

FACULTE DES SCIENCES DE TUNIS
DEPARTEMENT DES SCIENCES DE
L'INFORMATIQUE



RAPPORT DE PROJET DE FIN D'ÉTUDES

Conception et développement d'une
application de recrutement

Présenté par :
Arafet AZZABI

Encadré par :
Mr. Faouzi MOUSSA

Organisme d'accueil :



Année Universitaire 2016 - 2017

Table des matières

Introduction	7
1 ÉTUDE PRÉALABLE	9
1.1 Introduction	9
1.2 Présentation générale	9
1.2.1 Cadre générale	9
1.2.2 Étude de l'existant	9
1.3 État de l'art	11
1.4 Choix technique	12
1.4.1 Phase de conception	12
1.4.2 SGBD	13
1.4.3 Phase de réalisation	14
1.5 Choix méthodologique : SCRUM	16
1.6 Conclusion	16
2 ANALYSE ET SPÉCIFICATION DES BESOINS	17
2.1 Introduction	17
2.2 Identification des besoins	17
2.2.1 Besoins fonctionnelles	17
2.2.2 Besoins non fonctionnelles	18
2.3 Présentation de la solution	18
2.4 Diagramme de cas d'utilisation	19
2.4.1 Les acteurs	19
2.4.2 Diagramme de cas d'utilisation	19
2.5 Conclusion	22
3 CONCEPTION	23
3.1 Introduction	23
3.2 Diagramme des classes	23
3.3 Diagramme de séquence	26
3.3.1 Diagramme de séquence : « Authentification »	26
3.3.2 Diagramme de séquence : « Ajout d'un profil »	27

3.3.3	Diagramme de séquence : « Ajout d'une candidature »	28
3.3.4	Diagramme de séquence : « Ajout d'un poste »	28
3.3.5	Diagramme de séquence : « Ajout d'une étape »	29
3.4	Diagramme d'activité	29
3.5	Conclusion	31
4	RÉALISATION DU LOGICIEL	32
4.1	Introduction	32
4.2	Structure du projet	32
4.2.1	Arborescence du code source	32
4.2.2	Dans Symfony, tout est bundle	33
4.3	Diagramme de déploiement	34
4.4	Processus de travail	35
4.4.1	Déroulement de la méthode Scrum	35
4.4.2	Travail collaboratif avec GIT	35
4.5	Réalisation	36
4.5.1	Critères ergonomiques	36
4.5.2	Interfaces générale	37
4.5.3	Les pages concernant la gestion de Profil	39
4.5.4	Les pages concernant la gestion de Candidature	43
4.5.5	Les pages concernant la gestion des Postes	46
4.6	Conclusion	47
	CONCLUSION GÉNÉRALE	48
	ANNEXE	51

Table des figures

1.1	Méthode de présentation des données des candidats	10
1.2	Le logo de Inser RH	11
1.3	Le logo de Candidatus	11
1.4	Le logo de Beetwen Software	12
1.5	L'architecture MVC	13
1.6	Interface de Bitbucket	14
2.1	Waller	18
2.2	Diagramme des cas d'utilisation Général	19
2.3	Diagramme des cas d'utilisation Gestion des Profils	20
2.4	Diagramme des cas d'utilisation Gestion des Postes	21
2.5	Diagramme des cas d'utilisation Gestion des Étapes	21
2.6	Diagramme des cas d'utilisation Gestion des Candidatures	22
3.1	Diagramme des classes	24
3.2	Diagramme de séquence de l'Authentification	26
3.3	Diagramme de séquence de l'Ajout d'un profil	27
3.4	Diagramme de séquence de l'Ajout d'une candidature	28
3.5	Diagramme de séquence de l'Ajout d'un poste	28
3.6	Diagramme de séquence de l'Ajout d'une étape	29
3.7	Diagramme d'activité générale	30
3.8	Diagramme d'activité du processus de recrutement	31
4.1	Structure globale du projet	32
4.2	Structure du répertoire Src	33
4.3	Diagramme de déploiement	34
4.4	Interface de l'authentification	37
4.5	Interface du menu	38
4.6	La page de la liste des profils	39
4.7	La page de l'ajout d'un profil	40
4.8	La page de l'affichage du profil	41
4.9	La page de recherche des profils	42

4.10	La page de l'ajout d'une candidature	43
4.11	La page de l'édit de la candidature 1	44
4.12	La page de l'édit de la candidature 2	45
4.13	La page de l'ajout d'un poste	46
4.14	La page de l'affichage d'un poste	47
4.15	Déroulement du premier sprint	49
4.16	Déroulement du deuxième sprint	50
4.17	Déroulement du troisième sprint	51

Dédicaces

A la mémoire de mon défunt père.

À la plus belle créature que Dieu a créée sur terre .

À cet source de tendresse, de patience et de générosité ... À ma mère !

À tous mes frères et sœur, pour leurs encouragements permanents, leur soutien moral, et leur encouragement.

À mes chères neveux et nièce et à toute ma famille.

À tous ceux et toutes celles qui m'ont accompagné et soutenu durant mes années d'étude.

Remerciements

C'est avec un grand plaisir que je réserve cette page en signe de gratitude et de profonde reconnaissance à tous ceux qui m'ont aidé de près ou de loin à la réalisation de ce modeste travail.

Je tiens à exprimer mes sincères gratitude et respects à mon encadrant Mr.Faouzi Moussa, pour son encouragement et ses précieux conseils qu'il n'a cessé de me prodiguer tout au long de ce projet.

Je remercie également toute l'équipe de Smart Team pour leur accueil, leur esprit d'équipe, et leur encouragement, ce qui m'a beaucoup aidé à concrétiser ce projet .

Je n'omettrai pas d'exprimer toute ma gratitude à tout le staff de la Faculté des sciences de Tunis (FST) qui n'a épargné aucun effort pour que nos travaux se terminent dans de bonnes conditions.

Enfin, mes meilleurs et vifs remerciements s'adressent aux membres du jury pour avoir accepté d'évaluer ce projet.

INTRODUCTION

Le marché du recrutement vit de nos jours, avec l'essor du domaine informatique, des changements qui exigent un perpétuel développement. Pour faire face à ces nombreuses tensions observées sur le marché de l'emploi, les entreprises sont désormais dans l'obligation de répondre rapidement aux nouveaux enjeux et exigences liés à ce processus, tout en suivant cette évolution.

Trouver les profils qui répondent parfaitement aux besoins des sociétés au bons moments, et avoir une vue globale sur les compétences des profils, est une exigence du métier . Ce qui garantit l'équilibre nécessaire au développement social et économique de l'entreprise .

Dans ce cadre, Smart Team a décidé de mettre en place une application facilitant leur travail. Le but est d'avoir les meilleurs profils avec les bonnes compétences. Ces profils là doivent généralement suivre un long processus de recrutement pour être triés, filtrés et choisis. Un tel outil apportera une vision claire sur tous les candidats d'où une facilité dans la prise de décision.

Le but de ce rapport est de vous présenter la démarche suivie pour arriver à la mise en place de cette nouvelle solution. Il est structuré comme suit :

La première partie « **ÉTUDE PRÉALABLE** » sera consacrée à la présentation du cadre de mon stage de PFE, à savoir, la société Smart Team, et l'objectif de mon stage. Je ferai par la suite l'analyse et la critique de l'existant dans l'entreprise. Je finirai ce chapitre par l'étude de l'état de l'art et l'illustration de quelques exemples de solutions de logiciel de recrutement.

Puis, j'analyserai, dans la seconde partie « **ANALYSE ET SPÉCIFICATION** », les besoins exprimés par les responsables de la société, tout en présentant la solution réalisée. Ensuite, je présenterai les interactions possibles des acteurs avec l'application, à l'aide du diagramme de cas d'utilisation.

Dans la troisième partie, intitulée la « **CONCEPTION** », je mettrai en exergue l'importance de cette phase pour préparer un projet de développement bien consolidé . Je

présenterai donc, le diagramme de classe complet, le diagramme d'activité générale et spécifique au processus de recrutement et les diagrammes de séquence détaillés.

Dans le dernier chapitre intitulé « **RÉALISATION** », je présenterai la structure générale du projet . Ensuite, je détaillerai le diagramme de déploiement. Enfin, je donnerai des aperçus sur l'application en mettant le point sur l'importance des critères ergonomiques dans le choix de ces interfaces.

Je conclurais sur une analyse de cette première expérience professionnelle. J'exposerai également quelques perspectives sur l'avenir de cette solution.

Chapitre 1

ÉTUDE PRÉALABLE

1.1 Introduction

Je vais consacrer ce chapitre à la mise en contexte général de mon projet. Tout d'abord, je présente le cadre du projet et l'organisme d'accueil. Ensuite, j'enchaînerai avec l'étude de l'existant au sein de la société ainsi que sa critique. Après, je détaillerai l'état de l'art du projet en exposant quelques solutions logicielles de recrutement existantes. Enfin, je finirai ce chapitre avec les choix techniques et méthodologiques.

1.2 Présentation générale

1.2.1 Cadre générale

Dans le cadre du projet de fin d'étude de troisième année de licence fondamentale en science de l'informatique (LFI3) de la Faculté des sciences mathématiques, physiques et naturelles de Tunis, s'est déroulé mon stage au sein de la société Smart Team, une jeune Start-up travaillant dans le domaine de développement logiciel Agile & Offshore. L'objectif de ce stage était de concevoir et de développer une application web qui sert à faciliter et à informatiser d'une façon innovatrice le processus de recrutement, ainsi que la prise des décisions envers les candidats au sein de l'entreprise.

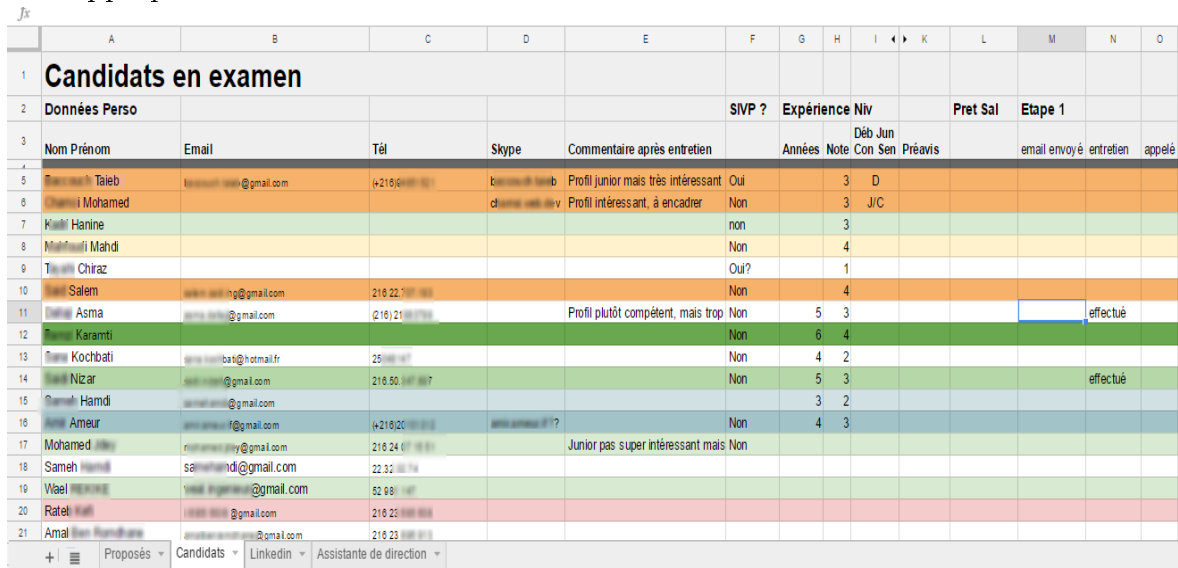
1.2.2 Étude de l'existant

Dans un premier temps, pour des raisons de compréhension du système, je vais m'intéresser au processus de recrutement actuel de Smart Team, de faire sa critique et de d'expliquer ses risques potentiels, dans le but de savoir ce qui est nécessaire pour mettre en place le futur outil de gestion des candidatures et du recrutement.

Modalité et processus de recrutement

A présent, Smart Team gère le recrutement de la façon suivante :

- les information concernant le profil, les compétences, et les coordonnées des candidats sont stockés dans des fichiers " Google sheets " .
- L'accès à ces fichiers est partagés entre les membres de l'équipe et clients de l'entreprise.
- Les profils sont alors appelés par téléphone, pour fixer une date d'entretien.
- La gestion des entretiens se fait à l'aide de l'outil "Trello ²" .
- Lors des entretiens, les commentaires et les évaluation sont rajoutés dans les colonnes appropriées.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Candidats en examen														
2	Données Perso					SVP ?	Expérience Niv		Pret Sal		Etape 1				
3	Nom Prénom	Email	Tél	Skype	Commentaire après entretien		Années	Note	Déb Jun	Con Sen	Préavis		email envoyé	entretien	appelé
4															
5	Youssef Taieb	youssef.taieb@gmail.com	(+216) 91 11 11 11	youssef.taieb	Profil junior mais très intéressant	Oui		3	D						
6	Chaimi Mohamed	chaimi.mohamed@gmail.com	(+216) 91 11 11 11	chaimi.mohamed	Profil intéressant, à encadrer	Non		3	J/C						
7	Khalil Hanine					Non		3							
8	Mohamed Mahdi					Non		4							
9	Toufik Chiraz					Oui?		1							
10	Youssef Salem	youssef.salem@gmail.com	216 22 111 111			Non		4							
11	Youssef Asma	youssef.asma@gmail.com	(216) 21 11 11 11		Profil plutôt compétent, mais trop	Non		5	3					effectué	
12	Youssef Karamti					Non		6	4						
13	Youssef Kochbati	youssef.kochbati@hotmail.fr	25 111 111			Non		4	2						
14	Youssef Nizar	youssef.nizar@gmail.com	216 50 111 111			Non		5	3					effectué	
15	Youssef Hamdi	youssef.hamdi@gmail.com						3	2						
16	Youssef Amour	youssef.amour@gmail.com	(+216) 20 111 111	youssef.amour		Non		4	3						
17	Mohamed Jalel	mohamed.jalel@gmail.com	216 24 111 111		Junior pas super intéressant mais	Non									
18	Samir Hamdi	samir.hamdi@gmail.com	22 32 111 111												
19	Wael Hachem	wael.hachem@gmail.com	52 08 111 111												
20	Rafael Kati	rafael.kati@gmail.com	216 21 111 111												
21	Amal Ben Houdhane	amal.ben.houdhane@gmail.com	216 23 111 111												

FIGURE 1.1 – Méthode de présentation des données des candidats

Points forts de cette méthode

- Profils accessibles à tout moment sur Google Sheet .
- Décisions et notes visibles aux acteurs.
- Ordre flexible des étapes

Points faibles de cette méthode

- Risques de pertes des données.
- Problèmes liés au droits d'accès.
- Absence de contrôle de saisie.

2. www.trello.com

- L'organisation : pas de gestion de visibilité aux utilisateurs pour certains données, protection, contrôle et suivie des documents.
- Visualisation des données des candidats difficile, voire épuisante.
- Difficultés dans la consultation des historiques des entretiens.
- Difficulté dans la gestion (ordres et décisions).

1.3 État de l'art

Inser Rh : [1]

Le logiciel Inser Recrutement est un logiciel Ressources Humaines, permettant de gérer l'ensemble de l'activité recrutement des établissements.

Ses principales fonctions :

- Gestion des entretiens, alertes, relances, courriers...
- Gestion des contrats et périodes d'essai.
- Reporting et analyses via les états standards fournis.



