

Mémoire de Projet Cadre

Sujet :

Conception et Réalisation d’une application de recrutement et CVthèque

Filière d’ingénieur :

**I**ngénierie **L**ogicielle et **I**ntégration des **S**ystèmes **I**nformatiques

****



Réalisé par :

* ABOUMOUSSA Zainaba
* AKERKAOU Rizki

Encadrants

* Pr K. DOUZI Encadrant Pédagogique
* Mr O QAROUI Encadrant Entreprise

Jury :

* Pr C. LEGHRISS Examinateur
* Pr O. EL BEGGAR Rapporteur

Année universitaire : 2018/2019

# Dédicace

**À LA MEMOIRE DE MON PERE**

Vous avez peiné des jours, veillé des nuits et consenti tous les sacrifices pour nous élever et mener à bien notre éducation. J’aurais tant souhaité vous avoir à mes côtés, mais Dieu a décidé autrement. Ni la mort, ni le temps ne vous feront oublier.

Que ce travail soit une prière pour le repos de votre âme.

**À MA CHERE MERE**

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez faits pour mon instruction et mon bien être.

Je vous remercie pour tout le soutien et l’amour que vous me témoignez depuis ma naissance et j’espère que votre bénédiction m’accompagnera pour toujours.

**À MES CHERS FRERES ET SOEURS**

En témoignage de mon respect, de ma profonde tendresse et reconnaissance, je vous souhaite une vie pleine de bonheur et de succès, et que Dieu, le Tout Puissant, vous protège et vous garde.

**À MES CHERS PROFESSEURS**

En gratitude à votre encouragement et votre patience, un remerciement particulier et sincère pour tous vos efforts avec moi, souhaits de bonheur, de santé et de succès.

**À TOUS MES CHERS AMIS SANS EXECEPTION**

En souvenir de notre sincère et profonde amitié et des moments agréables que nous avons passés ensemble.

Veuillez trouver dans ce travail l’expression de mon respect le plus profond et mon affection la plus sincère.

ABOUMOUSSA ZAINABA

Je dédie cet humble travail :

**A MES CHERS PARENTS**

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez faits pour mon instruction et mon bien être.

Je vous remercie pour tout le soutien et l’amour que vous me témoignez depuis ma naissance et j’espère que votre bénédiction m’accompagnera pour toujours.

**A MES CHERS FRERES ET SŒURS**

En témoignage de mon respect, de ma profonde tendresse et reconnaissance, je vous souhaite une vie pleine de bonheur et de succès, et que Dieu, le Tout Puissant, vous protège et vous garde.

**À MES CHERS PROFESSEURS**

En gratitude à votre encouragement et votre patience, un remerciement particulier et sincère pour tous vos efforts avec moi, souhaits de bonheur, de santé et de succès.

**A TOUS MES CHERS AMIS ET GRANDES FAMMILES**

En souvenir de notre sincère et profonde amitié et des moments agréables que nous avons passés ensemble.

**À TOUTES LES PERSONNES QUI NOUS CONNAISSENT DE PRES OU DE LOIN**

Seulement pour leur existence

Veuillez trouver dans ce travail l’expression de mon respect le plus profond et mon affection la plus sincère..

AKERKAOU RIZKI

# Remerciement

Au terme de ce travail, Il nous est agréable d’exprimer notre reconnaissance auprès de toutes les personnes, dont l’intervention au cours de ce projet, a favorisé son aboutissement.

Nous commençons par « Pr K. DOUZI », professeur universitaire à la FST de Mohammedia, nous tenons à vous remercier pour votre disponibilité, votre gentillesse, et pour vous patience et vous soutien qui nous ont été précieux afin de mener notre travail à bon port et d'avoir partagé avec nous vos connaissances et pour votre engagement à nos côtés.

En second lieu, nous tenons à remercier notre encadrant au sein de l’entreprise, Mr O. QUAROUI directeur de la société HR\_PATH pour nous avoir donné l’opportunité de passer ce stage dans les meilleures conditions de professionnalisme, matériels et morales et pour son aide qui nous consacrait durant cette période.

Nos remerciements vont également à tous les collaborateurs du Département informatique de la FSTM et le corps professoral de la filière ILISI.

Que tous ceux et celles qui ont contribué de près ou de loin à l’accomplissement de ce travail trouvent l’expression de nos sincères remerciements.

# Liste des tableaux

[Tableau 1 : Prodact Backlog 20](#_Toc11614588)

[Tableau 2 : Planification des sprints 21](#_Toc11614589)

[Tableau 3: Backlog du premier sprint (RELEASE1) 30](#_Toc11614590)

[Tableau 4 : Description textuelle du cas d'utilisation « Authentification » 31](#_Toc11614591)

[Tableau 5: Backlog du deuxième sprint (RELEASE1) 35](#_Toc11614592)

[Tableau 6 Description textuelle du cas d'utilisation « Mise à jour critère » 36](#_Toc11614593)

[Tableau 7 : Description textuelle du cas d'utilisation « Ajout utilisateur » 37](#_Toc11614594)

[Tableau 8 :Backlog du premier sprint (RELEASE2) 42](#_Toc11614595)

[Tableau 9 Description textuelle du cas d'utilisation « Créer CV automatique » 43](#_Toc11614596)

[Tableau 10 Description textuelle du cas d'utilisation « Charger des profils CSV » 43](#_Toc11614597)

[Tableau 11 Backlog du deuxième sprint (RELEASE2) 48](#_Toc11614598)

[Tableau 12 Description textuelle du cas d'utilisation « Affecter poste » 49](#_Toc11614599)

[Tableau 13 Description textuelle du cas d'utilisation « « Recherche Avancée» 49](#_Toc11614600)

# Table des figures

[Figure 1 : logo HR Path 11](#_Toc11614553)

[Figure 2 : Agences HR Path 11](#_Toc11614554)

[Figure 3 : Le processus Scrum 13](#_Toc11614555)

[Figure 4:Diagramme de contexte de l'application 17](#_Toc11614556)

[Figure 5:Diagramme de cas d'utilisations global 21](#_Toc11614557)

[Figure 6: Diagramme de classe global 22](#_Toc11614558)

[Figure 7: Architecture logicielle 25](#_Toc11614559)

[Figure 8 : Diagramme de déploiement 26](#_Toc11614560)

[Figure 9 : Diagramme de composants 27](#_Toc11614561)

[Figure 10 Diagram de cas d'utilisation du sprint 1 30](#_Toc11614562)

[Figure 11 Diagramme d'activité d'authentification 31](#_Toc11614563)

[Figure 12 : Diagramme de classe du sprint 1(RELEASE 1) 32](#_Toc11614564)

[Figure 13 : ARC capture login 1-1 33](#_Toc11614565)

[Figure 14: ARC capture login 1-2 33](#_Toc11614566)

[Figure 15 : Interface authentification 34](#_Toc11614567)

[Figure 16:Interface de Logout 34](#_Toc11614568)

[Figure 17 Diagram de cas d'utilisation du sprint 2 36](#_Toc11614569)

[Figure 18 Diagramme d'activité Ajout Utilisateurs 37](#_Toc11614570)

[Figure 19 Diagramme d'activité définition rôle 38](#_Toc11614571)

[Figure 20: Diagramme de classe du sprint 2 39](#_Toc11614572)

[Figure 21: Nouveau utilisateur 39](#_Toc11614573)

[Figure 22 : Affichage et modification utilisateurs 40](#_Toc11614574)

[Figure 23 : Gestion critères 40](#_Toc11614575)

[Figure 24 Diagram de cas d'utilisation du sprint 1 Release 2 44](#_Toc11614576)

[Figure 25:Diagramme Activité Charger Profils CSV 44](#_Toc11614577)

[Figure 26 Diagramme d'activité Créer CV automatique 45](#_Toc11614578)

[Figure 27 : Diagramme de classe sprint 1 Release 2 46](#_Toc11614579)

[Figure 28: Diagramme Cas d'utilisation Recrutement 48](#_Toc11614580)

[Figure 29 : Diagramme Cas d'utilisations recherche avancé 48](#_Toc11614581)

[Figure 30 : Diagramme de Cas d'utilisation reporting 49](#_Toc11614582)

[Figure 31 : Diagramme Activité Reporting 50](#_Toc11614583)

[Figure 32 : Diagramme Activité Recherche avancé 50](#_Toc11614584)

[Figure 33: Diagramme de classe sprint 2 Release 2 51](#_Toc11614585)

[Figure 34: Nouveau recrutement 51](#_Toc11614586)

[Figure 35: Recherche avancée 52](#_Toc11614587)

# Table des matières

[Liste des tableaux 2](#_Toc11602832)

[Table des figures 6](#_Toc11602833)

[Table des matières 7](#_Toc11602834)

[Introduction générale 9](#_Toc11602835)

[Chapitre 1 : Contexte général du projet 10](#_Toc11602836)

[**I.** **Présentation de l’organisme d’accueil** 11](#_Toc11602837)

[**II.** **Présentation du projet** 12](#_Toc11602838)

[1. Problématique 12](#_Toc11602839)

[2. Solution proposée 12](#_Toc11602840)

[3. Méthodologie adoptée 12](#_Toc11602841)

[4. Justification de choix 13](#_Toc11602842)

[5. Pilotage du projet avec Scrum 14](#_Toc11602843)

[5.1 Planification d’un projet par Scrum 14](#_Toc11602844)

[5.2 Equipe et rôles 14](#_Toc11602845)

[**Conclusion** 15](#_Toc11602846)

[Chapitre 2 : Étude Préliminaire 16](#_Toc11602847)

[**I.** **Identification des acteurs** 17](#_Toc11602848)

[1. Diagramme de contexte : 17](#_Toc11602849)

[**II.** **Spécification des besoins** 17](#_Toc11602850)

[1. Les besoins fonctionnels 17](#_Toc11602851)

[2. Les besoins non fonctionnels 18](#_Toc11602852)

[**III.** **Le backlog du produit :** 19](#_Toc11602853)

[**IV.** **Planification des sprints** 20](#_Toc11602854)

[**V.** **Modélisation des besoins fonctionnels** 21](#_Toc11602855)

[1. Diagramme de cas d’utilisations global : 21](#_Toc11602856)

[2. Diagramme de classe global : 21](#_Toc11602857)

[Conclusion 23](#_Toc11602858)

[Chapitre 3 : Étude technique. 24](#_Toc11602859)

[**I.** **Architecture logicielle** 25](#_Toc11602860)

[**II.** **Architecture physique** 25](#_Toc11602861)

[1. *Diagramme de déploiement* 26](#_Toc11602862)

[*2.* *Diagramme de composants* 27](#_Toc11602863)

[**III.** **Environnements de travail** 27](#_Toc11602864)

[*1.* *Environnement matériel* 27](#_Toc11602865)

[*2.* *Environnement technique* 28](#_Toc11602866)

[**IV.** **Conclusion** 28](#_Toc11602867)

[Chapitre 4 : Release 1 : Authentification, Paramétrage et Gestion des utilisateurs. 29](#_Toc11602868)

[**I.** **Sprint 1** 30](#_Toc11602869)

[***1.*** Backlog du sprint 1 *:* 30](#_Toc11602870)

[***2.*** Diagramme de cas d’utilisation *:* 30](#_Toc11602871)

[*3.* *Diagramme d’activités* 31](#_Toc11602872)

[4. Diagramme des classes 31](#_Toc11602873)

[5. Test et réalisation : 32](#_Toc11602874)

[4.1 Test unitaire : 32](#_Toc11602880)

[**II.** Sprint 2 35](#_Toc11602881)

[***1.*** Backlog du sprint 2 *:* 35](#_Toc11602882)

[***2.*** Diagramme de cas d’utilisation *:* 35](#_Toc11602883)

[*3.* *Diagramme d’activités* 37](#_Toc11602884)

[4. Diagramme des classes 38](#_Toc11602885)

[5. Réalisation : 39](#_Toc11602886)

[Chapitre 5 : Release 2 : gestion de recrutement et CVthèque. 41](#_Toc11602887)

[**I.** **Sprint 1** 42](#_Toc11602888)

[***1.*** Backlog du sprint 1 *:* 42](#_Toc11602889)

[*2.* Diagramme de cas d’utilisation *:* 42](#_Toc11602890)

[ ***Description du cas d’utilisation « Charger des profils CSV »*** 43](#_Toc11602891)

[*3.* *Diagramme d’activités* 44](#_Toc11602892)

[4. Diagramme des classes 45](#_Toc11602893)

[5. Réalisation : 46](#_Toc11602894)

[**II.** **Sprint 2** 47](#_Toc11602895)

[***1.*** Backlog du sprint 2 *:* 48](#_Toc11602896)

[*2.* *Diagramme de cas d’utilisation :* 48](#_Toc11602897)

[ ***Description du cas d’utilisation « Affecter poste»*** 49](#_Toc11602898)

[ ***Description du cas d’utilisation « Recherche Avancée »*** 49](#_Toc11602899)

[*3.* *Diagramme d’activités* 49](#_Toc11602900)

[4. Diagramme des classes 51](#_Toc11602901)

[5. Réalisation : 51](#_Toc11602902)

[Conclusion générale 53](#_Toc11602903)

# Introduction générale

Le recrutement consiste à embaucher la bonne personne au bon poste dans le bon contexte, il n'est pas une science exacte. C’est une démarche décisive dans la politique des ressources humaines, car elle peut avoir un effet contraire aux buts recherchés. Tout recrutement présente un risque, l'enjeu n'est pas de supprimer ce risque mais de le minimiser.

