

# Mémoire de Fin d'Etudes

Pour l'obtention du diplôme

## D'Ingénieur d'Etat

Génie Informatique  
Option : Système D'information

Promotion 2011 – 2016



Réalisation et développement d'une plateforme  
pour la gestion et le suivi des projets  
des Entreprises.



Mlle. Joairia Lafhal

Mlle. Soumaya Moussati

Soutenance le 24 Juin 2016

### Membres de jury :

M. LAZAAR Mohamed  
M. TABII Youness  
M. Al ACHHAB Mohamed  
M. MAHBOUB Oussama

Encadrant Société  
Encadrant ENSATé  
Encadrant ENSATé  
Enseignant ENSATé



# Dédicace

*À Ma très chère Maman,  
Mme. Laaziz Souad*

*Ma mère, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.*

*À Mon très cher Papa,  
M. Lafhal Abderrahim*

*Qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Dieu faire en sorte que ce travail porte son fruit ; Merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.*

*À Mon Fiancé,  
M. Bendris Abderrahim*

*Qui m'a soutenu toutes ces années, et m'a fait pousser la limite pour être une meilleure personne, je remercie Dieu chaque jour que tu as été là à mes côtés*

*À Soumaya Moussati,*

*Pour avoir été avant tout une très bonne amie et une coéquipière travailleuse, patiente et sérieuse*

*À tous mes amis et la famille,  
Merci de croire en moi*

*Joairia*

*A Mes très chers parents,*

*Je ne pourrais exprimer l'étendue de ma gratitude en ces quelques lignes. Toutefois je vous remercie pour votre soutien continu, pour vos sacrifices et pour l'excellent modèle que vous avez été pour moi le long de toutes ces années. Je vous dédie ce travail.*

*A Mes chers frères et sœurs,*

*En témoignage de mes sincères reconnaissances pour les efforts qu'ils ont consenti pour l'accomplissement de mes études. Je leur dédie ce modeste travail en témoignage de mon grand amour et ma gratitude infinie.*

*A Hicham El Mouilti,*

*Je te remercie pour ton soutien continu, pour ton foi en mes capacités et pour les encouragements sincères que tu n'as pas manqué de m'offrir tout au long de mon parcours et pendant les moments où j'en avais le plus besoin. Je te dédie ce travail.*

*A Joairia Lafhal,*

*Pour avoir été une coéquipière travailleuse et sérieuse, mais qui s'est également révélée être une grande amie.*

*Soumaya*

# Remerciements

C'est pour nous un plaisir autant qu'un devoir de remercier toutes les personnes qui ont pu contribuer de près ou de loin à l'élaboration de ce projet, qui nous ont aidé, nous ont soutenu et ont fait en sorte que ce travail ait lieu.

Nos remerciements les plus sincères vont être adressés à nos encadrants pédagogiques à l'Ecole Nationale des Sciences Appliquées de Tétouan :

**M. TABII Youness** pour ses directives, ses conseils et son soutien tout au long du stage. Nous vous remercions pour votre suivi et votre encadrement durant ce mois-ci.

**M. AL ACHHAB Mohamed** qui n'a épargné aucun moyen pour nous aider et soutenir, et aussi pour son judicieux encadrement ainsi que sa disponibilité afin de nous aider à élaborer ce travail

Ainsi, nous manifestons notre profonde gratitude à **M. MAACH Mohamed** directeur de Adréaline Holding pour nous avoir accueillis au sein de cette société. Nous remercions également notre encadrant **M. BOUZERGANE Smail** pour son soutien. Sa générosité, son encadrement, ses explications, sa disponibilité et ses conseils prodigues tout au long de période de notre stage.

Nous tenons à remercier tout le personnel de la société, pour leur soutien et pour leur générosité considérable quant à l'offre de l'information.

Nous tenons également à adresser nos plus sincères remerciements à l'ensemble du corps enseignant de l'ENSATé, pour avoir porté un vif intérêt à notre formation, et pour nous avoir accordé de plus clair de leur temps, leur attention et leur énergie et ce dans un cadre agréable de complicité et de respect.

# Résumé

Le présent rapport est le fruit d'un travail effectué au sein de la société Adrénaline Holding à Tanger, en vue de l'obtention du diplôme d'ingénieur d'état en génie informatique de l'Ecole Nationale de Sciences Appliquées de Tétouan. L'objectif de ce projet est la conception le développement et la mise en place d'une plate-forme de gestion et suivi des projets des entreprises qu'on l'a appelé PMSTalker.

Le système à réaliser comporte plusieurs fonctionnalités :

- Gestion des projets dont l'insertion des nouveaux projets leur modifications et aussi la consultation des projets sur lequel l'entreprise travaille.
- Gestion des tâches qui nous permet d'insérer les tâches de chaque projet de les consulter et modifier et aussi les affecter aux employés concernés.
- Gestion des équipes où on peut créer des équipes de travail les mis à jour et les affecter aux tâches des projets.
- Gestion des employés pour la consultation des employés de l'entreprise ajouter des nouveaux employés afin de savoir l'effectif de l'entreprise et les ressources qu'elle a lors d'un projet.
- Gestion des dépenses qui va permettre à savoir tous les charges que l'entreprise a dépensé dans ses projets
- Gestion de la Facturation pour facturer les projets terminés et les livrer au client.
- Gestion des Clients permet de gérer et savoir les clients ou les collaborateurs avec lesquels l'entreprise réalise ses projets
- Suivi temporel sous forme d'un calendrier qui permet à l'administrateur d'avoir une vision globale sur le timing des projets et des tâches.

Pour la réalisation du projet, nous avons suivi le processus de développement SCRUM et la modélisation du système était avec le langage UML (Unified Modeling language).

En ce qui concerne de développement de l'application qui respecte l'architecture JAVA EE (JAVA Entreprise Edition), nous avons utilisé les frameworks suivants :

- Spring MVC pour respecter une architecture logicielle qui permet une meilleure réutilisabilité et une grande facilité de maintenance du code.
- Spring Security pour sécuriser l'accès au système.
- Hibernate pour le mapping objet/relationnel. Par ailleurs, nous avons utilisé le SGBD (Système de Gestion de Bases de données) MYSQL.

Les tests de l'application étaient réalisés en utilisant le framework JUnit.

**Mots-clés :** PMSTalker, gestion projet, architecture logicielle

# Abstract

This report is the result of four months of work carried out within the company Adrenaline Holding - Tangier, as part of our final project in order to obtain the engineering degree in computer science.

The project was about creating and developing an application for the projects management called PMSTalker.

This application to be made must contain several features:

- Projects Management allows to insert new projects editing them and consulting the list of projects that the society works on.
- Tasks Management allows to insert, edit and delete the project's tasks and affect them to the concerned employees.
- Teams Management where we can create work teams update the old ones and affect them to the project's tasks.
- Employees Management to view and consult all the society's employees, add new ones so that the society can preview all the ressources it has for a project.
- Expenditures Management allows to know all the expenditures that the society had to pay for its projects.
- Invoicing Management to invoice the projects that have been done and deliver them to the client.
- Clients Management to manage and know all the clients with whome the society is collaborating to realize a specific project.
- Timing Management it's a calendar that allows to the administrator to have a global vision about timing of all the projects and tasks.

To achieve this project, we have followed the development process in Scrum and we modeled the system with UML (Unified Modeling Language).

Regarding the development of the application that respects JAVE EE architecture (JAVA Enterprise Edition), we have used the following frameworks: Spring MVC to respect a software architecture that allows for better reusability and eases the code maintenance. Spring Security to secure the access to the system. Hibernate framework for object/relational mapping between the system and the DBMS. In addition, we used MySQL DBMS (Database Management System).

We also used JUnit Test for unit testing and Jasper Report for extracting list and files in PDF Format among the system.

**Keywords:** PMSTalker , project Management , architecture

# تلخيص

يستعرض هذا التقرير عرضا مفصلا لمشروع التخرج الذي قمنا ب انجازه في شركة "أدريلانين هولدينغ بطنجة" بهدف حصولنا على دبلوم مهندس الدولة في المعلومات عن المدرسة الوطنية للعلوم التطبيقية بتطوان.

يتلخص الهدف من هذا المشروع في انشاء نظام معلوماتي من أجل تدبير و تتبع المشاريع التي تنجزهم الشركات و ايضا خاصية تمكن من التعمق في تحليل هذه البيانات بهدف مساعدة ادارة الشركة في اتخاذ القرارات PMSTalker .

و يتوفر هذا النظام على عدة خصائص و مميزات :

- تدبير المشاريع من خلال رؤية ، إضافة وتعديل المشاريع التي تشتغل عليهم الشركة .
- تدبير و ادارة البيانات الأساسية.
- تدبير المعطيات المتعلقة بتتبع سيرورة المشاريع.
- تدبير المهام المكونة للمشروع و تعيينها للموظف الذي سيقوم بإنجازها.
- تدبير فرق العمل التي ستشتغل على المشروع.
- تدبير الموظفين الذين يشتغلون في الشركة.
- تدبير نفقات الشركة لإنجاز المشروع.
- تدبير الفواتير
- تدبير الزبائن التي تشتغل معهم الشركة مشروعها.