FIGURE 1.2 – Le logo de Inser RH

Candidatus : [2]

est un logiciel en mode SaaS, accessible par Internet. Il centralise les candidatures multicanales : email, site web, formulaire, papier, sourcing...

Avantages : Le traitement des candidatures est homogène et rapide ; la sélection pertinente ; l'accueil des candidats visiblement amélioré.



FIGURE 1.3 – Le logo de Candidatus

***Between software* : [3]**

Logiciel de recrutement digital mobile et cloud : disponible sur tout supports, fournissant la possibilité des entretiens en vidéo, et consultation de l'activité des candidatures .



FIGURE 1.4 – Le logo de Beetween Software

Recrutor, ProfilSoft, TalentSoft, etc ..

1.4 Choix technique

1.4.1 Phase de conception

UML [4]

Le langage de modélisation unifié, est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système.

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

StarUML

Lors de la phase de conception j'ai utilisé le logiciel StarUML pour la réalisation des diagrammes.

Définition [5]

C'est un logiciel de modélisation UML, cédé comme open source par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL.

Design pattern

Symfony, étant un framework PHP, se base essentiellement sur le desing pattern MVC¹ .

1. MVC « Modèle / Vue / Contrôleur » : découpage répandu pour développer les sites Internet, qui sépare les couches selon leur logique propre

Le Contrôleur (ou Controller) : ayant comme rôle de générer la réponse à la requête HTTP demandée par les visiteurs. C'est la couche qui se charge d'analyser et de traiter la requête de l'utilisateur.

Le Modèle (ou Model) : C'est l'élément contenant les données (Base de données, données XML, JSON...) ainsi que de la logique en rapport avec les données : validation, lecture et enregistrement.

La Vue (ou View) : C'est la partie visible d'une interface graphique. Une vue contient des éléments visuels ainsi que la logique nécessaire pour afficher les données provenant du modèle.

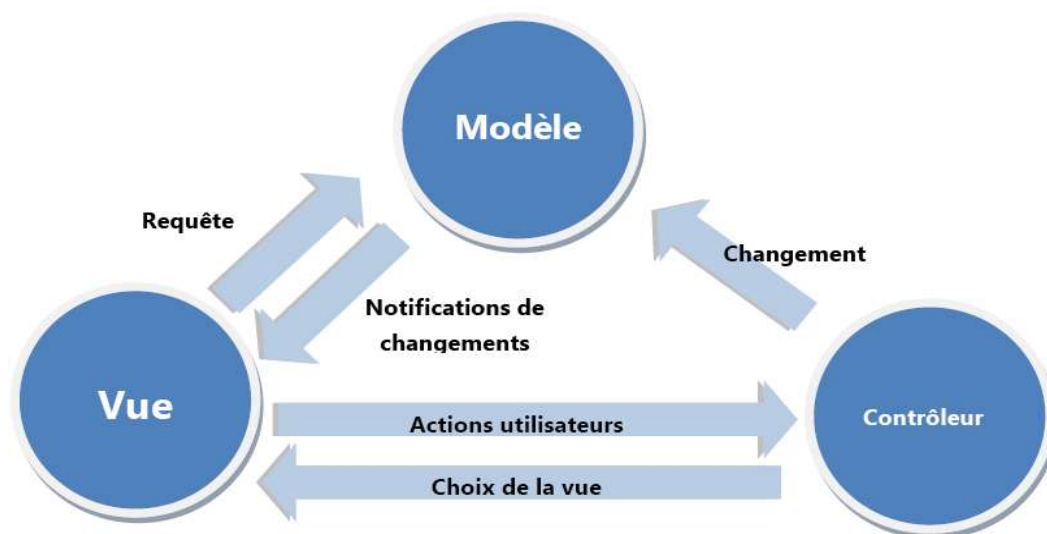


FIGURE 1.5 – L'architecture MVC

1.4.2 SGBD

Pour réaliser l'implémentation de la base de données j'ai eu recours à :

SQL [6]

C'est un langage informatique normalisé servant à exploiter des bases de données relationnelles. La partie langage de manipulation des données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données relationnelles.

phpMyAdmin : c'est une interface de gestion et d'administration d'une base de donnée permettant d'exécuter et de manipuler très facilement cette dernière.

MySQL Workbench : [7] est un logiciel de gestion et d'administration de bases de données MySQL créé en 2004. Via une interface graphique intuitive, il permet, entre

autres, de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données. Pour ce faire, il doit être connecté à un serveur MySQL.

1.4.3 Phase de réalisation

Architecture GIT

GIT : [8]

C'est un logiciel de gestion de versions décentralisé. Sa principale tâche est de gérer l'évolution du contenu d'une arborescence.

Le travail avec Git consiste donc à avoir la possibilité de gérer les versions du code source du projet tout le long de la phase de réalisation.

Ce code sera enregistré dans un dépôt distant (Exemple : GitLab, Bitbucket..) .

Dépôts distants : représente versions du projet, hébergées sur Internet ou sur un réseau local (d'une entreprise par exemple).

Au cours de ce projet j'ai utilisé Bitbucket .

Bitbucket [9] : est un service web d'hébergement et de gestion de développement logiciel utilisant les logiciels de gestion de versions Git et Mercurial.

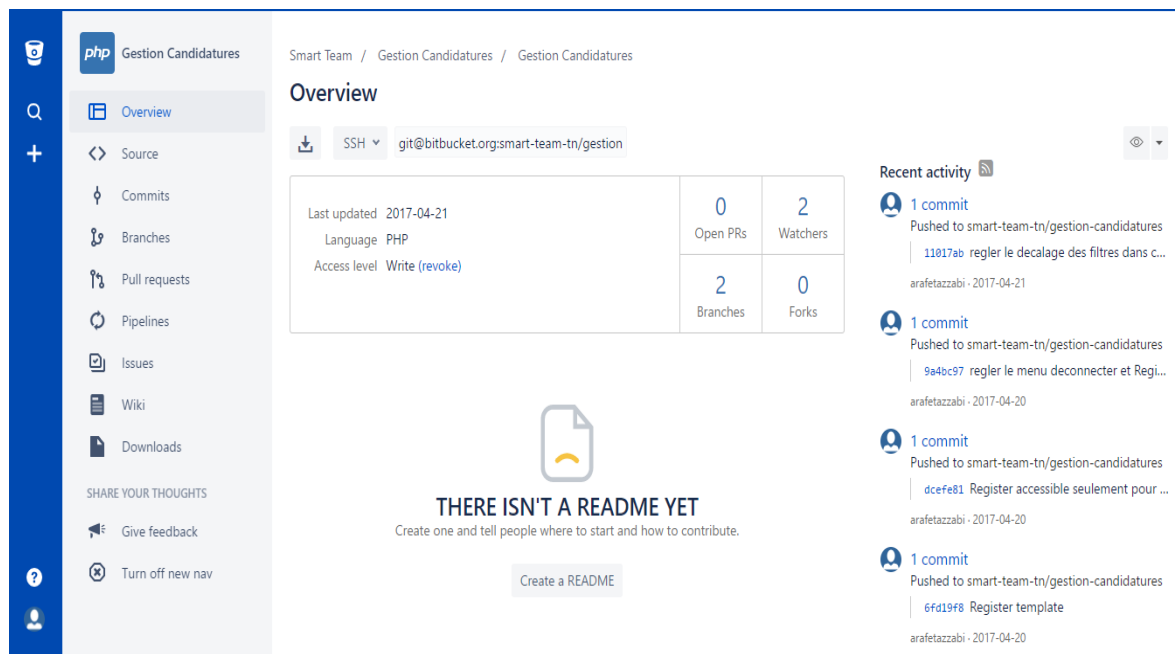


FIGURE 1.6 – Interface de Bitbucket

Les enregistrements réalisés au cours de développement du projet se fait sur des branches.

Une branche : représente une “ligne de développement”. Cette dernière est formé par un ensemble de commit.

Une commit : représente une étape dans l’historique du projet, et identifié par un message.

La branche par défaut dans Git s’appelle master. Cette branche contiendra la version finale du projet.

Technologie utilisés

SYMFONY 3 : [10] et [11]

est un ensemble de composants PHP ainsi qu’un framework MVC libre écrit en PHP. Symfony est un kit de composants destinés à faciliter le développement de sites internet riches ou d’applications web.

Pour cela, le code est séparé en trois couches selon le modèle MVC qui sépare le modèle de données (M), l’interface utilisateur ou vue (V), et le contrôleur (C) qui gère les évènements, la synchronisation, etc.

Conçu pour gérer :

- Des sites professionnels.
- Des problématiques complexes.
- Dans des environnements exigeants.

Doctrine : [12] C’est un ORM (Object-Relational Mapping) composé d’énorme fonctionnalités ; à commencer par le DQL (Doctrine Query Language). Le DQL vous permet de créer et d’exécuter vos requêtes via le paradigme de la programmation orientée objet.

Twig : [13] C’est un moteur de template PHP directement intégré dans Symfony depuis la version 2 . Twig permettra de gérer de l’héritage entre templates et layout, séparer les couches de présentation et couches métiers.

jQuery : [14] C’est une bibliothèque JavaScript libre et multi-plateforme créée pour faciliter l’écriture de scripts côté client dans le code HTML des pages web.

Au cours de mon projet j’ai utilisé des plugins Datatables.

Logiciel utilisés

PhpStorm : [15] C’est un éditeur pour PHP, HTML et JavaScript, édité par JetBrains.

1.5 Choix méthodologique : SCRUM

Avant la réalisation d'un projet informatique, il est nécessaire de fixer une méthodologie de travail et un processus de développement afin d'aboutir à la réalisation d'un logiciel fiable. Cela nous permettra de ne pas se trouver coincé à la moindre complication.

Définition : [16]

Scrum est un schéma d'organisation de développement de produits complexes. Il est défini par ses créateurs comme un « cadre de travail holistique itératif qui se concentre sur les buts communs en livrant de manière productive et créative des produits de la plus grande valeur possible ». Scrum est considéré comme un cadre méthodologique et non à proprement parler comme une méthode agile.

Principe :

Ce schéma d'organisation s'appuie sur le découpage d'un projet en boîtes de temps, nommées « sprints ». Les sprints peuvent durer entre quelques heures et un mois (avec une préférence pour deux semaines). Chaque sprint commence par une estimation suivie d'une planification opérationnelle. Le sprint se termine par une démonstration de ce qui a été achevé.

Hierarchie :

Scrum définit trois rôles : le propriétaire du produit (product owner), le scrum master et le développeur.

- *Propriétaire du produit (Product owner) :*
- *Maître de mêlée (Scrum master)*
- *Équipe de développement*

1.6 Conclusion

Dans ce chapitre introductif, j'ai présenté le cadre dans lequel s'est déroulé le projet ; à savoir l'étude de l'existant, et j'ai fait une étude sur les logiciels de recrutement existants. Et j'ai fini par une bibliographie sur les choix techniques et méthodologiques.

Chapitre 2

ANALYSE ET SPÉCIFICATION DES BESOINS

2.1 Introduction

Dans ce chapitre, je vais faire l'analyse des besoins des utilisateurs, ainsi que leurs objectifs dans mise en place de la solution. Ensuite, je vais faire la présentation des différents rôles (des acteurs). Enfin, je vais finir ce chapitre par le diagramme de cas d'utilisation global et détaillé.

2.2 Identification des besoins

L'objectif de mon application est de fournir un outil qui gère d'une façon informatisée les profils, les candidatures, les postes, ainsi que des étapes par lesquelles passe une candidature. Cette dernière offre alors une informatisation de tout le processus de recrutement pour différents postes, donc facilitant la prise de décision envers les candidats.

2.2.1 Besoins fonctionnelles

Cet outil, intitulé “ Waller ”, permettra à la société Smart Team, ainsi qu'à leurs clients de satisfaire les besoins fonctionnels suivants :

- Gestion des profils.
- Gestion des candidatures.
- Gestion des postes.
- Gestion des étapes du workflow de la candidature.
- *Recherche avancée* : L'application offre la possibilité de faire des recherches avancées sur les profils et les candidats.

2.2.2 Besoins non fonctionnelles

Ce sont les besoins qui permettraient d'améliorer la qualité des services du site comme la convivialité et l'ergonomie des interfaces et l'amélioration du temps de réponse. Parmi ces besoins on cite :

- *La performance* : l'application doit être performante c'est-à-dire il répond, avec ses fonctionnalités, à toutes les exigences des usagers d'une manière optimale.
- *La sécurité* : Besoins d'établissement de la connexion, il faut avoir une interface d'authentification qui permet à chaque utilisateur de se connecter afin de réaliser seulement les actions autorisées.
- *La disponibilité* : Lorsque n'importe quel utilisateur désire consulter le site, ce dernier doit être disponible.
- *La convivialité et la Simplicité* : Une application doit être simple et facile à utiliser. Il doit présenter un enchaînement logique et simple entre les interfaces, et des liens pour assurer une navigation rapide, et intuitive.
- *Ergonomie* : Les interface doivent respecter les standards d'ergonomie

2.3 Présentation de la solution

Après l'étude qu'a effectué Smart Team, et les besoins manifestés par leurs clients, il est apparu nécessaire de développer une application web, à accès limité (entre les membres de Smart Team et ces clients), qui mettra une fin aux problèmes rencontrés dans le chapitre précédent.

Cette application web aura pour objectifs :

- Un accès pour les clients qui peuvent ajouter des postes dont ils auront besoin de les combler, et en choisissant ensuite des profils à convoquer pour passer par les suites des entretiens (représente le workflow de recrutement).
- Un espace administrateur qui a les fonctionnalités suivantes :
 - La gestion des profils : l'ajout ou la suppression des profils.
 - La gestion des candidatures (convocation ,annulation des candidatures ...)
 - La gestion des postes (ajout des postes pour la société ou pour les clients).
 - La gestion des étape : par lesquelles passera une candidature.
 - La gestion des membres qui auront un accès à l'application.



FIGURE 2.1 – Waller

2.4 Diagramme de cas d'utilisation

2.4.1 Les acteurs

Un acteur représente l'abstraction d'un rôle joué par des entités externes. Dans notre application on distingue principalement trois acteurs qui sont les suivants :

- *L'administrateur* : représente le gérant(ou le propriétaire de la société), ayant tout les privilèges d'accès et de manipulation possible dans l'application, ainsi que la prise de décision envers les candidats.
- *L'assistante* : représente la responsable de communication au sein de la société, ayant un accès limité dans la manipulation des données.
- *Le(s) client(s)* : qui interviennent dans la prise des décisions envers les candidats lors des entretiens . Ils peuvent aussi consulter la liste des profils (et des candidats qui l'ont convoqués), aussi de rajouter des postes dont ils auront besoin.

2.4.2 Diagramme de cas d'utilisation

Définition [17] utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel.

