|  |
| --- |
|  |
| Rapport de stage |
| Projet Suivi de stagiaire |
|  |
| **Harlé Aurélien** |
| **09/06/2017** |

|  |
| --- |
| Ce document présente le stage effectué en entreprise par Mr. HARLÉ Aurélien, on y abordera la société qui m’a accueillie pour mon stage, le projet que j’ai réalisé pendant ce stage ainsi que l’apport de ce stage au niveau personnel. |

Contenu

[La société 2](#_Toc488677123)

[Remerciement 2](#_Toc488677124)

[Présentation 2](#_Toc488677125)

[Organigramme 2](#_Toc488677126)

[Contexte de stage 2](#_Toc488677127)

[Le projet 3](#_Toc488677128)

[Contexte du projet 3](#_Toc488677129)

[Présentation Projet 3](#_Toc488677130)

[Suivi chronologique 4](#_Toc488677131)

[Planification 5](#_Toc488677132)

[Cahier des charges 6](#_Toc488677133)

[Documentation 6](#_Toc488677134)

[Analyse 6](#_Toc488677135)

[La fonctionnalité auto-évaluation 6](#_Toc488677136)

[Conception 8](#_Toc488677137)

[Les tests unitaires 8](#_Toc488677138)

[Conclusion 8](#_Toc488677139)

[L’apport personnel 9](#_Toc488677140)

[Retour d’expérience 9](#_Toc488677141)

[Axes d’amélioration 9](#_Toc488677142)

[Les Annexes 9](#_Toc488677143)

# La société

## Remerciement

Je tiens à remercier toutes les personnes qui ont contribué au succès de mon stage et qui ont su m’aider lors de celui-ci.

Je tiens à remercier vivement mon maitre de stage, Mme Dominique Muller, Formatrice de la formation CDI au sein de l’agence AFPA, pour son accueil, le temps passé ensemble et le partage de son expertise au quotidien. Grâce aussi à sa confiance j'ai pu m'accomplir totalement dans mes missions. Il fut d'une aide précieuse dans les moments les plus délicats.

Je remercie également toute l’AFPA pour leur accueil, leur grande dévotion, ce qui m'a permis de pouvoir réaliser ce stage sans encombre.

## Présentation

L'Agence pour la formation professionnelle des adultes ou AFPA est un organisme de formation professionnelle dont la création remonte en 1949, même si à l’heure actuelle l’AFPA vie une nouvelle étape dans son histoire, du fait de sa transformation en EPIC, Etablissement Public Industrielle et Commercial, l’Association nationale pour la formation professionnelle des adultes devient l’Agence nationale pour la formation professionnelle des adultes, malgré sa transformation elle reste compétitive sur le marché, car elle engendre un chiffre d’affaire de pas moins de 750M€ environ, chaque année elle voit passer environ 150.000 stagiaires qui ont pour but d’être formé, pour cela l’AFPA compte pas moins de 8000 employés.

Aujourd’hui, cette agence est l'un des principaux intervenants dans les dispositifs de formation professionnelle : plan de formation, contrat et période de professionnalisation, VAE, CIF et DIF …

Elle accueille à la fois des demandeurs d’emploi à la recherche d’une reconversion professionnelle et des salariés en quête de perfectionnement dans leur métier, mais aussi des actifs en situation particulière (personnes handicapées, détenus…).

Du fait de l’ampleur de cette société, nous allons plus principalement nous intéresser à l’Afpa de Marseille Saint-Jérôme et à la formation CDI concepteur développeur en informatique, orchestrer par Dominique Muller où j’ai réalisé mon stage.

## Organigramme

## Contexte de stage

Du fait du contexte de mon stage, nous avons décidé pour bien séparer les différents rôles que joue mon tuteur de stage, client, formatrice et de passer obligatoirement par e-mail pour tous les parties concernant le client, les différentes validations fonctionnelles, prise de rendez-vous, organisation de réunion et présentation des release livrable.

# Le projet

## Contexte du projet

Lors de mon arrivée en stage et en connaissant bien le contexte de la formation CDI, je savais que plusieurs projets étaient susceptibles d’être réalisé, afin de facilité le quotidien de Mme Dominique Muller, je lui ai donc fait la proposition de trois projets réalisable, qui pourrait l’intéresser, du fait de son rôle de formateur elle a décidé de prioriser le projet suivi de stagiaire, qui lui permettra de simplifier son quotidien, mais surtout de simplifier le quotidien des stagiaires des prochaines formations.