لإنشاء هذا المشروع قمنا باستخدام منهجية SCRUM التي تركز على مبدأ Sprints . كما قمنا باستخدام لغة النمذجة الموحدة UML لتصميمه.

فيما يخص لغات البرمجة والتقنيات التي اعتمدنا عليها في انشاء المشروع ، فقد استخدمنا لغة الجافا مع الإطار الخاص بالشركات JAVA Enterprise Edition . هذا بالإضافة الى مجموعة من المكتبات و الإضافات الأساسية : ك Spring MVC, Spring Security, Hibernate, JasperReport.

سنقوم في هذا التقرير بعرض مفصل للمراحل التي قمنا بها لإنشاء هذا المشروع بدءا من الدراسة ومرورا بالتصميم وختاما بالبناء والتفعيل.

كلمات مفاتيح: PMSTalker ، تدبير المشاريع .



# Liste des Abréviations

Abréviation	Désignation
<b>PMSTalker</b>	Project Management Stalker
<b>UML</b>	Unified Modeling language
<b>Java EE</b>	Java Entreprise Edition
<b>DAO</b>	Data Access Object
<b>SGBD</b>	Système de gestion de base de données
<b>MVC</b>	Model View Controller
<b>IHM</b>	Interface Homme Machine
<b>JSP</b>	Java Server Pages
<b>SQL</b>	Structured Query Language
<b>AJAX</b>	Asynchronous JavaScript and XML
<b>MVC</b>	Model View Controller
<b>ORM</b>	Object Relational Mapping
<b>IDE</b>	Integrated Development Environment
<b>XML</b>	eXtensible Markup Language
<b>HTML</b>	Hypertext Markup Language
<b>STS</b>	Spring Tool Suite

*Tableau 1: liste des abréviations*

# Liste des Figures

Figure 1: logo Adrénaline .....	19
Figure 2: L'organigramme de la société .....	21
Figure 3: Planification d'un projet en Excel .....	24
Figure 4: Diagramme de Gantt d'un projet en utilisant un logiciel.....	24
Figure 5: L'interface web d'ASANA.....	25
Figure 6: L'interface web d'Aceproject .....	26
Figure 7: L'interface graphique de GanttProject.....	27
Figure 8: logo du système PMStalker .....	28
Figure 9: la méthode Scrum .....	37
Figure 10: Diagramme de GANTT de notre projet.....	41
Figure 11: Architecture 3tiers .....	45
Figure 12 : Comparaison entre java et .net.....	47
Figure 13: Comparaison entre java et LARAVEL.....	47
Figure 14: logo de Spring .....	48
Figure 15: logo de hibernate.....	49
Figure 16: logo de JSP .....	50
Figure 17: logo de spring security.....	50
Figure 18: Interface de STS .....	51
Figure 19: Architecture vFabric tc Server.....	52
Figure 20: logo MySQL .....	53
Figure 21: logo d'Ajax.....	54
Figure 22: logo de Visio.....	54
Figure 23: Diagramme cas d'utilisation pour la gestion des projets, taches et utilisateur .....	63
Figure 24: Diagramme cas d'utilisation pour le directeur.....	64
Figure 25: Diagramme de cas d'utilisation pour l'employé.....	64
Figure 26: Diagramme d'activité .....	66
Figure 27: Diagramme de séquence du scénario : « Authentification » .....	67
Figure 28: Diagramme de séquence : « Consulter diagramme de Gantt ».....	67
Figure 29: Diagramme de séquence pour la gestion des utilisateurs .....	68
Figure 30: Diagramme de séquence pour la gestion des projets .....	69
Figure 31: Diagramme de classe de notre système .....	72
Figure 32: Diagramme de déploiement.....	73
Figure 33: Processus des Test unitaires.....	76
Figure 34: Création d'un test unitaire .....	76
Figure 35: Test MetierClient .....	77
Figure 36: Test MetierDepense .....	77
Figure 37: Test MetierEmploye .....	78
Figure 38: Page d'authentification .....	79
Figure 39: Page d'accueil.....	80
Figure 40: Interface gestion des clients .....	81
Figure 41: Ajouter client .....	81
Figure 42: Interface gestion des employés .....	82
Figure 43: Ajouter employé .....	82
Figure 44: Rapport des employés.....	83
Figure 45: Interface gestion des utilisateurs.....	84
Figure 46: Ajouter utilisateur .....	84

Figure 47: Interface gestion des équipes .....	85
Figure 48: Ajouter équipe .....	85
Figure 49: Affectation des membres aux équipes .....	86
Figure 50: Interface gestion des projets .....	86
Figure 51: Ajouter un projet.....	87
Figure 52: Liste des documents de projets .....	87
Figure 53: Ajouter document d'un projet.....	88
Figure 54: Diagramme de Gantt d'un projet .....	88
Figure 55: Détail de projet-informations générales- .....	89
Figure 56: Détail de projet-taches de projet-.....	89
Figure 57: Détail de projet-Gantt de projet-.....	90
Figure 58: Liste des tâches .....	90
Figure 59: Ajouter une tâche .....	91
Figure 60: Gestion des catégories des dépenses.....	91
Figure 61: Ajouter une catégorie des dépenses .....	92
Figure 62: liste des dépenses .....	92
Figure 63: Ajouter dépense .....	93
Figure 64: Rapport des dépenses.....	93
Figure 65:Liste des factures .....	94
Figure 66:Details d'une facture.....	94
Figure 67: Planning des projets.....	95
Figure 68:Planning des taches.....	96
Figure 69: les projets de l'employé.....	96
Figure 70: liste des tâches de l'employé .....	97

# Liste des Tableaux

Tableau 1: liste des abréviations .....	9
Tableau 2 : Fiche technique d'Adrénaline .....	20
Tableau 3: Tableau comparative des différents cycles de vie .....	35
Tableau 4: Planification du projet .....	39
Tableau 5: Fiche technique de Spring .....	48
Tableau 6: Fiche Technique de Hibernate.....	49
Tableau 7: Identification des acteurs.....	57
Tableau 8 : Cas d'utilisation « Authentification » .....	58
Tableau 9: Cas d'utilisation « Créer projet » .....	58
Tableau 10: Cas d'utilisation « Supprimer projet » .....	59
Tableau 11: Cas d'utilisation « Consulter de diagramme de Gantt ».....	59
Tableau 12: Cas d'utilisation « Consulter projet ».....	59
Tableau 13: Cas d'utilisation « Créer une équipe » .....	60
Tableau 14: Cas d'utilisation « Créer employé » .....	60
Tableau 15: Cas d'utilisation « Modifier employé ».....	60
Tableau 16: Cas d'utilisation « Supprimer employé ».....	61
Tableau 17: Cas d'utilisation « Créer Tâche » .....	61
Tableau 18: Cas d'utilisation « Ajouter Client ».....	62
Tableau 19: Cas d'utilisation « Modifier client » .....	62
Tableau 20 : Dictionnaire de données .....	70

# Table des Matières

Dédicace .....	3
Remerciements .....	5
Résumé .....	6
Abstract .....	7
تلخيص.....	8
Liste des Abréviations .....	9
Liste des Figures.....	10
Liste des Tableaux.....	12
Table des Matières .....	13
Introduction générale.....	16
Chapitre 1 .....	18
Contexte général du projet .....	18
Introduction .....	19
1- Présentation de l'organisme d'accueil : Adrénaline .....	19
1-1 Fiche Technique de la société .....	20
1-2 L'organigramme de la société.....	20
1-3 Prestations et Services.....	21
2- Présentation générale du projet.....	22
2-1 L'étude de l'existant .....	22
2-2 La problématique .....	28
2-3 Présentation du projet PMSTalker .....	28
Conclusion.....	29
Chapitre 2 .....	30
Etude Fonctionnelle du projet .....	30
Introduction .....	31
1- La finalité du projet.....	31
2- Objectifs du projet.....	31
3- Les fonctionnalités de PMSTalker .....	31
4- Démarche de développement.....	33
5- Planification du projet.....	37
5-1 Planification .....	38
5-2 Diagramme de Gantt .....	40

Conclusion.....	42
Chapitre 3 .....	43
Etude Technique du projet .....	43
Introduction .....	44
1- Architecture logicielle.....	44
2- Méthodologie de développement : Démarche MVC .....	45
3- La plateforme Java EE.....	46
4- Frameworks.....	47
4-1 Spring Framework .....	48
4-2 Hibernate Framework .....	49
4-3 JSP.....	50
4-4 Spring Security.....	50
4-5 JUnit.....	51
5- Environnement de travail.....	51
5-1 Spring Source Tool Suite (STS) .....	51
5-2 vFabric tc Server .....	52
6- Système de Gestion de Base de Données .....	53
7- Les Outils Utilisés.....	53
7-1 XML.....	53
7-2 AJAX .....	54
7-3 La bibliothèque JQuery .....	54
8- Outils de dessin .....	54
Conclusion.....	55
Chapitre 4 .....	56
Conception détaillée du projet.....	56
Introduction .....	57
1-Diagramme de cas d'utilisation .....	57
1-1 Identification des acteurs .....	57
1-2 Identification des cas d'utilisation .....	58
1-3 Diagramme de cas d'utilisation :Gestion des projets , taches et utilisateurs .....	62
1-4 Diagramme de cas d'utilisation : Pour le directeur.....	63
1-5 Diagramme de cas d'utilisation : pour un employé .....	64
2- Diagramme d'Activité .....	65
3- Diagramme de séquence .....	66
4- Diagramme de classe .....	70
5- Diagramme de déploiement .....	73

Conclusion.....	73
Chapitre 5 .....	74
Mise en œuvre du projet.....	74
Introduction .....	75
1- Déploiement de PMSTalker.....	75
2- Les Tests d'intégration.....	75
2-1 Tests unitaires .....	75
2-2 Architecture du système.....	78
3- Les interfaces graphiques.....	78
3-1 Partie administrateur .....	79
3-2 Partie Employé.....	96
Conclusion.....	97
Conclusion Général et Perspectives .....	98
Référence.....	99
Bibliographie .....	100
Webographie .....	100

# Introduction générale

Aujourd'hui le domaine informatique est devenu de plus en plus vaste et diversifié, et les systèmes d'informations ont répondu à un besoin vif pour n'importe quel type d'entreprise, c'est la gestion des informations qui est parmi les enjeux les plus primordiaux pour les entreprises et touche pratiquement toutes les activités.