Diagramme de cas d'utilisation : Générale

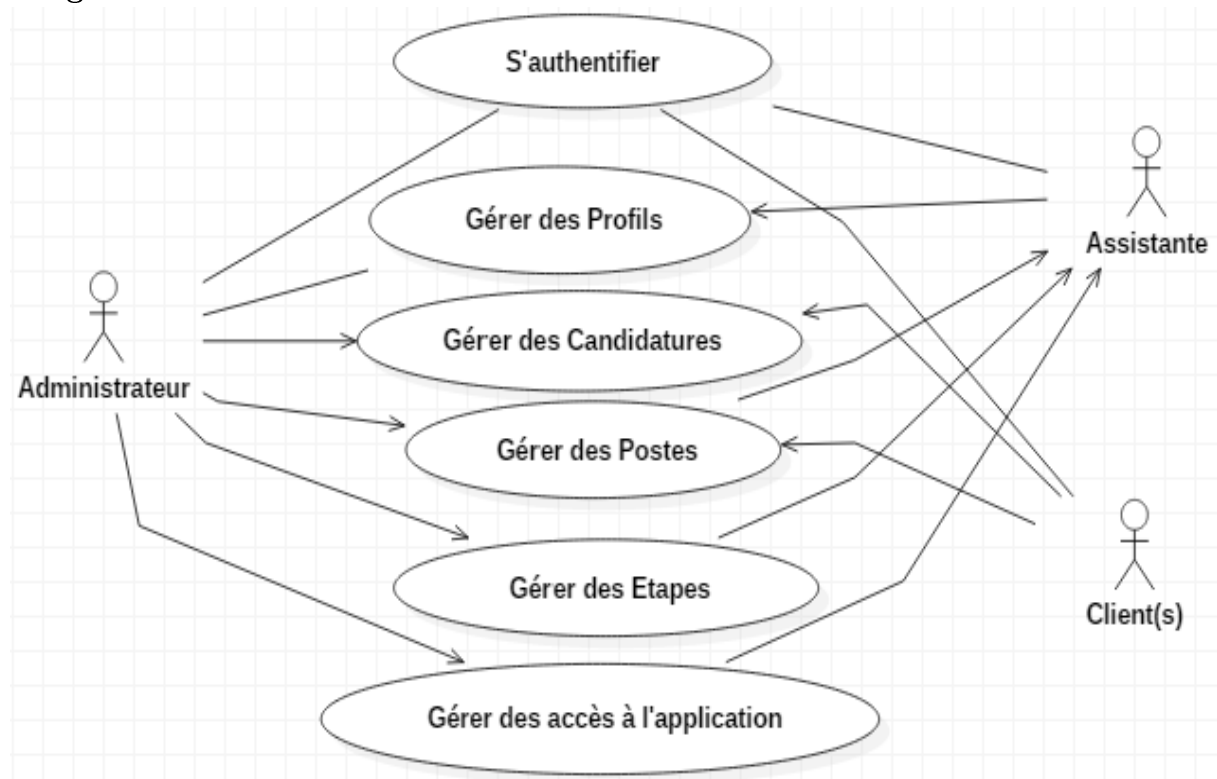


FIGURE 2.2 – Diagramme des cas d'utilisation Général

Diagramme de cas d'utilisation : Gestion des Profils

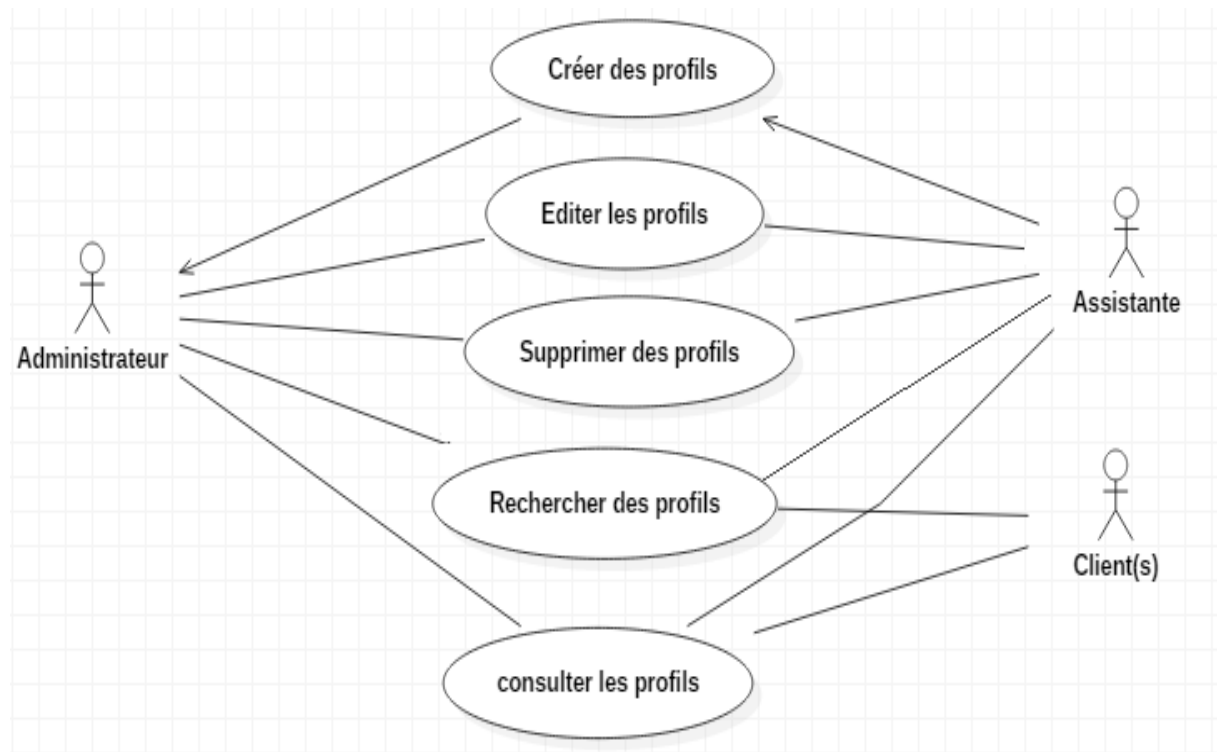


FIGURE 2.3 – Diagramme des cas d'utilisation Gestion des Profils

Cette figure montre le diagramme de cas d'utilisation général de la gestion des profils. En tant qu'administrateur, le directeur de l'entreprise aura comme privilège d'ajouter, modifier, supprimer et consulter des profils. Il peut également faire une recherche pour choisir un (des) profil(s) à recruter. L'assistante, en tant que responsable RH, aura les mêmes privilèges que le directeur de l'entreprise. Elle représente l'acteur principale dans l'ajout des profils. Le client, ayant un accès limité, il ne peut que rechercher et consulter des profils pour les convoquer.

Diagramme de cas d'utilisation : Gestion des Postes

Cette figure montre le diagramme de cas d'utilisation général de la gestion des Postes. L'administrateur et l'assistante ont tous les deux tous les privilèges de gestion des postes (Ajout, modification, suppression et consultation). Le client peut ajouter des profils. Il peut également modifier, supprimer, et consulter les postes qui les appartiennent. Il représente ainsi que l'administrateur les acteurs principaux dans l'ajout d'un nouveau poste.

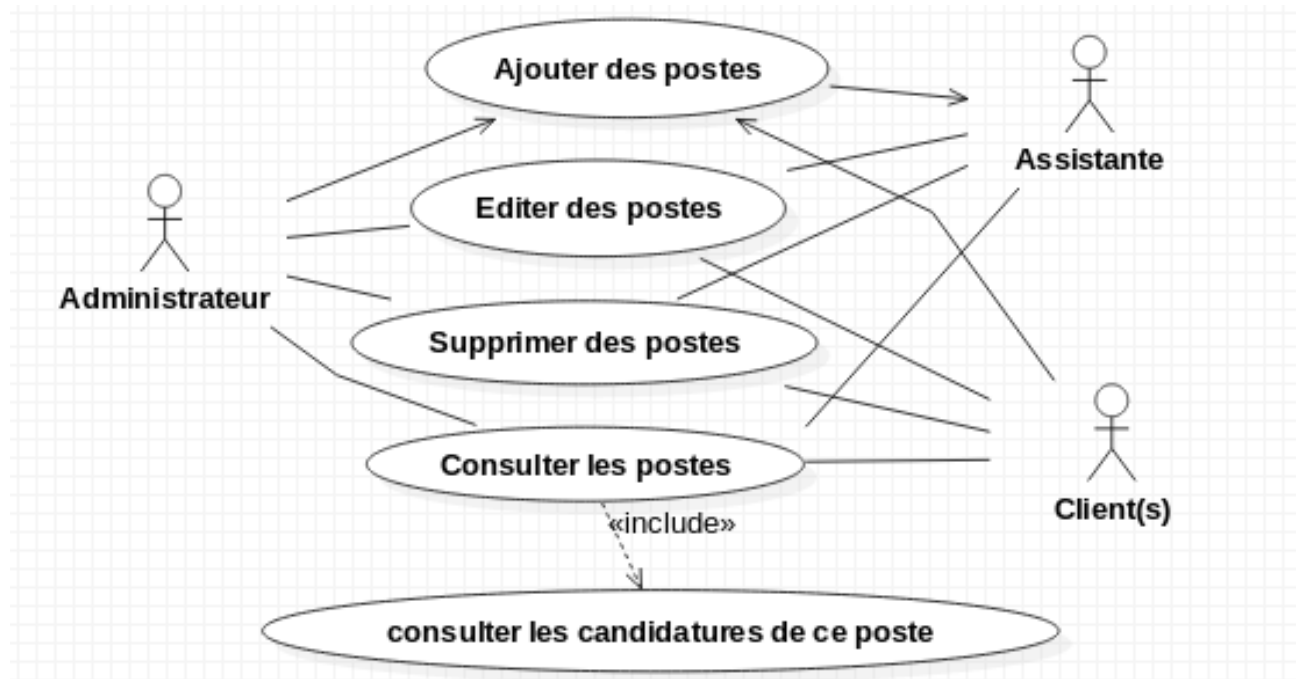


FIGURE 2.4 – Diagramme des cas d'utilisation Gestion des Postes

Diagramme de cas d'utilisation : Gestion des Étapes

Cette figure montre le diagramme de cas d'utilisation général de la gestion des étapes du workflow. Au niveau de la gestion des étapes, sauf les membres de l'entreprise ont tout les privilèges d'accès. Ils peuvent ajouter, supprimer, et modifier les étapes. Ils peuvent également modifier l'ordre de ces dernières au sein d'un workflow.

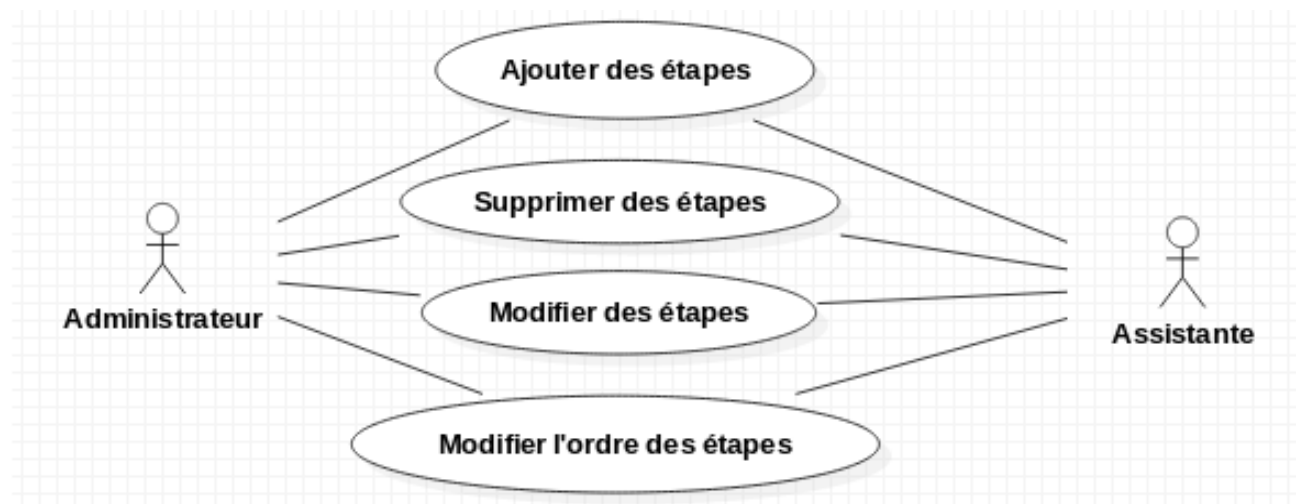


FIGURE 2.5 – Diagramme des cas d'utilisation Gestion des Étapes

Diagramme de cas d'utilisation : Gestion des Candidatures

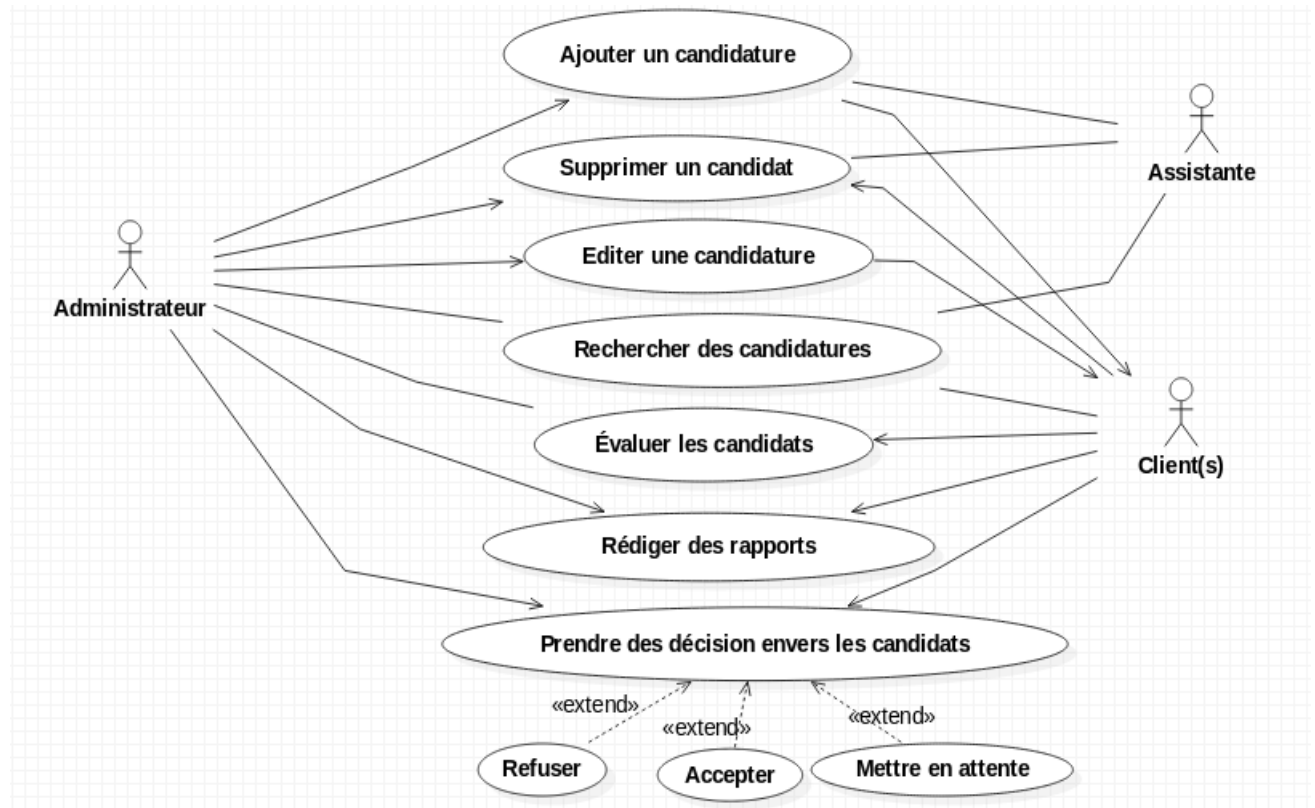


FIGURE 2.6 – Diagramme des cas d'utilisation Gestion des Candidatures

Le directeur peut ajouter, supprimer, chercher, et éditer les candidatures. Le client, et l'assistante peuvent aussi ajouter des nouveaux candidats, et de supprimer des autres. L'évaluation et la prise de décision envers les candidats représente le rôle décisionnel des clients et du directeur. A chaque prise de décision un rapport doit être rédigé, expliquant ces décisions et les conditions de l'entretien. Ces décisions peuvent être soit d'accepter, soit de refuser, soit de mettre en attente la candidature.