Aujourd'hui, le marché du recrutement a changé et en conséquence, la conception du recrutement à évoluée elle aussi. Pour faire face à ces très nombreuses tensions observées sur le marché de l'emploi, les entreprises sont désormais dans l'obligation de répondre rapidement aux nouveaux enjeux liés à ce processus, la chose qui n’est pas facile avec l’utilisation des emails et des tableurs Excel qui manque de fluidité et génère une perte de temps.

C’est dans ce cadre que s’inscrit notre projet cadre effectué au sein de HR\_PATH. Il s’agit de la mise en œuvre d’une application centralisée, qui regroupe la totalité des modules de recrutement et de changement des carrières au sein de la société Ciments du Maroc.

Le but de l’application est de faciliter la gestion des CV et de simplifier la procédure de recrutement en ayant des indicateurs bien déterminée pour pouvoir effectuer des recherches sur des profils à la demande, de façon rapide et précise.

Le présent rapport décrit les différentes étapes de conception et réalisation de ce projet en cinq chapitres :

* Le première chapitre **« Contexte général » :** C’est un chapitre introductif dans lequel nous effectuons une description de l'entreprise « HR\_PATH », Ensuite, nous exposons le cadre général du projet et la solution proposée, Enfin, nous décrivons la méthodologie adoptée pour la mise en place de la solution.
* Le deuxième chapitre **« Étude Préliminaire »** : Dans ce chapitre nous spécifions les acteurs interagissant avec l'application, nous dégageons également les besoins fonctionnels dans un Backlog de produit. Par la suite, nous définissons le planning estimé du projet.
* Le troisième chapitre **« Étude technique »** : expose une étude technique de l'environnement logiciels les différents outils et technologies de développement misent en ouvre pour la réalisation de l’application.
* Les deux derniers chapitres sont dédiés aux différentes itérations suivies lors du développement de la solution.
* Nous clôturons ce rapport par une conclusion générale dans laquelle nous résumerons l’ensemble de travail et nous exposerons les éventuelles perspectives du présent projet.

# Chapitre 1 : Contexte général du projet

Ce chapitre a pour objectif de situer le projet dans son environnement organisationnel et contextuel. Il présente dans la première section l’organisme d’accueil, tandis que dans la deuxième partie défini la problématique et les objectifs du projet. Enfin, il expose la démarche choisie pour la réalisation de la solution.

## **Présentation de l’organisme d’accueil**

**Le groupe HR Path** créé en 2001, est spécialisé dans les SIRH (Systèmes d'Information pour les Ressources Humaines). Spécialiste global des solutions RH, HR Path propose une prestation complète allant de la réflexion stratégique au déploiement de progiciels intégrés dans le domaine des ressources humaines.



Figure : logo HR Path

HR Path exerce principalement quatre métiers :

* Le conseil métier RH et SIRH : accompagnement des entreprises pour des prestations de Conseil Métier RH et SIRH.
* L’implémentation de SIRH : mise en place technique et fonctionnelle de solutions RH et SIRH des principaux éditeurs.
* L'externalisation de la paie et de la fonction RH.
* L'édition et la distribution de ses propres logiciels RH (Pandore, BiHRdy, Virtualia, Geef).

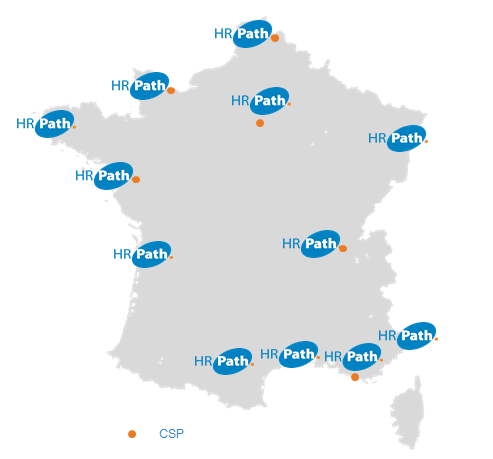


Figure : Agences HR Path

Le groupe compte un peu plus de 700 salariés et réalise un chiffre d’affaires de 80 millions d’euros. Le siège basé à Paris, est soutenu par la présence de 12 agences régionales (cf. Figure 2) et plusieurs antennes internationales dans 15 pays au total.

## **Présentation du projet**

### Problématique

L’entreprise actuellement recrutent sans aucun outil adapté. Les outils utilisés sont Excel pour gérer la liste des recrutements, voire la liste des candidats, et les boites emails pour recevoir tous les CVs et échanger de l’information en interne sur les candidats en question. Ce mode de fonctionnement, même s’il permet d’avancer tant bien que mal, montre vite ses limites :

* Perte de temps: nécessité de saisir des informations dans des tableurs, difficulté pour chercher CVs et informations,
* Perte d’information : aucun stockage d’information sur les candidats et sur les recrutements,
* Pas de fonctionnalité métier pour optimiser le processus de recrutement,
* Pas de centralisation : les informations sont éclatées sur les postes de travail des collaborateurs et dans les boites email,
* Pas de partage d’information en collaboratif,
* Pas de mesure de performance de l’activité de recrutement.

### Solution proposée

Afin de répondre à cette problématique, HR\_path a décidé de mettre au profit des recruteurs un outil sécurisé pour simplifier la procédure de recrutement à travers des indicateurs bien déterminés pour pouvoir effectuer des recherches sur des profils à la demande, de façon rapide et précise.

La plateforme ERecrute permet de gérer, de structurer, d’organiser et de centraliser les recrutements. En fait pour chaque recrutement, on peut définir un processus adéquat afin d’optimiser la perfomance de recrutement.

Cette application permet d’affecter une ressource au poste adéquat et permettre aux recruteurs de trouver facilement les CVs contenant les compétences exigées pour le poste. Ainsi, l’équipe de développement a mis en place un système de gestion de ressources couvrant plusieurs fonctionnalités telles que : Gestion de profil, Gestion de CV, Gestion des recrutements.

### Méthodologie adoptée

Le choix d'une méthode de développement constitue une étape très importante permettant d'avoir une meilleure organisation des différentes phases du projet.

Nous avons opté pour une méthodologie agile (SCRUM) vu qu’il propose au client de vérifier au fur et à mesure que le projet évolue dans la bonne direction tout en tolérant le changement des besoins.

Après le choix de la méthodologie, nous avons besoins d’un langage de modélisation unifiée pour la modélisation de notre projet. Pour concevoir notre système, nous avons choisi UML[[1]](#footnote-1) comme un langage de modélisation.

Notre choix s'est basé sur les points forts de ce langage notamment sa standardisation et les divers diagrammes qu’il propose. Aussi UML présente le meilleur outil pour schématiser des systèmes complexes sous un format graphique, textuel, simplifié et normalisé

### Justification de choix

*« Scrum signifie mêlée au rugby. Scrum utilise les valeurs et l’esprit du rugby et les adapte aux projets de développement. Comme le pack lors d’un ballon porté au rugby, l’équipe chargée du développement travaille de façon collective, soudée vers un objectif précis. Comme un demi de mêlée, le Scrum Master aiguillonne les membres de l’équipe, les repositionne dans la bonne direction et donne le temps pour assurer la réussite du projet. »*

Scrum est issu des travaux de deux des signataires du Manifeste Agile[[2]](#footnote-2), Ken Schwaber et Jeff Sutherland, au début des années 1990.Il appartient à la famille des méthodologies itératives et incrémentales et repose sur les principes et les valeurs agiles.

Le plus souvent, les experts de Scrum, même ses fondateurs, le décrivent comme un cadre ou un patron de processus orienté gestion de projet et qui peut incorporer différentes méthodes ou pratiques d’ingénierie. S’il est difficile de définir la nature de Scrum, sa mise en place est beaucoup plus simple et peut être résumée par la figure 3.

Le principe de base de Scrum est le suivant :

* Dégager dans un premier lieu le maximum des fonctionnalités à réaliser pour former le backlog du produit,
* En second lieu définir les priorités des fonctionnalités et choisir lesquelles seront réalisé dans chaque itération,
* Par la suite focaliser l'équipe de façon itérative sur l’ensemble de fonctionnalités à réaliser, dans des itérations appelées Sprints,
* Un Sprint aboutit toujours sur la livraison d’un produit partiel fonctionnel appelé incrément.

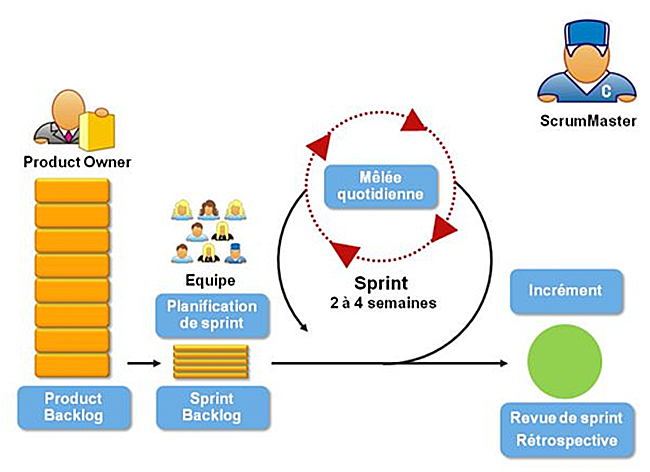


Figure : Le processus Scrum

Le choix de s’est basé sur les atouts de ce dernier. Il se résume comme suit :

* Plus de souplesse et de réactivité,
* Simplicité et légèreté,
* La grande capacité d’adaptation au changement grâce à des itérations courtes,
* La chose la plus importante, c’est que Scrum rassemble les deux cotés théorique et pratique et se rapproche beaucoup de la réalité.

Vu que Scrum ne couvre que les aspects de gestion de projet, et pour compléter le vide laissé en matière de pratiques de développement, nous avons pris la décision de coupler Scrum avec une autre méthodologie agile qui est l’extrême programming et qui couvre les bonnes pratiques d’ingénierie logicielle notamment le développement dirigé par le test, etc. Alors c'est d'ici d'où vient l'idée d'adopter une méthode hybride SCRUM-XP pour combiner les deux aspects : technique et organisationnel.

### Pilotage du projet avec Scrum

#### Planification d’un projet par Scrum

Planification du sprint : Elle s’appuie sur la planification de la « release » réalisée en pente. La première réunion du sprint ne se limite pas à planifier, on y trouve les activités suivantes :

* Valider les « stories » du Backlog pris en compte dans le sprint, concevoir les solutions.
* Identifier et estimer les tâches.
* Prise des tâches par chacun des membres de l’équipe …

Revue du sprint : Elle permet de montrer les résultats du développement effectués au cours du sprint, seule une version opérationnelle est montrée.

Rétrospective : Elle est faite en interne en équipe (avec la présence du Scrum Master), l’objectif est de comprendre ce qui n’a pas bien fonctionné dans le sprint, les erreurs commises et de prendre des décisions pour procéder aux améliorations.

Scrum quotidien : il s’agit d’une réunion de synchronisation de l’équipe de développement qui se fait debout en 15 minutes maximum au cours de laquelle chacun répond principalement à 3 questions :

1. Qu’est-ce que j’ai fait hier ?
2. Qu’est-ce que je ferai aujourd’hui ?
3. Quels obstacles me retardent ?

#### Equipe et rôles

« L’équipe a un rôle capital dans Scrum : elle est constituée dans le but d’optimiser la flexibilité et la productivité ; pour cela, elle s’organise elle-même et doit avoir toutes les compétences nécessaires au développement du produit. Elle est investie avec le pouvoir et l’autorité pour faire ce qu’elle a à faire ».[[3]](#footnote-3)

Scrum définit trois rôles qui sont :

**Le Product Owner (le propriétaire du produit)**: c’est une personne qui porte la vision du produit à réaliser, généralement c’est un expert dans le domaine.

**Le Scrum Master (le directeur de produit)** : c'est la personne qui doit assurer le bon déroulement des différents sprints du release, et qui doit impérativement maitriser Scrum.

**Le Scrum Team (l’équipe de Scrum) :** constitué des personnes qui seront chargées d’implémenter les différents besoins du client. Bien évidemment, cette équipe sera constituée des développeurs, des testeurs, etc.

Dans le contexte de notre projet, M. Oussama QAROUI sera le propriétaire et M. Zakaria DAHOUNE le directeur de produit et nous formons Zainaba ABOUMOUSSA et Rizki AKERKAOU les deux membres de l’équipe Scrum.

## **Conclusion**

Dans ce chapitre nous avons présenté le cadre général de notre projet en déterminant la problématique et en proposant la solution envisagée pour faire face à la situation courante. Nous avons dévoilé le langage et la méthodologie de conception qui seront utilisés dans les prochains chapitres de ce rapport et nous avons argumenté notre choix.

# Chapitre 2 : Étude Préliminaire

Après la définition du contexte général de notre projet, l'objectif de ce chapitre est de construire une bonne vision du produit, d’identifier les rôles des utilisateurs et de dégager les fonctionnalités principales afin de produire le backlog initial ainsi qu'une première planification des sprints

## **Identification des acteurs**

Un acteur représente l’abstraction d’un rôle joué par des entités qui interagissent directement avec le système étudié. Dans cette partie, nous allons énumérer les acteurs susceptibles d’interagir avec le système en les assignant aux fonctionnalités du système. On distingue trois acteurs pour cette application :

* Administrateur : Gère la plateforme, gère les utilisateurs et leurs rôles et gère les critères des profils.
* Recruteur : Gère les recrutements et les départs des salariés.
* Responsible: Gère les profils.

### Diagramme de contexte :

Ce diagramme d’UML, présenté dans la figure4, permet simplement de montrer la relation des différents acteurs avec le système.



