences en développement de systèmes XML, en validation de données, et en conception d'interfaces dynamiques .

# Inroduction

Avant de plonger dans le rapport détaillé sur notre projet intitulé Développement d'une application de gestion des employés et des offres d'emploi basée sur XML avec validation des données via XSD et utilisant le design pattern Prototype, nous souhaitons présenter une brève introduction.

## 1 XML

XML (eXtensible Markup Language) est un langage de balisage conçu pour stocker et transporter des données de manière lisible par les machines et les humains. Dans notre projet, XML est utilisé pour la gestion et le stockage structuré des données. L'intégration de XSLT (eXtensible Stylesheet Language Transformations) permet d'appliquer des feuilles de style aux données XML, séparant ainsi la structure des données de leur présentation visuelle. Parallèlement, le Document Object Model (DOM) en PHP, notamment avec DOM Document, offre une interface pour la manipulation dynamique des documents XML, renforçant la flexibilité de notre application web dans le traitement des données structurées.

## 2 UML

UML (Unified Modeling Language) est un langage de modélisation standard utilisé dans l'ingénierie logicielle pour visualiser, concevoir et documenter les systèmes logiciels. Il permet de représenter graphiquement les structures, comportements et interactions d'un système, facilitant ainsi la communication entre les développeurs, analystes et parties prenantes, tout en servant de guide pour le développement et la maintenance des logiciels.

## 3 Design Pattern Prototype

Le patron de conception Prototype est un patron de création qui permet de créer de nouveaux objets en copiant des instances existantes, au lieu de les instancier directement via une classe. Chaque objet cloné est une copie indépendante, mais conserve les caractéristiques et propriétés de l'objet original.

## **Rôle du Design Pattern Prototype**

Le rôle principal de ce design pattern est de :

1. Réduire les coûts de création d'objets complexes ou coûteux à initialiser.
2. Simplifier la duplication d'objets ayant des configurations ou états spécifiques.
3. Encapsuler la logique de création dans chaque classe, rendant le système plus extensible et flexible.

## **Avantages du Design Pattern Prototype**

### **1. Réduction des coûts de création**

Le clonage est souvent plus rapide que l'instanciation via un constructeur, surtout pour des objets complexes.

### **2. Flexibilité**

Permet de créer de nouveaux objets dynamiquement, sans connaître leur type exact à l'exécution.

### **3. Encapsulation des détails de création**

La logique de création est déléguée à l'objet, rendant le code client plus simple et moins couplé aux classes concrètes.

### **4. Facilité d'ajout de nouveaux types**

Il suffit d'ajouter une nouvelle classe qui implémente l'interface Prototype et la méthode clone().

### **5. Économie de mémoire**

En évitant la répétition de processus d'initialisation lourds, le patron Prototype optimise l'utilisation des ressources.

### **6. Création d'objets complexes**

Il permet de répliquer des objets ayant des états spécifiques ou configurés de manière complexe, sans devoir répéter tout le processus de configuration.



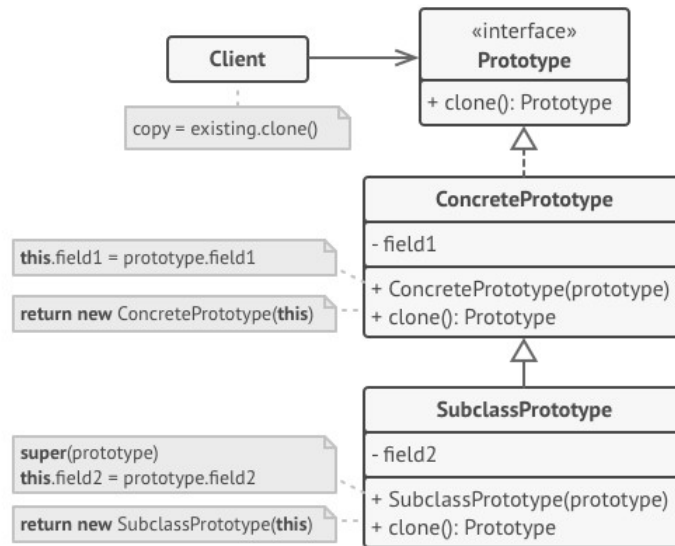


FIGURE 1 – Structure du Design Pattern Prototype

## 4 Objectifs du Projet

1. **Simplifier la gestion des ressources humaines :** Offrir une solution numérique pour centraliser et automatiser les processus d'ajout, de modification et de suppression des informations des employés.
2. **Optimiser la gestion des congés :** Mettre en place un système pour la soumission, la validation et le suivi des demandes de congés.
3. **Faciliter la gestion des offres d'emploi :** Permettre la création, l'affichage et la mise à jour des offres d'emploi dans un format structuré et standardisé.
4. **Assurer la gestion des données via XML :** Développer un système permettant de gérer les informations des employés, des congés et des offres d'emploi en utilisant XML pour le stockage et la validation des données.
5. **Utiliser le design pattern Prototype :** Implémenter le design pattern Prototype pour faciliter la création de nouveaux objets en clonant des objets existants, ce qui permettra une gestion plus flexible et évolutive des données.

Ces objectifs visent à créer une application efficace, intuitive et adaptée aux besoins des utilisateurs, tout en exploitant les standards modernes et les bonnes pratiques en matière de gestion des données XML et de conception logicielle.

## 5 Méthodologie

1. **Développement par XML, Java et Design Patterns :** Le projet repose sur l'utilisation des technologies XML pour le stockage, la validation et la transformation des données, ainsi que sur Java pour le développement de l'application. L'architecture de l'application est basée sur l'utilisation du design pattern Prototype, qui permet de créer de nouveaux objets par clonage d'objets existants. Cette approche facilite la gestion flexible et efficace des entités, telles que les employés et les offres d'emploi. Le design pattern Prototype permet également d'optimiser la création d'objets complexes en réduisant la duplication du code et en garantissant une meilleure performance du système.
2. **Processus de Gestion des Employés et des Offres** Une description détaillée des processus sera fournie, mettant l'accent sur l'intuitivité et l'efficacité. L'application est conçue pour permettre une gestion fluide et centralisée des données des employés, des congés, et des offres d'emploi, tout en validant les données via des schémas XSD.

## 6 Résultats Attendus

1. **Application Web Intuitive :** Le projet vise à fournir une application web intuitive pour la gestion des employés et des offres d'emploi, facilitant les processus tels que l'ajout, la modification, et la suppression de données, ainsi que la soumission et la gestion des congés.
2. **Efficacité de la Gestion :** Du côté administratif, les résultats attendus incluent une gestion optimisée des employés et des offres grâce à des outils conviviaux intégrés dans l'application, avec une prise en charge de la fonctionnalité de multi-rôle, permettant à chaque utilisateur d'accéder à des fonctionnalités adaptées à ses responsabilités.
3. **Gestion des Données et Validation :** L'utilisation de XML et de schémas XSD garantit des données structurées et validées, assurant ainsi l'intégrité et la conformité des informations au sein du système.

# Concept & Prérequis

## 1 Introduction

L'évolution des technologies numériques a profondément modifié la manière dont les entreprises et les organisations gèrent leurs ressources humaines et leurs offres d'emploi. Dans ce contexte, la nécessité d'une plateforme efficace pour superviser et simplifier les interactions entre les administrateurs et les employés est devenue essentielle.