Par ailleurs, l'informatique doit être aperçue comme un outil de gestion, de communication, d'aide à la prise des décisions et d'unification des systèmes d'information, permettant à l'utilisateur de fournir des informations fiables, au bon moment et dans un espace donné. Ainsi, tout organisme souhaitant réussir, doit aujourd'hui s'équiper des instruments les plus modernes répondant de manière optimale à ses exigences et s'adaptant totalement à sa structure.

Ce projet s'intègre dans le cadre du stage de fin d'études pour l'obtention du diplôme d'ingénieur d'état en informatique de l'Ecole Nationale des Sciences appliquées de Tétouan, visant à aboutir une meilleure adéquation entre la formation aussi bien théorique que pratique. Ceci a pour objectif principal l'intégration des futurs ingénieurs d'état dans le monde professionnel, ainsi que la pratique de toutes les connaissances théoriques cumulées pendant le parcours académique.

L'entreprise **Adrénaline Holding** afin d'élargir ses services a décidé de faire sortir au marché un nouveau produit qui va viser les entreprises et les sociétés de n'importe quel domaine et qui ont des problèmes et des difficultés pour gérer les projets sur lesquels ils travaillent. Dans cette optique, ce projet de fin d'études, vient répondre à ces besoins en réalisant une solution (PMSTalker) d'automatisation du processus de la gestion et le suivi de l'avancement des projets en temps réel, tout en assurant une bonne communication et une bonne diffusion des informations entre les différents acteurs du futur système.

Notre mémoire de fin d'étude est structurée comme suit :

- Le premier chapitre définit le contexte général du projet et se compose de deux parties. Une partie donne une présentation de la société au sein de laquelle nous avons effectué notre stage. Quant à la deuxième partie, elle est consacrée à faire une étude préalable qui consistait à étudier l'existant et les applications de gestion des projets qui existent déjà dans le marché
- Le deuxième chapitre contiendra l'étude fonctionnelle du projet où on va définir la finalité, les objectifs et les spécifications générales du projet qui contiennent le cahier de charge, les spécifications des besoins, les scénarios proposés.
- Le troisième chapitre est dédié à l'étude technique du projet qui va contenir des présentations des plateformes et technologies utilisées ainsi que les ressources dont le projet a besoin.
- Le quatrième chapitre est entièrement consacré à la conception détaillée du projet. Il contient l'ensemble des diagrammes nécessaires à la bonne compréhension et cohésion du système, du diagramme de cas d'utilisation au diagramme de classe. Le tout est accompagné des scénarios et descriptions nécessaires.



- Le dernier chapitre donne une vue globale de la solution ainsi que les tests d'intégration et détaille la réalisation des applications tout en proposant des aperçus de celles-ci.
- Pour terminer, une conclusion permettant de faire une synthèse du travail réalisé et de donner les principales perspectives.

# Chapitre 1

---

## Contexte général du projet

Dans ce chapitre, nous présenterons le contexte général du projet. Nous allons commencer par présenter l'organisme d'accueil et ses prestations, pour passer ensuite à une présentation générale du projet contenant une étude de l'existant, la problématique et la présentation du projet, pour détailler la planification de celui-ci et sa méthode de développement.

## Introduction

Dans ce présent chapitre, nous allons présenter la société d'accueil, Adréaline Holding, sa création, ses prestations et services, sa fiche technique et les différentes activités qu'elle présente. Comme nous allons le contexte de notre projet ainsi qu'une étude de l'existant.

### 1- Présentation de l'organisme d'accueil : Adréaline



*Figure 1: logo Adréaline*

**ADRENALINE** est une structure de communication globale développé qui accompagne les stratégies de communication institutionnelle, de communication opérationnelle et produit & de marketing direct. L'agence est également soutenue par un réseau de partenaires experts, constitué sur la base de collaborations déjà menées avec succès.

Les services de la société Adréaline sont destinés aux sociétés souhaitant optimiser leurs performances.

Pour communiquer, création et design graphique sont essentiels. Les notions d'écoute et de confiance sont primordiales. Son conseil est enrichi par une solide expérience et des expertises ciblées permettant la réalisation de tous supports de communication.

Créé en août 2015 par le gérant associé Mohammed MAACH sous la forme juridique S.A.R.L, ADRENALINE est une agence de communication et d'événementiel qui compte deux autres associés assurant à leur tour, chacun, une telle fonction au sein de l'agence.

La société est créée d'un capital de 100.000 Dh, dont 30% sous forme de monnaie et 70% sous forme de matériaux (bureaux, fournitures et MIFO, matériel informatique et outillage).

La participation au capital est de 60% par le gérant associé Mr Mohammed MAACH, 30% par l'associé Souhir MAACH et 10% par l'associé BOUZERGANE Ismail.

ADRENALINE propose une multitude de services indispensables au développement de la société moderne, des services associés dédiés d'une part aux entreprises qui tendent à sous-traiter une partie de leurs tâches, et d'autre part aux particuliers.

Attaquer une cible diversifiée est la politique adoptée par ADRENALINE et ce afin d'éviter les risques d'échec éventuels, une telle logique semble nécessaire pour une jeune entreprise, TPE ou ce qu'on appelle aujourd'hui « START-UP », il s'agit de très petites entreprises dont le personnel ne dépasse généralement pas 10 personnes et qui viennent de se lancer dans un contexte d'affaires compliqué, le risque d'échec pour ce type d'entreprises est jugé très élevé face à un marché en permanente mutation, d'une demande exigeante et d'une concurrence acharnée.

Pour concrétiser cette politique, ADRENALINE prépare actuellement le projet ADRENALINE Holding, la création de plusieurs entités de services opérant sous et pour le compte de l'agence permettra de minimiser les risques et d'absorber les faillites éventuelles de l'une de ces entités.

## 1-1 Fiche Technique de la société

### Fiche Technique d'Adrénaline

Domination Social	<b>Adrénaline Holding</b>
Date d'implantation	<b>Aout 2015</b>
Siège Social	<b>Technopark Tanger</b>
Forme Juridique	<b>Société à Responsabilité limité (SARL)</b>
Secteur D'activité	<b>Communication global et évènementiel</b>
Directeur Général	<b>M.MAACH Mohamed</b>
Effectif Total	<b>5 personnes</b>
Téléphone	<b>+212539324668 +212660151107</b>

Tableau 2 : Fiche technique d'Adrénaline

## 1-2 L'organigramme de la société

L'équipe ADRENALINE se compose de 3 associés dont 2 assurent, chacun, une fonction au sein de l'agence.

**Mohammed MAACH** : ex-fondateur d'une agence web-médiatique, Mr MAACH est le gérant de la société ADRENALINE, il profite de son expérience dans le domaine et du réseau qu'il a pu établir pour mettre l'agence sur la bonne voie, guider l'équipe au bon chemin et assurer la survie.

**Smail BOUZERGUANE** : détenant d'un diplôme d'ingénieur d'état en Génie Informatique, Mr BOUZERGUANE est le responsable sur le volet informatique et développement web de la société, il s'occupe de la création et du développement des applications web et/ou mobiles pour le compte de l'agence même et de ses clients, veille au bon fonctionnement de ses applications et assure leur maintenance (modification et réparation) et veille à ce que les procédés de l'agence soient conformes aux évolutions technologiques .