2.5 Conclusion

Dans ce chapitre, j'ai fait l'analyse des besoins des utilisateurs, ainsi que les objectifs de la solution, ce qui m'a aidé à présenter la solution. Ensuite, j'ai fait l'étude sur les rôles des acteurs suivi des diagrammes de cas d'utilisation global et détaillé.

Chapitre 3

CONCEPTION

3.1 Introduction

Dans le but d'obtenir une bonne représentation de l'architecture du future application, on a choisi la méthode objet pour bien décrire et spécifier les composants de notre projet, en utilisant le Langage de Modélisation UML (Unified Modeling Language). Cette dernière permet d'analyser et de comprendre facilement comment va être le fonctionnement du système.

En fait UML est conçu pour représenter, spécifier et documenter les applications, et il offre donc une meilleure compréhension des applications , permettant de répondre aux objectifs définis par les clients

3.2 Diagramme des classes

Définition [18]

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes, ainsi que les différentes relations entre eux.

Toute l'application est basé sur deux classes : Profil et Candidature.

La classe « Profil » : elle définit l'entité principale du programme. Elle représente les données des personnes qui ont envoyé leurs CV à la société. Cette entité est définis par un nom, un prénom, un nombre d'années d'expérience, des coordonnées (numéro téléphone, réseaux sociaux et Skype), ainsi que des informations sur le type du contrat, la prestation salariale et le niveau.

La classe « Candidature » : elle représente une exécution du workflow de candidature. Elle définit le choix d'un profil et sa convocation à une suite d'entretiens pour un poste donné. Elle relie un profil à un poste et un groupe. Le groupe dans ce cas c'est l'entité qui va faire l'initiation de la candidature, pour le compte du groupe détenteur du poste. Une candidature est défini par un profil, une évaluation, et un groupe qui as convoqué ce profil à un poste. Cette dernière suit un Workflow d'étapes.

La classe « Poste » : représente le poste pour lequel les sociétés veulent recruter des profils.

Il est décrit par un libellé, des liens d'annonce pour ce poste (sur les réseaux sociaux et sites d'annonces). Il est proposé par un groupe représentant la société qui a crée ce poste là(dans notre cas c'est soit Smarteam soit l'un de ses clients).

La classe « Group » : cette classe fait partie de FOSUserBundle, elle est utilisée généralement pour regrouper un ensemble d'utilisateurs ayant les mêmes droits. Un groupe est soit l'entité détentrice du poste, et pour laquelle sont adressées les candidatures (dans ce cas, ce sont les clients de Smart Team), soit Smart Team elle même. Un groupe est définie dans la base de donnée par un nom, et un rôle.

La classe « User » : représente la partie élémentaire d'un groupe. Un utilisateur dans Symfony est définie par un nom, un login, un mot de passe, un email, et un rôle. .

La classe « Workflow » : représente le processus et la succession des étapes par lesquels passe une candidature. Il est formé par une collection d'étapes triés selon un critère ordre. L'ordre des étapes varies d'un workflow à un autre, ce qui donne un aspect de flexibilité au recrutement. Un workflow est définie par un libellé , et une description.

Les classes « Étape » et « Action » : l'étape représente l'état par laquelle passe la candidature. Elle est définie par un libellé, un ordre, et appartenant à un workflow. L'action est la transition entre une étape source et une étape destination. Elle est définie par un libellé, une étape source, une étape destination, et une affection ¹. On peut transiter d'une étape 'A' à une étape 'B', seulement s'il y a une action programmée, qui a pour

1. booléen, indique si suite à la transition vers l'étape destination, la visibilité de la candidature sera affectée au groupe initiateur du workflow ou celui détenteur du poste

étape source 'A', et pour étape destination 'B'. De cette façon, on peut concevoir notre workflow, avec les règles qui le régissent et les transitions possibles (Par exemple : à partir de l'étape 'A', on ne peut aller que vers l'étape 'B' ou 'C' seulement).

La classe « Note » : représente les commentaires décrivant les candidats, qui peuvent être prises durant les étapes. Ces notes serviront à faciliter la prise de décision dans les prochains entretiens. Une note est définie par un commentaire, une évaluation, appartenant à un seul candidat, et prise durant une étape.

La classe « Rapport » : est un commentaire qui accompagne un changement d'étape. Lorsque la candidature passe d'une étape à une autre, le rapport servira à la prochaine personne qui lui est affectée l'étape (préférences sur les prises de rendez-vous, disponibilités, raisons de refus, etc..). Un rapport est défini par un libellé, un candidat, et liée à une action.

3.3 Diagramme de séquence

Définition C'est une représentation des objets et leurs interactions selon une ligne temporelle. Elle représente la vue fonctionnelle d'une demande, actions, ou d'une activité donnée.

Afin d'assurer une bonne compréhension du comportement voulu du système nous allons traiter tous les cas en détails dans l'ordre de leurs importance :

3.3.1 Diagramme de séquence : « Authentification »

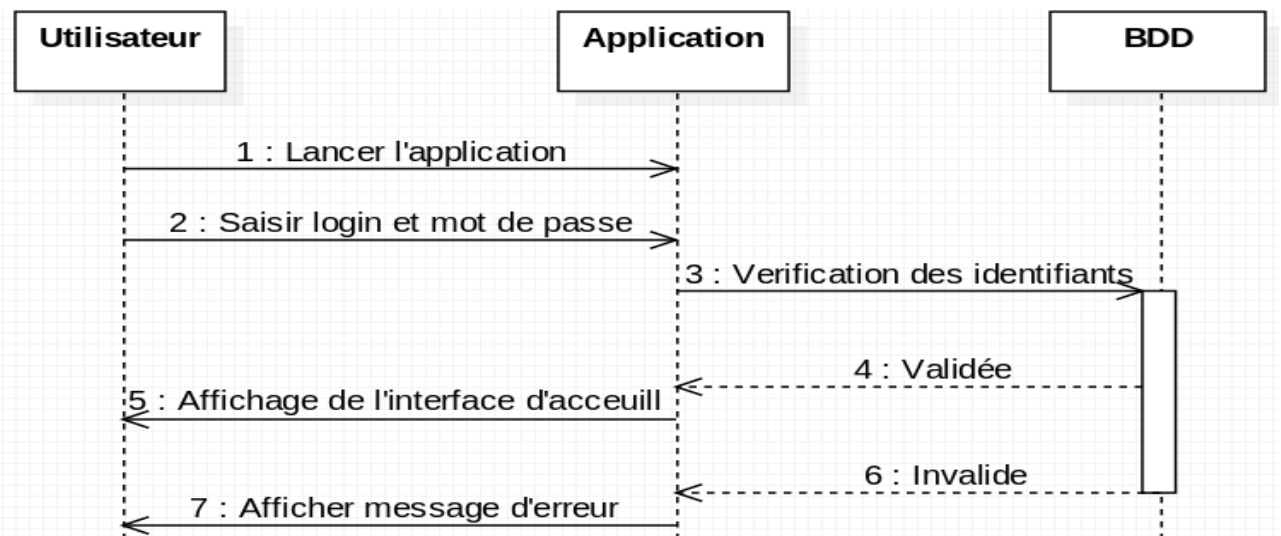


FIGURE 3.2 – Diagramme de séquence de l'Authentification

En s'authentifiant au système, l'utilisateur accède à son compte via le mot de passe et le nom d'utilisateur. Ces données sont stockées dans la table « Users », et c'est avec

laquelle le contrôleur effectue une opération de lecture pour rechercher l'utilisateur en question avant de valider l'accès ou le refuser.

3.3.2 Diagramme de séquence : « Ajout d'un profil »

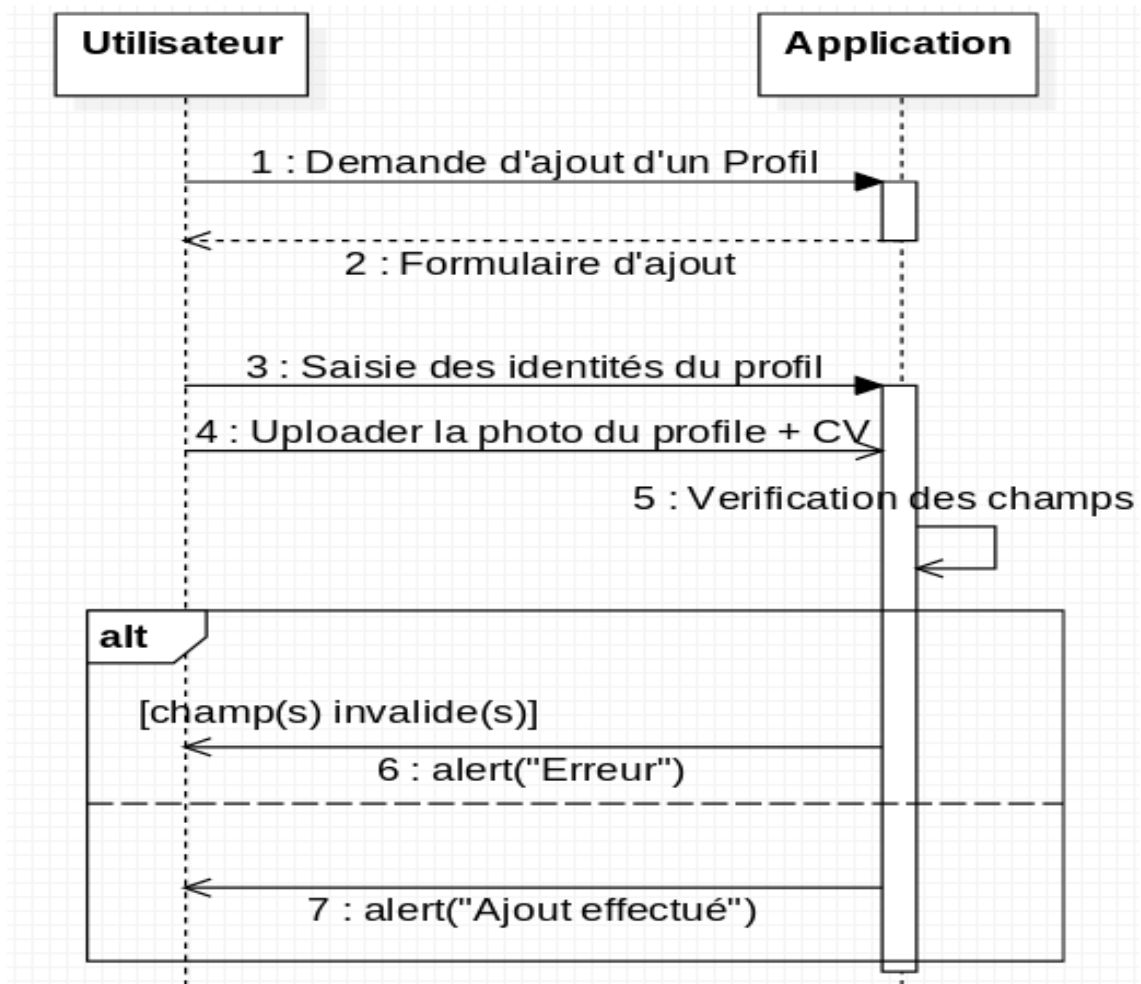


FIGURE 3.3 – Diagramme de séquence de l'Ajout d'un profil

Description : Après avoir s'authentifier au système, l'utilisateur accède à la page des profils, et demande l'ajout d'un nouveau profil. Il remplit les informations nécessaire (identité du profil, ses coordonnées, ses compétences, et sa situation).

Si les informations sont valides le profil sera enregistré dans la base de données, sinon le système lance un message d'erreur avec l'indication du type d'erreur et le(s) champ(s)

invalide(s).

3.3.3 Diagramme de séquence : « Ajout d'une candidature »

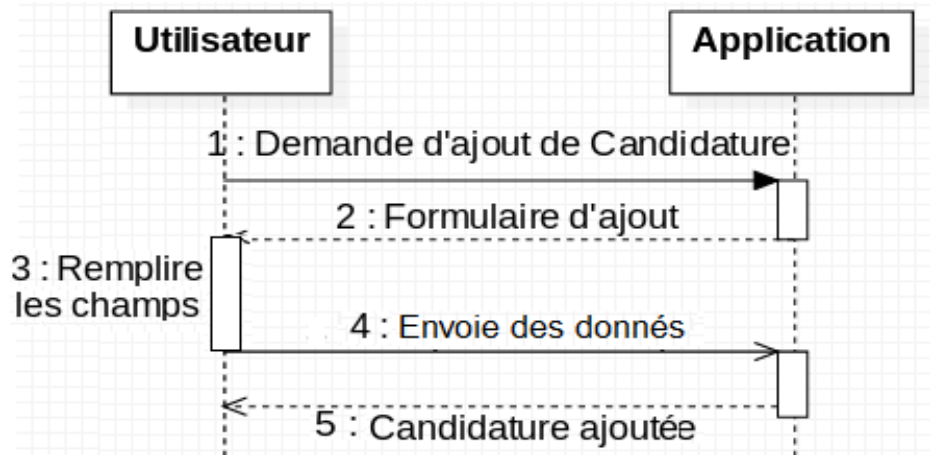


FIGURE 3.4 – Diagramme de séquence de l'Ajout d'une candidature

Description : Après avoir s'authentifier au système, l'utilisateur accède à la page des candidatures, et demande l'ajout d'un nouveau candidature.

Il commence par remplir les données nécessaire pour ajouter une nouvelle candidature. Ensuite, il choisit le profil à recruter ainsi que le poste et la société pour laquelle ce candidat va être recruter. Puis il finis par définir l'étape par laquelle la candidature va commencer.

3.3.4 Diagramme de séquence : « Ajout d'un poste »

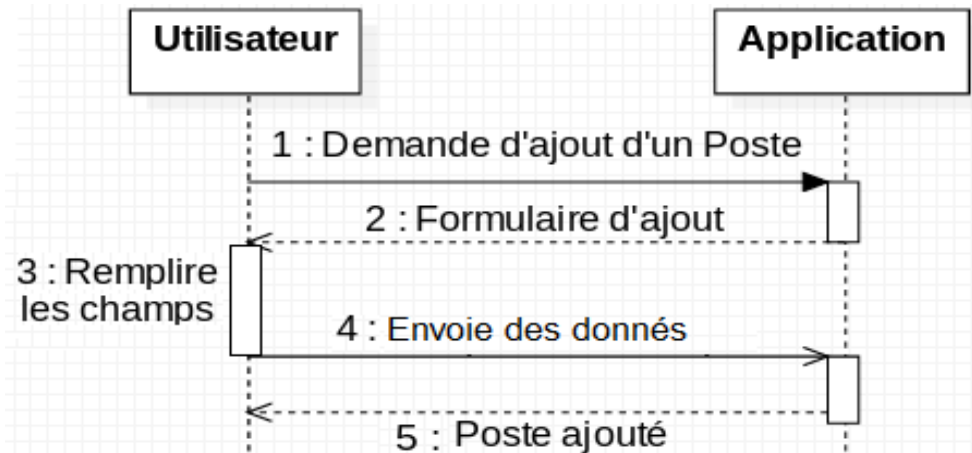


FIGURE 3.5 – Diagramme de séquence de l'Ajout d'un poste

Description : Après avoir s'authentifier au système, l'utilisateur accède à la page des postes, et demande l'ajout d'un nouveau poste.