Ce chapitre documente le processus de conception et de développement d'une application innovante, spécialement conçue pour répondre aux besoins de gestion des employés et des offres d'emploi. Cette application, développée en utilisant Java, XML, et des design patterns, offre une solution complète et intuitive permettant aux administrateurs d'avoir un contrôle total sur les processus de gestion des employés, des congés, et des offres d'emploi.

La réalisation de cette application marque une avancée significative dans l'optimisation des processus de gestion interne, tout en garantissant une expérience utilisateur fluide et efficace. Grâce à des fonctionnalités telles que la validation des données via XSD, et la génération de rapports dynamiques à l'aide de XSLT, cette application vise à améliorer les opérations quotidiennes et à renforcer la collaboration entre les administrateurs, les gestionnaires, et les employés.

Elle fournit également un système flexible et sécurisé basé sur des rôles, permettant à chaque utilisateur d'accéder aux fonctionnalités adaptées à ses responsabilités, contribuant ainsi à une expérience utilisateur optimale et à une gestion efficace.

## 2 Besoins Fonctionnels de l'Application

### Pour l'Employé :

**1. Consultation des informations personnelles :** Chaque employé peut accéder et modifier ses informations personnelles et professionnelles, telles que son nom, adresse, rôle dans l'entreprise, et autres données perti-

nentes.

**2. Demande de congé :** Les employés peuvent soumettre des demandes de congé, spécifier les dates, et suivre l'état de leurs demandes (en attente, approuvée, rejetée).

**3. Accès aux offres d'emploi internes :** L'employé peut consulter les offres d'emploi ouvertes dans l'entreprise et postuler directement à celles qui correspondent à ses qualifications et à ses intérêts.

**4. Suivi des candidatures internes :** L'employé peut suivre l'état de sa candidature pour les postes internes et recevoir des notifications sur le statut de sa demande.

## **Pour l'Administrateur :**

**1. Gestion des informations des employés :** L'administrateur peut créer, mettre à jour ou supprimer les profils des employés, gérer leurs rôles et permissions, et consulter leurs informations et historiques.

**2. Validation des demandes de congé :** L'administrateur valide ou rejette les demandes de congé soumises par les employés, et peut également ajuster les périodes de congé selon les besoins de l'entreprise.

**3. Gestion des offres d'emploi :** L'administrateur peut créer, publier, et gérer les offres d'emploi disponibles dans l'entreprise, y compris les descriptions de poste, les qualifications requises, et les processus de candidature.

**4. Communication avec les employés :** L'administrateur peut envoyer des notifications ou des messages aux employés concernant l'état de leurs demandes de congé, leur candidature interne, ou toute autre information pertinente.

## **3 Besoins Non-Fonctionnels de l'Application**

**-Performances :** Assurer une application réactive avec des temps de réponse rapides, notamment pour la consultation des informations des employés, la soumission des candidatures, et la gestion des congés. L'objectif est de garantir une expérience utilisateur fluide, même lors de pics de trafic.

**-Évolutivité :** Concevoir l'application pour qu'elle puisse évoluer facilement afin de supporter un nombre croissant d'employés, de candidats et de demandes de congé, sans compromettre les performances ou la stabilité du système.

**-Facilité de maintenance :** L'architecture de l'application doit être modulaire et claire pour faciliter la maintenance, la mise à jour et la correction des bogues, ainsi que l'ajout de nouvelles fonctionnalités sans perturber les opérations en cours.

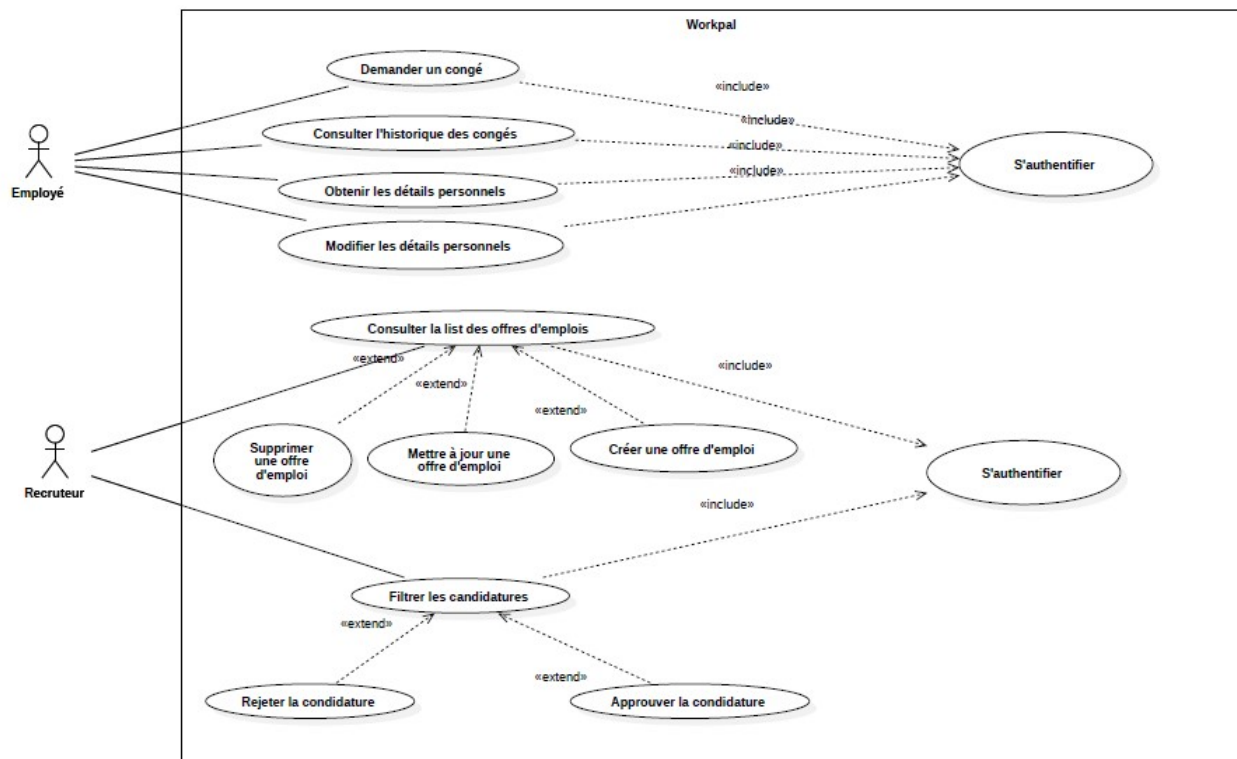
**-Documentation :** Fournir une documentation complète pour les administrateurs, les utilisateurs finaux (employés, candidats), et les développeurs. Cela permet d'assurer une prise en main facile et une gestion optimisée du système, ainsi que de clarifier les processus d'utilisation et d'administration.

**-Extensibilité :** Concevoir l'application pour qu'elle puisse facilement intégrer de nouvelles fonctionnalités ou être étendue pour répondre à de futurs besoins, tels que l'ajout de nouveaux rôles, de nouvelles options de gestion de congés, ou de nouveaux types d'offres d'emploi.