Ce projet est réalisable et simplifierai son quotidien car à l’heure actuelle le suivi des stagiaires au cours de leurs formations n’est pas informatisés sur une plateforme WEB, permettant de n’avoir qu’un seul point d’entrée pour tout le suivi des stagiaires.

(Annexe 1)

## Présentation Projet

Le projet suivi de stagiaire, est une plateforme web qui permettra au stagiaire ainsi qu’au formateur référent, un suivi simplifié et complet de leur formation, pour leurs permettent de suivre la formation dans les meilleures conditions possibles.

C’est dans cet objectif que ce projet comprendra, le bilan des différents ECF, avec un système de remarques et d’évaluation, les notes des ECF, ainsi que leur progression personnelle et leur niveau d’acquisition des cours qu’ils ont reçus, et de leurs progressions sur l’exercice en cours.

Tout ceci leur permettra de mieux ciblé les compétences et les efforts qu’ils vont devoir accomplir afin d’être en capacité d’obtenir leur certification.

Un système de rendez-vous entre les stagiaires et le formateur réfèrent seras aussi mis en place, afin de faciliter les échanges entre eux.

Ce projet contiendra aussi toute la partie administrative qui peut incomber un stagiaire, afin de leur faciliter, leurs échanges avec l’AFPA, il y aura aussi un formulaire d’absence mise à la disposition des stagiaires.

Une feuille interactive sera mise à disposition des stagiaires, afin de pouvoir faciliter leurs suivis de recherches de stage, elle permettra notamment, un suivi des entreprises contacter, avec leurs noms, leurs sites, l’adresse e-mail du contact, le numéro de téléphone du contact, le nom du contact et un suivi de progression des échanges avec l’entreprise, un système de couleur seras mis en place, selon l’état de progression des échanges avec l’entreprise, les stagiaires pourront aussi mettre les différents CV et lettre de motivation utiliser pour leur recherche de stage.

Une liste des entreprises qui recrutent des stagiaires sera mise à disposition du formateur, pour lui permettre de la tenir à jour et de pouvoir la divulguer aux personnes en difficulté dans leurs recherches de stage.

Il serrât mis à disposition des stagiaires une zone bac à sable, afin de publier leur idée et retour sur la formation.

Un système de notification directement intégré au site ou par envoie d’e-mail seras mis en place, afin de pouvoir informer les stagiaires ainsi que le formateur, qu’il y a des points à prendre en compte sur la plateforme.

## Suivi chronologique

J’ai dans un premier temps, mis en place une organisation de dossier, afin de m’y retrouver facilement et d’éviter tous les problèmes d’organisation que je pourrais rencontrer,

J’ai ensuite mis en place un script de sauvegarde de tout le projet, qui est exécuté toutes les heures, afin d’éviter tous soucis de perte de données, aussi afin de rendre ces sauvegardes effectives je les ai récupérées manuellement sur un matériel déporté.

J’ai ensuite écris une première version du planning général pour le client afin de le tenir au courant des dates prévus en rapport à la visibilité que j’avais sur le projet, ainsi qu’une premier itération de mon planning détaillé comprenant la première itération nommez « Documentation du projet ».

J’ai ensuite commencé la rédaction du cahier des charges afin de pouvoir le faire valider par le client, ce qui s’est fait lors d’une première réunion de présentation du premier cahier des charges qui fut ensuite corrigé et validé à la suite de plusieurs e-mail.

J’ai par la suite décidé d’étudier l’outil de versioning Jenkins, que je souhaitai mettre en place sur le projet, l’ayant entrevue pendant mon temps libre et trouvant ses fonctionnalités intéressantes au niveau de sa remontée sur le serveur avec automatisation des tests unitaires et des différents plugin qu’il peut proposer, mais je souhaitai aussi pouvoir le relier à la plateforme Github pour me permettre d’avoir accès au projet depuis internet mais l’API Github qui est atteint par Jenkins, est très limité au niveau des remontés depuis un serveur tiers pour les comptes Github non professionnelle, j’ai donc dû abandonner de mettre en place un serveur Jenkins.

C’est pour cela que je n’utiliserais sur ce projet seulement Github comme outil de versioning.