L'utilisateur remplit les données nécessaire pour ajouter un nouveau poste.

Il commence par remplir le libelle, la description du poste et la description du profil demandé.

3.3.5 Diagramme de séquence : « Ajout d'une étape »

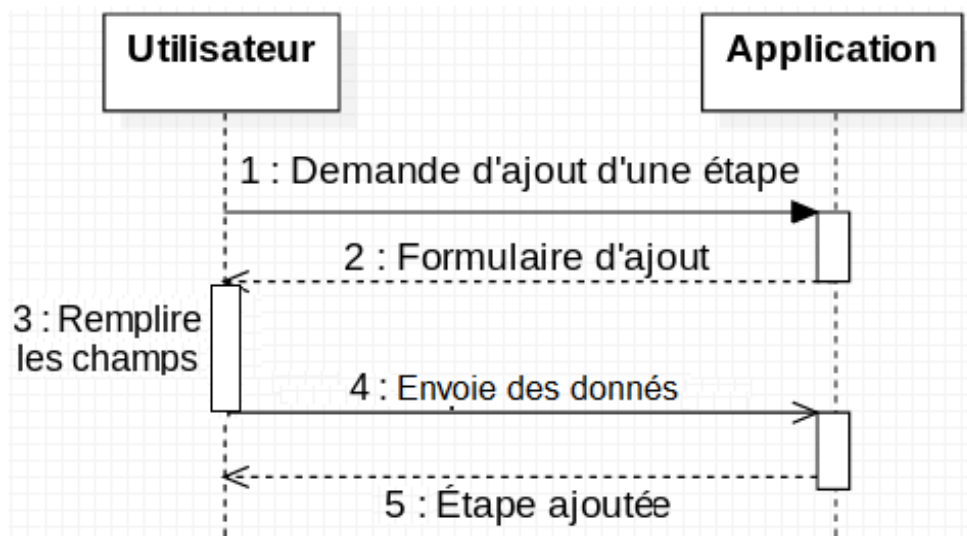


FIGURE 3.6 – Diagramme de séquence de l'Ajout d'une étape

Description : L'utilisateur remplit les données nécessaire pour ajouter une nouvelle étape.

Il commence par choisir le libellé de l'étape, ensuite il choisie à quelle workflow qu'il va ajouter cette étape. Enfin, il choisie l'ordre de cette étape par rapport au autres étapes de ce Workflow.

3.4 Diagramme d'activité

Définition [19]

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallélisables.

Diagramme d'activité générale

La figure ci-dessous représente le diagramme d'activité général de l'application.

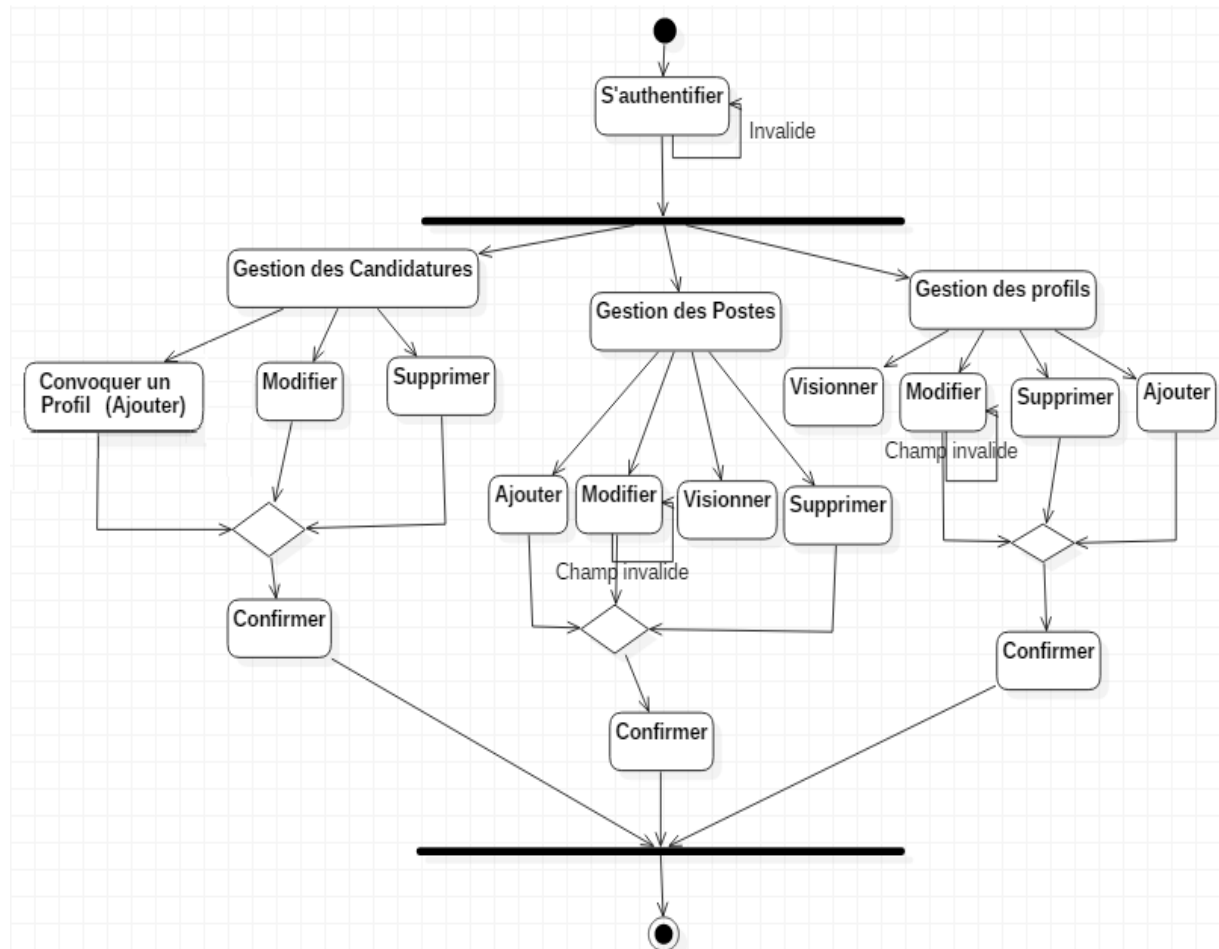


FIGURE 3.7 – Diagramme d'activité générale

Description : Après avoir été authentifié, l'utilisateur peut alors accéder aux différentes pages qui lui sont permis (selon les utilisateurs). En générale, l'utilisateur aura l'accès sur 3 volé dans l'application. Le premier se base sur la gestion des profils, avec l'ajout, le visionnage, la modification, ou la suppression des profils. Le deuxième représente l'ajout (convocation d'un profil), la modification, ou la suppression d'une candidature.

Diagramme d'activité du processus de recrutement

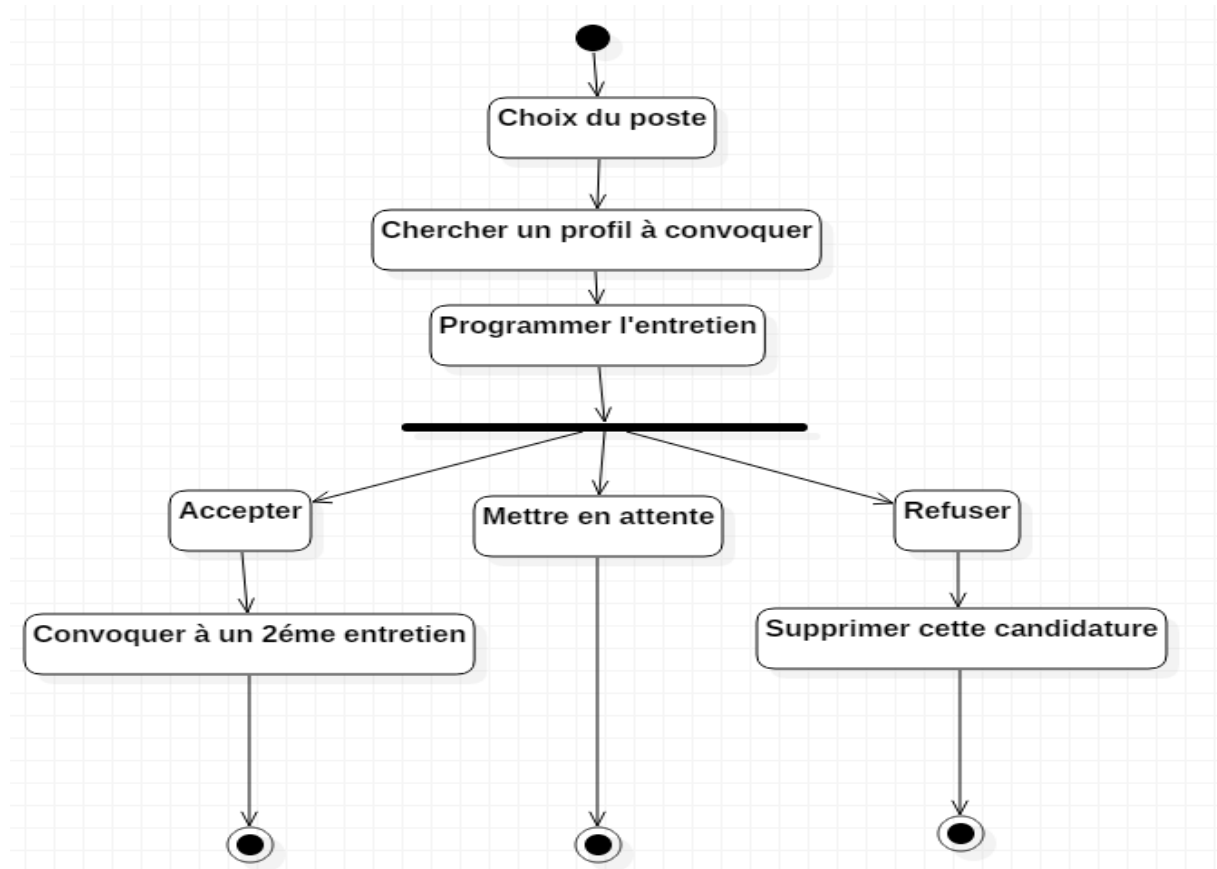


FIGURE 3.8 – Diagramme d'activité du processus de recrutement

Description : Le processus de recrutement commence tout d'abord par l'ajout d'un poste, dont l'entreprise aura besoin de couvrir. Ensuite, ils choisissent parmi les profils disponibles, celui (ou ceux) qui convient (conviennent) le mieux au poste. Enfin, il fixe une date du premier entretien. Durant l'entretien, une décision parmi 3 possibles peut être prise. La première possibilité c'est d'accepter ce profil, et donc lui convoquer au prochain entretien. La deuxième possibilité est de mettre sa candidature en attente en vue de le convoquer pour d'autres postes (conserver le profil dans la liste des profils). La dernière possibilité est que la candidature sera refusée et cette candidature sera supprimée.

3.5 Conclusion

Au terme de ce chapitre, j'ai présenté l'aspect fonctionnel de la solution réalisée à travers les diagramme de classes, de séquence, et d'activité. Cette phase me servira pour la partie réalisation qui sera traitée dans le chapitre suivant.

Chapitre 4

RÉALISATION DU LOGICIEL

4.1 Introduction

Cette partie du rapport a pour objectif d'exposer l'implémentation du système conçu, présenté dans le chapitre précédent. Je vais commencer tout d'abord par la présentation de la structure globale du projet tout en détaillant les parties les plus importantes d'un projet Symfony avec le rôle de chacun d'eux. Ensuite, je vais parler du principe des bundles dans Symfony et leur importance dans le projet.

Après, je vais présenter le diagramme de déploiement, avant de finir avec la présentation des interfaces de l'application, en décrivant brièvement les critères ergonomique prises en compte.

4.2 Structure du projet

Dans cette partie, je vais détailler les différentes étapes permettant le développement de mon application.

4.2.1 Arborescence du code source

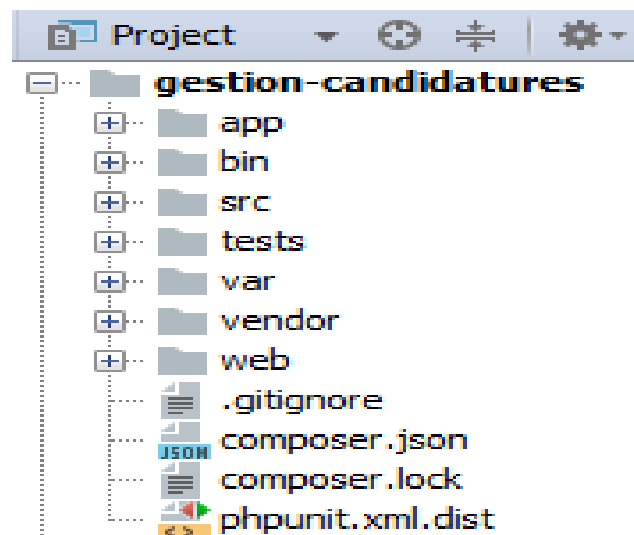


FIGURE 4.1 – Structure globale du projet

- **app** : contient les ressources, configurations et les traductions.
- **bin** : contient les scripts de gestion symfony.
- **src** : contient les bundles du projet.
- **var** : contient les logs et le dossier de cache.
- **vendor** : contient les bundles externes.
- **web** : contient les fichiers "publics" du projet.

4.2.2 Dans Symfony, tout est bundle

Un Bundle est un ensemble de fichiers et répertoires qui permet aux utilisateur d'implémenter une ou plusieurs fonctionnalités.

Dans les bonnes pratiques de Symfony, tout le code source du projet doit être placé dans un bundle spécial, nommé AppBundle. Ce bundle est placé sous le répertoire src, et contient les Contrôleurs, les Entités, les Services, etc.

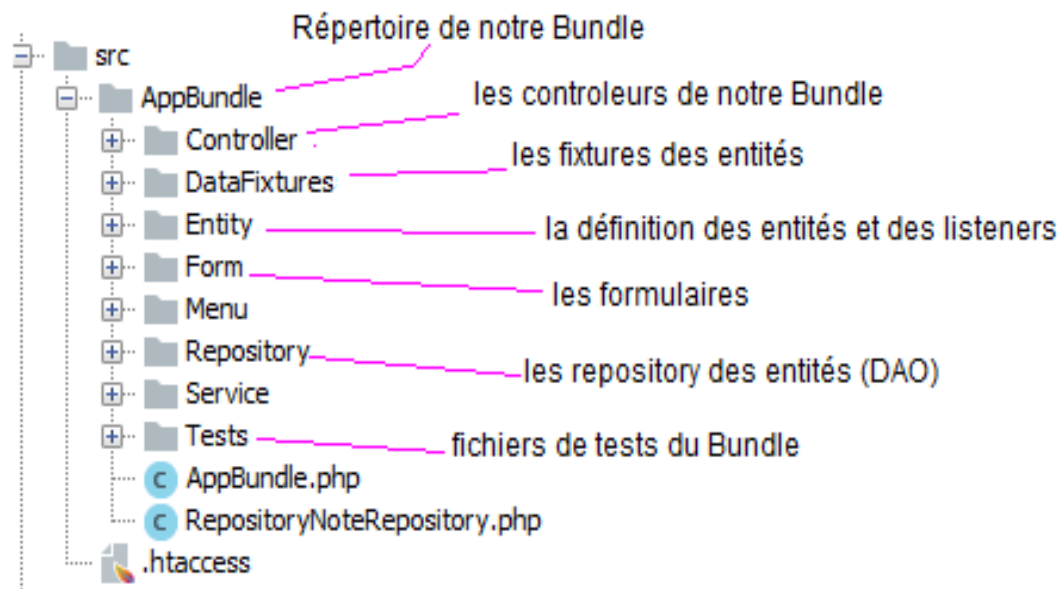


FIGURE 4.2 – Structure du répertoire Src

Il existe une très large panoplie de bundles, développés par des communautés Open Source, qui servent à apporter de nouvelles fonctionnalités au framework, ou à améliorer des fonctionnalités existantes. Ces Bundles sont installés via Composer, dans un répertoire nommé 'vendor'. Au cours de la réalisation de mon projet, j'ai utilisé ces bundles :

- FOSUserBundle : ajoute un support pour un système d'utilisateur pris en charge par la base de données dans Symfony. Il fournit un framework flexible pour la gestion des utilisateurs qui vise à gérer les tâches courantes telles que l'enregistrement de l'utilisateur et la récupération du mot de passe.