**-Disponibilité :** Assurer une disponibilité élevée de l'application avec des mécanismes de sauvegarde et de redondance pour minimiser les interruptions de service, en mettant en place des mesures de surveillance et de détection précoce des pannes pour une résolution rapide.

# Conception

## 1 Diagramme de Cas d'Utilisation



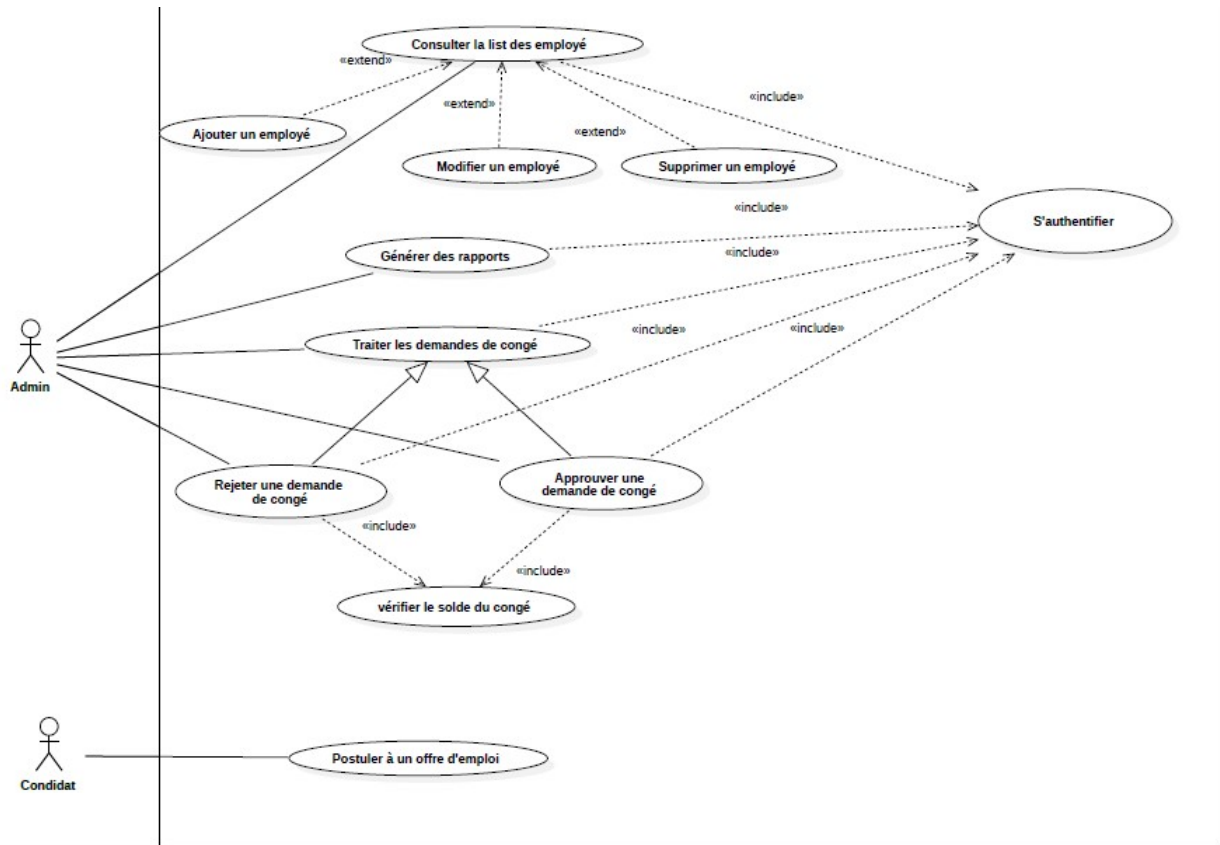


FIGURE 2 – Diagramme de Cas d'Utilisation

Le diagramme de cas d'utilisation pour le système Workpal illustre les interactions entre les acteurs (Employé, Recruteur, Candidat, Admin) et leurs actions respectives telles que l'obtention ou la modification des détails personnels, la gestion des congés, la création et la mise à jour des offres d'emploi, la gestion des candidatures, et la génération de rapports, avec une dépendance commune à l'authentification représentée par « include ».

## 2 Diagramme de Classe

Le diagramme de classe pour la gestion des employés et des offres d'emploi, illustre les rôles principaux (Admin, Employé, Recruteur), leurs responsabilités respectives (gestion des congés, création et suivi des offres d'emploi, candidatures), ainsi que les entités métiers (Congés, Offres d'emploi, Candidatures) avec leurs relations, dans une architecture orientée objet conçue pour assurer une gestion efficace et structurée des processus.