Ainsi l'organigramme de l'agence ADRENALINE se présente comme suit :

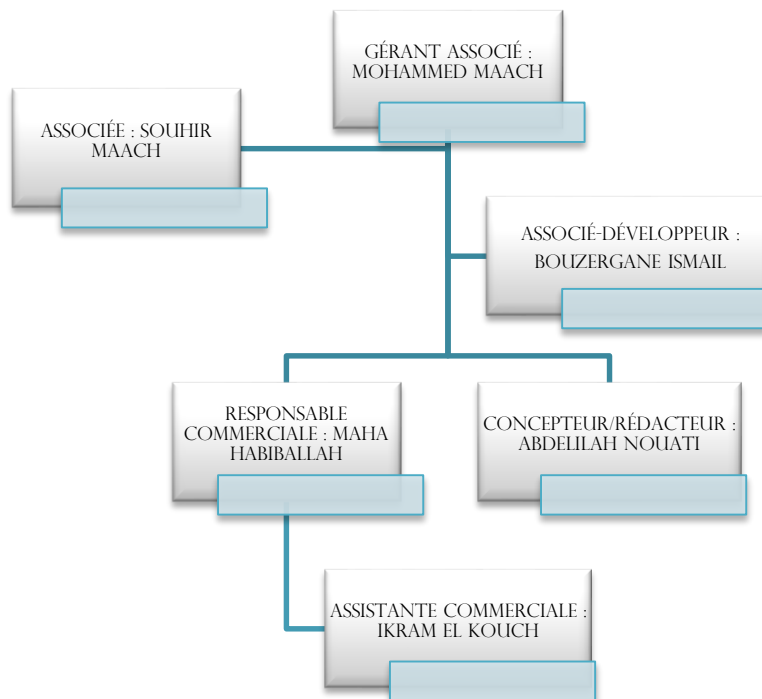


Figure 2: L'organigramme de la société

### 1-3 Prestations et Services

Divisé en 3 catégories, le portefeuille services ADRENALINE se présente comme suit :

#### i. Communication

La communication est un domaine pluridisciplinaire dont l'objectif essentiel est de révéler avec pertinence la personnalité, les services et les produits, il s'agit de communiquer une image de marque aussi parfaite qu'elle pourrait être. ADRENALINE emploie pour le compte de ses clients l'essentiel de ses compétences et veille à assurer avec pertinence :

- La conception
- L'analyse
- La stratégie de marque
- Le positionnement Marketing
- La communication globale et opérationnelle
- Le design graphique
- L'identité visuelle
- Le webdesign
- L'édition-mise en page
- Le packaging
- Le design environnemental

- Le design sonore

### ii. Web et Multimédia

ADRENALINE propose des solutions avancées afin de garder sa clientèle à la pointe de technologie. Il s'agit de développer, pour le compte des clients, des :

- Sites web.
- Sites web institutionnels, boutiques en ligne, sites événementiels, sites mobiles.
- Sites web WordPress.
- Sites web e-commerce.
- Stratégie digitale.
- Référencement.
- Applications web.
- Applications mobiles pour (Android, IOS, Windows Phone et BlackBerry).

### iii. Événementiel

ADRENALINE s'attache à mettre en scène des événements uniques afin de marquer la communication et de donner une meilleure image de marque. Que ce soit un salon, forum, conférence, exposition, séminaire ou convention, l'équipe ADRENALINE se penche, pour la réalisation d'un événement marquant, sur la création, la qualité et la cohérence des contenus de la communication événementielle. Il s'agit de mettre en œuvre tout moyen et action liée au projet événementiel :

- Team building.
- Voyages sur mesure.
- Animation (animation cocktail, animation de convention et de stand).
- Cadeau d'affaires.
- Photo shooting (professionnel et personnel).

## 2- Présentation générale du projet

Nous procédons maintenant à une présentation générale du projet : elle contiendra la problématique que notre projet va essayer de résoudre ainsi qu'une étude de l'existant et les applications de gestion de projet qui existent déjà dans le marché ce qui nous permettra de présenter la bonne solution.

### 2-1 L'étude de l'existant

Avant d'entamer la réalisation du projet qui consiste à réaliser une plate-forme web pour la gestion et le suivi en temps réel des projets des entreprises, il faut d'abord savoir qu'est-ce qu'un projet de société comment les entreprises gèrent leurs projets et ensuite savoir quelles sont les difficultés et les problématiques que ces entreprises rencontrent lors de la

gestion d'un projet pour qu'on puisse donner une solution valable à ces problèmes. Pour cela nous devons nécessairement savoir en premier lieu qu'est qu'un projet de société ?

Un projet représente l'ensemble des actions à entreprendre afin de répondre à un besoin défini dans des délais fixés. Ainsi un projet étant une action temporaire avec un début et une fin, mobilisant des ressources identifiées (humaines et matérielles) durant sa réalisation, celui-ci possède également un coût et fait donc l'objet d'une budgétisation de moyens et d'un bilan indépendant de celui de l'entreprise.

La difficulté dans la conduite du projet réside en grande partie dans la multiplicité des acteurs qu'il mobilise. En effet, contrairement aux projets personnels ou aux projets internes à faible envergure pour lesquels le besoin et la réponse à ce besoin peuvent être réalisés par la même personne ou par un nombre limité d'intervenants, dans un projet au sens professionnel du terme, l'expression du besoin et la satisfaction de ce besoin sont portés par des acteurs généralement distincts.

De cette manière, il est nécessaire de s'assurer tout au long du projet, que le produit en cours de réalisation correspond clairement aux attentes du « client ».

Pour cela chaque entreprise suit ces trois étapes en conduisant ses projets :

**La phase de définition ou de cadrage :** il s'agit de toute la partie d'analyse, de formalisation et de planification amont. C'est le tout début du processus avec la prise en compte de la demande et la préparation des travaux à venir (constitution de l'équipe, mise au point du planning, définition des ressources...).

**L'exécution :** la réalisation de ce qui a été planifié précédemment avec les séances de travail, les réunions avec les parties prenantes, les validations intermédiaires, la communication, la (très importante) conduite du changement...

**La conclusion :** l'heure du bilan et l'organisation de la fin des travaux. Avec un l'objectif de capitaliser sur l'expérience récemment acquise.

Mais la question qui se pose est comment ces entreprises peuvent planifier leurs projets et les tâches de ces projets tout en assurant la bonne communication entre les équipes de travail et les parties prenantes de l'entreprise, comment le directeur peut savoir l'avancement de son projet sans qu'il ait recourt à faire des réunions à chaque fois, ici nous devons savoir quels sont les outils que ses entreprises utilisent pour une bonne gestion des projets et une bonne communication entre les acteurs du projet.

La majorité des entreprises au Maroc et même ailleurs utilisent toujours les papiers et les tableaux pour planifier leurs projets et affecter les tâches du projet aux équipes de travail et parfois ils ont recours à des logiciels comme GANTT Project le MS Project ou tout simplement à l'Excel pour la planification de leur projet et la gestion des ressources, comme on peut remarquer dans les figures suivantes (figure 3 et 4):

Projet Business Intelligence : Planning														
Tâches	Acteur	Date Début prévue	Ressource prévue	Durée prévue	Date Fin prévue	Date Début estim./réel	Ressource affectée	Durée estimée	Jours écoulés	Jours à faire	Jours indispo.	Dépass. ement	Récupé ration	Date Fin réelle
7. Archivage des données N-2 et N-3	SBO	26/03/2013	1	1	27/03/2013	26/03/2013	1	1					1	26/03/2013
8. Validation des VE	PBA	08/04/2013	1	1	09/04/2013	10/04/2013	1	1	12		4	11		28/04/2013
9. Modifications analytiques	PBA	09/04/2013	1	5	16/04/2013	26/04/2013	1	5	2		2		3	30/04/2013
10. Intégration des données N-1	PBA	16/04/2013	1	5	23/04/2013	30/04/2013	1	5	1				4	01/05/2013
11. Intégration des données N	PBA	23/04/2013	2	4	29/04/2013	01/05/2013	2	4	12		4	8		17/05/2013
12. Calcul et intégrations des intra-groupes	PBA	29/04/2013	1	1	30/04/2013	17/05/2013	1	1		1	2			
13. Synthèse	PBA	30/04/2013	1	1	01/05/2013	20/05/2013	1	1		1				
14. Soldes intermédiaires de gestion	SBO	01/05/2013	1	1	02/05/2013	21/05/2013	1	1		1				
15. Réalisation du tableau de bord	SBO	02/05/2013	1	3	07/05/2013	22/05/2013	1	3		3	2			
16. Création des tables import/consultation	SBO	07/05/2013	1	3	10/05/2013	27/05/2013	1	3		3				
17. Procédure stockée d'import/cube	SBO	10/05/2013	1	2	14/05/2013	30/05/2013	1	2		2	2			
18. Optimisation programme répartition	SBO	14/05/2013	1	5	21/05/2013	03/06/2013	1	5		5	2			
19. Rédaction procédures d'administration	PBA	21/05/2013	1	2	23/05/2013	10/06/2013	1	2		2				
20. Rapport par Marché et par Produit	PBA	23/05/2013	1	4	29/05/2013	12/06/2013	1	4		4	2			
Bilan Projet :										Jours à faire : 22 jours	Dépassement : 11 jours			