Une fois ceci fait, j’ai commencé l’analyse et la conception de l’application (voir section analyse et conception pour plus d’information), une fois l’analyse réalisée j’ai rédigé les droits des différents utilisateur sur les différentes fonctionnalités de l’application et demander au client de valider les différents droits ce qui s’est fait à la cour de différents e-mail.

Par la suite j’ai commencé l’installation et la configuration de mon serveur de base de données, j’ai décidé d’utiliser Oracle XE l’ayant étudié sur toute la période de ma formation, j’ai donc créé une database et un tablespace dédier au projet suivi de stagiaire dans un souci de gain de temps j’ai créé qu’un unique utilisateur ayant tous les droits sur ceci, il faudrait par la suite créer, un compte dédier à l’application avec les autorisations nécessaires.

Puis une fois l’analyse terminée et la base de données créé, j’ai mis en place l’outil de versioning Github, ensuite j’ai commencé à organiser le projet selon une architecture que je n’avais pas encore expérimenté, c’est-à-dire de créer un package par fonctionnalité et des packages généraux, ce qui ne fut pas convaincant et modifier par la suite dans le projet par une réorganisation par entité du projet (exemple : auto-évaluation). Une fois l’organisation de l’architecture du projet mis en place j’ai ensuite commencé à coder les différentes entités du projet, ne me limitant pas seulement à l’auto-évaluation, mais aussi à tout ce qui pouvait l’entourer ce qui fut une erreur de ma part et changé par la suite, une fois toutes les entités misent en place j’ai ensuite commencé à coder la couche d’accès aux données, mais comme pour les entités je ne me suis pas concentré seulement sur l’auto-évaluation, en transverse à chaque fois que j’ai fini une méthode de la couche d’accès aux données j’ai réalisé les tests unitaires lié à cette fonctionnalité et ainsi directement corriger les erreurs et mis en place les différents contrôles nécessaire pour répondre aux différentes règle métier lié à l’auto-évaluation en même temps que la couche DAO j’ai mis en place toute la couche service ainsi que les différentes façades à chaque couche de mon application, mes façades à l’heure actuelle ne me conviennent pas et mériterai d’être retravaillé afin de pouvoir découplés les différentes couches de l’application, en mettant en place le design pattern du contrôleur, mais ceci n’est pas réalisé par manque de temps.

Une fois, les différentes couches mises en place, je me suis affairé à la mise en place de la couche présentation, qui permet à l’utilisateur d’utiliser les différentes fonctionnalités de l’application, toutes les interactions de l’utilisateur avec l’application sont aux préalable contrôler afin d’éviter le plus d’erreurs possible.

(Annexe 2 à 4)

## Planification

#### Méthodologie

J’ai décidé sur ce projet, de réaliser une planification générale qui permet au client d’avoir une vue global sur l’avancement du projet et des délais que je me suis fixé, mais aussi une planification détaillée qui me permet de gérer l’état d’avancement du projet de façon plus réel.

Cette planification détaillée a était réalisé en plusieurs itérations la premier étant toute la documentation et la prise d’information sur le projet, la deuxième itération étant tout l’aspect analyse et conception et enfin une troisième itération sur tout ce qui est aspect technique du projet, cette manière d’aborder le planning détaillée me permet de mieux entrevoir le projet a chaque itération et ainsi d’être plus juste au niveau de ma planification, j’ai mis en place cette méthodologie après mes différentes expériences et des problèmes que j’ai plus rencontrés sur les précédentes planifications de projet que j’ai plus réalisée.

(Annexe 5 à 6)

#### Les bienfaits de la planification

La planification m’a permis de réaliser que je n’étais pas dans les temps au niveau de l’analyse de l’application, ce qui m’a permis de prévoir un entretien avec mon tuteur de stage à la suite de cette entretien, j’ai réalisé qu’étant donné que je ne me limitais pas à la méthodologie Merise ainsi que de réaliser de petite itération, je perdais énormément de temps, tout ceci m’a permis de recadrer l’analyse du projet, par plus petite fonctionnalité plutôt que de vouloir analyser toute l’application en une seule foi.

J’ai aussi réalisé grâce à la planification que le temps que je m’accordai pour développer une fonctionnalité était bien trop cours, j’ai donc décidé suite à un entretien avec mon tuteur de réalisé la fonctionnalité a son plus petit besoin fonctionnel et de ne pas développer tout ce qui peut l’entouré grâce à cela j’ai plus réalisé une première itération sans retard.