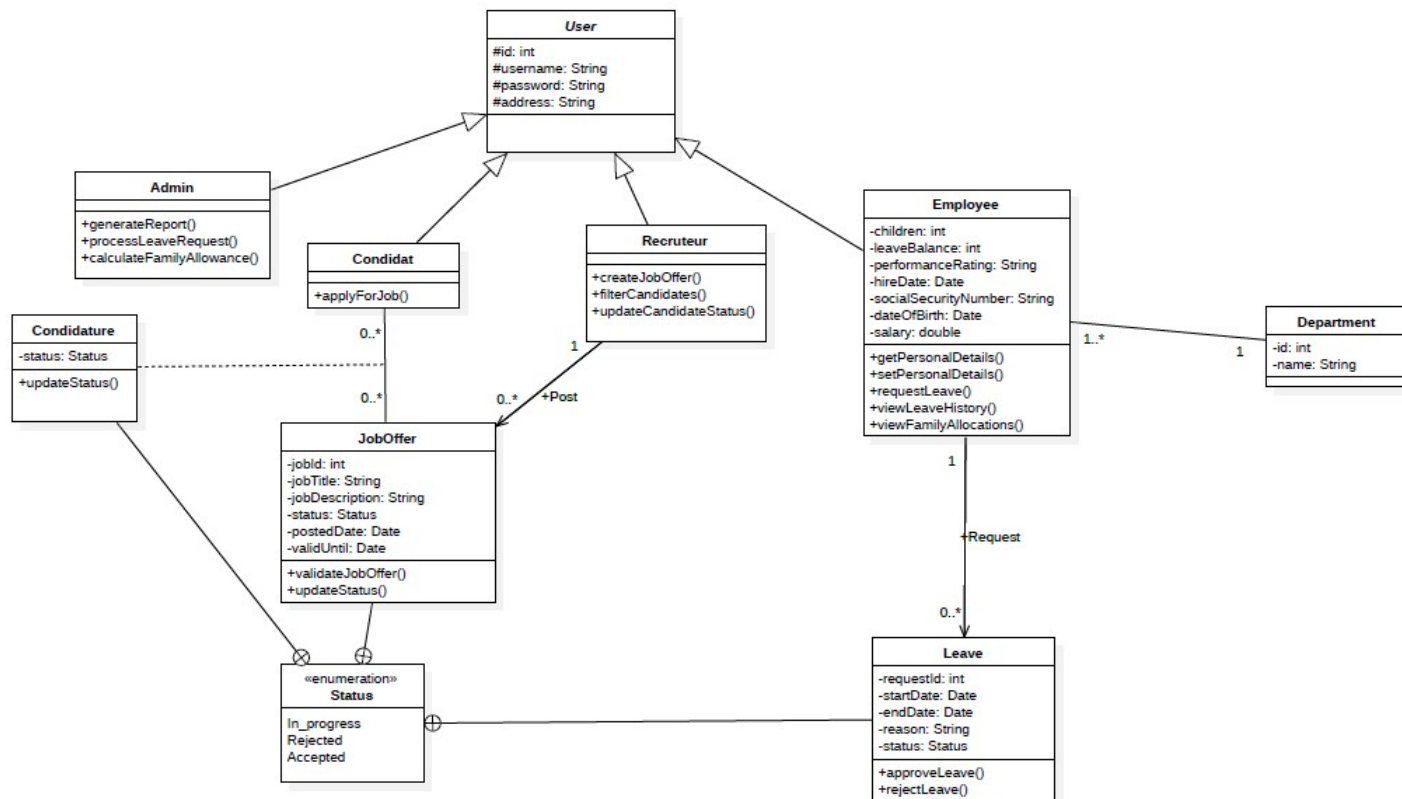


FIGURE 3 – Diagramme de Classe

### 3 Diagramme de Séquence

Le diagramme de séquence pour le cas d'utilisation "Demander un congé" décrit l'interaction de l'employé avec le système, où il s'authentifie, soumet une demande de congé via un formulaire. Les données sont d'abord vérifiées pour les erreurs, puis le système vérifie la disponibilité des dates demandées et valide le solde de congés disponible. Si toutes les conditions sont remplies, la demande est enregistrée et un message de succès est affiché.



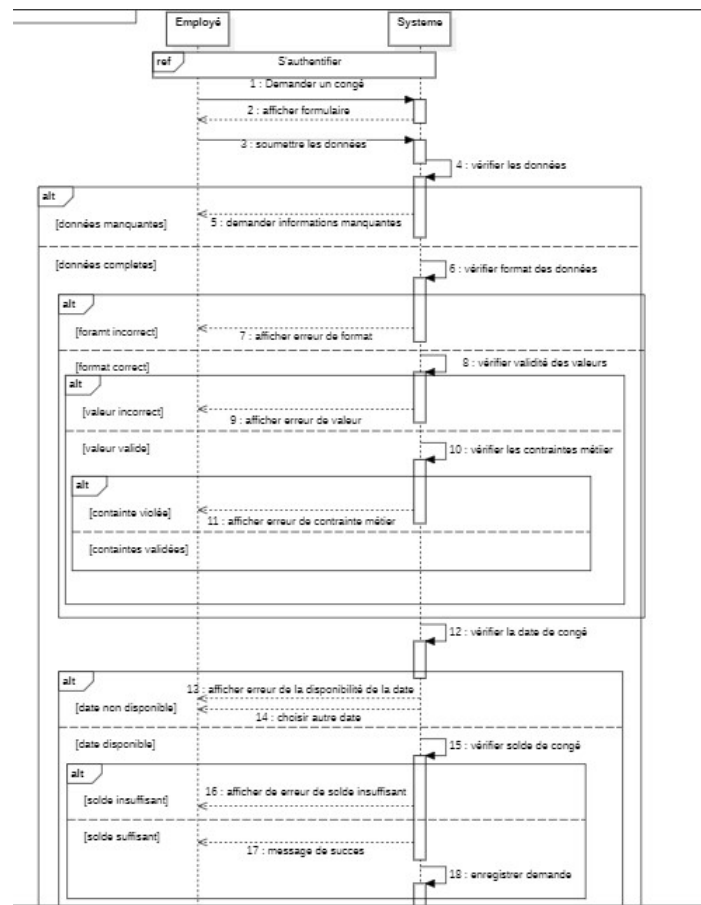


FIGURE 4 – Diagramme de Séquence

## 4 Diagramme d'activité

Ce diagramme d'activité décrit le processus de demande de congé d'un employé, incluant l'authentification, la soumission des informations, la vérification du solde des congés, l'enregistrement de la demande avec des messages d'erreur en cas d'authentification échouée ou de solde insuffisant.

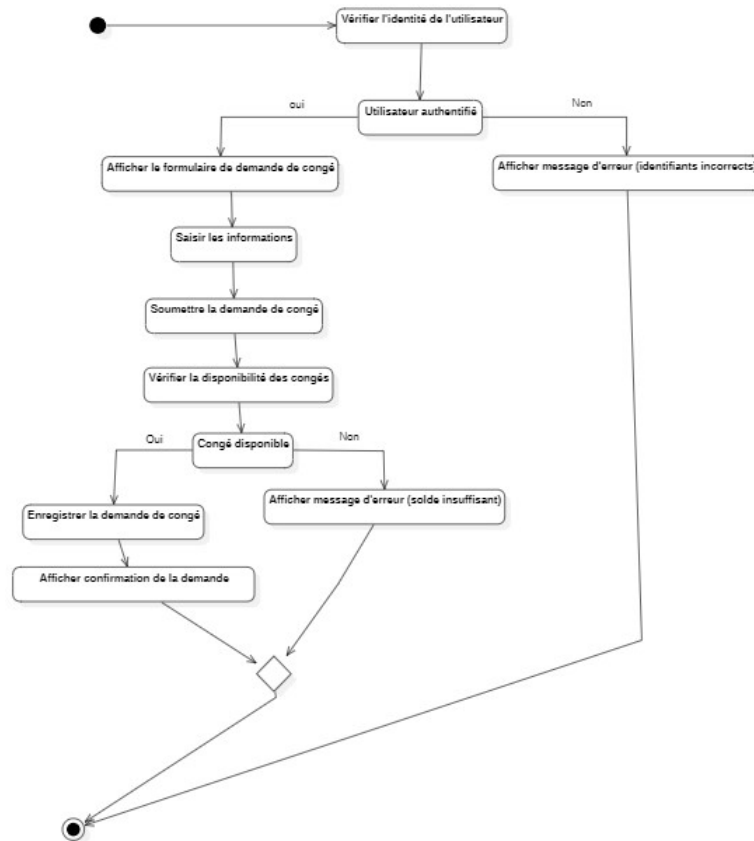


FIGURE 5 – Diagramme d'activité

## 5 Diagramme de Classe avec le Design Pattern Prototype

Ce diagramme UML illustre l'utilisation du **design pattern Prototype** pour modéliser un système de gestion des employés et des offres d'emploi, où chaque entité clé (comme les utilisateurs : administrateurs, employés, recruteurs ; les départements ; les demandes de congés ; les offres d'emploi ; et les candidatures) implémente une méthode de clonage via l'interface 'Prototype', permettant la duplication d'objets avec leurs états et configurations spécifiques. Les relations détaillées incluent, par exemple, l'association des employés à leurs demandes de congés et à leur département, des recruteurs aux offres d'emploi qu'ils créent, et des candidats aux offres auxquelles ils postulent, tout en gérant les états grâce à une énumération 'Status'. Ce modèle assure une gestion flexible, évolutive et optimisée des processus liés aux employés et au recrutement.