Figure :Diagramme de contexte de l'application

## **Spécification des besoins**

### Les besoins fonctionnels

En se basant sur l’étude de l’existant et sur les différentes réunions effectuées, notre application intègre les modules suivants :

* Authentification : L’authentification se fait par login et mot de passe, Chaque compte a un ou plusieurs rôles pour gérer l’application et ses volets.
* Module de recrutement: Le recrutement des profils se fait facilement à l’aide d’une interface qui regroupe les informations les plus intéressantes:
  + Type du profil: Salarié, Intérim, Stagiaire…,
  + Type du contrat: ANAPEC, CDD, CDI…,
  + Secteur d’activité (Informatique, Finance...),
  + Nom et prénom,
  + Date de naissance ou l’âge,
  + Civilité,
  + Situation familiale,
  + Permis de conduire,
  + Adresse (Pays, Ville, Adresse long format),
  + RIB,
  + Informations sociales (CIN, CIMR, CNSS),
  + Poste actuel,
  + Poste désiré,
  + Expérience,
  + Compétences,
  + Langues.
* Module d’importation des données :  l’importation des données peut se faire soit par un fichier CSV soit par une connexion directe aux bases de données où se situe ces données.
* Module de création de CV automatique : Sur chaque fiche de salarié/profil, la possibilité de générer automatiquement un CV sous format PDF est possible, avec les critères et le format désiré. Ce document est téléchargé lors du clic sur le bouton de génération automatique.
* Module de gestion des fiches salariés/profils : ce module est prévu pour gérer les profils existants. Les tâches que l’on peut effectuer sont: Recherche par critère (simplifiée), Modification, Suppression, Départ (avec le type et motif du départ), Changement de poste.
* Module de recherche avancé: Une recherche avancée peut être réalisée en saisissant des critères de sélection bien précis, ce qui permet à l’utilisateur d’avoir les profils/CV qui répondent exactement à son besoin. Cette recherche englobe tous les critères que l’utilisateur souhaite utiliser pour filtrer les profils dont il dispose, afin d’avoir comme résultat les bons profils que ce soit pour un changement de poste, un départ ou un recrutement.
* Module de reporting: Le reporting à réaliser englobe les périmètres suivants:
  + Suivi des entrants et sortants par période,
  + Reporting par poste,
  + Reporting par domaine d’activité,
  + Reporting par type de contrat...

### Les besoins non fonctionnels

Cette partie regroupe les différents besoins non fonctionnels et les contraintes liés à l’implémentation, afin de réaliser une solution adéquate qui répond le mieux aux attentes du client.

Ces besoins sont les suivant :

* **La facilitation d’utilisation** : L’interface de l’application doit être simple et intuitive afin que l’utilisateur puisse l’exploiter sans se référer à des connaissances particulières, en d’autres termes, notre application doit être facile à manipuler par n’importe quel utilisateur.
* **L’évolutivité** : C’est-à-dire qu’il doit y avoir une possibilité d’ajouter de nouvelles fonctionnalités ou de modifier celles existantes dans l’application tout en maintenant ses performances en cas de forte demande.
* **La sécurité** : L’application doit être sécurisée. D’où la nécessité de procéder à l’authentification des agents et administrateurs tout en assurant la confidentialité de leurs données.
* **La rapidité et intégrabilité dans d’autres applications** : L’application doit être rapide et assure que les modules seront intégrables et utilisables dans d’autres applications.

En plus des contraintes techniques suivantes :

* Le langage de programmation côté serveur sera Java/JEE en utilisant un Framework Spring MVC,
* Le Système de Gestion de la Base Des données devra être SqlServer,
* Le framework côté client sera Angular,
* La solution sera hébergée dans un serveur dédié propre à l’entreprise.

## **Le backlog du produit :**

Le backlog du produit est l’artefact le plus important de Scrum, c’est l’ensemble des caractéristiques fonctionnelles ou techniques qui constituent le produit souhaité. Les caractéristiques fonctionnelles sont appelées des « histoires utilisateur » (user story) et les caractéristiques techniques sont appelées des « histoires techniques » (technical story).

Le tableau ci-dessous résume le backlog produit de notre application. Il est à noter que nous n’avons pas cité les histoires techniques comme la préparation de la maquette graphique, les travaux de conception et les jeux de tests, etc.

Dans le tableau suivant, chaque histoire utilisateur est caractérisée par sa priorité expliquée dans la section 2 de ce même chapitre. Pour le traitement de nos « histoires utilisateur » nous faisons le choix de commencer avec les cas d’utilisation les plus prioritaires et plus importante.

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | En tant que | Je veux que’ | Pour |
| Authentification | Utilisateurs | Il soit capable de se connecter à l’application grâce à un login et mot de passe. | Avoir accès à l’application selon le rôle. |
| Gestion Recrutement | Recruteur | Il soit possible de sélectionner un poste et un profil | Affecter les profils au postes disponibles |
| Il soit possible de sélectionner un profil | Affecter un nouveau poste au profil sélectionné |
| Enregistrer un départ du poste |
| Gestion CVthèque | Recruteur | Il soit possible de créer un CV et modifier sa mise en forme et son contenu | Télécharger le CV et l’enregistrer au choix |
| Il soit possible de consulter les CVs |  |
| Gestion Salariés et profils | Responsable | Il soit possible d’ajouter, modifier et supprimer un profil | Mettre à jour les profils |
| Il soit possible de charger des profils existants à partir d’un fichier CSV | Alimenter la base de données |
| Recruteur | Il soit possible de consulter les profils |  |
| Recherche avancée | Recruteur | Il soit possible de personnaliser la recherche en choisissant les critères de recherche | Filtrer les profils |
| Reporting | Responsable | Il soit possible de suivre les entrants et les sortants par période | Réaliser des statistiques |
| Paramétrage | Admin | Il soit possible  D’ajouter supprimer et modifier les critères du profils | Définir un profil |
| Il soit possible de mettre à ajour les formats de Cvs | Mettre à disposition plusieurs modèles |
| Il soit possible de définir les rôles | Les affecter aux utilisateurs |
| Gestion utilisateurs | Il soit possible de modifier, ajouter supprimer utilisateurs | Mettre à jour utilisateur |

Tableau  : Prodact Backlog

## **Planification des sprints**

Pour réaliser l’ensemble des fonctionnalités cité auparavant, Une réunion de planification des sprints est réalisée.

La réunion de planification des sprints est l’événement le plus important dans Scrum. Pour ce projet nous avons choisi de développer deux release. Le tableau suivant résume notre planning de travail opté pour notre release planning.

|  |  |  |  |
| --- | --- | --- | --- |
| Plan du Release 1 | | Plan de release 2 | |
| Sprint 1 : Préparation du projet | **Sprint 2 : Paramétrage et gestion utilisateurs** | **Sprint 3 : gestion profils et CVthèque** | **Sprint 4 : gestion de recrutement** |
| Préparation de l’environnements  Authentification  Création de la Base de données | **Paramétrage**  **Gestion utilisateurs** | **Gestion Salariés et profils**  **Gestion CVthèque** | **Gestion recrutement**  **Recherche avancée**  **Reporting** |
| DE 22/04 au 3/05 | **DE 06/05 au 10/05** | **DE 13/05 au 31/05** | **DE 03/06 au 21/6** |

Tableau : Planification des sprints

## **Modélisation des besoins fonctionnels**

### Diagramme de cas d’utilisations global :

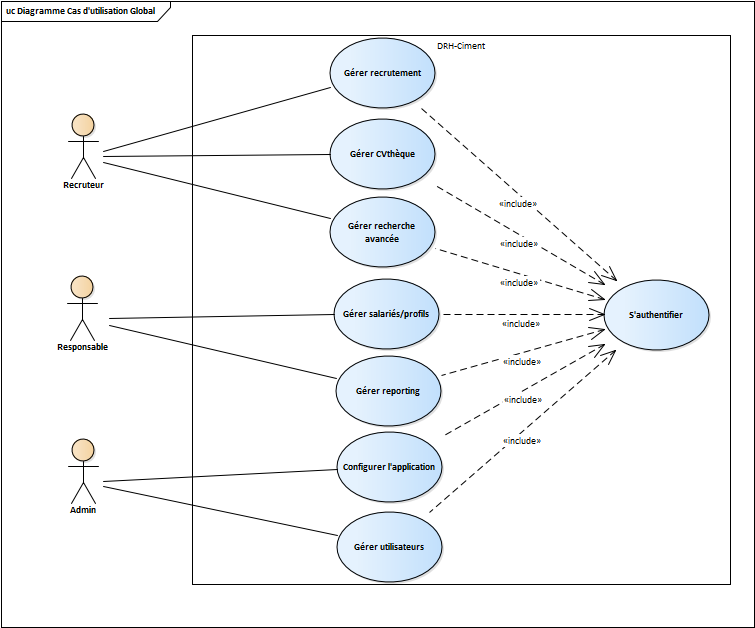


Figure :Diagramme de cas d'utilisations global

### Diagramme de classe global :

La table User regroupe les informations d’authentification à l’application. Il est lié à un profil, un profil peut ne pas avoir d’user, un User peut avoir différents rôles.

La table profil regroupe l’ensemble d’informations les plus intéressantes des profils (il peut être un salarié interne à l’entreprise ou une personne externe) à savoir : Nom et prénom, Date de naissance, Civilité, Permis de conduire, Adresse (Pays, Ville, Adresse long format), RIB, Informations sociales (CIN, CIMR, CNSS).

Un profil peut avoir une ou plusieurs Formation, peut être lier à plusieurs sociétés dans des dates différentes et peut avoir un ou plusieurs Cv.

Un cv concerne un et un seul profil et il est lié dans différentes dates à plusieurs : Expérience, Situation familiale, Compétences, Langues, Type du profil (Salarié, Intérim, Stagiaire…), Type du contrat (ANAPEC, CDD, CDI…), Secteur d’activité (Informatique, Finance...).

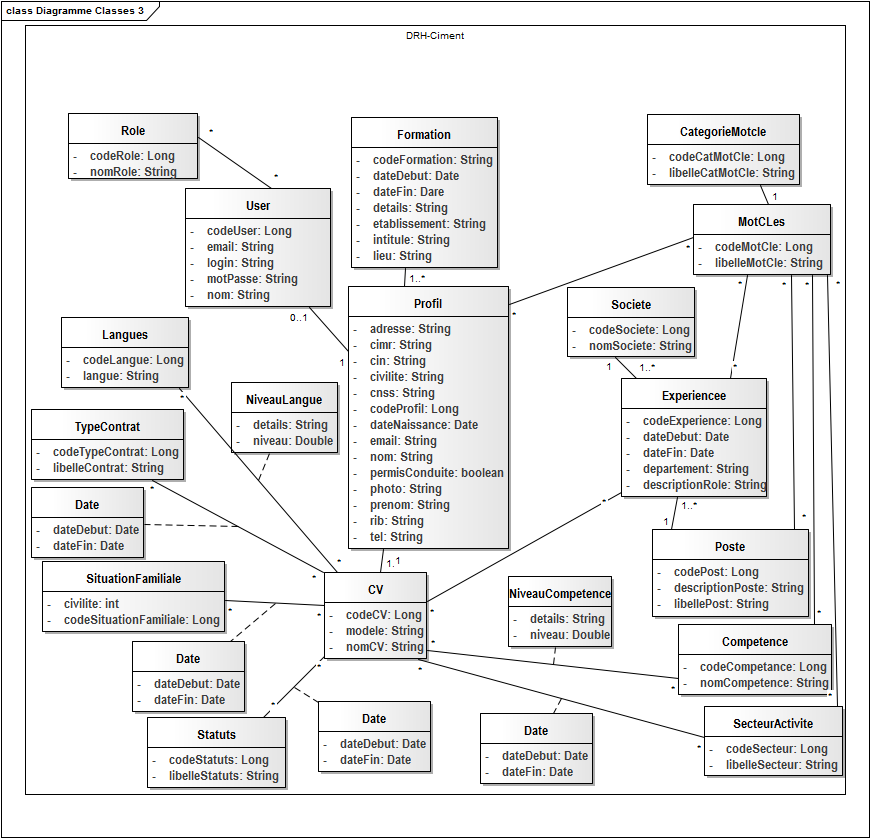


Figure : Diagramme de classe global

## Conclusion

Dans ce chapitre nous avons préparé notre plan de travail. Nous avons capturé les besoins fonctionnels de notre application, les rôles des utilisateurs, par la suite nous avons préparé le plan de release de notre projet.

# Chapitre 3 : Étude technique.

Cette phase est souvent appelée sprint de stabilisation. Les tâches effectuées pendant cette phase ne sont pas claires, et elles dépendent fortement du type de déploiement du logiciel (mise en production à chaud, packaging du produit, mise à disposition par téléchargement en ligne...).

Dans cette partie, nous allons décrire l’architecture technique et logicielle pour la mise en place de l’application. Ce chapitre élabore l’ensemble des éléments de l’application.

## **Architecture logicielle**

L’architecture logicielle s’intéresse au découpage logique de l’application et la façon de regrouper les composants selon le type de fonction et traitements qu’ils effectuent. Pour notre solution nous avons opté pour une architecture multicouche.

Couche présentation (Client) : correspond à la partie de l’application visible et interactive avec les utilisateurs. Elle communique avec la couche métier par les web services et le protocole HTTP.

Couche métier (Business Layer) : correspond à la partie fonctionnelle de l’application, qui est responsable de l’implémentation de la « logique », décrivant les opérations que l’application opère sur les données en fonction des requêtes des utilisateurs, effectuées au travers de la couche présentation.