## Cahier des charges

J’ai commencé à écrire un premier jet du cahier des charges une fois satisfait par ce cahier des charges, j’ai organisé une réunion avec le client afin de pouvoir valider celui-ci, au cours de cette réunion le client a plus pointé les choses qui n’allaient pas sur ce cahier des charges et aussi mieux m’expliquait le domaine métier de l’application j’ai donc le cahier des charges, selon les attentes du client et lui ai retourné la version modifier de celui-ci afin de le faire valider, après quelque petite modification sur ce cahier des charges nous avons plus le valider et j’ai donc commencé par réunir la documentation liée au projet.

(Annexe 4)

## Documentation

J’ai donc demandé au client de me fournir tous les documents qu’il pouvait m’apporter qui était en lien avec l’application, une fois ces documents reçus je les ai étudiés afin de comprendre le domaine métier de l’application, ce qui m’a permis de réaliser que certain document manqué et j’ai donc demandé les documents manquant au client une fois les documents analysés et compris j’ai commencé l’analyse de l’application.

(Annexe 7 à 12)

## Analyse

J’ai commencé une première analyse, en suivant le même schéma que décrit par la suite, mais sans découpé mon analyse en petite itération de l’application, cette analyse c’est donc révélée fausse infructueuse et laborieuse, c’est pourquoi après une réunion avec mon tuteur de stage, il m’a conseillé de découper mon analyse en plus petite itération et ma conseiller de commencer par l’étude de la fonctionnalité d’auto-évaluation.

(Annexe 13 à 27)

### La fonctionnalité auto-évaluation

#### Liste des exigences

J’ai commencé avant d’analyser la fonctionnalité par écrire la liste des exigences, basé sur les règles métiers que j’ai plus identifiés, j’ai tout au long du développement de cette fonctionnalité tenue à jour cette liste avec les différentes règle qui me sont remontées et qui on était validé par le client au préalable.

(Annexe 13)

#### Cas d’utilisation

Je suis ensuite passé à l’écriture des cas d’utilisation, j’ai tout d’abord construit le niveau le plus haut des cas d’utilisations (nommé UC0 dans mon analyse) et ensuite créer les sous UC se trouvant dans mon UC0, et détailler seulement l’UC qui m’intéressait c’est-à-dire l’UC1.3 - Auto-évaluation du stagiaire sur les compétences requises, j’ai limité l’étude des cas d’utilisation au cas nominaux à cause des délais fournis pour développer cette application.

(Annexe 14 à 22)

#### Dictionnaire de données

Pour analyser la fonctionnalité auto-évaluation j’ai commencé par écrire le dictionnaire de données, en étudiant tous les documents que j’avais à ma disposition pour effectuer l’analyse, j’ai ressorti, selon moi tout ce qui devait être persisté dans l’application, j’ai aussi tenu à jour ce document lorsque j’ai rajouté ou renommé des attribues.

(Annexe 23)

#### Modèle conceptuelle de données

Grâce à ce dictionnaire de données j’ai ensuite construit le MCD de la fonctionnalité, j’ai réalisé les liens entre les différentes entités et posé leur cardinalité le nombre d’entité étant plutôt restreint je n’ai pas eu de mal à le réaliser, par la suite comme pour le dictionnaire de données j’ai tenu à jour le MCD.

(Annexe 24)

#### Modèle logique de données

Une fois le MCD construit, j’ai réalisé le MLD à ce moment-là, j’ai décidé comment les jointures entre mes entités devraient être, j’ai mis en place un lien identifiant entre les séquences et les modules (vers séquence) ainsi qu’un lien entre compétences et séquences (vers compétence), afin de coller au plus à l’existant les autres jointures sont quant à elle simple.

(Annexe 25)

#### Modèle physique de données

Par la suite j’ai réalisé le MPD où j’ai plus, mettre mon modèle en correspondance avec le type de base de données que j’allais utiliser (oracle XE), et ainsi typé mes colonnes.

(Annexe 26)

#### Diagramme de classe

Une fois tous les modèles créés, je suis passé à la creation de mon diagramme de classe, où j’ai repris toutes mes entités avec la convention de nommage du langage Java, ainsi que définit toutes les méthodes que c’est entités devrais avoir.

(Annexe 27)

## Conception

### Architecture

Sur ce projet j’ai décidé de mettre en place une conception N-tiers, afin de pouvoir profiter de toute la robustesse que cette conception apporte tant au niveau de la maintenabilité du projet cas l’évolution de celui-ci.