- VichUploaderBundle : permet de faciliter le téléchargement des fichiers attachés aux entités. Dans le cas de mon projet, je ai eu recours à ce bundle dans la gestion des photos et des CV des profils.
- FOSJsRoutingBundle.
- *Les Bundles pré-existant* : TwigBundle, DoctrineBundle, SecurityBundle, SwiftMailerBundle...

4.3 Diagramme de déploiement

Définition [20]

C'est une vue statique qui sert à représenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que leurs relations entre eux.

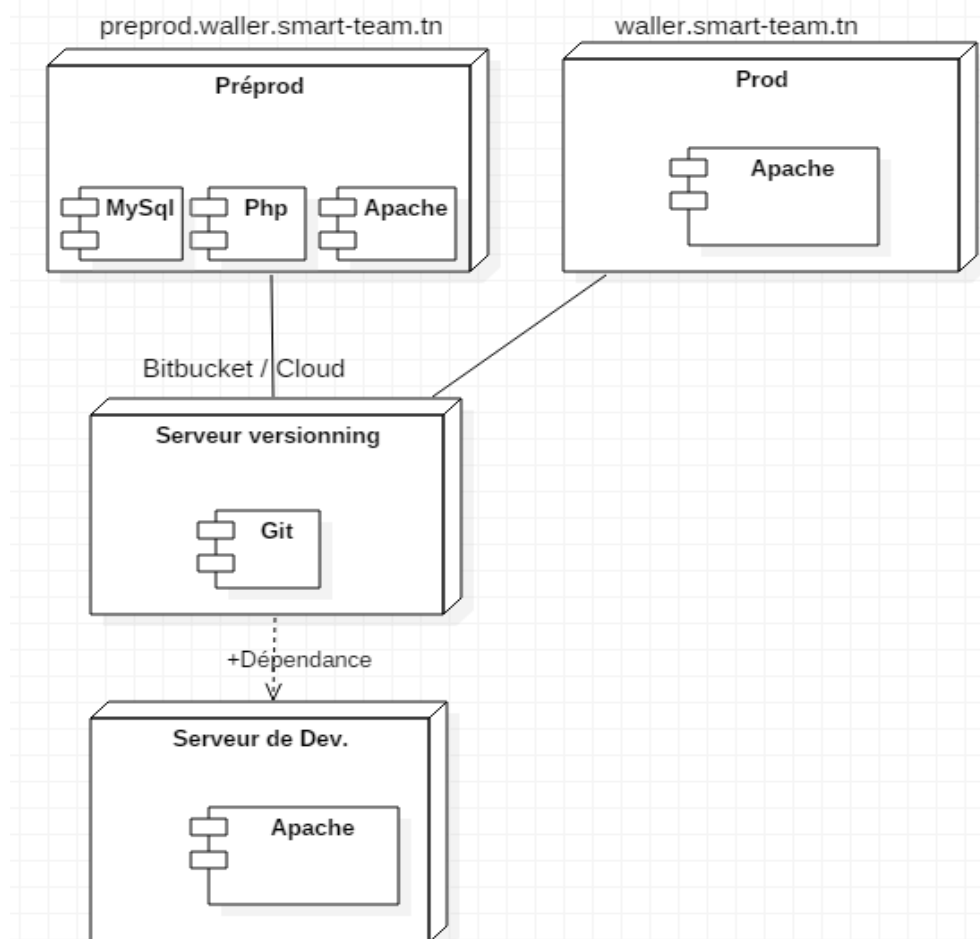


FIGURE 4.3 – Diagramme de déploiement

L'illustration de la réalisation du projet avec le diagramme de déploiement l'en résulte 4 nœuds : un représente le serveur de pré-production, un autre pour le serveur de production, un serveur de versionning (GIT/Bitbucket) et le dernier représentant le serveur de développement.

Ces 4 nœuds sont reliés entre eux à l'aide des association simple de communication et autres de dépendance.

Les serveurs de pré-production, de production et le serveur de développement sont dotés par un serveur Apache2.4 Php7.0 et MySql5.7 .

4.4 Processus de travail

4.4.1 Déroulement de la méthode Scrum

Pour bien démarrer un projet dans le cadre de Scrum, une réunion doit être programmée . Cette réunion aura comme but de définir le produit final à obtenir.

Durant mes premiers jours du stage, une réunion s'est tenue entre les différents hiérarchies Scrum de la société . A la fin cette réunion, la liste des fonctionnalités du prochaine application a été traduite en une liste détaillée de "US" (User Stories).

Ces US ont été enregistrés dans un Backlog, et estimés indépendamment avec Fibonacci¹. Ensuite, elles sont triées selon leur priorité .

Au cours de l'avancement de la réalisation, des réunions quotidiens (appelées daily Scrum) ont eu lieu pour appliquer le principe de défis . Cela m'a permis de garder toujours le courage et l'enthousiasme pour accomplir le projet dans de bonnes conditions.

A la fin de tout sprint une démo s'est déroulé, afin de vérifier l'achèvement des fonctionnalités, et de faire la rétrospective de sprint . Ces rétrospectives servent à faire la critique du travail accomplie et de détecter les anomalies dans le processus s'il y'en a pour l'améliorer.

4.4.2 Travail collaboratif avec GIT

Le travail avec des dépôts distant, est basé sur 2 branches :

- **master** : représente la dernière version qui est sur la production.
- **develop** : représente la dernière version qui est sur la pré-production.

Au cours de la réalisation du projet, une autre branche que la master et develop, sera créé pour tout User story . Cette branche nouvellement créée portera le nom de l'US .

1. La suite de Fibonacci : 0 1 2 3 5 8 13 21 34 55 89

Ces branches permet de garder des versions du code.

Pour créer une branche, il faut executer cette commande : ***git checkout -b nom_US***

Toutefois une US validée fonctionnellement (par le Product owner) et techniquement (par le Lead developer), sera fusionnée à la branche develop . La fusion de deux branche se fait à travers des Pull Requests ².

Une fois mise à jour, la branche develop sera mise sur le serveur du pré-production.

A la fin de chaque sprint, le contenu de la dernière version (validée) de la branche develop sera fusionnée à la branche master. Ensuite, ce contenu sera mis à son tour sur le serveur de production.

4.5 Réalisation

4.5.1 Critères ergonomiques

Lors de la création d'un site web, le coté esthétique est compté parmi les premiers critères de sa réussite. Ce dernier doit être facile et utile. Pour ce faire, il doit respecter l'ergonomie du web.

L'ergonomie du web se base sur 3 règles principaux, le site doit être Utile(répondre au besoin), Utilisable(par tous) et Agréable(beau à voir).

Au cours de la réalisation de ce projet, je me suis basé, dans la mise en œuvre des interfaces de l'application, sur les critères ergonomiques de Bastien et Scapin.

Guidage : à ce niveau j'ai essayé de créer une application web fournissant des *Feedback immédiat* (des réponses de validation d'actions doivent être lancer) et des informations *Lisible*.

Charge de travail : Les information fournit sont présenté en toute brièveté (limiter le travail de lecture).

Gestion des erreurs : L'utilisateur est guider à ne pas faire des erreurs. A titre d'exemple, le champs de numéro de téléphone dans le formulaire du profil, ne peut pas contenir une chaine de caractères. D'autre part, les messages d'erreurs fournissent une identification exacte du lieu et du type d'erreurs.

Controle explicite , Adaptabilité, Homogénéité, etc...

2. ou autrement appelées merge Request

4.5.2 Interfaces générale

Page d'authentification



FIGURE 4.4 – Interface de l'authentification

Cette page permet l'authentification des utilisateurs. L'authentification se fait par le nom de l'utilisateur et son mot de passe. Si l'utilisateur veut enregistrer son login et mot de passe il doit cocher la case "Se souvenir de moi". Après avoir remplie les champs nécessaire, l'utilisateur doit cliquer sur le bouton "Connexion". En cas d'une erreur de saisie ou les données saisie sont invalide, une notification s'affiche en haut de la page.

Le menu de navigation

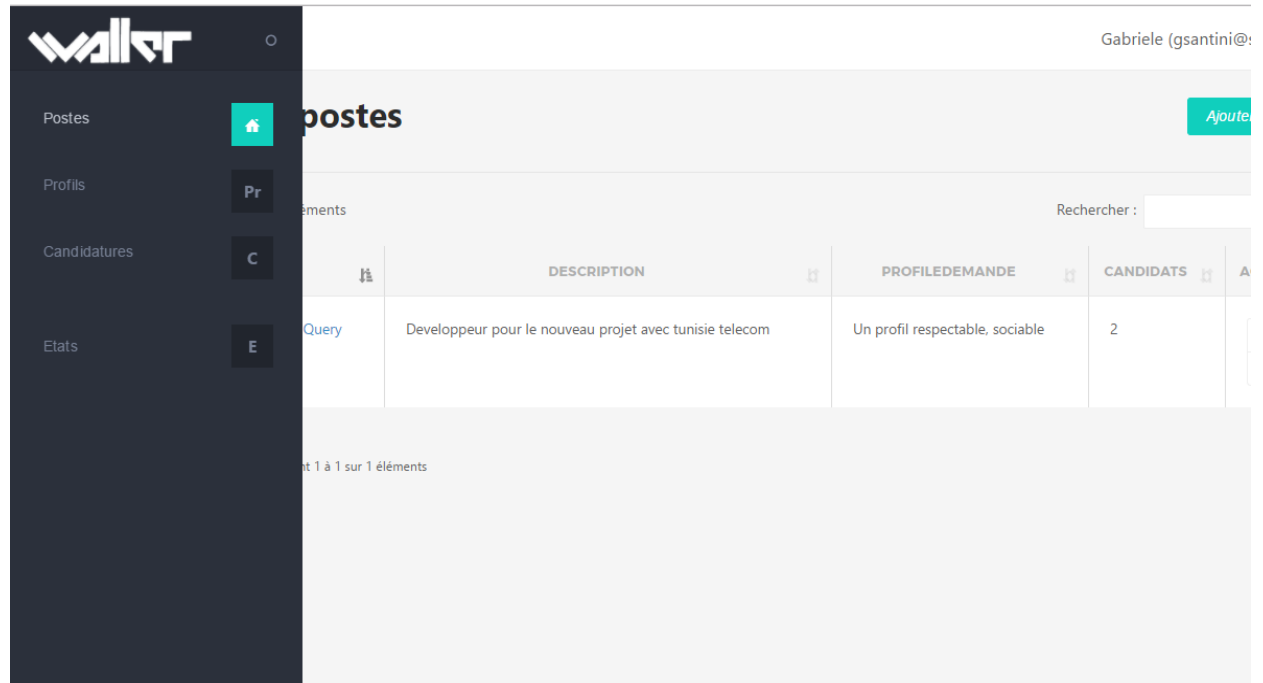


FIGURE 4.5 – Interface du menu

La densité des tableaux et du contenu des pages avait posé un gros problème dans le choix du menu de navigation. Donc, un menu avec un slidebar qui fait le va et viens selon la position du curseur était une solution pour ce problème.

4.5.3 Les pages concernant la gestion de Profil

La page de la liste des profils

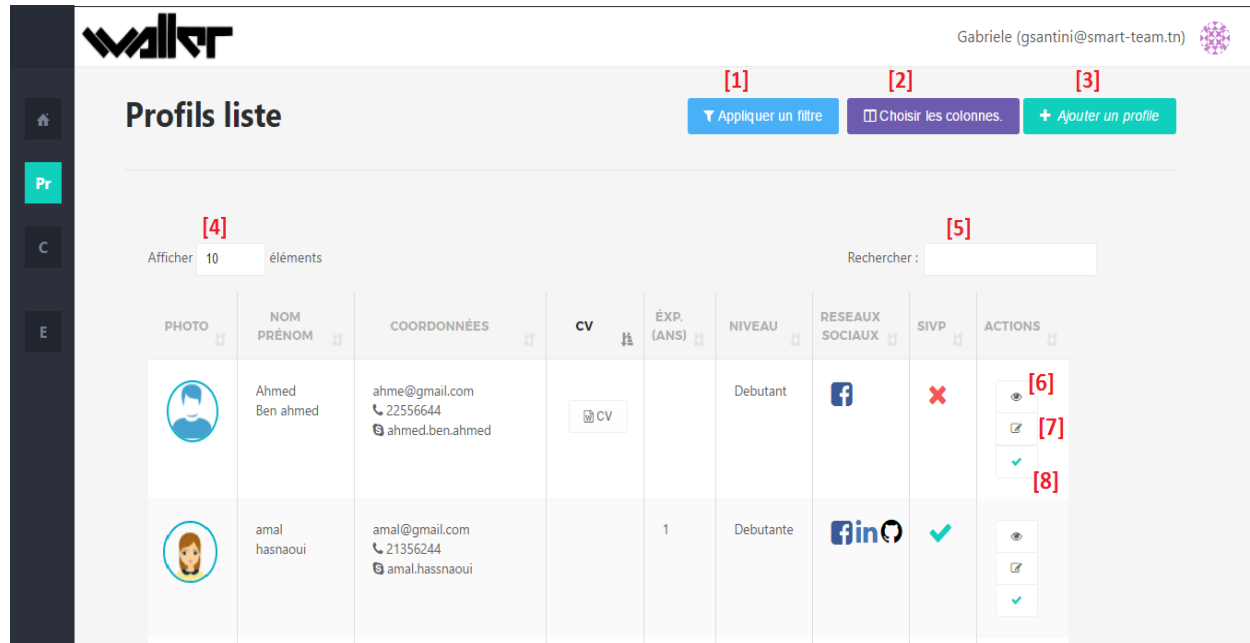


FIGURE 4.6 – La page de la liste des profils

Cette page représente la liste des profils. Les données de ces profils sont présentées dans un Datatable. Le nombre de profils dans chaque page peut être présenté avec 10 ou 25 ou 50 ou 100 profils par page [4]. On peut également faire une recherche sur le contenu des tables avec le champ "Rechercher" [5].

Le bouton "Appliquer un filtre"[1] permet d'appliquer un filtre multi-critère (voir Fig 4.9) sur les profils. Ce filtre permet bien évidemment d'obtenir une vision plus claire, et donc une facilité dans le choix des prochains candidats.