Figure 3: Planification d'un projet en Excel

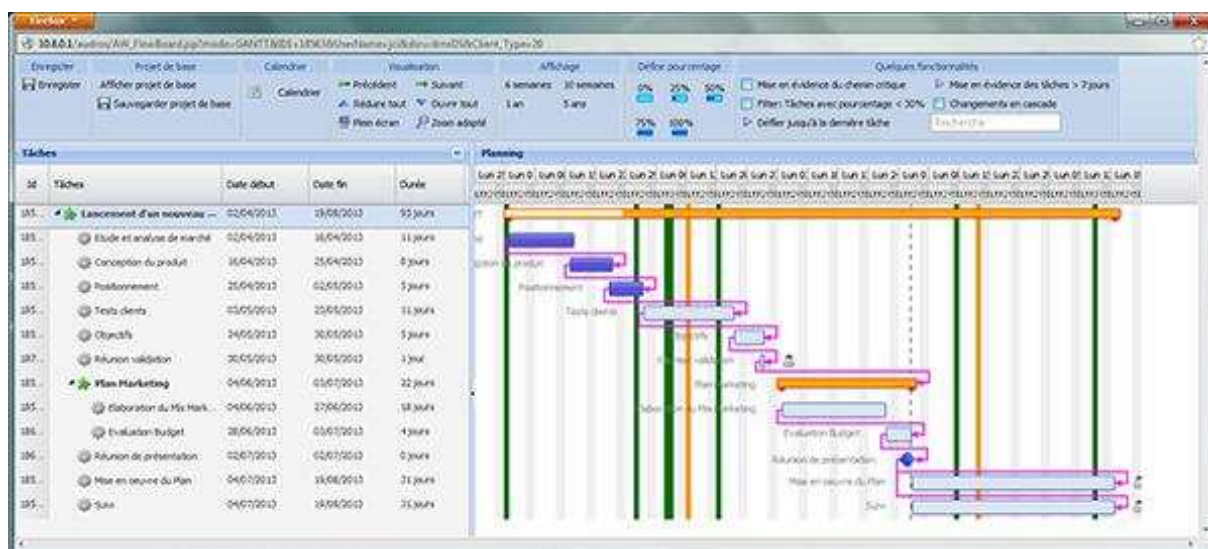


Figure 4: Diagramme de Gantt d'un projet en utilisant un logiciel

Sous ses conditions, et pour faciliter la gestion des projets au entreprises ; plusieurs développeurs web ont développé des applications web et d'autre bureau dont la majorité sont open source pour les entreprises dont ces derniers peuvent s'y abonner mensuellement ou annuellement et les utiliser pour une meilleure gestion des projets.

Et parmi ces applications nous avons trouvé les trois les plus répandu et les plus utilisé au monde :



## ■ ASANA :

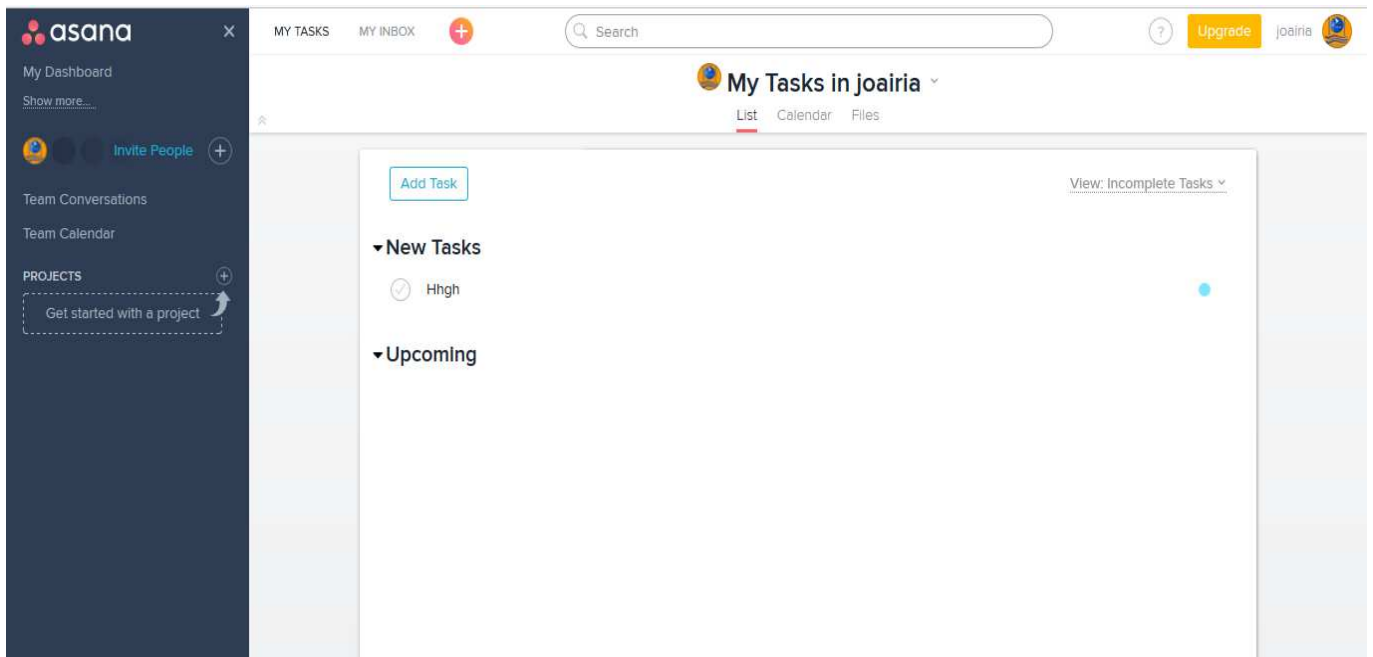


Figure 5: L'interface web d'ASANA

Asana est une application dédiée à gérer les projets où on peut insérer un nouveau projet et ajouter les tâches de ce projet comme on peut visualiser le timing du projet dès un calendrier, aussi asana a intégré la discussion instantanée où les employés du même projet peuvent se communiquer. Et comme toutes les applications Asana a des avantages et des inconvénients :

### Les avantages :

- Belle interface qui est intuitive pour naviguer.
- La hiérarchie est bien organisée, ce qui vous permet de classer les choses dans des espaces de travail, des projets, des tâches, des sous-tâches.
- Facile à ajouter des membres de l'équipe et a de grandes fonctions pour la collaboration, y compris la possibilité d'assigner des tâches, et attribuer des dates d'échéance.

### Les inconvénients :

- Absence de l'aspect des notifications, on ne peut pas recevoir des e-mails de notification sur les tâches qui vous sont assignées.
- Absence de la gestion des équipes.
- Absence de tableau de bord.
- On ne peut pas savoir l'état d'avancement d'un projet ou d'une tâche.

### ▪ AceProject :

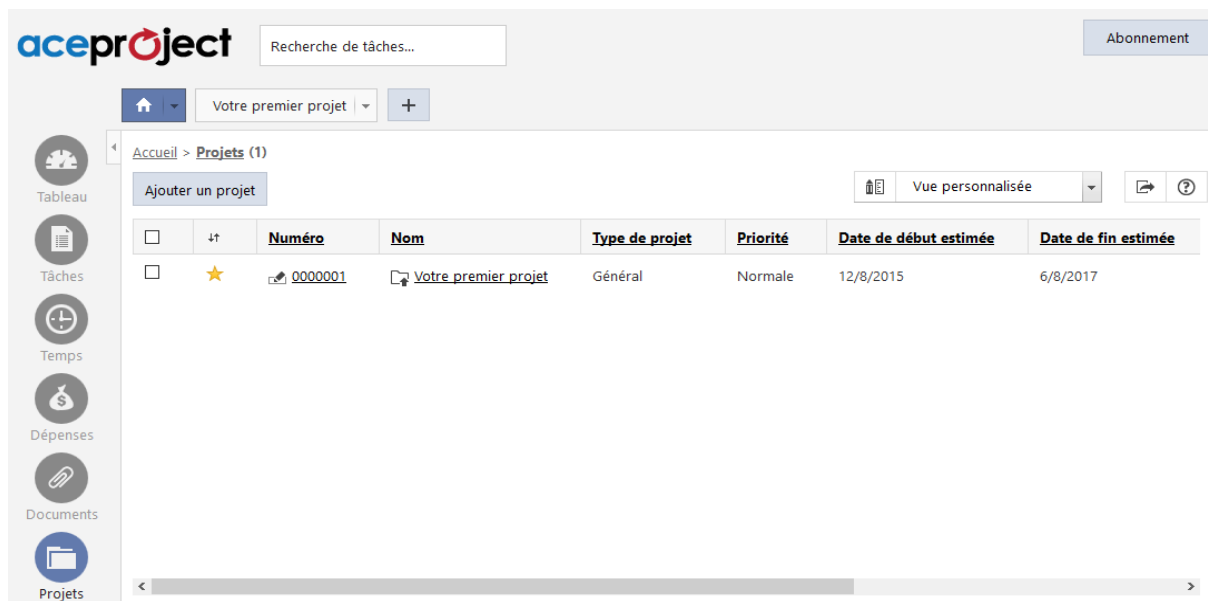


Figure 6: L'interface web d'Aceproject

AceProject a été lancé en 2001 en tant Free Task Manager, qui était un outil de gestion de tâches simple et gratuite, mais plus tard on a ajouté le suivi et la gestion des projets avec la possibilité de gérer le temps, et on l'a rebaptisé AceProject le 1er Mars 2003.

#### Les avantages :

- AceProject est beaucoup plus rapide que certains des autres logiciels Web 2.0 de gestion de projet sur le marché.
- Le logiciel fait un bon travail en abordant la phase de planification et fournit de solides fonctionnalités pour le suivi des projets et des rapports.
- Il contient la possibilité de réaliser les diagrammes de Gantt pour les projets.
- Il propose la possibilité de suivi des dépenses des projets.
- Toutes les communications sont centralisées en un seul endroit.