Couche accès aux données (Data Layer) : chargée de l'accès aux données et de leur manipulation, indépendamment du SGBD choisi.

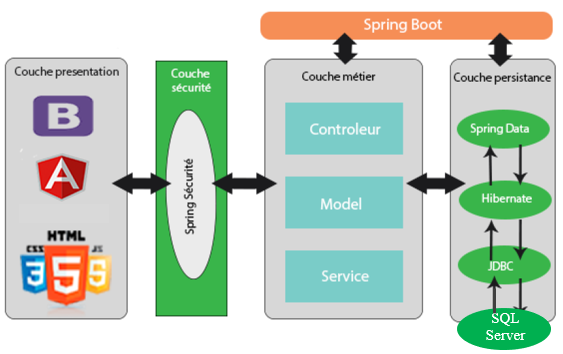


Figure : Architecture logicielle

## **Architecture physique**

Le diagramme de déploiement est utilisé pour décrire l’architecture physique d’un système. Il présente les périphériques utilisés et la répartition du système sur ces différents éléments. Il montre aussi les liens de communication entre ces diverses entités.

Notre système gère un seul type de client qui peut accéder à l’application par son navigateur web.

### *Diagramme de déploiement*

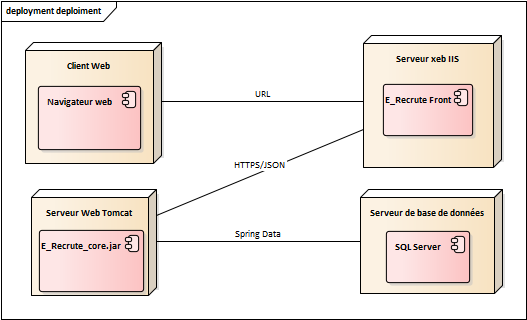


Figure : Diagramme de déploiement

Notre application est basée sur des API CRUD RESTFul à l'aide de JPA Spring Boot 2 et d'une base de données MS-SQL. Afin de développer Un micro service qui va être consommer par une application de front Angular. Cette application est sécurisée avec l'authentification JWT (JSON Web Token) et Spring Security.

* SGBD : Permet de stocker les données de l’application.
* Serveur d’application : Permet de gérer le cycle de vie des applications, fournie l’infrastructure nécessaire pour faire fonctionner les applications de bonnes qualités et offre un Framework d’Inversion de Contrôle (Spring IOC).
* Couche Métier : Permet d’implémenter la logique métier de l’application en utilisant une approche orienté objet.
* Couche DAO : couche technique qui représente la couche d’accès aux données de l’application, utilise Hibernate le Framework de Mapping Objet Relationnel implémentant la spécification JPA
* Couche Web Service REST full : Permet de définir un web service qui permet à l’application de front développée avec Angular de faire appel à distance, aux fonctionnalités de l’application, le service REST recoit des requêtes HTTP, exécute des traitements et renvoie le résultat au client avec le format JSON, pour l’implémenter en utilise la spécification JAXRS.

### *Diagramme de composants*

Un diagramme de composants montre les composants, les interfaces fournies et les relations entre eux. Les développeurs trouvent que le diagramme de composant utile car il leur offre une vue architecturale, de niveau supérieur du système. Ce diagramme permet de mettre en évidence les dépendances entre les composants.

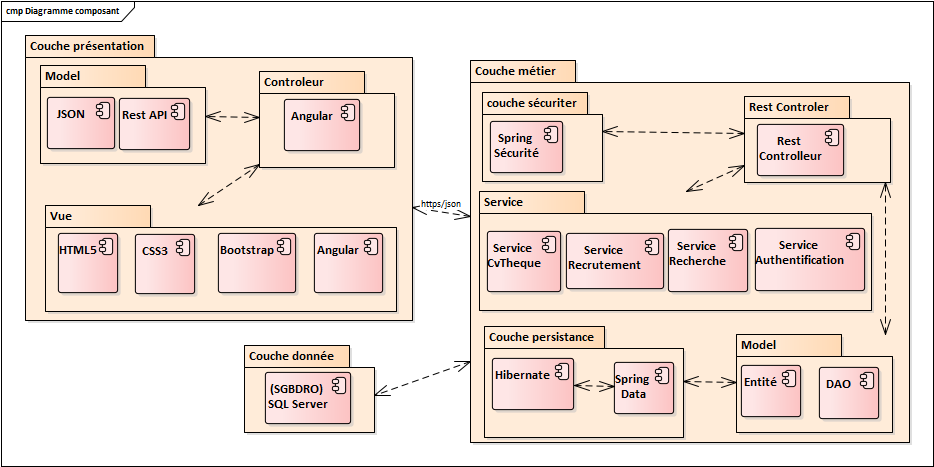


Figure : Diagramme de composants

## **Environnements de travail**

Dans cette partie, nous présenterons l'environnement matériel et technique relatif à la réalisation du projet.

### *Environnement matériel*

Durant la réalisation du présent travail, on dispose de deux ordinateurs DELL et HP embarquant un système d'exploitation Windows 10 et ayant les caractéristiques suivantes :

* **DELL :**
  + Un processeur Intel® Core i5-2520M CPU @ 2.50GHz 2.50 GHz
  + Une mémoire vive de 4,0 GO
  + Une carte graphique Intel® HD Graphics 3000
  + Un disque dur de 250 GO
* **HP :**
  + Un processeur Intel® Celeron CPU N3060 @ 1.60GHz 1.60 GHz
  + Une mémoire vive de 4,0 GO
  + Une carte graphique ATI Mobility Radeon HD 5000 Series
  + Un disque dur de 250GO

### *Environnement technique*

Dans cette partie, nous allons présenter les différents outils et langages de programmation que nous avons utilisé pour la mise en place de notre système.

* **JPA (Java Persistence API)**, est un standard de la plateforme Java EE gérant la persistance des données. Il permet la gestion de correspondance entre des objets Java (entity) et une base de données.
* **Spring Boot**, Spring Boot est un nouveau Framework créé pour simplifier le démarrage et le développement des nouvelles applications Spring. Le Framework propose une approche dogmatique de la configuration, qui permet d'éviter aux développeurs de redéfinir la même configuration à plusieurs endroits du code.
* **Java 8**, est la dernière version de Java et offre notamment la possibilité d’utiliser des lambda expressions, une nouvelle API de gestion de dates et heures ainsi que l'utilisation des “Optionals” qui permettent une meilleure gestion des références null.
* **Angular 7**, est un cadriciel côté client open source basé sur TypeScript dirigée par l'équipe du projet Angular à Google et par une communauté de particuliers et de sociétés. Angular est une réécriture complète de AngularJS, cadriciel construit par la même équipe.
* **Bootstrap,** est une collection d’outils utiles à la création 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.

## **Conclusion**

Au cours de ce chapitre nous avons présenté l'architecture logique et physique de notre application, ainsi que l'environnement de travail.

Dans le chapitre suivant, nous allons présenter les itérations suivies lors de la réalisation de la première release du système.

# Chapitre 4 : Release 1 : Authentification, Paramétrage et Gestion des utilisateurs.

Le terme release peut-être défini comme une version distribuée d'une application ou une période de temps qui permet de la produire. Peu importe quelle définition nous utilisons, une release est constituée d'une suite d'itérations (sprint) qui se terminent quand les incréments de ces derniers construisent un produit présentant suffisamment de valeur aux utilisateurs finaux.

Tout au long de ce chapitre, nous allons traiter les histoires utilisateurs de nos sprints pour produire un incrément potentiellement livrable

## **Sprint 1**

Le sprint est le cœur de Scrum. Il s’agit d’un bloc de temps durant lequel un incrément du produit sera réalisé. Tous les sprints d’une release ont une durée constante et ne se chevauchent jamais, c'est-à-dire qu’un sprint ne peut pas démarrer tant que le précédent n’est pas encore terminé.

Avant de se lancer dans un sprint, l’équipe Scrum doit obligatoirement définir le but de ce dernier. Il s’agit de répondre à une question fondamentale « pourquoi faisons-nous ce sprint ? ». Suite à une étude sur la division du projet, nous avons décidé le but suivant : **« Préparer l’environnement et les fonctionnalités qui sont liées à l’authentification. »**.

### Backlog du sprint 1 *:*

Une fois, nous avons défini le but de notre sprint, il est temps de décider quelles sont les fonctionnalités et leurs degrés d’importance incluent dans ce dernier. Plus précisément, quels noms de notre backlog du produit seront inclus dans le backlog du sprint. Ce qui le résume donc le Tableau 3 ci-dessous, backlog de notre premier sprint :

|  |  |
| --- | --- |
| Histoire Utilisateurs | Priorité |
| Se connecter | Élevée |
| S’inscrire | Élevée |
| Mot de passe oubliée | Faible |
| Se déconnecter | Moyenne |

Tableau : Backlog du premier sprint (RELEASE1)

### Diagramme de cas d’utilisation *:*

La spécification fonctionnelle dans notre cas se traduit par le diagramme des cas d’utilisation d’UML et la description textuelle de ces derniers.

Le schéma suivant synthétise le cas d’utilisation du sprint 1 :

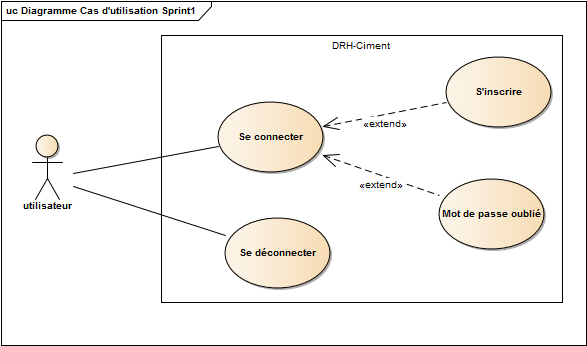


Figure Diagram de cas d'utilisation du sprint 1

Cette description décrit le cas d’utilisation « s’authentifier » :

|  |  |  |  |
| --- | --- | --- | --- |
| **Intitulé** |  | **Se connecter** | |
| Acteur principal |  | Utilisateur | |
| Objectif |  | Se connecter à l’application | |
|  | - | | L’application en ligne est fonctionnelle. |
| Pré-conditions | - | | L’utilisateur est déjà inscrit. |
| Scénario nominal |  | 1-L’utilisarteur remplit les champs d’authentification et clique sur le bouton « Connexion » ; | |
|  |  | 2-Le système vérifie les informations saisies par l’utilisateur et affiche l’interface suivante selon son rôle ; | |
| Post-condition |  | L’utilisateur est identifié. | |
| Exception |  | Si un des champs est invalide, le système affiche un message d’erreur. | |

Tableau  : Description textuelle du cas d'utilisation « Authentification »

### *Diagramme d’activités*

La figure 11 montre le diagramme d’activité d’authentification d’un utilisateur.

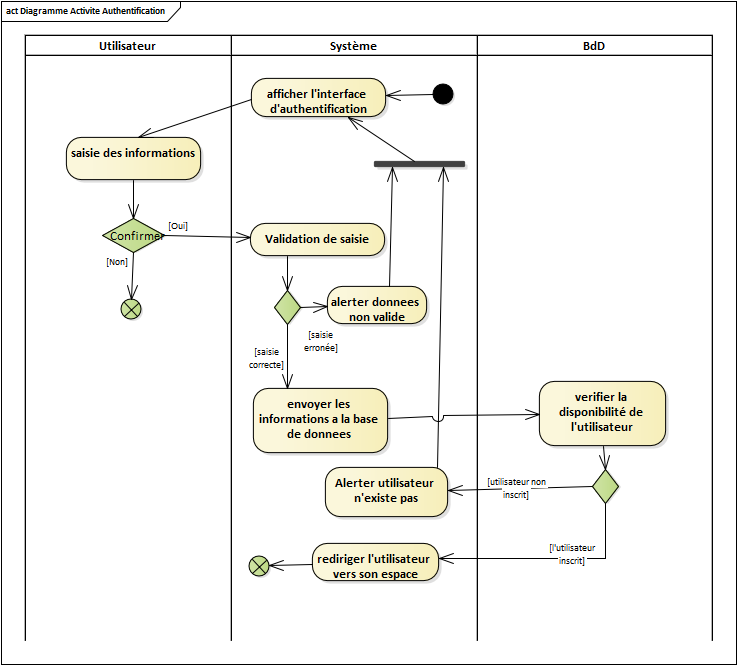


Figure Diagramme d'activité d'authentification

### Diagramme des classes

Le diagramme de classe est l’un des diagrammes statiques d'UML. Il permet de décrire la structure d'un système informatique tout en montrant les différentes classes, leurs attributs, leurs méthodes ainsi que les relations entre eux. Tout au long de nos sprints, nous essayerons de construire ce diagramme au fur et mesure en ajoutant les différentes classes déduites.

La figure ci-dessous illustre le diagramme de classe de conception de ce premier sprint :

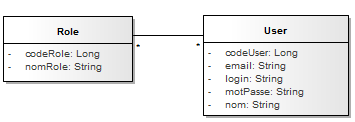


Figure : Diagramme de classe du sprint 1(RELEASE 1)

### Test et réalisation :

Le test est un processus manuel ou automatique, qui vise à établir qu’un système vérifie les propriétés exigées par sa spécification, ou à détecter des différences entre les résultats engendrés par le système et ceux qui sont attendus par la spécification. (Définition issue de la norme IEEE-STD729, 1983).

Les activités de test constituent un axe très important dans le cycle de développement d’un logiciel. Ils permettent de détecter les erreurs afin de les corriger et d’assurer la qualité du logiciel fourni.