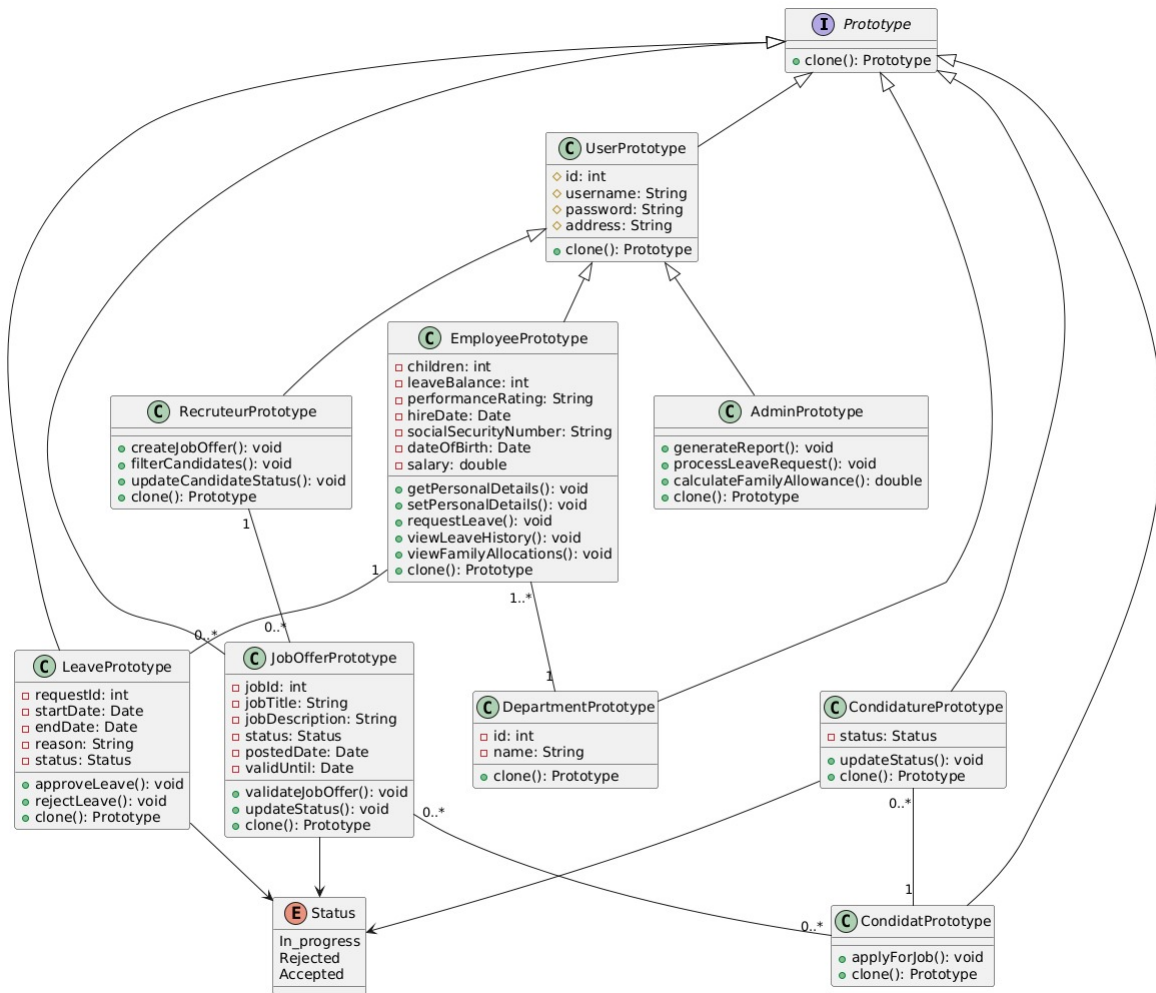


FIGURE 6 – Diagramme de Classe avec le Design Pattern Prototype

# Outils et Technologies de Travail

## 1 Visual Studio Code

Visual Studio Code (VS Code) est un éditeur de code source open source développé par Microsoft. Il est conçu pour être léger et puissant, avec un support pour de nombreux langages de programmation, tels que JavaScript, Python, C++, Java, et bien d'autres. VS Code offre des fonctionnalités telles que l'autocomplétion, la coloration syntaxique, la gestion de projets, le débogage intégré, l'intégration avec Git, et une vaste bibliothèque d'extensions pour personnaliser l'éditeur selon les besoins de l'utilisateur. Il fonctionne sur les systèmes d'exploitation Windows, macOS et Linux.



FIGURE 7 – Logo de Visual Studio Code

## 2 XML Copy Editor

Un XML Copy Editor est un éditeur spécialisé pour créer et modifier des fichiers XML, offrant des outils pour vérifier la validité de la syntaxe, valider contre des schémas (XSD) et faciliter l'édition des données structurées en XML.

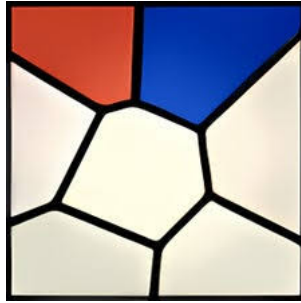


FIGURE 8 – Logo de XML Copy Editor

### 3 StarUML

StarUML est un logiciel de modélisation UML (Unified Modeling Language) qui permet de concevoir des diagrammes pour représenter visuellement des systèmes logiciels. Il prend en charge plusieurs types de diagrammes UML, tels que les diagrammes de classes, de séquences, d'activités, etc. StarUML est utilisé pour créer des modèles de logiciels, améliorer la communication entre les développeurs et les parties prenantes, et faciliter la gestion de la complexité dans le développement logiciel. Il offre une interface intuitive et des fonctionnalités avancées comme la génération de code à partir de modèles.



FIGURE 9 – Logo de StarUML

### 4 Java

Java est un langage de programmation orienté objet, développé par Sun Microsystems (aujourd'hui propriété d'Oracle). Il est conçu pour être indépendant de la plateforme, ce qui signifie qu'un programme Java peut s'exécuter sur n'importe quel appareil ou système d'exploitation qui dispose d'une machine virtuelle Java (JVM). Java est largement utilisé dans le développement d'applications web, mobiles (notamment pour Android), ainsi que pour des applications d'entreprise et des systèmes embarqués. Il est apprécié pour sa robustesse, sa sécurité et sa portabilité.



FIGURE 10 – Logo de Java

## 5 XML

XML (eXtensible Markup Language) est un langage de balisage utilisé pour structurer, stocker et échanger des données de manière lisible par l'homme et la machine. Contrairement à HTML, qui est destiné à la présentation de pages web, XML est conçu pour décrire la structure des données de façon flexible. Il permet de créer des documents avec des balises personnalisées, facilitant l'organisation et la transmission des informations entre différentes applications ou systèmes. XML est largement utilisé dans les échanges de données entre systèmes informatiques, les configurations de logiciels, et les formats de données comme RSS, SOAP et SVG.



FIGURE 11 – Logo de XML

## 6 Docker

Docker est une plateforme de virtualisation légère qui permet de développer, de déployer et d'exécuter des applications dans des conteneurs. Un conteneur Docker est un environnement isolé qui regroupe une application et toutes ses dépendances (bibliothèques, configurations, etc.) pour garantir qu'elle fonctionne de manière cohérente sur n'importe quel système. Docker permet ainsi de simplifier la gestion des environnements de développement et de production, en offrant une portabilité entre différentes machines et une meilleure gestion des versions d'applications. Il est largement utilisé pour l'orchestration des applications, le déploiement continu, et les environnements cloud.