#### Les inconvénients :

- Le tableau de bord est propre et simple, mais d'un point de vue esthétique, la police de caractères est très faible qui ne sont pas visuellement attrayant.
- Les icônes et les textes sont plus lisibles et beaucoup moins écrasante.
- L'interface devrait être plus simpliste. En période de stress élevé, que plusieurs projets peuvent provoquer, les membres de l'équipe ne seront pas gravités vers un logiciel qui est source de confusion ou écrasante.
- Absence de gestion des clients
- Absence de la facturation.

### ■ GANTT Project :

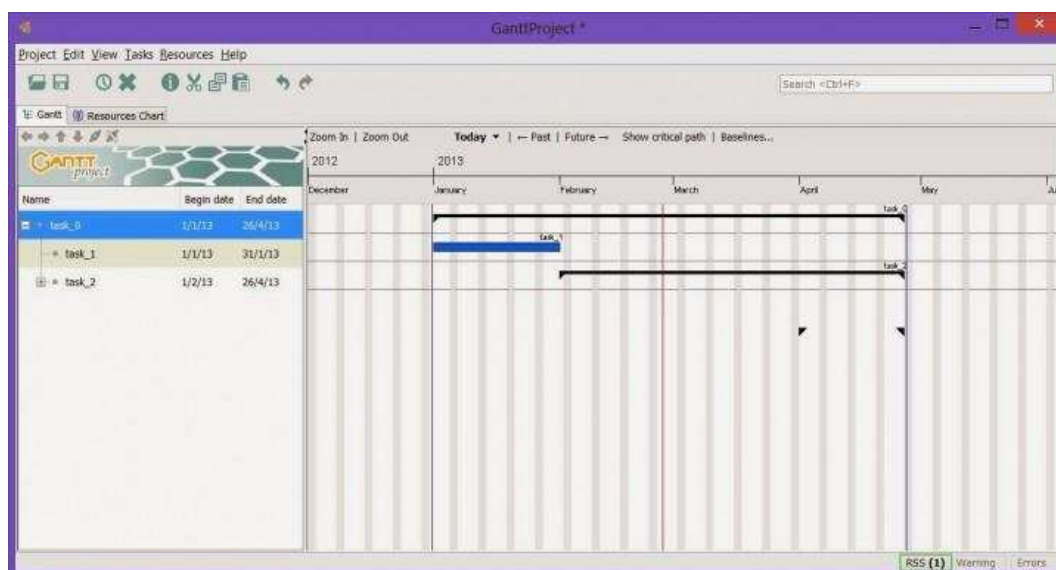


Figure 7: L'interface graphique de GanttProject

GanttProject permet la planification d'un projet à travers la réalisation d'un diagramme de Gantt. L'outil permet de créer des diagrammes de Gantt, des diagrammes de ressources et des réseaux PERT. Il convient de noter qu'en l'état actuel le logiciel ne permet pas de concevoir des tâches durant moins d'une journée.

Voici la liste des fonctionnalités principales du GanttProject:

- Description de la structure de découpage du projet (en anglais "Work Breakdown Structure"). C'est une décomposition hiérarchique, axée sur les tâches, du travail que l'équipe de projet doit exécuter pour atteindre les objectifs du projet et produire les livrables voulus.
- Liens multiples de précédences entre les tâches (prédécesseurs/successeurs).
- Visualisation du chemin critique.
- Sauvegarde des données au format XML .
- Gestion des jours fériés et des vacances pour les ressources.

#### Les avantages :

- Il permet de visualiser la complexité des projets.
- Il aide à définir un planning réaliste.
- C'est un bon outil de communication sur l'avancement du projet.

#### Les inconvénients :

A part le fait que Gantt Project permet de planifier les projets à travers le diagramme de Gantt qu'il propose et la gestion des ressources, Gantt Project ne propose aucune solution pour le suivi en temps réel de l'avancement des projets ni l'avancement des tâches.

## 2-2 La problématique

Comme nous l'avons spécifié précédemment, les entreprises ont plus recours à des applications web qui leur permettent de bien gérer leurs projets et suivre leur avancement. Ce qui nous amène à la problématique suivante :

Les grandes entreprises souhaitent suivre l'avancement de chaque projet sur lesquels ils travaillent et aussi visualiser à chaque étape du projet l'évolution des indicateurs de performance de l'entreprise ce qui va permettre aux parties prenantes de l'entreprise à prendre les bonnes décisions et qui auront un effet positive sur le gain de la société.

Cependant, les applications et les logiciels de gestion des projets ne sont pas propices à le suivi et la gestion des projets qui va permettre l'étude des indicateurs de performance de l'entreprise, car celles-ci présentent également quelques lacunes au niveau des interfaces graphiques , ils ne sont pas simple à utiliser aussi on remarque qu'il y a une absence de gestion des employés, des équipes , des clients et de la facturation, De ce fait, un client n'a pas une possibilité pour consulter et voir l'avancement de ses projets , aussi l'employé n'est pas notifié lorsqu'on lui affecte une tâche, ce qui peut tarder le démarrage de la tâche.

Pour finir, ces applications ne présentent pas un tableau de bord qui peut contenir des graphes qui calculent l'évolution des indicateurs de performance de l'entreprise lors la réalisation d'un projet.

Ainsi, il s'avère important de trouver une solution pour la problématique se présentant.

## 2-3 Présentation du projet PMStalker



Figure 8: logo du système PMStalker

Ce projet a pour but est de concevoir une solution informatique permettant la gestion et le suivi des projets d'une entreprise, afin d'aider à la décision a l'aide d'un tableau de bord dynamique tout en assurant en vue globale de l'état d'avancement de chaque projet selon les taches planifiées.

**PMStalker** doit assurer la fiabilité des informations fournies par les utilisateurs à l'aide d'un système de pré-validation. La solution doit assurer une bonne communication entre les différents acteurs. En plus ce système doit avoir une meilleure sécurité en termes de gestion des droits d'accès et aussi une interface ergonomique permettant aux utilisateurs de se guider sans besoin d'aide.

### Conclusion

Dans ce chapitre, nous avons présenté le contexte général du projet, en introduisant l'organisme d'accueil : Adrenaline Holding. Nous avons ensuite effectué une étude de l'existant, grâce à laquelle nous avons pu cerner les problématiques se présentant.

# Chapitre 2

---

## Etude Fonctionnelle du projet

Dans ce chapitre, nous présenterons le projet en détail, sa finalité, ses objectifs, ainsi qu'une étude fonctionnelle de ce projet qui contiendra la solution que cette application viendra avec et les différentes caractéristiques et fonctionnalités de ce système, pour détailler enfin la planification de celui-ci et sa méthode de développement.

### Introduction

Nous procédons dans ce chapitre à la présentation de l'application PMStalker, sa finalité, ses objectifs, ensuite nous allons détailler les spécifications de ce projet son cahier de charge et ses fonctionnalités, ainsi qu'une présentation de la planification suivra.

### 1- La finalité du projet

La mission ultime de notre système est d'aider les parties prenantes (les directeurs, les managers et les chefs de projets) des entreprises de prendre les bonnes décisions qui vont influencer positivement l'évolution de l'entreprise et augmenter ses gains, tout en suivant l'avancement de ses projets et l'évolution de ses indicateurs de performance.

### 2- Objectifs du projet

L'objectif de ce projet est la création et la mise en place d'une solution de gestion et suivi en temps réel des projets de l'entreprise afin d'aider à la décision.

La solution doit assurer en premier lieu l'ensemble des fonctionnalités suivantes :

- La gestion des données de base.
- Gestion et suivi en temps réel l'ensemble des projets selon les domaines d'activités.
- Gestion et suivi de l'avancement et la réalisation des tâches de chaque projet.
- Réalisation d'un système permettant à chaque employé de consulter, modifier, et valider ses tâches et ses projets.
- Un tableau de bord dynamique contenant les indicateurs de performance des projets, des employés et des dépenses.
- Un module de gestion des employés et l'affectation des tâches à chaque employé selon ses compétences.
- Un module de gestion des clients, pour qu'on puisse savoir et gérer les clients avec lesquels l'entreprise collabore en travaillant sur ce projet.

### 3- Les fonctionnalités de PMStalker

Dans cette partie nous allons présenter les différents modules qui vont constituer le système

#### ▪ Module de gestion des projets :

Notre système a comme premier but permet la gestion des projets sur lesquels travaille les entreprises. Cette composante a pour objectif d'offrir aux utilisateurs (le directeur, le chef de projet) des moyens pour :

- ✓ Ajouter, modifier ou supprimer les projets.
- ✓ Suivre l'avancement de chaque projet.
- ✓ Consulter le rapport des projets.
- ✓ Visualiser le diagramme Gantt de chaque projet.

Les autres acteurs qui sont les employés peuvent aussi effectuer les différentes opérations qui se suivent :

- ✓ Consulter la liste des projets de l'entreprise.
- ✓ Consulter leurs projets.
- ✓ Suivre l'avancement des projets.

### ▪ **Module de gestion des tâches :**

Ce module est le complément du précédent car chaque projet est divisé en plusieurs tâches, alors on ne peut pas créer un projet sans tâche. La gestion des tâches a pour objectif d'aider l'administrateur du système qui peut être le chef du projet de :

- ✓ Ajouter, modifier ou supprimer les tâches d'un projet.
- ✓ Affecter les tâches à leur projet.
- ✓ Affecter la tâche à l'équipe de travail concerné.
- ✓ Affecter à chaque tâche sa dépense concernée.