Les colonnes du tableau sont filtrables. On peut cacher des colonnes dont on ne veut pas les consulter. Ce filtre est obtenu en cliquant sur le bouton "Choisir les colonnes"[2].

Le bouton "Ajouter un profil"[3] permet de renvoyer la page du formulaire d'ajout d'un nouveau profil (voir Fig 4.7).

Le bouton [6] permet de renvoyer la page d'affichage détaillé de ce profil (Fig 4.8). En appuyant sur le bouton [7] une page de modification des données de ce profil sera ouverte. Par contre le bouton [8] permet de convoquer ce profil à un poste, et d'initier le Workflow de candidature.

La page de l'ajout d'un profil

The screenshot shows the 'Ajout d'un nouveau profil' page in the Walker application. The page has a dark sidebar on the left with icons for home, profile, and other sections. The main content area is titled 'Ajout d'un nouveau profil' and features a green 'Confirmer' button and a 'Retour à la liste' button. The form is divided into sections: 'IDENTITÉ' (with fields for 'PRENOM' and 'NOM'), 'PHOTO' (with a file upload button), 'CURRICULUM VITAE' (with a file upload button), 'CONTACT', 'COMPÉTENCES', and 'SITUATION'. Each section has a plus icon on the right side, indicating it can be expanded or collapsed.

FIGURE 4.7 – La page de l'ajout d'un profil

Cette page permet d'ajouter un nouveau profil. Les données d'un profil sont regroupées en 4 sections (4 Modales en accordéon) .

La première section regroupe les données identifiant un profil (prénom, nom, photo et CV). La deuxième section les moyens avec lesquelles on peut contacter ce nouveau profil (numéro téléphone, e-mail, skype, Facebook, github, et LinkedIn). La troisième section donne une information sur ses compétences, ses années d'expérience et son niveau. Enfin, la section situation, permettant de savoir sa disponibilité et la prestation salariale.

La bouton "Confirmer" permet de valider l'ajout du nouveau profil. Par contre le bouton "Retour à la liste" permet l'annulation de l'ajout et de retourner à la page de liste des profils (*Fig 4.6*).

La page de l’affichage du profil

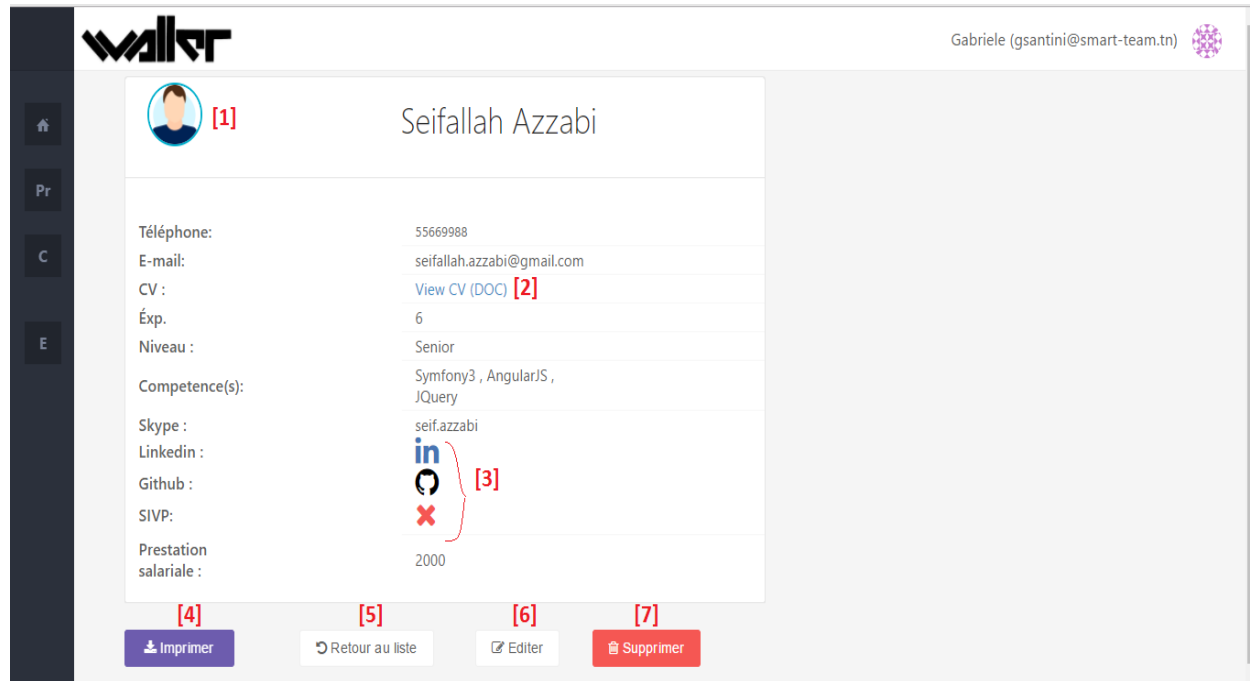


FIGURE 4.8 – La page de l’affichage du profil

Cette page donne une information détaillé sur le profil. Cette page est présenté sous forme d’une fiche de données imprimable en appuyant sur le bouton "Imprimer" [4]. On peut bien évidemment consulter le CV de ce profil (s’il y en a) à l’aide du lien [2]. D’autre part, un lien vers les réseaux sociaux [3] peuvent être obtenus (s’ils existent).

La photo de ce profil apparait en haut [1] pour faire se rappeler du profil lors des entretien . Si ce profil n’a pas de photo, une photo par défaut lui sera affectée (en anglais Placeholder).

Pour modifier quelques informations de cette fiche, on peut appuyer sur le bouton "Editer" [6], par contre pour supprimer ce dernier il faut choisir le bouton sur "Supprimer" [7] . Enfin, le bouton "Retour à la liste" [5] permettra de retourner à la page de départ (Fig 4.6) en annulant toutes les modifications non enregistrées.

La page de recherche des profils

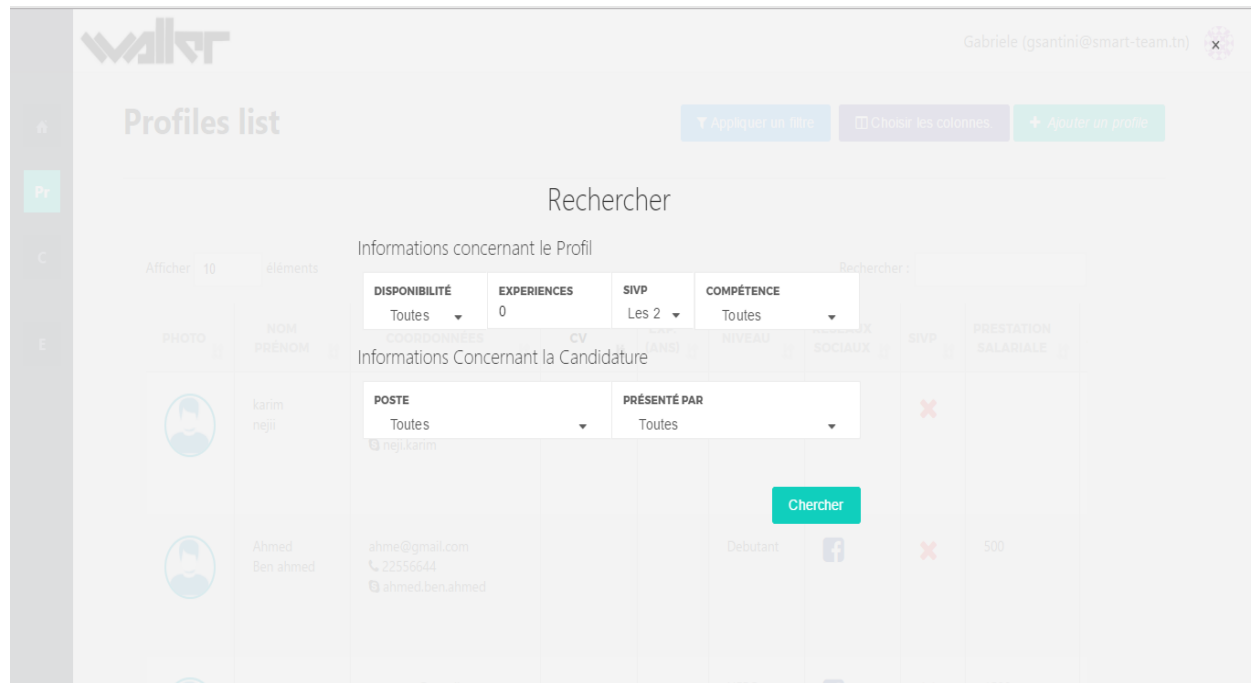
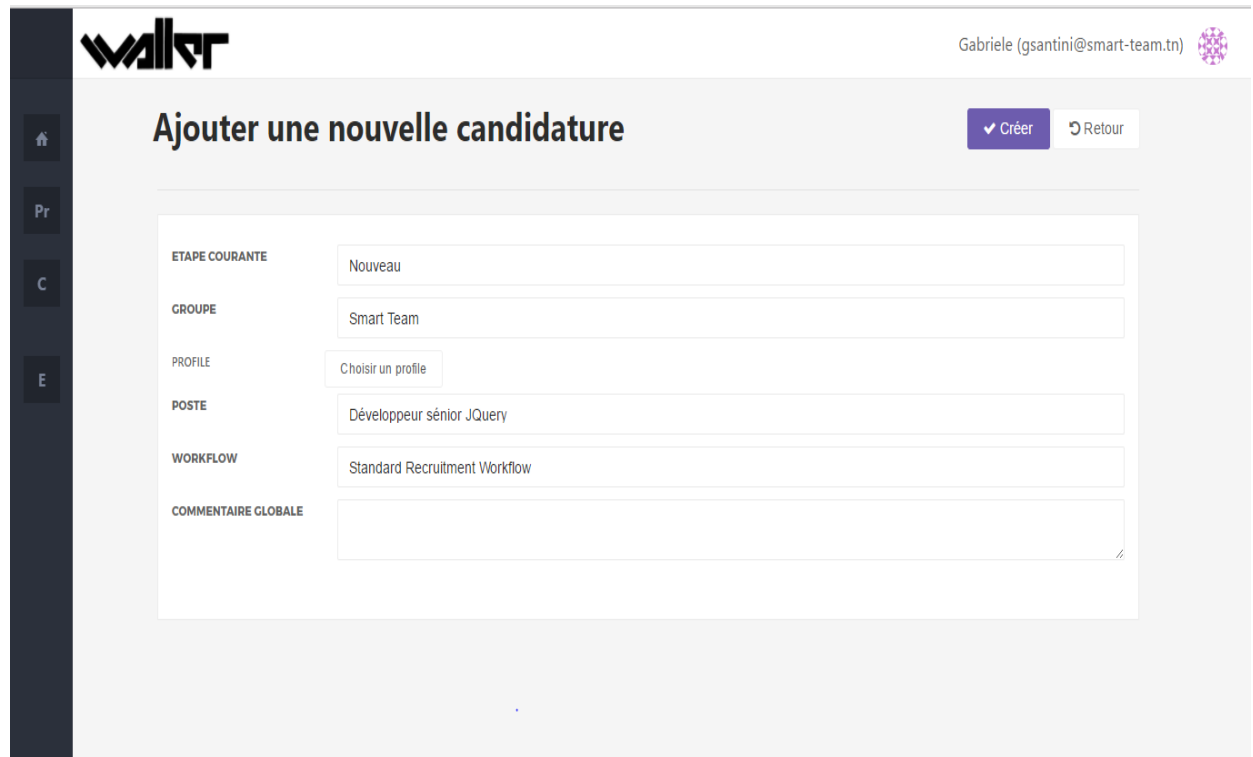


FIGURE 4.9 – La page de recherche des profils

Cette page présente la recherche multi-critère qu'on peut faire une recherche sur la liste des profils. La recherche peut se faire en définissant un ou plusieurs critères. Cette dernière peut s'appliquer soit sur les données du profil (Disponibilité, expérience, SIVP, ou compétence) ou sur des données des candidatures des profils convoqué au moins une fois (poste ou le groupe pour lequel ils ont été convoqué).

4.5.4 Les pages concernant la gestion de Candidature

La page de l'ajout d'une candidature



The screenshot shows the 'Ajouter une nouvelle candidature' (Add new application) page in the Walker application. The page has a dark sidebar on the left with navigation icons for Home, Profile, Candidates, and Employees. The main content area has a header with the Walker logo, the user's name 'Gabriele (gsantini@smart-team.tn)', and a profile icon. Below the header, the title 'Ajouter une nouvelle candidature' is displayed, followed by two buttons: 'Créer' (Create) and 'Retour' (Return). The form itself is titled 'ETAPE COURANTE' (Current Step) and contains several fields: 'GROUPE' (Group) with the value 'Smart Team', 'PROFILE' (Profile) with a dropdown menu showing 'Choisir un profil' (Choose a profile), 'POSTE' (Position) with the value 'Développeur sénior JQuery', 'WORKFLOW' (Workflow) with the value 'Standard Recruitment Workflow', and 'COMMENTAIRE GLOBALE' (Global Comment) with a text area.

FIGURE 4.10 – La page de l'ajout d'une candidature

Cette page permet d'ajouter une nouvelle candidature, en commençant par le choix de l'étape de départ, le profil et le groupe auquel le profil sera affecté. Ensuite, un choix du poste et du workflow doivent être fait. Enfin, le champs commentaire global permettra de joindre une remarque ou un commentaire sur la candidature.

Pour confirmer l'ajout, il faut cliquer sur le bouton créer, sinon pour quitter cette page il faut cliquer "Retour" et annuler l'ajout.

La page de l'édit de la candidature

FIGURE 4.11 – La page de l'édit de la candidature 1

Cette page permet la modification de la page de candidature. Elle est formé par deux formulaire :

- Le premier, celui du gauche présente le formulaire de modification des données du profil.
- Le deuxième consiste à modifier les données de la candidature. Cette modification symbolise la prise de décision envers ce candidat (soit pour aller à l'étape suivante, ou simplement pour saisir les notes).

Un historique des Notes et des Rapports est obtenue à l'aide des sections spécifiées . Cela donnera une information pour les décideur, sur le déroulement des entretiens précédents.

Pour confirmer l'édit, il faut cliquer sur le bouton "Enregistrer", pour supprimer cette candidature on peut cliquer sur "Supprimer", sinon pour quitter cette page le bouton "Retour à la liste".

Modification de la candidature

CANDIDATURE

ETAPE COURANTE

CV intéressant

COMMENTAIRE GLOBAL

Disponible durant les matinés

NOTE D'ETAPE

EVALUATION

0

LIBELLÉ RAPPORT DE CHANGEMENT D'ETAPE *

Historiques des Notes et Rapport de ce candidat

NOTES

Nouveau

Profil intéressante

Évaluation : 4

05-23-2017 20:08:29

RAPPORTS

Nouveau to Proposition de recrutement 2017-05-23

Profil sociale, expérience intéressante

FIGURE 4.12 – La page de l'édit de la candidature 2

Lors du changement de "étape courante" de la candidature, le champ "libellé rapport de changement d'étape" avec un contour de couleur rouge. Cette couleur indique l'obligation pour remplir ce champ.

L'historique des notes et des rapports sont représentés d'une façon lisible. Le nom de l'étape est écrit en premier et en gros, suivi de la libellé de la note, de l'évaluation, et de la date de prise de cette note. Tous les notes sont séparées les uns des autres par une ligne horizontale fin facilitant la différenciation entre elles.