Contrairement aux cycles de développement séquentiel[[4]](#footnote-4), avec la méthodologie agile, le test n'est pas une phase qui se déroule après la fin de développement. En effet, les tests seront intégrés dès le début du premier sprint jusqu’à la livraison du produit final. En outre, la qualité du logiciel n’est pas négligeable, c’est dans ce cadre que Scrum doit être complété par les bonnes pratiques d’ingénierie techniques du logiciel. Parmi ces pratiques[[5]](#footnote-5), seulement deux qui nous intéressent et qui est le pilotage par les tests centrés sur les tests unitaires et les interfaces de l’application.

La figure suivante illustre le cas de succès pour l’envoie d’une demande. Cette figure nous montre que notre méthode est bien fonctionnelle et que la demande a bien été envoyée.



#### Test unitaire :

Le principe de cette pratique est d’écrire les tests avant même d’écrire le code et de profiter par la suite de l’existence des tests automatiques pour l’amélioration et le remaniement du code. Cette technique permet aux programmeurs de rester simples au niveau du code et de s’assurer de son bon fonctionnement après des changements.

Les figures suivantes illustrent le cas de succès pour l’authentification. Cette figure nous montre que notre méthode est bien fonctionnelle et que l’authentification a réussi.

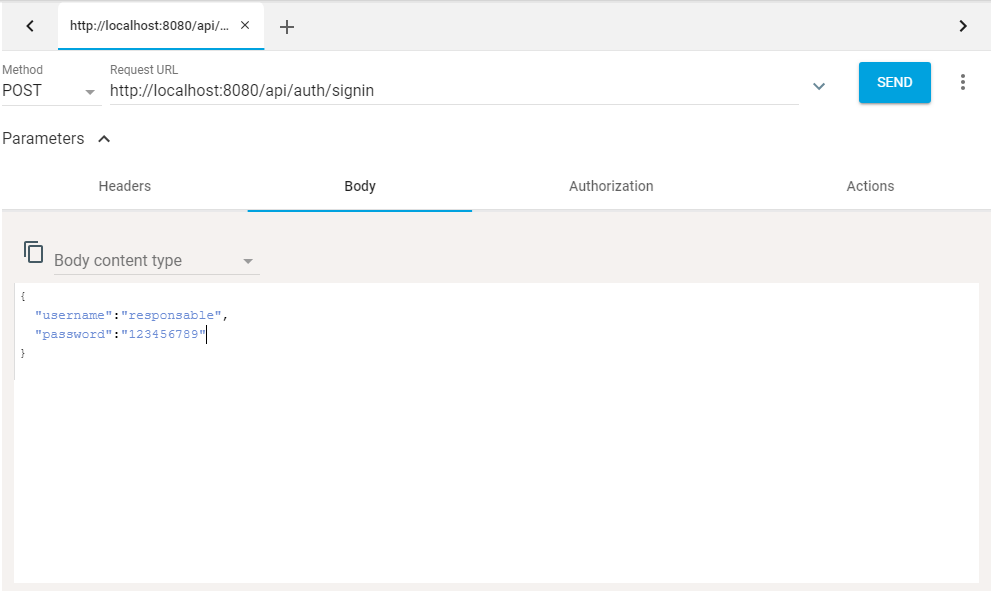


Figure : ARC capture login 1-1

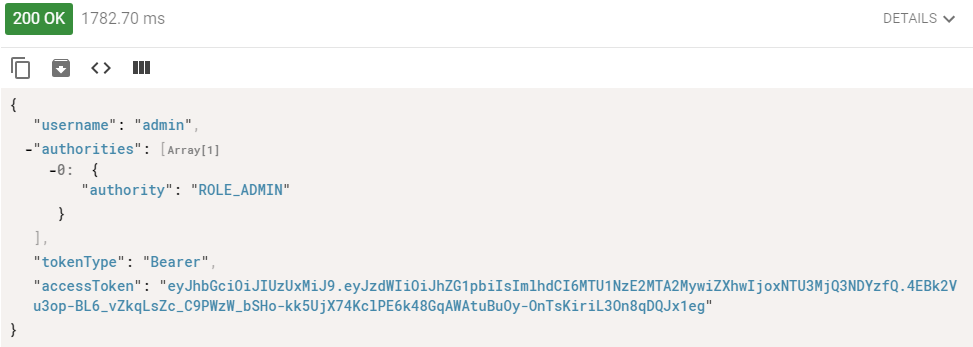


Figure : ARC capture login 1-2

À ce stade, nous exposons quelques interfaces de notre application à travers différents imprimes écrans réalisés. En essayant à chaque fois de décrire les différents objets interactifs mis à la disposition de l'utilisateur.

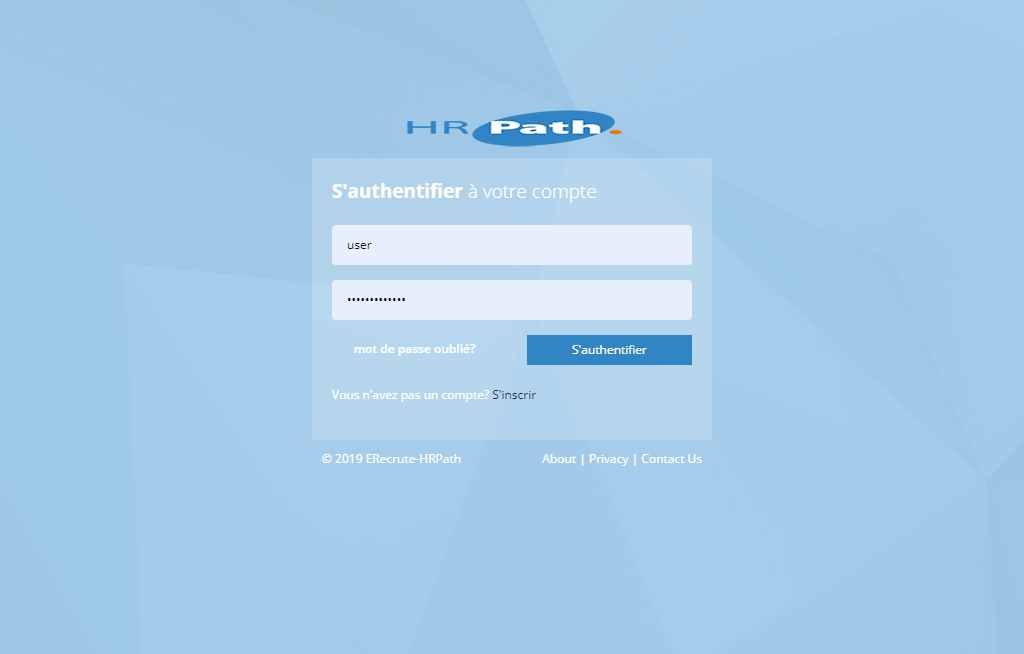


Figure : Interface authentification

La figure 12 présente l’interface où l’utilisateur doit saisir ses coordonnées pour accéder à l’application.

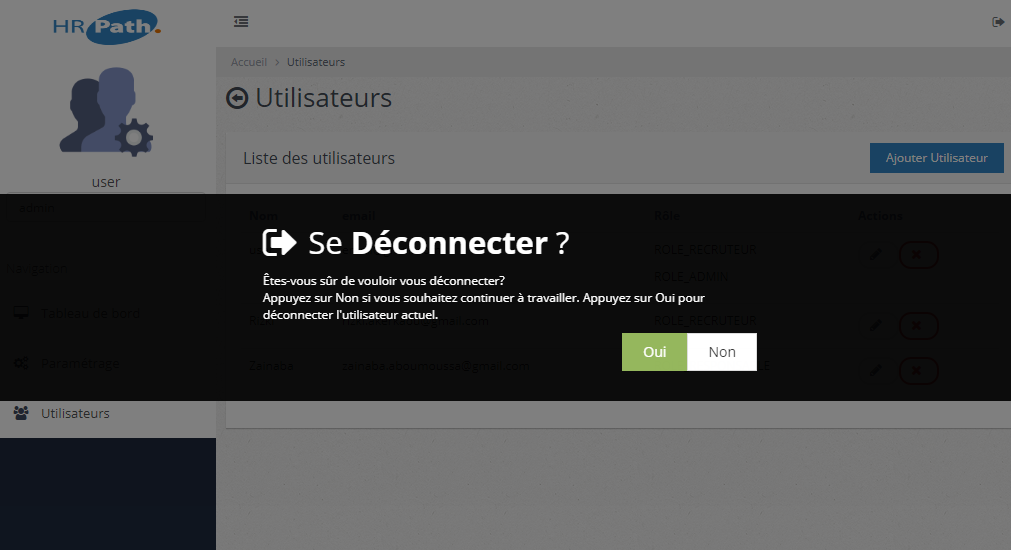


Figure :Interface de Logout

La figure 13 présente l’interface de déconnection de l’application.

## Sprint 2

En partant sur le même principe que le sprint précédent, nous commençons par définir le but de notre second sprint. Suite à une conversation entre le Product Owner et l’équipe Scrum, nous avons décidé le but suivant : « terminer la partie qui concerne paramétrage et gestion des utilisateurs ».

### Backlog du sprint 2 *:*

Une fois que le but de notre sprint est défini, il est temps de décider quelles histoires inclure dans ce dernier. Le tableau ci-dessous résume le backlog de notre second sprint.

|  |  |
| --- | --- |
| Histoire Utilisateurs | Priorité |
| Mise à jour critères | Moyenne |
| Affectation des rôles | Faible |
| Mise à jour utilisateur | Élevée |

Tableau : Backlog du deuxième sprint (RELEASE1)

### Diagramme de cas d’utilisation *:*

Pour la spécification fonctionnelle de ce sprint, nous commençons par le diagramme des cas d’utilisation global du second sprint, illustré dans la figure 13 ci-dessous, en respectant toujours le même principe que le sprint précédent, nous avons découpé certaines histoires en un ensemble de tâches.



Figure Diagram de cas d'utilisation du sprint 2

Nous allons maintenant décrire chacun des cas d’utilisation énumérés dans le paragraphe précédent en identifiant les acteurs et les différents scénarios possibles.

* Description du cas d’utilisation « Mise à jour critère »

|  |  |  |  |
| --- | --- | --- | --- |
| **Intitulé** |  | **Mise à jour critère** | |
| Acteur principal |  | Admin | |
| Objectif |  | Mettre à jour les critères du profil | |
|  | - | | L’application en ligne est fonctionnelle. |
| Pré-conditions | - | | L’admin est déjà authentifié. |
| Scénario nominal |  | 1-L’admin choisit "mise à jour critères profil" | |
|  |  | 2. le système affiche liste critères  3. l'admin choisit le critère à mettre à jour  4. système affiche formulaire relatif au critère  5. l'admin remplie formulaire et enregistre mise à jour | |
| Post-condition |  | Le critère est mis à jour | |
| Exception |  | 1. a. L'admin annule mise à jour 2. b. Le système le redirige vers sa page d'accueil | |

Tableau Description textuelle du cas d'utilisation « Mise à jour critère »

* Description du cas d’utilisation « Ajout utilisateur»

|  |  |  |  |
| --- | --- | --- | --- |
| **Intitulé** |  | **Ajout utilisateur** | |
| Acteur principal |  | Admin | |
| Objectif |  | Ajout utilisateur | |
|  | - | | L’application en ligne est fonctionnelle. |
| Pré-conditions | - | | L’admin est déjà authentifié. |
| Scénario nominal |  | 1-L’admin choisit "Ajout User" | |
|  |  | 2. le système affiche formulaire d’ajout  5. l'admin remplie formulaire et enregistre User | |
| Post-condition |  | L’utilisateur est ajouté | |
| Exception |  | 1. a. L'admin annule l’ajout 2. b. Le système le redirige vers sa page d'accueil | |

Tableau  : Description textuelle du cas d'utilisation « Ajout utilisateur »

### *Diagramme d’activités*

Nous effectuons dans ce qui suit ( figure 18 et figure 19) le diagramme d’activités de certains cas d’utilisation déjà expliqués dans la section précédente.



Figure Diagramme d'activité Ajout Utilisateurs



Figure Diagramme d'activité définition rôle

### Diagramme des classes

Après tout le travail de spécification et de conception, nous pouvons maintenant construire le nouvel incrément de notre diagramme des classes en ajoutant les différents éléments (classes, associations, attributs, etc.) déduits à partir des activités précédentes.

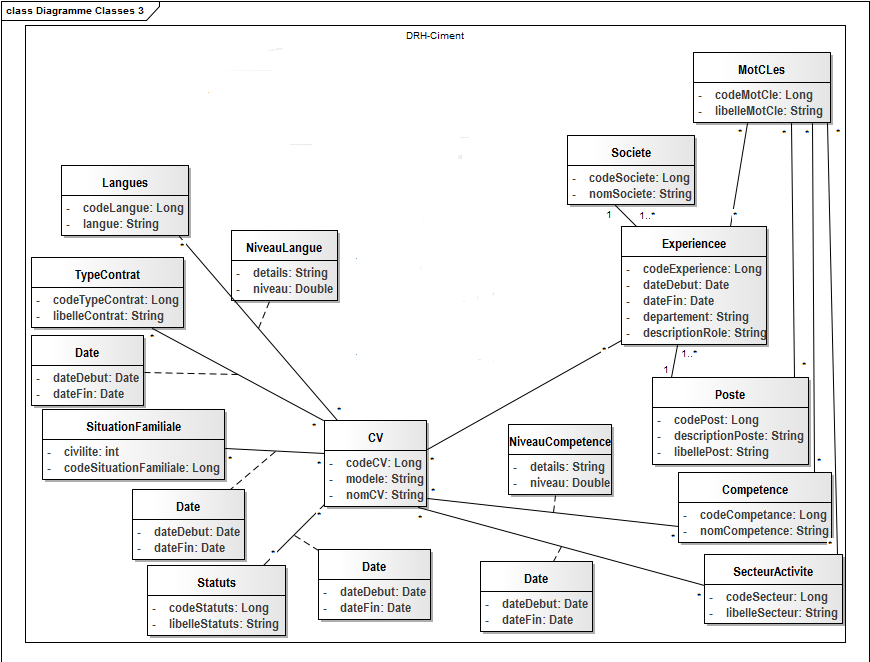


Figure : Diagramme de classe du sprint 2

### Réalisation :

La figure 17 illustre l’interface d’ajout d’un nouvel utilisateur.