Par contre les autres employés peuvent :

- ✓ Consulter la liste des tâches.
- ✓ Consulter les tâches qui leur sont affectées.
- ✓ Modifier l'état (en préparation, en cours, terminé) de leurs tâches.

### ▪ **Module de gestion des employés :**

Ce module est aussi important pour la gestion et suivi des projets, car les employés sont des acteurs prenants dans l'entreprise et l'exécution des projets la gestion des employés est consacrée seulement pour l'administrateur du système qui est le directeur ou le chef du projet de l'entreprise où il peut :

- ✓ Ajouter, modifier ou supprimer un employé.
- ✓ Affecter les tâches aux employés.
- ✓ Affecter les employés aux équipes du travail.

### ▪ **Module de gestion des équipes :**

La gestion des équipes complète la gestion des tâches, car pour chaque il faut affecter une équipe de travail qui va s'occuper de sa réalisation, le chef de projet qui s'authentifie en tant que rôle d'administrateur qui va s'occuper de :

- ✓ Ajouter, modifier ou supprimer une équipe.
- ✓ Désigner le chef d'équipe ou le responsable de cette équipe.
- ✓ Affecter les employés aux équipes du travail.
- ✓ Affecter les tâches aux équipes

### ▪ **Module de gestion des clients :**

Ce module permet à l'entreprise de consulter et savoir ses clients avec lesquels elle travaille ses projets, l'administrateur qui peut être le directeur où le chef de projet qui va :



- ✓ Ajouter, modifier ou supprimer un client.
- ✓ Affecter un projet à un client.

### ▪ Module de gestion de la Facturation :

La gestion de la facturation permet de facturer les projets pour envoyer la facture au client, ce module concerne seulement de directeur qui aura le rôle d'administrateur et qui va permettre de :

- ✓ Ajouter, modifier ou supprimer une facture.
- ✓ Affecter la facture à un projet.
- ✓ Consulter la liste des factures.
- ✓ Générer la facture en format PDF pour l'envoyer au client.

### ▪ Module de gestion des dépenses :

Ce module permet gérer les dépenses de chaque tâche du projet, chaque dépense a une catégorie. La gestion des dépenses est la responsabilité de l'administrateur il peut :

- ✓ Ajouter, modifier ou supprimer une dépense.
- ✓ Désigner la catégorie de la dépense.
- ✓ Affecter la dépense à une tâche.

## 4- Démarche de développement

Le processus de développement constitue un facteur déterminant dans la réussite d'un projet, du fait qu'il cadre ses différentes phases et caractérise les principaux traits de sa conduite.

Pour cela, le choix d'une méthode de développement, qui soit adéquate aux particularités et exigences d'un projet, doit être élaboré au préalable afin d'obtenir un produit de qualité qui répond aux besoins et aux attentes des utilisateurs.

Parmi les processus de développement les plus répandus on trouve :

	Description	Points forts	Points faibles
<b>Cascade</b>	<ul style="list-style-type: none"> <li>Propose de dérouler les phases projet de manière séquentielle.</li> </ul>	<ul style="list-style-type: none"> <li>Distingue clairement les phases projet.</li> </ul>	<ul style="list-style-type: none"> <li>Non itératif.</li> <li>Ne propose pas de modèles de documents.</li> </ul>
<b>2TUP Two Track Unified</b>	<ul style="list-style-type: none"> <li>S'articule autour de l'architecture</li> <li>Propose un cycle de développement en Y Détaillé</li> </ul>	<ul style="list-style-type: none"> <li>Itératif.</li> <li>Fait une large place à la technologie et à la gestion du risque Définit les profils des</li> </ul>	<ul style="list-style-type: none"> <li>Plutôt superficiel sur les phases situées en amont et en aval du développement : capture des besoins, support,</li> </ul>

Process	dans « UML en action » (voir références).	intervenants, les livrables, les plannings, les prototypes.	maintenance, gestion du changement.
<b>En V</b>	<ul style="list-style-type: none"> <li>Il est représenté par un V dont la branche descendante contient toutes les étapes du projet, et la branche montante toutes les étapes de tests du projet.</li> </ul>	<ul style="list-style-type: none"> <li>Facile à utiliser.</li> <li>Les tests sont effectués à chaque étape.</li> <li>Le contrôle se fait progressivement à chaque étape.</li> <li>Les phases de validation sont prises en main très tôt dans le processus de développement.</li> </ul>	<ul style="list-style-type: none"> <li>Le processus n'est pas itératif.</li> <li>Une mauvaise prise en compte des changements de la spécification des besoins.</li> <li>Ne contient pas les activités d'analyses de risques.</li> </ul>
<b>EN SPIRALE</b>	<p>Le cycle en spirale reprend les étapes du cycle en V, mais prévoit l'implémentation de n versions successives, ce qui permet de mettre l'accent sur la gestion des risques, la première phase de chaque itération étant dédiée à ce poste.</p>	<ul style="list-style-type: none"> <li>Sans coût élevé, donne des indications sur les risques majeurs.</li> <li>Les fonctions critiques à haut risque sont développées en premier lieu.</li> <li>La conception ne doit pas forcément être terminée.</li> <li>Les utilisateurs ont dès le départ une vue globale du système.</li> </ul>	<ul style="list-style-type: none"> <li>Le temps consacré à l'évaluation des risques est trop élevé pour des petits projets.</li> <li>Le temps mis à planifier, évaluer les risques, fixer les objectifs, les prototypes peut être excessif.</li> <li>Ce modèle est complexe</li> </ul> <p>Une expertise en évaluation des risques est nécessaire.</p>
	<p>Un schéma d'organisation de développement de produits complexes.</p> <p>Il est défini par ses créateurs comme un « cadre de travail permettant de répondre à des problèmes</p>	<ul style="list-style-type: none"> <li>Scrum est centrée sur le produit.</li> <li>Progrès est incrémental, facilement mesurable et clairement visible pour les acteurs économiques.</li> <li>Les développeurs fixeront le rythme, ne sont pas surchargés de</li> </ul>	<ul style="list-style-type: none"> <li>Projets Scrum sans Product owner actif et sans acteurs économiques engagés sera probablement fiasco.</li> <li>Si le propriétaire du produit n'a pas apprécié Scrum, ou ne comprend pas le rôle du</li> </ul>

<p><b>SCRUM</b></p>	<p>complexes et changeants, tout en livrant de manière productive et créative des produits de la plus grande valeur possible ».</p>	<p>travail, et bénéficient d'un rôle accru.</p> <ul style="list-style-type: none"> <li>• La charge du travail est réglable, basé sur la capacité de l'équipe et la priorité des tâches à réaliser.</li> <li>• Les questions sont généralement découvertes avant qu'ils ne deviennent endémiques.</li> <li>• L'équipe a l'autorité de prendre des décisions.</li> <li>• L'équipe est encouragée à « consulter » avec les entreprises Les tâches ont tendance à être granulaire, et par conséquent, plus facilement testables.</li> </ul>	<p>propriétaire de produit, le succès est plus difficile.</p> <ul style="list-style-type: none"> <li>• Si le propriétaire du produit ne se présente pas où apprécier dette technique articulée par l'équipe, des problèmes techniques peuvent se supprimer.</li> <li>• Si les développeurs ne pas co-localisé ou à portée de voix du client et d'affaires, la productivité va en souffrir.</li> <li>• Les normes organisationnelles.</li> </ul>
---------------------	---	---	---

Tableau 3: Tableau comparative des différents cycles de vie

Après avoir étudié la nature de notre projet, ce dernier incite à suivre une méthode agile et plus précisément la méthode Scrum. En effet les spécifications du projet n'ont pas cessé d'être modifiées tout au long du projet et les outils et les technologies n'ont pas été explicitées depuis le départ. Scrum permet d'entamer le projet avec un minimum d'informations sur les travaux à faire et qui seront ajustées régulièrement au cours du projet.

### i. Méthode de SCRUM :

Les méthodes agiles sont des méthodologies essentiellement dédiées à la gestion de projets informatiques. Elles reposent sur des cycles de développement itératifs et adaptatifs en fonction des besoins évolutifs du client. Elles permettent notamment d'impliquer l'ensemble des collaborateurs ainsi que le client dans le développement du projet.

### ii. Approche Agile :

Depuis une quinzaine d'années, la majorité des développements des logiciels s'appuie sur des méthodes dites "agiles". Sous cette bannière se regroupent plusieurs méthodes basées sur un développement itératif et incrémental, dans lequel la recherche de solutions aux problèmes rencontrés s'appuie sur la collaboration de pair à pair. Elle promeut des réponses rapides et flexibles, une planification des tâches adaptatives dans des laps de temps très courts permettant une très grande réactivité.

Les approches plus classiques tels que cycle en V ou le modèle en cascade sont souvent mises en œuvre pour les projets répondant à un imposant cahier des charges. Le client et le prestataire s'entendent alors sur un contrat et si le projet prend du retard ou ne répond pas à tous les besoins à la date butoir, des pénalités sont alors facturées au prestataire. De la même manière si le client s'aperçoit en cours de route que certains besoins ont été omis dans le cahier des charges, il devra alors renégocier avec le prestataire le contrat.