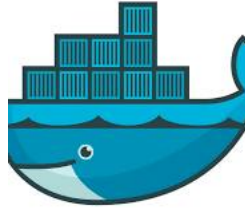


FIGURE 12 – Logo de Docker

## 7 PostgreSQL

PostgreSQL est un système de gestion de base de données relationnelle (SGBDR) open-source, réputé pour sa fiabilité, sa performance et sa conformité aux standards SQL. Il prend en charge des fonctionnalités avancées comme les transactions ACID, la gestion de données géospatiales avec l'extension PostGIS, et la possibilité de définir des types de données personnalisés. PostgreSQL permet également l'extension avec des fonctions et des procédures stockées écrites en plusieurs langages, comme SQL, PL/pgSQL, et d'autres. Il est utilisé pour une large gamme d'applications, des petits projets aux grandes entreprises, grâce à sa robustesse, sa flexibilité et son évolutivité.



FIGURE 13 – Logo de PostgreSQL

# Réalisation

## 1 Page Login

Permet aux utilisateurs de se connecter à l'application en entrant leur nom d'utilisateur, mot de passe et rôle (admin, recruteur ou employé).

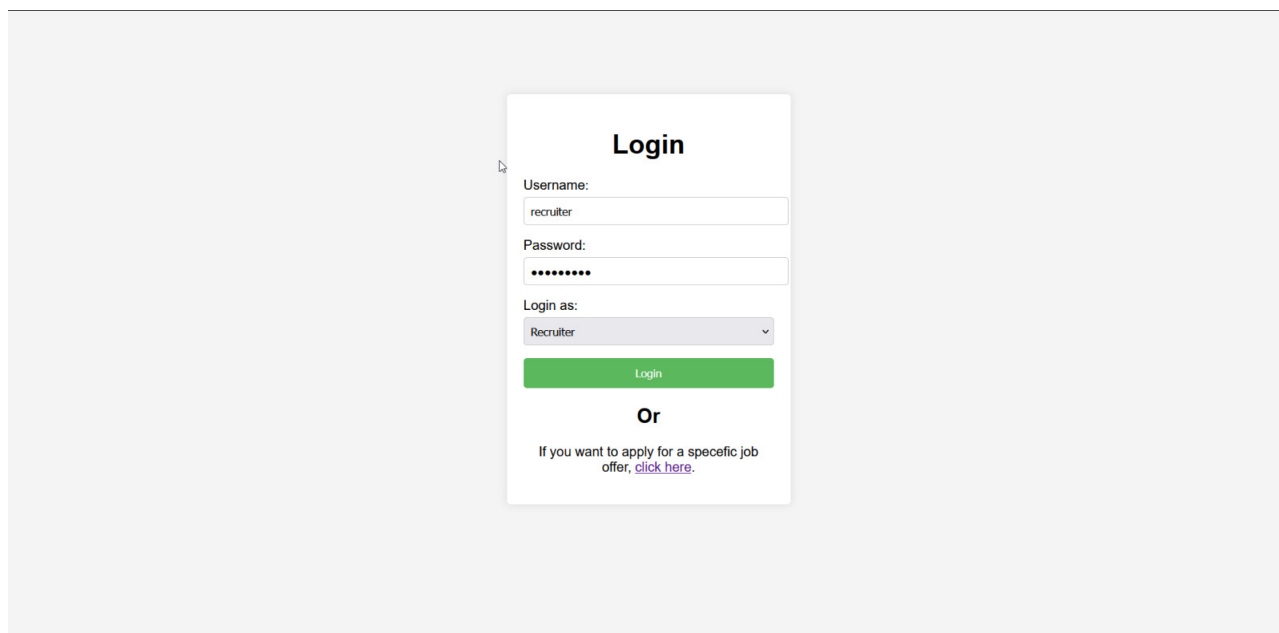
A screenshot of a web application's login page. The page has a light gray background. In the center, there is a white rectangular box with a thin gray border. Inside this box, the word "Login" is written in bold black text at the top. Below it, there are three input fields: "Username:" with the text "recruter" entered, "Password:" with a masked password of ten dots, and "Login as:" with a dropdown menu showing "Recruter". Below these fields is a green button with the text "Login". Under the button, the word "Or" is centered. At the bottom of the box, there is a line of text: "If you want to apply for a specefic job offer, [click here](#)." The word "specefic" is misspelled. A mouse cursor is visible over the "Login" button.

FIGURE 14 – Login Page



## 2 Dashboard de l'Administrateur

Présente une vue d'ensemble des statistiques et données clés, comme le nombre d'employés, les offres d'emploi, et d'autres fonctionnalités pour la gestion globale.

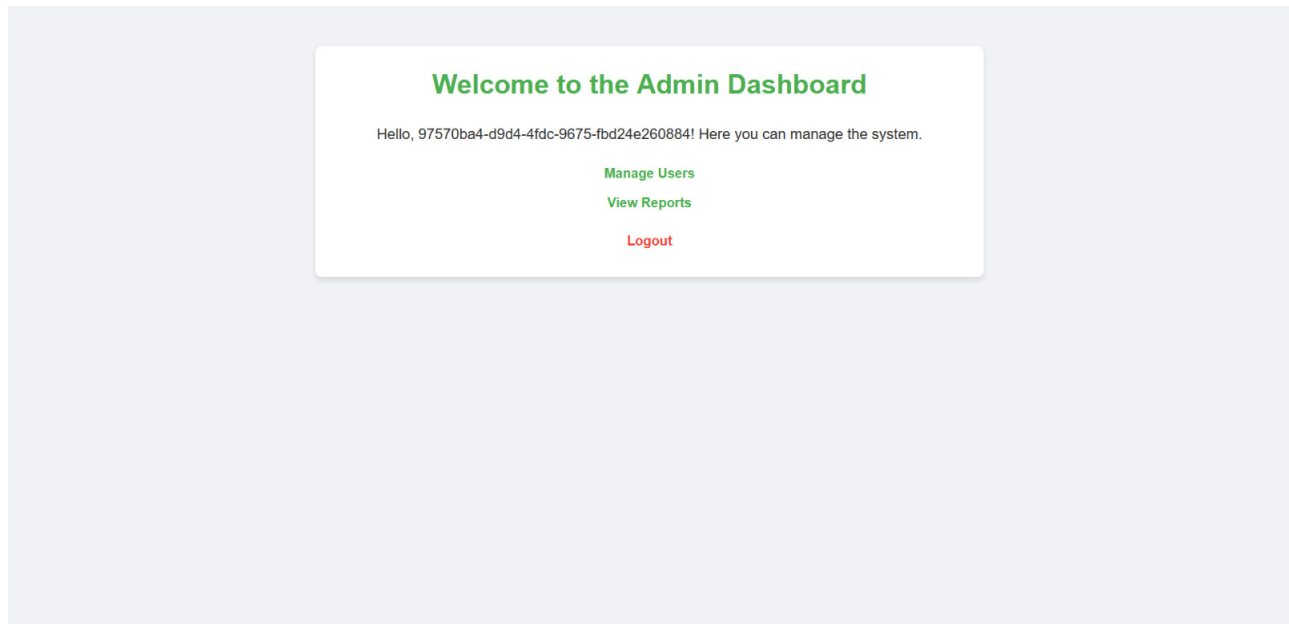
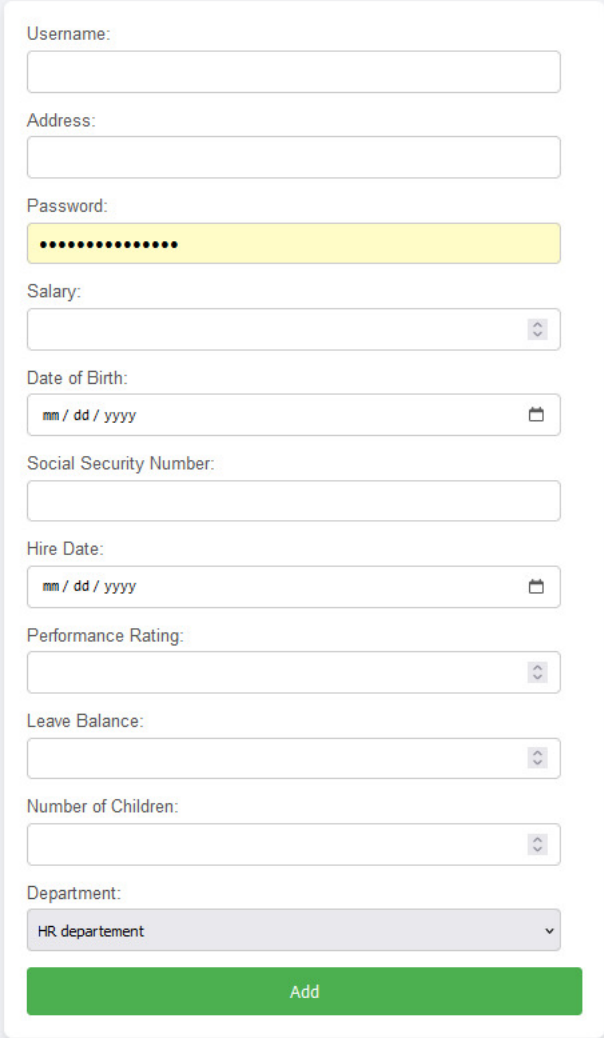