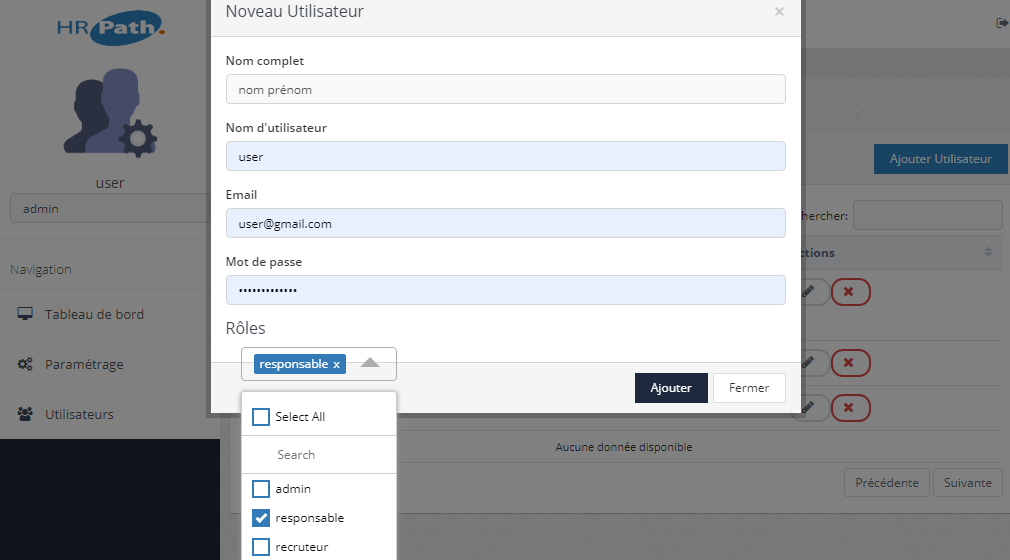


Figure : Nouveau utilisateur

La figure 18 illustre l’interface d’affichage et de modification des utilisateurs.

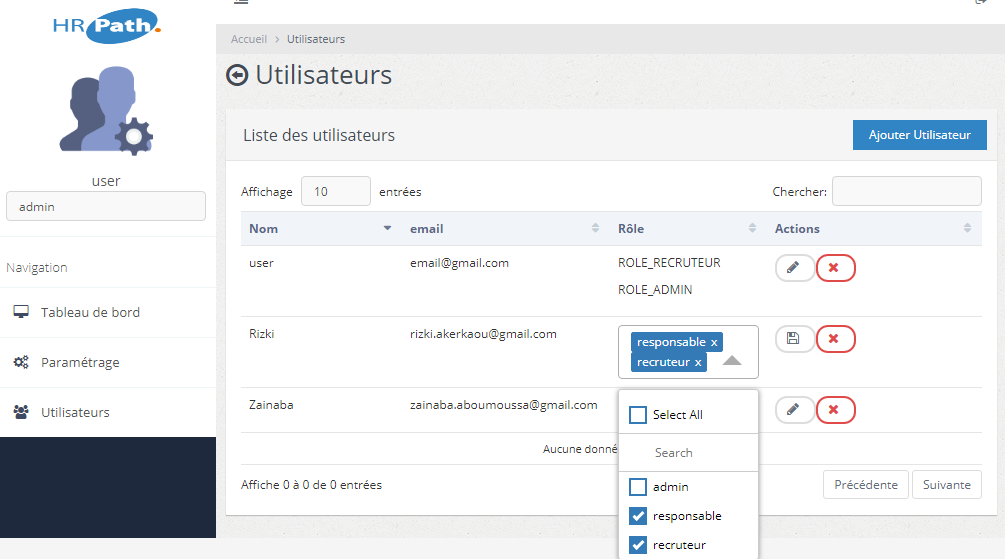


Figure : Affichage et modification utilisateurs

La figure 19 illustre l’interface de gestion des critères de recrutement.

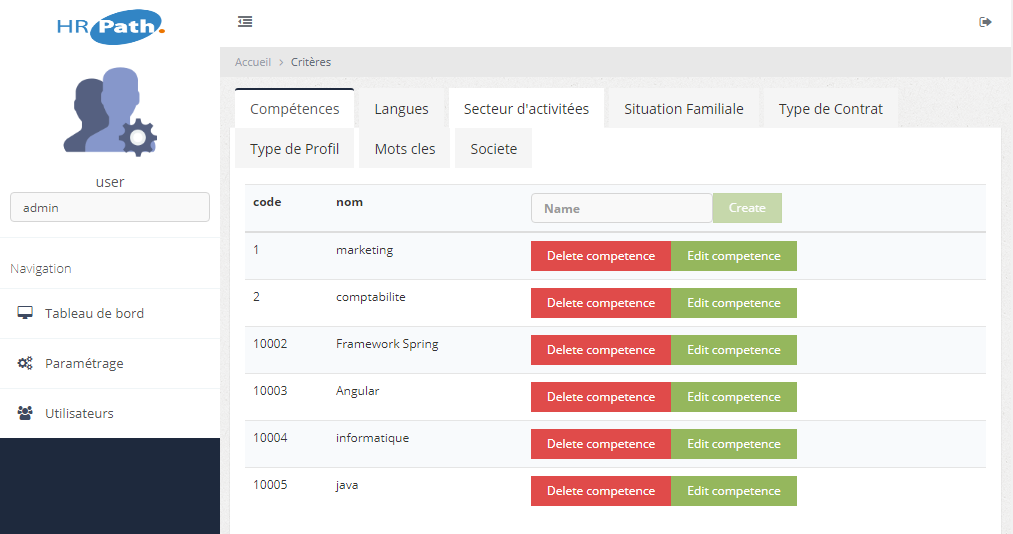


Figure : Gestion critères

# Chapitre 5 : Release 2 : gestion de recrutement et CVthèque.

Après avoir terminé le premier release de notre système informatique, nous pouvons maintenant nous lancer dans les travaux nécessaires pour produire le second release qui concerne les différentes fonctionnalités de recrutement et CVthèque. En effet les méthodologies agiles, et Scrum en particulier, sont caractérisées par un rythme régulier. Tout au long du chapitre suivant, nous aurons deux sprints ayant une durée plus importante que les sprints précédents, et nous allons traiter les histoires utilisateurs de ces derniers pour avoir à la fin de cet release le logiciel complet, livrable et fonctionnel.

## **Sprint 1**

En partant sur le même principe que les sprints précédents, nous commençons par définir le but de notre premier sprint pour ce release. Suite à une conversation entre le Product Owner et l’équipe Scrum, nous avons décidé le but suivant : « **terminer la partie qui concerne la gestion des profils et CVthèques** ».

### Backlog du sprint 1 *:*

Une fois, nous avons défini le but de notre sprint, il est temps de décider quelles histoires inclure dans ce dernier. Le tableau 8 résume donc le backlog de notre sprint :

|  |  |
| --- | --- |
| Histoire Utilisateurs | Priorité |
| Mise à jour de CV | Élevée |
| Mise à jour de profils | Élevée |
| Chargement en masse des profils | Moyenne |
| Télécharement de CV | Élevée |
| Consulter profils | Élevée |
| Consulter CVs | Élevée |

Tableau :Backlog du premier sprint (RELEASE2)

### Diagramme de cas d’utilisation *:*

Dans la figure 20 nous illustrons le diagramme des cas d’utilisation raffiné pour ce premier sprint.

En respectant toujours le même principe que les sprints précédents, nous avons découpé certaines histoires utilisateurs en un ensemble des tâches.

Nous allons maintenant décrire chacun des cas d'utilisation énuméré dans le paragraphe précédent en identifiant les acteurs et les différents scénarios possibles.

Chaque tableau présente la description détaillée du fonctionnement du cas d’utilisation. Dans ce Sprint nous allons nous intéresser aux deux cas d’utilisations suivantes :

Dans ce sprint, il existe des cas d’utilisations qui ont la même fonctionnalité tels que « Supprimer Profil », « Supprimer CV », …

Dans ce qui suit, nous présentons la description de quelques cas d’utilisation.

* Description du cas d’utilisation « Créer CV automatique»

|  |  |  |  |
| --- | --- | --- | --- |
| **Intitulé** |  | **Créer CV automatique** | |
| Acteur principal |  | Recruteur | |
| Objectif |  | Créer un cv automatiquement | |
|  | - | | L’application en ligne est fonctionnelle. |
| Pré-conditions |  | | - Le recruteur est déjà authentifié.  - Le recruteur a choisi un profile spécifique  - Système affiche formulaire relatif au critère  L'admin remplie formulaire et enregistre mise à jour |
| Scénario nominal |  | 1. Le recruteur choisit « Créer un nouveau CV » ; | |
|  |  | 2. Le système affiche le formulaire des informations relative au CV ;  3. Le recruteur rempli les informations nécessaires ;  4. Le système affiche des informations du profil ;  5. Le recruteur choisi les informations du profil à générer dans le CV ;  6. Le recruteur choisi de voir le CV ou de l’enregistrer ;  7. Le système affiche le CV créé ; | |
| Post-condition |  | Création d’un CV relative à un profil. | |

Tableau Description textuelle du cas d'utilisation « Créer CV automatique »

* *Description du cas d’utilisation « Charger des profils CSV »*

|  |  |  |  |
| --- | --- | --- | --- |
| **Intitulé** |  | **Charger des profils CSV** | |
| Acteur principal |  | Responsable | |
| Objectif |  | Chargement en masse des profils vers la base de données | |
|  | - | | L’application en ligne est fonctionnelle. |
| Pré-conditions |  | | - Le responsable est déjà authentifié.  Le responsable possède une ou plusieurs listes de salariés CSV. |
| Scénario nominal |  | 1. Le responsable choisit « Importer des salariés » ; | |
|  |  | 2. Le système affiche le formulaire relatif à la première étape contenant l’ensemble des listes des salariés ;  3. Le responsable choisi le fichier des salariés à importer ;  4. Le responsable confirme l’importation ;  5. Le système insère les informations des salariés vers la base de données ;  6. Le système redirige le responsable vers son tableau de bord ; | |
| Post-condition |  | Création et insertion des salariés appartenant au fichier importé. | |
| Exception |  | En cas d’erreurs au niveau des données, le système annule l’importation du fichier des salariés et réaffiche le formulaire relatif à l’étape courante. | |

Tableau Description textuelle du cas d'utilisation « Charger des profils CSV »

Le schéma suivant synthétise le cas d’utilisation du sprint 1 Release 2 :

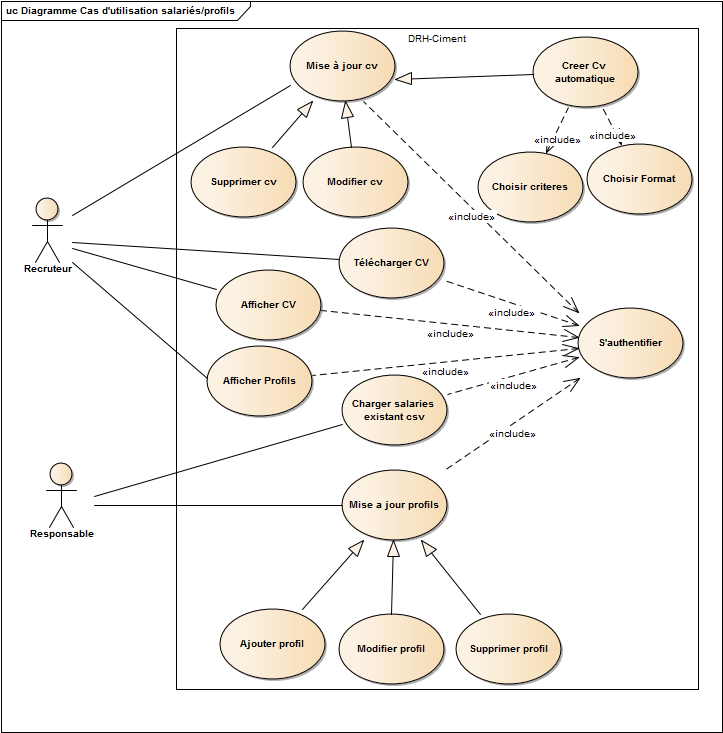


Figure Diagram de cas d'utilisation du sprint 1 Release 2

### *Diagramme d’activités*

Nous effectuons dans ce qui suit le diagramme d’activités de certains cas d’utilisation déjà expliqués dans la section précédente.