La description d'un rapport commence par la libellé de l'ordre (*le libellé de l'étape source vers l'étape destination*), suivie par la date de prise de la rédaction du rapport, et le corps du rapport. Tous les descriptions des rapport sont séparées les unes des autres par une ligne fine facilitant la différenciation entre elles.

4.5.5 Les pages concernant la gestion des Postes

La page de l'ajout d'un poste

The screenshot shows the 'Création d'un nouveau poste' (Create new job) page in the Walker application. The page has a dark sidebar on the left with icons for home, profile, create, and edit. The main content area is titled 'Création d'un nouveau poste' and includes a 'Créer' button and a 'Retour à la liste' link. The form contains several input fields: 'LIBELLÉ POSTE', 'DESCRIPTION', 'PROFIL DEMANDÉ', 'PROPOSÉ PAR' (pre-filled with 'Smart Team'), and 'LIENS DE RECRUTEMENT'. The 'LIENS DE RECRUTEMENT' section has two sub-fields: 'LIBELLÉ LIEN' (pre-filled with 'Keejob') and 'LIEN DU POSTE' (pre-filled with 'http://www.keejob.tn'). A plus sign button is located below these fields.

FIGURE 4.13 – La page de l'ajout d'un poste

Cette page permet d'ajouter de nouveaux postes. L'ajout d'un nouveau poste commence par la libellé du poste, sa description, et la description du poste. On devra bien évidemment mentionner le groupe qui a demandé ce poste, ainsi que d'ajouter quelques liens de recrutement dans les sites d'offre d'emplois. Le bouton (+) permet d'ajouter d'autres liens. Ces liens permettront de garder une trace, où le poste a été déposé dans les sites d'emploi.

Pour confirmer l'ajout, il faut cliquer sur le bouton créer, sinon pour quitter cette page il faut cliquer "Retour à la liste".

La page de l’affichage d’un poste

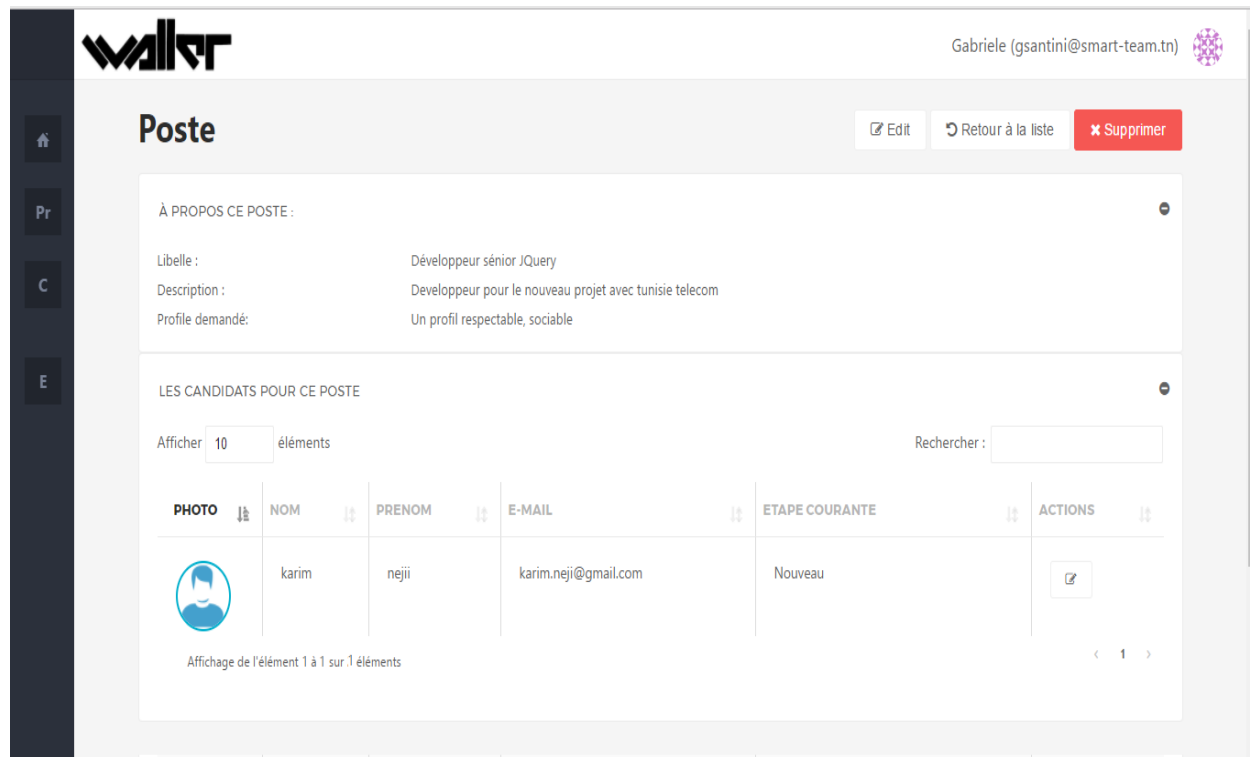


FIGURE 4.14 – La page de l’affichage d’un poste

Cette page donne une information détaillé sur le poste . Elle est dotée aussi d’un tableau représentant les candidats convoqués pour ce poste, avec leurs informations (nom, prénom, e-mail, et étape courante).

Pour éditer ce poste, il faut cliquer sur le bouton "Edit", pour supprimer ce poste on peut cliquer sur "Supprimer", sinon pour quitter cette page le bouton "Retour à la liste".

4.6 Conclusion

La partie réalisation est le fruit des études et des travaux effectués tout au long de ce projet. Ce dernier chapitre a été consacré à l’explication de la structure du projet ainsi que son diagramme de déploiement. J’ai clôturé ce chapitre par des imprimes écran de ma réalisation, tout en mettant en valeur les critères ergonomiques dans ces choix des pages ainsi que leurs styles.

CONCLUSION GÉNÉRALE

Ce projet de fin d'étude, qui s'est déroulé au sein de la société Smart Team, avait pour but la conception et le développement d'une application Web de recrutement. Cette solution apportera une valeur ajoutée à la société, en lui garantissant de faire le choix idéal des futurs membres.

La réalisation de ce projet avec PHP m'a permis non seulement d'enrichir mes connaissances dans le développement d'applications Web, mais aussi de me confronter aux bonnes pratiques et les différentes normes de développement, et de les respecter. Cela m'a permis aussi de maîtriser les Framework Symfony et la librairie jQuery, ce qui a été un réel challenge pour moi.

Ce stage était une occasion pour approfondir mes connaissances théoriques, acquises durant mon parcours universitaire, par la pratique de nouvelles technologies et méthodologies. D'autre part, cette expérience m'a offert la chance de maîtriser Git et Bitbucket, et de découvrir les bases du travail collaboratif avec Scrum.

D'un point de vue personnel, l'expérience au sein de Smart Team était pour moi une occasion pour développer les capacités de communication dans un environnement professionnel.

A titre de conclusion, l'application que j'ai développée pourrait être enrichie par l'ajout d'un module de gestion du workflow. Ce dernier, va permettre à l'entreprise de créer de différents workflow d'entretiens spécifique à chaque poste ou entreprise.

Cette application pourra aussi fournir des questionnaires d'évaluation de compétence en ligne, que suite à des contraintes de temps j'ai pas eu l'occasion de le développer, mais c'est un objectif qui reste prioritaire dans l'avenir. Ces questionnaire seront proposés au candidats durant des entretiens spécifique, pour évaluer leurs compétences. Ces test de compétences, une fois corrigé ils vont donner une vision plus claire au décideurs.

ANNEXE

Premier sprint

ID	Sprint	Libellé	Détail	Complexité	Statut
#1	1	En tant que utilisateur, je dois avoir la possibilité d'ajouter un nouveau profile	champs selon feuille excel	5	Recette ▼
#2	1	En tant que utilisateur / client, je dois pouvoir éditer le profil d'un candidat.	-Ajouter des information. -Supprimer le candidat.	3	Recette ▼
#3	1	En tant que utilisateur / client, je dois pouvoir visualiser une liste de candidature	filtrer les colonnes à visualiser	5	Recette ▼
#4	1	En tant que utilisateur, j'affecte un candidat à un poste.	Démarrage de la candidature	2	Recette ▼
#5	1	En tant qu'utilisateur, je peux faire une recherche selon l'état du candidat	- Nouveau - Refusé - En attente - En attente d'entretien - Entretien programmé - Proposé au client - Recalé - Pas intéressé - En étude - CV proposé - Entretien planifié - CV intéressant - Recturement demandé - Proposition de recrutement - Recruté	5	Recette ▼
#6	1	En tant que utilisateur, je dois pouvoir uploader des fichiers (pdf, word) dans le profil du candidat.		2	Recette ▼

FIGURE 4.15 – Déroulement du premier sprint

Date : Du 17 Février jusqu'au 28 Février .

Description : Après la mise en place des entités au sprint 0 .

En premier lieu, ce sprint a été consacré pour la mise en place des CRUD¹ des entités Profil et Candidature.

En deuxième lieu, l'adaptation des listes des profils et des candidatures au plugin Datatables (de jQuery), ainsi que l'utilisation du VichUploaderBundle pour la gestion des

1. CRUD : pour Create, Read, Update, Delete.

fichiers(Photo et CV) des profils.

Les points de complexité de ce sprint est compris entre 2 et 5 points.

Deuxième sprint

ID	Sprint	Libellé	Détail	Complexité	Statut
#7	2	Ajouter un champs text "commentaire" dans la candidature		1	Recette
#8	2	En tant que utilisateur, je dois pouvoir prendre des notes, commentaires, évaluations dans chaque étape.		3	Recette
#9	2	Amélioration du design sur Edit Profile	Ajouter 4 zones collapse : 1- Identité : Prénom - Nom Photo - CV 2- Contact : Téléphone - Email - Skype Facebook - Github - LinkedIn 3- Compétences Compétences - Expérience Niveau - Formation 4- Situation SIVP - Disponibilité Prestations Salariales	5	Recette
#10	2	Pouvoir visualiser le CV et la Photo lors de l'édit d'un profile (et candidature)		1	Recette
#11	2	Comportement "tags" pour les compétences, si une compétence n'existe pas, il faut l'ajouter en base		5	Recette
#12	2	Ajouter le CRUD de Poste et Etapes (dans Poste ajouter la possibilité de naviger vers les candidatures de ce poste là)		5	Recette
#13	2	Poste : Liste des liens de l'offre d'emploi (dans LinkedIn, TanitJob, Keejob, etc...)		5	Recette
#14	2	Ajout d'une note (rapport) à chaque changement d'étape	surtout pendant l'entretien!!	5	Recette
#15	2	En tant que utilisateur, lors de chaque étape, je peux voir les détails des rapports, notes et évaluations des précédentes étapes.	afficher l'historique des notes + rapports dans l'ordre	2	Recette

FIGURE 4.16 – Déroulement du deuxième sprint

Date : Du 1 Mars jusqu'au 22 Mars .

Description : En premier lieu, ce sprint a été consacré pour la correction des anomalies remarqué suite au au demo du premier sprint et appliquer des amélioration dans le design des pages réalisés.

En deuxième lieu, pour la mise en place des CRUD des entités Poste (avec possibilité

d'ajout des liens vers les publications) et Étape.

En troisième lieu, au cours de ce sprint j'ai ajouté la possibilité de rédiger des commentaires lors de la création de la candidature.

Les points de complexité de ce sprint est compris entre 1 et 5 points.

Troisième sprint

ID	Sprint	Libellé	Détail	Complexité	Statut
#16	3	En tant que utilisateur, je dois m'identifier afin d'accéder au logiciel.	Installation et configuration de FOS User Bundle Chargement de fixtures de users avec les bons rôles	3	Recette
#17	3	Je dois pouvoir cacher différentes informations du profile / candidature selon le profile de la personne connectée	MVP Géré en base selon les rôles / groupes	2	Recette
#18	3	En tant que administrateur, je peux faire une recherche selon la disponibilité du candidat	Recherche par Disponibilité	2	Recette
#19	3	En tant que administrateur, je peux faire une recherche selon les compétences du candidat	Recherche par Compétences	3	Recette
#20	3	En tant que administrateur, je peux faire une recherche selon le note globale du candidat	Recherche par Note globale	2	Recette
#21	3	En tant que administrateur, je peux faire une recherche selon l'expérience du candidat	Recherche par années d'expérience	1	Recette
#22	3	En tant que administrateur, je peux faire une recherche selon une fourchette des prestations salariales du candidat	Recherche par prestations salariales	1	Recette
#23	3	En tant que utilisateur, je dois évaluer les compétences du candidat.	Interface ergonomique (tagging) des compétences du candidat (disponible à chaque étape du workflow) Exemple : PHP(+), Javascript(++), MySQL(---)	5	Recette

FIGURE 4.17 – Déroulement du troisième sprint

Date : Du 28 Mars jusqu'au 13 Avril .

Description : Au cours de ce sprint, j'ai mis en place le Bundle FOSUserBundle qui assure la gestion des utilisateurs (authentification, registre et droits d'accès).
En suite, des recherches multiples ont été appliqués sur la liste des profils.

Quatrième sprint

Date : Du 18 Avril jusqu'au 29 Avril .

Description : Ce sprint consistait à appliquer des améliorations sur toute l'application.

Bibliographie

- [1] “inser rh.” <http://www.inserrh.com/>.
- [2] “Candidatus.” <http://www.candidatus.com/>.
- [3] “Beetween software.” <http://www.beetween.fr/>.
- [4] “Uml.” [https://fr.wikipedia.org/wiki/UML_\(informatique\)](https://fr.wikipedia.org/wiki/UML_(informatique)).
- [5] “Staruml.” <https://fr.wikipedia.org/wiki/StarUML>.
- [6] “sql.” https://fr.wikipedia.org/wiki/Structured_Query_Language.
- [7] “Mysqlworkbench.” https://fr.wikipedia.org/wiki/MySQL_Workbench.
- [8] “Git.” <https://fr.wikipedia.org/wiki/Git>.
- [9] “bitbucket.” <https://fr.wikipedia.org/wiki/Bitbucket>.
- [10] “Symfony3.” <https://fr.wikipedia.org/wiki/Symfony>.
- [11] F. POTENCIER, *Symfony ,Simplifier vos développement Web professionnels avec PHP*, 2007.
- [12] “doctrine.” <http://www.mti.epita.fr/blogs/2010/12/30/quest-ce-quun-orm-faisons-un-tutorial-surdoctrine/>.
- [13] “twig.” <http://www.siteduzero.com/informatique/tutoriels/utilisation-de-twig-un-moteur-detemplates>.
- [14] “jquery.” <https://fr.wikipedia.org/wiki/JQuery>.
- [15] “phpstorm.” <https://fr.wikipedia.org/wiki/PhpStorm>.
- [16] “Scrum.” [https://fr.wikipedia.org/wiki/scrum_\(Boite_à_outils\)](https://fr.wikipedia.org/wiki/scrum_(Boite_à_outils)).
- [17] “Diagramme de cas d’utilisation.” https://fr.wikipedia.org/wiki/Diagramme_des_cas_d%27utilisation.
- [18] “Diagramme des classes.” https://fr.wikipedia.org/wiki/Diagramme_de_classes.
- [19] “Diagramme d’activité.” https://fr.wikipedia.org/wiki/Diagramme_d%27activit%C3%A9.
- [20] “Diagramme de déploiement.” https://fr.wikipedia.org/wiki/Diagramme_de_d%C3%A9ploiement.