J’ai donc pour cela découper mon projet en plusieurs couches, une première couche de présentation qui est basé sur le modèle MVC qui permet toutes les interactions de l’utilisateur avec le système, ensuite une couche applicative qui permet de rediriger les demandes de l’utilisateur à une dernière couche, d’accès aux données qui permet de persister toutes les données de l’utilisateur vis-à-vis du système.

### Technologie

Du coter fonctionnelle il y a deux serveurs, un premier serveur de base de données sous Oracle XE, qui permet la persistance des données, et un serveur applicatif sous Wildfly 10, qui permet de gérer tout le côté applicatif du projet.

J’ai aussi utilisé les librairies suivantes

Bootstrap

Qui me permet d’appliquer un style graphique facilement à mes pages HTML.

JQuery

Me permet a réalisé des requête Ajax et un des prérequis pour utiliser certaine fonctionnalité de Bootstrap.

Struts2

Me permet de réalisé le MVC ainsi que le lien et donc le lien entre le client et le serveur.

Hibernate

Me permet de lié le coté applicatif de l’application à la base de données Oracle XE en simplifiant certaine requête SQL.

JUnit

Me permet de réalisé mes test unitaire sur l’application.

GSon

Me permet de transformer un String JSon en objet Java.

(Annexe X)

## Les tests unitaires

J’ai réalisé pour le projet des tests unitaires avec JUnit que j’ai mis en place directement lors du premier codage de l’application, avant tout j’ai construit une fiche de test, regroupent ce qui devrait être testé et par la suite créer les tests.

Une fois les tests réalisés j’ai modifié le programme pour qu’ils répondent à tous les tests, j’ai aussi utilisé ces tests après chaque modification de l’application qui toucheraient les fonctionnalités tester afin d’assurer la non-régression de celle-ci, les tests unitaires on était un gros apport pour contrôler l’intégrité de mon application ainsi que la debugger.

(Annexe X)

## Conclusion

De réaliser les tests unitaires en transverse du développement ainsi que de réaliser le développement et l’analyse en petite itération a été un gros gain de temps et de performance pour réaliser ces actions, je pense donc pour mes futures projet réutilisé cette méthode afin de ne pas me faire submerger par la charge de travail qu’un projet peut avoir aussi l’architecture N-Tiers bien que plus couteuse à mettre en place en termes de temps apporte une vrai visibilité sur le projet et facilite ainsi sa maintenabilité et son évolution c’est pour cela que je pense la réutiliser aussi souvent que possible sauf pour les « petits » projets qui demandent des délais courts.

# L’apport personnel

## Retour d’expérience

J’ai eu de réels problèmes sur la gestion de la charge de travail sur ce projet ainsi que pour la planifier, je pense aussi que d’utiliser struts2 pour faire le lien entre mes pages web et l’application java, est trop lourd à mettre en place, de plus au niveau de mes requêtes Ajax j’ai utilisé des objets JSon, mais qui je trouve ne sont pas assez lisibles pour pouvoir être maintenus par la suite, aussi chaque retour d’une requête Ajax m’oblige à mettre à jour la présentation que j’ai réalisée en codant une fonction JavaScript pour chaque retour.

L’utilisation de la librairie Bootstrap fut un réel gain de temps pour parfaire le style de la couche de présentation qui pouvait me prendre beaucoup de temps quand je le faisais seulement avec du CSS je pense la réutiliser pour mes futures projets sauf s’ils demandent un style graphique plus particulier que ce proposé par Bootstrap.

## Axes d’amélioration

* Étudier Angular et le mettre en place sur une couche de présentation afin de régler les problèmes de mise à jour de la couche de présentation sur un retour Ajax.
* Étudier la possibilité de création de XML depuis un objet java, si cela n’ajoute pas de complexité et de temps au développement plutôt que d’utiliser des objets JSon afin d’améliorer leur lisibilité.
* Utiliser de plus petite itération pour la réalisation de projet afin de mieux gérer la charge de travail.
* Conserver les planifications déjà réalisées afin de pouvoir mieux calculer les temps qu’une action me demande.
* Sur les contrôleurs utilisant struts2, mettre en place seulement des requêtes Ajax avec des objets JSon ou XML, car plus simple et rapide à mettre en place.

# Les Annexes