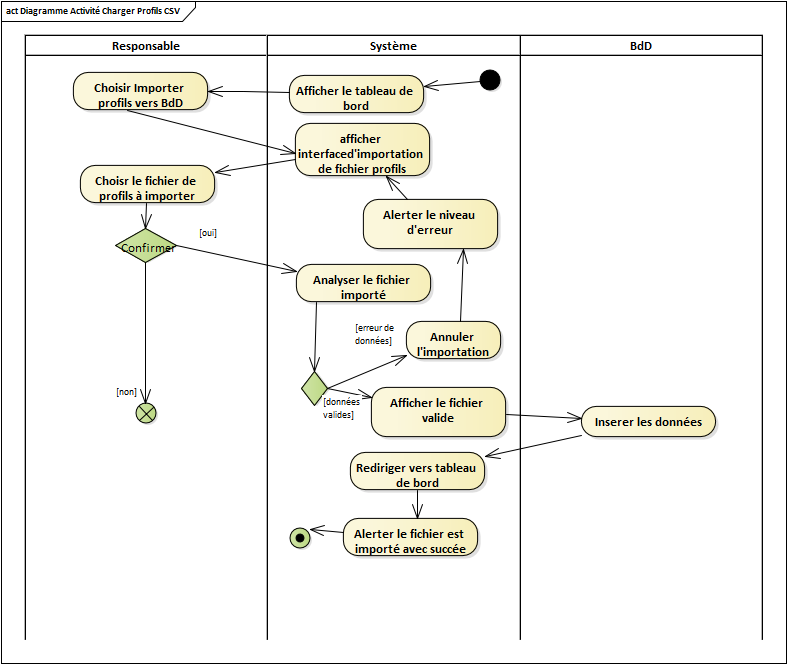


Figure :Diagramme Activité Charger Profils CSV

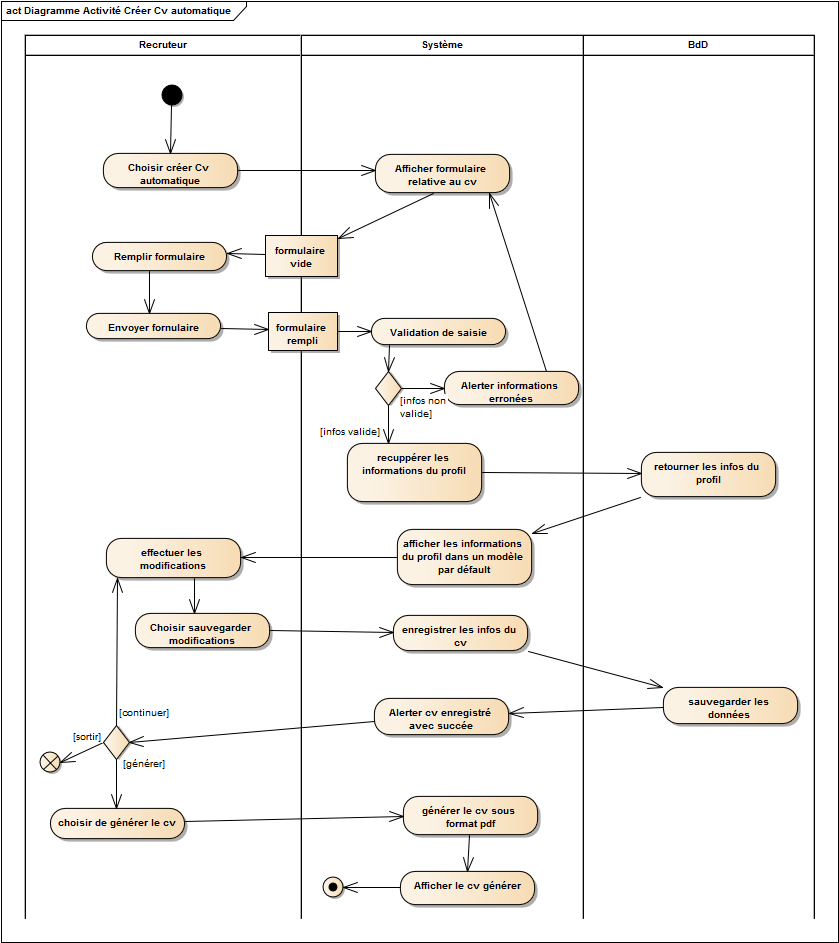


Figure Diagramme d'activité Créer CV automatique

### Diagramme des classes

Nous illustrons dans la figure le nouvel incrément de notre diagramme des classes en ajoutant les différents éléments (classes, associations, attributs, etc.) déduits à partir des activités précédentes.

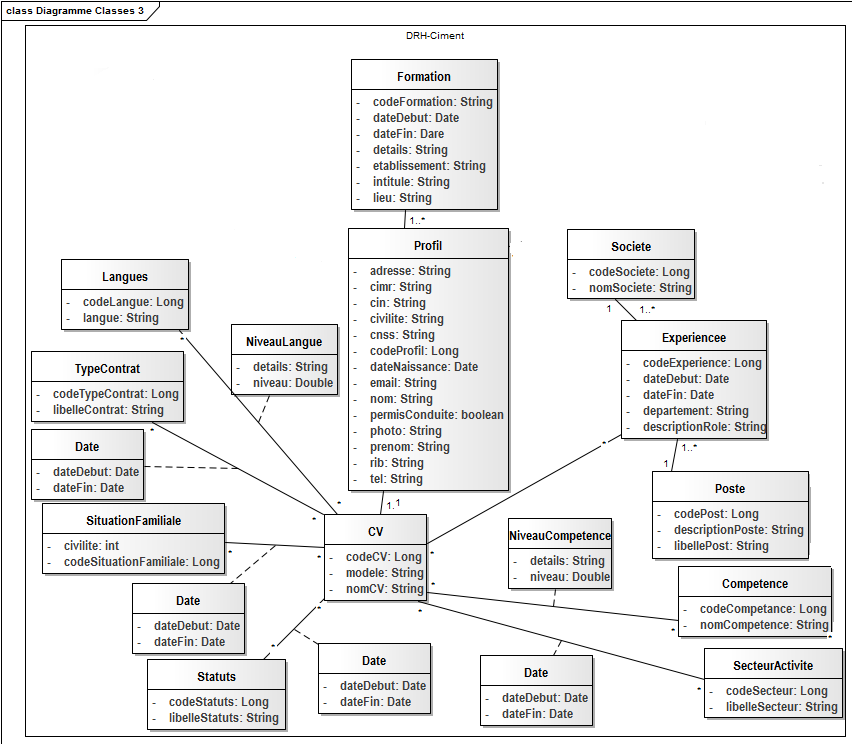
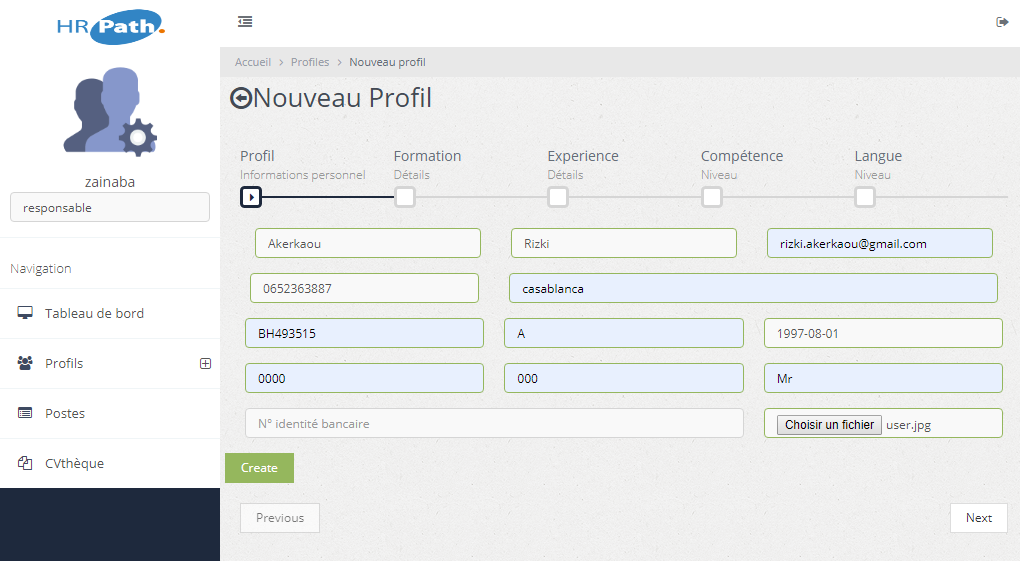


Figure : Diagramme de classe sprint 1 Release 2

### Réalisation :

La figure 28 illustre l’interface de création d’un nouveau profil.



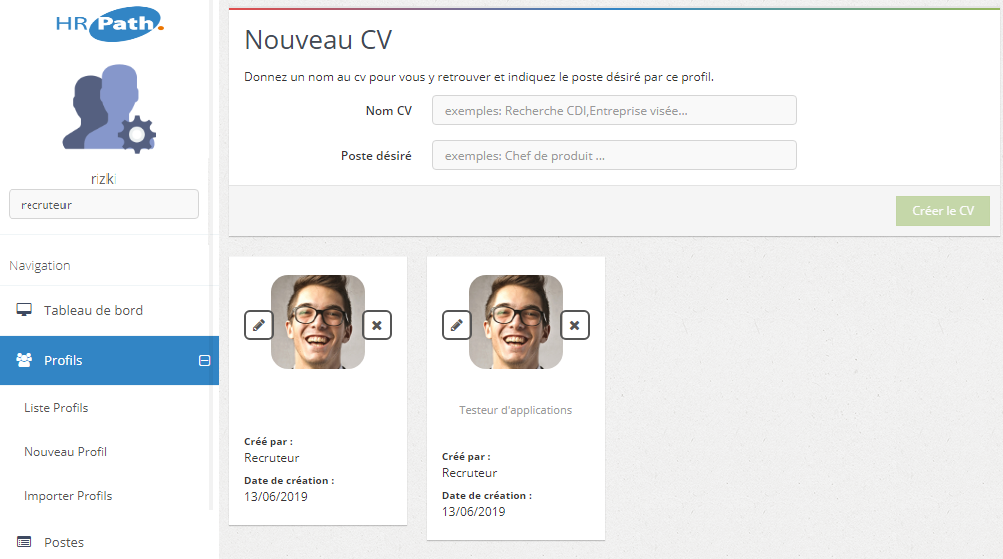
*Figure 28 : Nouveau Profil*

La figure 29 illustre l’interface de création de CV.



*Figure 29 : création CV*

La figure 30 illustre l’interface de CVthèque.



*Figure 30 : Ajout CV*

## **Sprint 2**

En partant sur le même principe que les sprints précédents, nous commençons par définir le but de notre premier sprint pour ce release. Suite à une conversation entre le Product Owner et l’équipe Scrum, nous avons décidé le but suivant : « **terminer la partie qui concerne la gestion du recrutement et Recherche avancée et Reporting** ».

### Backlog du sprint 2 *:*

Une fois, nous avons défini le but de notre sprint, il est temps de décider quelles histoires inclure dans ce dernier. Le tableau 11 résume donc le backlog de notre sprint :

|  |  |
| --- | --- |
| Histoire Utilisateurs | Priorité |
| Affecter les profils au poste disponibles | Élevée |
| Changer le poste d’un salarié | Élevée |
| Téléchargement de Rapport | Moyen |
| Consulter profils selon des critères détaillé | Élevé |
| Recherche avancée | Moyen |

Tableau Backlog du deuxième sprint (RELEASE2)

### *Diagramme de cas d’utilisation :*

Dans les figures suivantes nous illustrons les diagrammes des cas d’utilisation pour ce deuxième sprint.

En respectant toujours le même principe que les sprints précédents, nous avons découpé certaines histoires utilisateurs en un ensemble des tâches.

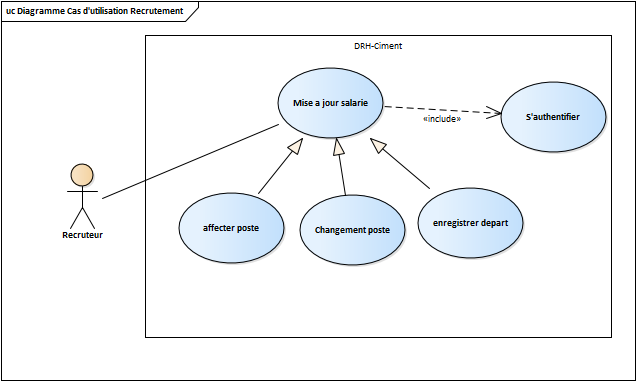


Figure : Diagramme Cas d'utilisation Recrutement

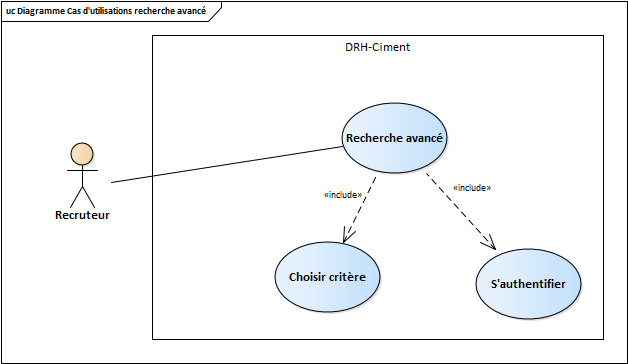


Figure : Diagramme Cas d'utilisations recherche avancé

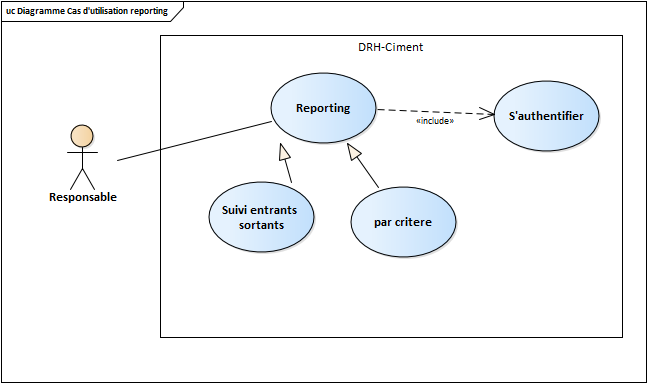


Figure : Diagramme de Cas d'utilisation reporting

Nous allons maintenant décrire chacun des cas d'utilisation énuméré dans le paragraphe précédent en identifiant les acteurs et les différents scénarios possibles.