FIGURE 15 – Dashboard de l'Administrateur

### 3 Page d'Ajout d'un Employé

Fournit un formulaire pour ajouter un nouvel employé en saisissant ses informations personnelles et professionnelles.

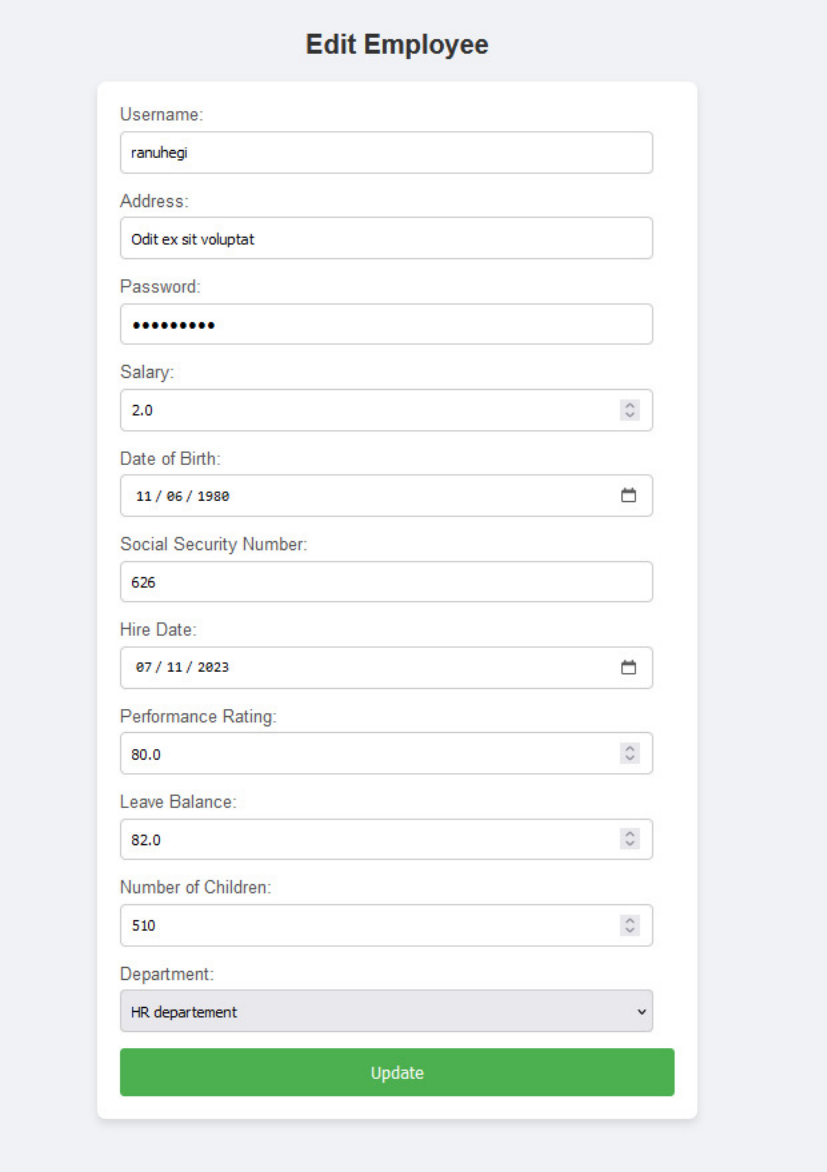


The image shows a web form titled "Add Employee" centered on a light gray background. The form itself is a white rounded rectangle containing several input fields and a green "Add" button at the bottom. The fields are labeled as follows: "Username:" with a text input; "Address:" with a text input; "Password:" with a yellow text input showing masked characters; "Salary:" with a text input and a small downward arrow icon; "Date of Birth:" with a text input showing "mm / dd / yyyy" and a calendar icon; "Social Security Number:" with a text input; "Hire Date:" with a text input showing "mm / dd / yyyy" and a calendar icon; "Performance Rating:" with a text input and a small downward arrow icon; "Leave Balance:" with a text input and a small downward arrow icon; "Number of Children:" with a text input and a small downward arrow icon; and "Department:" with a dropdown menu showing "HR departement" and a downward arrow icon. The green "Add" button is positioned at the bottom center of the form.

FIGURE 16 – Page d'Ajout d'un Employé

## 4 Page de Modification de l'Employé

Permet de mettre à jour les informations d'un employé existant via un formulaire pré-rempli.



The image shows a web form titled "Edit Employee" with a light gray background. The form is a white card with a shadow, containing several input fields and a green "Update" button at the bottom. The fields are labeled and pre-filled with the following data:

Field Label	Value	Field Type
Username:	ranuhegi	Text
Address:	Odit ex sit voluptat	Text
Password:	.....	Text (masked)
Salary:	2.0	Text (with spinner)
Date of Birth:	11 / 06 / 1980	Date (with calendar icon)
Social Security Number:	626	Text
Hire Date:	07 / 11 / 2023	Date (with calendar icon)
Performance Rating:	80.0	Text (with spinner)
Leave Balance:	82.0	Text (with spinner)
Number of Children:	510	Text (with spinner)
Department:	HR departement	Dropdown

At the bottom of the form is a green button labeled "Update".