### **Description du cas d’utilisation « Affecter poste»**

|  |  |  |  |
| --- | --- | --- | --- |
| **Intitulé** |  | **Affecter un poste à un profil** | |
| Acteur principal |  | Recruteur | |
| Objectif |  | Affecter un poste à un profil | |
|  | - | | L’application en ligne est fonctionnelle. |
| Pré-conditions | - | | - Le recruteur est déjà authentifié.  - Le recruteur a choisi un profile spécifique |
| Scénario nominal |  | 1. Le recruteur choisit « Affecter Poste » ; | |
|  |  | 2. Le système affiche la liste des postes disponnibles;  3. Le recruteur choisi le poste à affecter ;  4. Le système affiche le poste a été affecté ; | |
| Post-condition |  | Affectation d’un poste à un profil. | |

Tableau Description textuelle du cas d'utilisation « Affecter poste »

### **Description du cas d’utilisation** **« Recherche Avancée »**

|  |  |  |  |
| --- | --- | --- | --- |
| **Intitulé** |  | **Recherche avancée** | |
| Acteur principal |  | Recruteur | |
| Objectif |  | Afficher les profils selon des critères spécifiques | |
|  | - | | L’application en ligne est fonctionnelle. |
| Pré-conditions | - | | - Le recruteur est déjà authentifié. |
| Scénario nominal |  | 1. Le recruteur choisit « Recherche avancée » ; | |
|  |  | 2. Le système affiche l’ensemble des critères des profils ;  3. Le recruteur choisi les critères qu’ils veut ;  5. Le système affiche les profils avec les critères souhaités ; | |
| Post-condition |  | Affichage des profils selon les critères. | |

Tableau Description textuelle du cas d'utilisation « « Recherche Avancée»

### *Diagramme d’activités*

Nous effectuons dans ce qui suit le diagramme d’activités de certains cas d’utilisation déjà expliqués dans la section précédente.

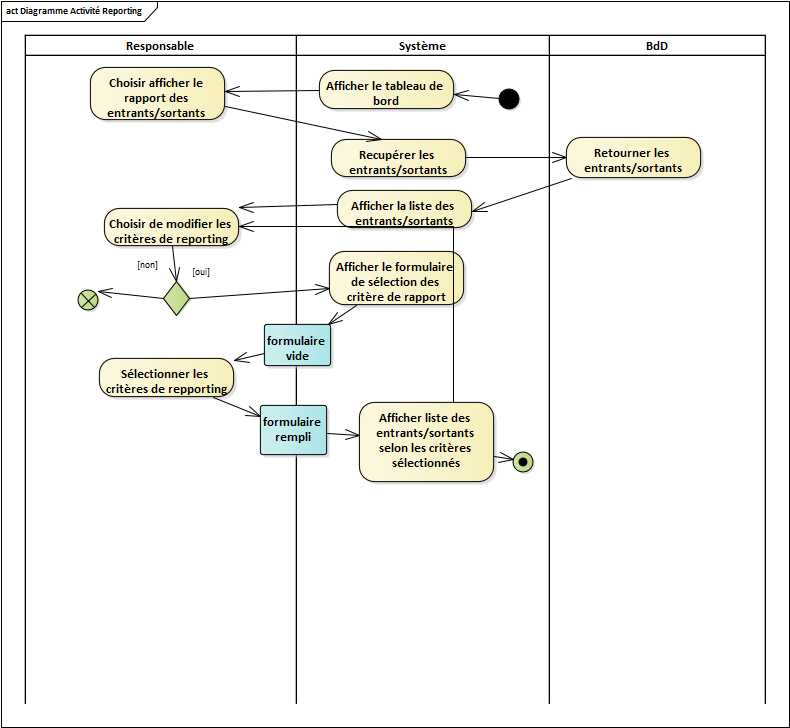


Figure 31 : Diagramme Activité Reporting

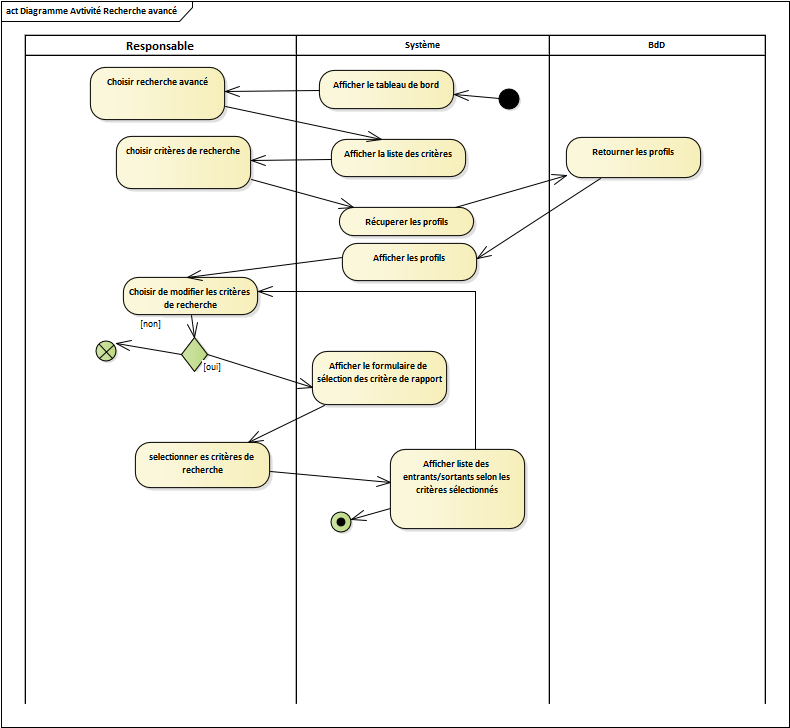


Figure 32 : Diagramme Activité Recherche avancé

### Diagramme des classes

Nous illustrons dans la figure le nouvel incrément de notre diagramme des classes en ajoutant les différents éléments (classes, associations, attributs, etc.) déduits à partir des activités précédentes.

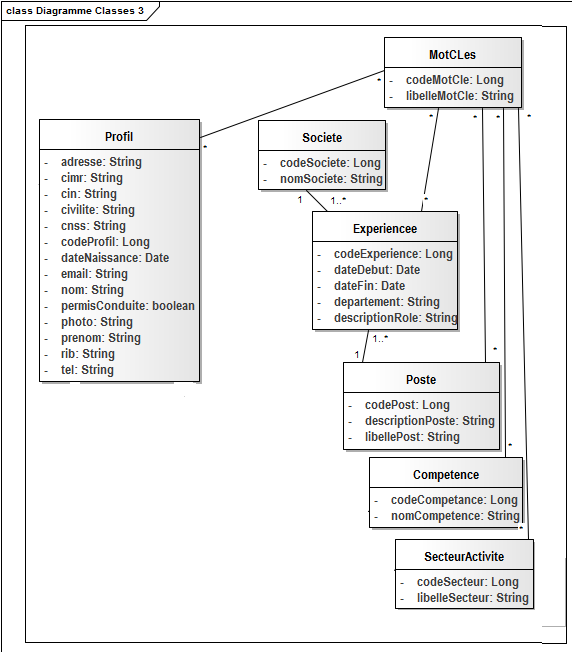


Figure : Diagramme de classe sprint 2 Release 2

### Réalisation :

La figure 34 illustre l’interface de recrutement.

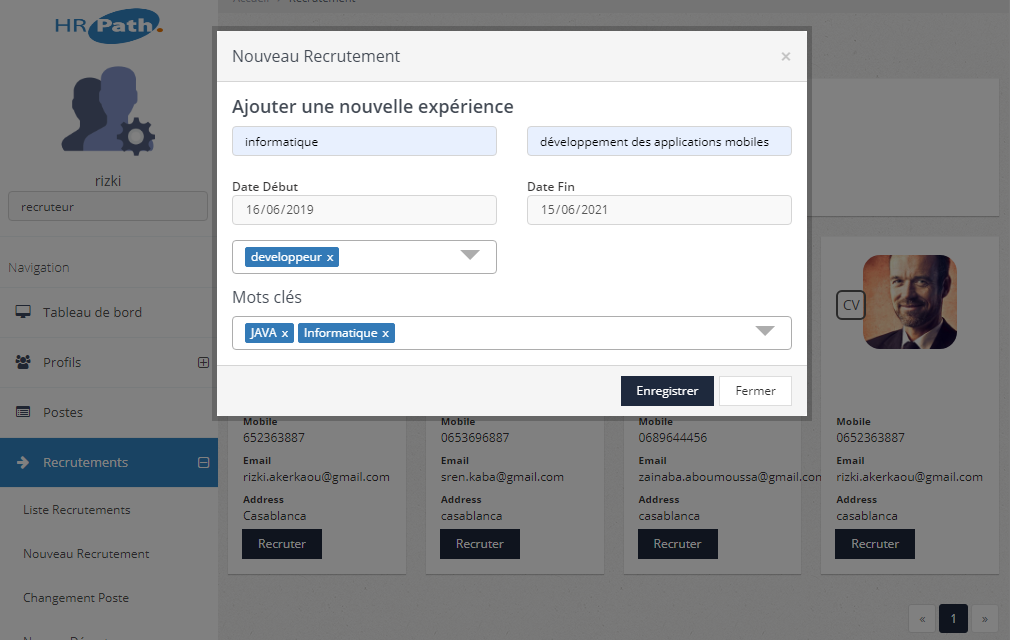


Figure : Nouveau recrutement

La figure 35 illustre l’interface de recherche avancée.

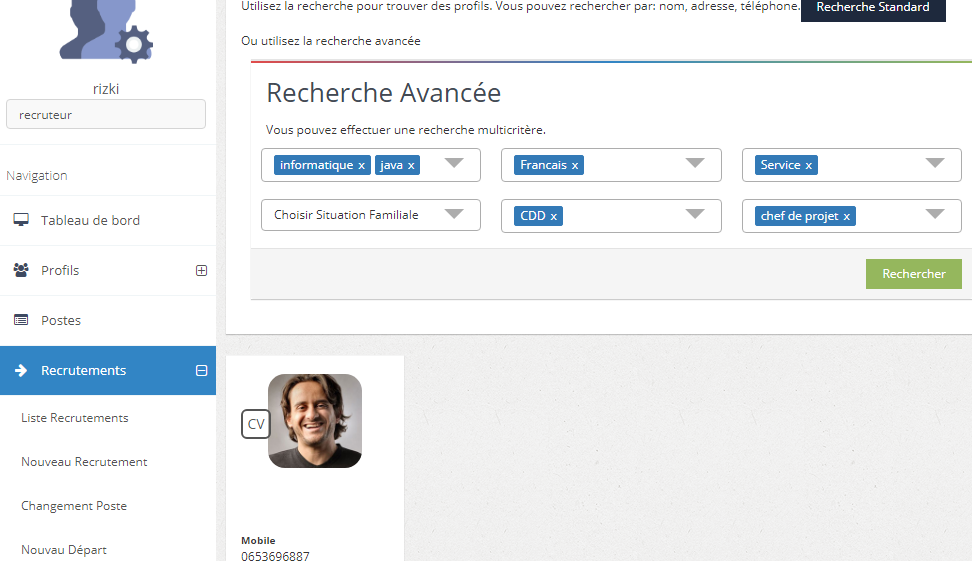


Figure : Recherche avancée

# Conclusion générale

L’expérience joue un rôle très important dans le développement et l’amélioration des compétences d’un individu du fait qu’elle est considérée comme étant un savoir pratique acquis après une longue durée d’effort et de travail.

Ainsi ce stage était une occasion précieuse qui nous a permis de découvrir le monde du travail, de clarifier certains problèmes, d’enrichir et d’améliorer les connaissances acquises pendant notre cursus d’études, à la fois sur le plan théorique et le plan pratique.

Notre projet avait pour but la réalisation d’une application de recrutement et CVthèque. Pour mener bien ce projet nous avons commencé par une étude détaillée des besoins.

Nous avons choisi la méthode SCRUM durent tous notre projet. Nous avons ainsi décomposé le projet en plusieurs itérations ou chacune d’elles suit le processus suivant : Analyse et conception, codage et puis tests.

1. Unified Modeling Language [↑](#footnote-ref-1)
2. Le manifeste agile est un texte rédigé et signé en 2001 par 17 experts dans le domaine de développement d’applications informatique. [↑](#footnote-ref-2)
3. C. Aubry, SCRUM le guide pratique de la méthode agile la plus populaire, Dunod, 2010. [↑](#footnote-ref-3)
4. Ce sont les méthodologies dont les activités de développement (spécification, conception, codage et test) se déroulent séquentiellement notamment le modèle en cascade ou en V. (Nommage de Claude Aubry) [↑](#footnote-ref-4)
5. Les pratiques les plus connues sont : l’intégration continue, la programmation en binôme, etc. [↑](#footnote-ref-5)