FIGURE 17 – Page de Modification de l'Employé

## 5 Page de Liste des Détails des Employés

Affiche une liste complète des employés avec leurs informations principales, permettant aux administrateurs ou recruteurs de consulter les détails.

### Manage Employees

[Add New Employee](#)

Username	Address	Hire Date	Salary	Department	Actions
ranuhegi	Odit ex sit voluptat	2023-07-11	2.0	HR departement	<a href="#">Edit</a> <a href="#">Delete</a>
edited employee	Nihil tempor occaec	2005-11-15	72.0	HR departement	<a href="#">Edit</a> <a href="#">Delete</a>
pabiq	Voluptates perferend	1975-02-01	76.0	HR departement	<a href="#">Edit</a> <a href="#">Delete</a>

FIGURE 18 – Page de Liste des Détails des Employés

## 6 Dashboard du Recruteur

Fournit des outils pour gérer les candidatures, suivre les offres d'emploi et superviser les recrutements.

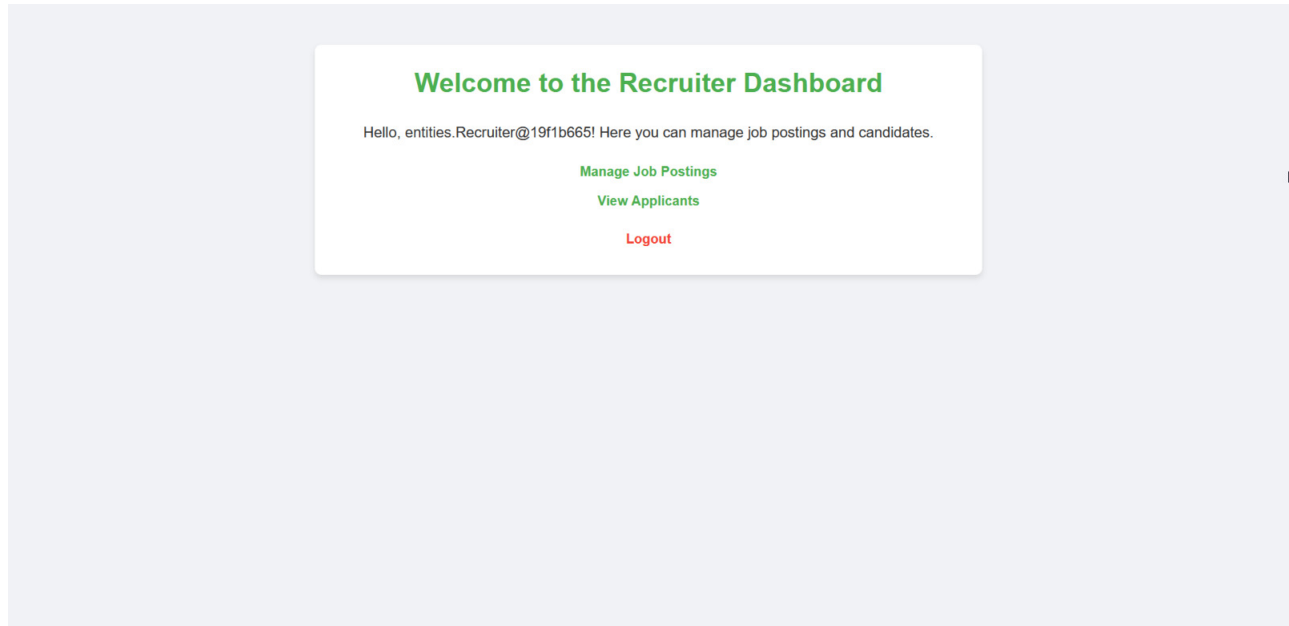


FIGURE 19 – Dashboard du Recruteur

## 7 Page d'Ajout d'une Offre d'Emploi

Permet aux recruteurs de créer une nouvelle offre d'emploi en renseignant des informations comme le titre, la description, et les qualifications requises.

**Add a New Job Offer**

Job Title:

Job Description:


Valid Until:  

FIGURE 20 – Page d'Ajout d'une Offre d'Emploi

## 8 Page d'Édition d'une Offre d'Emploi

Permet de modifier une offre d'emploi existante en mettant à jour ses détails selon les besoins.

Edit a New Job Offer

Job Title:

Nihil ipsum sit ea a

Job Description:

Adipiscing harum ve

Valid Until:

05 / 06 / 1986

Update Job Offer

FIGURE 21 – Page d'Édition d'une Offre d'Emploi

## 9 Page de Liste des Offres d'Emploi

Affiche toutes les offres d'emploi disponibles avec leurs descriptions, dates de publication, et autres détails pertinents.

Liste des Offres d'Emploi

Créer une Nouvelle Offre d'Emploi

ID	Titre	Description	Statut	Date Publication	Date Expiration	Actions
263374c5-cf8b-4af4-9f3f-6baeb0671dcf	Développeur Backend	Développer et maintenir des APIs backend sécurisées.	Ouverte	2025-01-01	2025-02-01	<div>ModifierDelete</div>
da8af4d9-c678-4f4f-b594-8f72597735f5	Et dolore suscipit i	Natus veritatis accu	IN_PROGRESS	2025-01-13	1976-03-15	<div>ModifierDelete</div>
e5eac15a-5987-4a91-9166-b88497afd17b	Mollit ut alias aspe updated	Dolorem sint eiusmod	IN_PROGRESS	2025-01-13	2018-08-01	<div>ModifierDelete</div>
492e0cc7-265d-4cd8-a6cc-31f3d58007fa	soufiane	Sint aut ullamco exc	IN_PROGRESS	2025-01-13	1973-06-15	<div>ModifierDelete</div>
c1186c45-aa6f-451c-9d70-aa18de9fcbf4	Nihil ipsum sit ea a	Adipiscing harum ve	IN_PROGRESS	2025-01-14	1986-05-06	<div>ModifierDelete</div>

FIGURE 22 – Page de Liste des Offres d'Emploi

# Conclusion

En résumé, le développement de notre application de gestion des employés et des offres d'emploi a été une expérience enrichissante et stimulante. En utilisant une combinaison de technologies modernes telles que XML, Java, et des Design Patterns, nous avons conçu une solution flexible et évolutive. L'application repose sur une architecture robuste, avec une prise en charge multilingue et une validation des données via XSD, permettant ainsi une gestion fluide et sécurisée des informations. La méthodologie adoptée a permis de garantir une gestion optimale des processus, tout en favorisant l'innovation et l'efficacité dans le développement d'applications web. Cette expérience nous a non seulement permis de maîtriser des outils techniques puissants, mais également de mettre en œuvre une approche structurée, assurant ainsi la réussite de notre projet.

# Références

- <https://www.w3schools.com/xml/>
- <https://refactoring.guru/design-patterns/prototype>
- <https://www.lucidchart.com/blog/fr/types-de-diagrammes-UML>