Les premières méthodes agiles apparues sont EVO (Evolutionary Project Management) (1976), RAD (développement rapide d'applications) (1991), puis DSDM, la version anglaise du RAD (1995). Les trois méthodes agiles désormais les plus utilisées sont : la méthode Kanban, issue de la méthode industrielle Lean, la méthode Scrum publiée en 2001 par Ken Schwaber et Jeff Sutherland, et la méthode XP (Extreme programming) publiée en 1999 par Kent Beck.

### iii. Pourquoi la méthode SCRUM ?

**SCRUM** est de très loin la méthodologie la plus utilisée parmi les méthodes Agile existantes. Elle est donc la plus éprouvée, documentée et supportée. Livres, blogs, formations, vidéos, associations, conférences traitant de Scrum ne manquent pas et bon nombre de ces ressources sont accessibles gratuitement. On pourrait pratiquement parler d'un standard Agile. Un autre atout important : Scrum est simple à comprendre. Sa maîtrise est en revanche difficile.

Les experts de Scrum, même ses fondateurs, le décrivent comme un « cadre de travail permettant de répondre à des problèmes complexes et changeants tout en livrant de manière productive et créative des produits de la plus grande valeur possible » Scrum propose un modèle de contrôle de processus basé sur l'empirisme.

Il s'appuie sur trois piliers :

- **Transparence** : Les aspects importants du processus doivent être visibles à ceux qui sont responsables des retombées. La transparence requiert la définition d'un standard commun pour ces aspects afin que les observateurs partagent une compréhension commune de ce qui est observé.
- **L'inspection** : Les utilisateurs de Scrum doivent fréquemment inspecter les artefacts Scrum et l'état d'avancement par rapport à un objectif de Sprint (Sprint Goal) afin de détecter les écarts indésirables.

- **L'adaptation :** Si un inspecteur détermine qu'un ou plusieurs aspects du processus dérivent hors des limites acceptables, et que le produit qui en résulte sera inacceptable, le processus ou le matériel utilisé par le processus doit être ajusté. Un ajustement doit être fait dès que possible afin de minimiser le risque d'autres dérives.

### iv. Fonctionnement de Scrum

La méthode Scrum s'appuie sur le découpage des projets en itérations encore nommées « sprints ». Un sprint peut avoir une durée qui varie généralement entre deux semaines et un mois.

Avant chaque sprint, les tâches sont estimées en temps et en complexité à l'aide de certaines pratiques comme le « planning poker », une manière ludique de chiffrer la complexité des tâches ou évolutions à l'aide de cartes à l'instar du célèbre jeu dont le nom est repris. Ces estimations permettent à la fois de planifier les livraisons, mais aussi d'estimer le coût de ces tâches auprès du client. Les fonctionnalités (encore appelées « user stories ») qui font l'objet d'un sprint constituent ce que l'on appelle un « sprint back log » du produit éventuellement livrable à la fin du sprint. Il est nécessaire de distinguer le sprint back log du « Product back log » qui lui correspond à l'ensemble des fonctionnalités attendues pour le produit sur l'ensemble des sprints.

La méthode Scrum est aussi caractérisée par une « mêlée » quotidienne, encore appelée « morning » ou « stand-up », dans laquelle les collaborateurs (chefs de projets, développeurs et responsables fonctionnels) indiquent tour à tour les tâches qu'ils ont effectuées la veille, les difficultés rencontrées et enfin ce sur quoi ils vont poursuivre leur travail le jour suivant. Cela permet d'évaluer l'avancement du projet, de mobiliser des ressources là où cela est le plus nécessaire, mais aussi de venir en aide aux collaborateurs rencontrant des difficultés lorsque celles-ci ont déjà été rencontrées auparavant par d'autres membres de l'équipe.

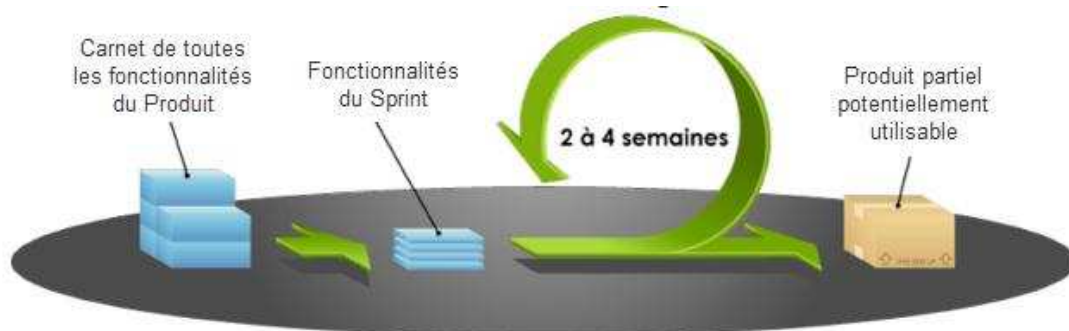


Figure 9: la méthode Scrum

## 5- Planification du projet

Cette partie a pour objectif de présenter la manière avec laquelle le projet a été géré. Nous allons présenter dans un premier temps comment le projet a été découpé en tâches afin d'assurer son bon déroulement.

### 5-1 Planification

La planification du projet est une étape importante d'avant-projet. Elle consiste à prévoir le déroulement du projet tout au long des phases constituant le cycle de développement.

Le tableau ci-dessous (tableau 4) montre le planning de notre projet que nous avons réalisé avec Gantt Project, nous l'avons réalisé avant de commencer à travailler sur notre projet lors de l'élaboration du cahier des charges, nous avons essayé de bien le respecter, mais parfois nous avons eu certaines contraintes qui nous ont obligées de retarder la réalisation de certaines tâches parfois pendant des semaines.

Activity	↕	Début	Fin	Jours	
<b>Rédaction du rapport</b>		15-02-16	07-06-16	82.0	82.0
<b>Etude préalable</b>		15-02-16	16-02-16		2.0
Cadre général du projet		15-02-16	15-02-16	1.0	
Etude de l'existant		16-02-16	16-02-16	1.0	
<b>Etude de projet</b>		17-02-16	03-03-16	12.0	24.0
Etude fonctionnelle		17-02-16	23-02-16	5.0	
Rédaction du cahier des charges		24-02-16	25-02-16	2.0	
Etude technique		26-02-16	03-03-16	5.0	
<b>Conception</b>		03-03-16	23-03-16	15.0	30.0
Diagramme de cas d'utilisation		03-03-16	09-03-16	5.0	
Diagramme de classes		10-03-16	14-03-16	3.0	
Diagramme de séquence		15-03-16	21-03-16	5.0	
Diagramme d'activité		22-03-16	23-03-16	2.0	
<b>Réalisation</b>		24-03-16	06-06-16	5.0	80.0
Sprint1: Module de gestion des clients		24-03-16	30-03-16	5.0	
▲ <b>Sprint 2: Module de gestion des employés</b>		31-03-16	06-04-16	1.0	6.0
Récupération sprint 1		31-03-16	31-03-16	1.0	
Développement : Employés +tests unitaires		01-04-16	06-04-16	4.0	
▲ <b>Sprint 3: Module de gestion des équipes</b>		07-04-16	13-04-16	5.0	10.0
Récupération sprint 2		07-04-16	07-04-16	1.0	
Développement : Equipes +tests unitaires		08-04-16	13-04-16	4.0	
▲ <b>Sprint 4: Module de gestion des projets</b>		14-04-16	21-04-16	6.0	12.0
Récupération sprint 3		14-04-16	14-04-16	1.0	
Développement : Projets +tests unitaires		15-04-16	21-04-16	5.0	
▲ <b>Sprint 5: Module de gestion des taches</b>		22-04-16	29-04-16	1.0	7.0
Récupération sprint 4		22-04-16	22-04-16	1.0	
Développement : Taches +tests unitaires		25-04-16	29-04-16	5.0	
▲ <b>Sprint 6: Module de gestion des dépenses</b>		02-05-16	06-05-16	1.0	6.0
Récupération sprint 5		02-05-16	02-05-16	1.0	
Développement : Dépenses +tests unitaires		03-05-16	06-05-16	4.0	
▲ <b>Sprint 7: Module de gestion des factures</b>		09-05-16	13-05-16	1.0	6.0
Récupération sprint 6		09-05-16	09-05-16	1.0	
Développement : Factures +tests unitaires		10-05-16	13-05-16	4.0	
▲ <b>Sprint 8: Module de gestion des utilisateurs</b>		16-05-16	20-05-16	5.0	10.0
Récupération sprint 7		16-05-16	16-05-16	1.0	
Développement : Utilisateurs +tests unitaires		17-05-16	20-05-16	4.0	
▲ <b>Sprint 9: Reporting</b>		17-05-16	06-06-16	1.0	13.0
Récupération sprint 8		23-05-16	23-05-16	1.0	
Reporting		24-05-16	26-05-16	3.0	
Tests fonctionnels		26-05-16	06-06-16	8.0	

Tableau 4: Planification du projet

## **5-2 Diagramme de Gantt**

Diagramme de GANTT est un outil permettant de modéliser la planification de tâches nécessaires à la réalisation du projet, il visualise dans le temps les diverses tâches liées composant le projet en fournissant une représentation graphique de son avancement.

La figure ci-dessous (figure 10) montre le diagramme de Gantt de notre projet :



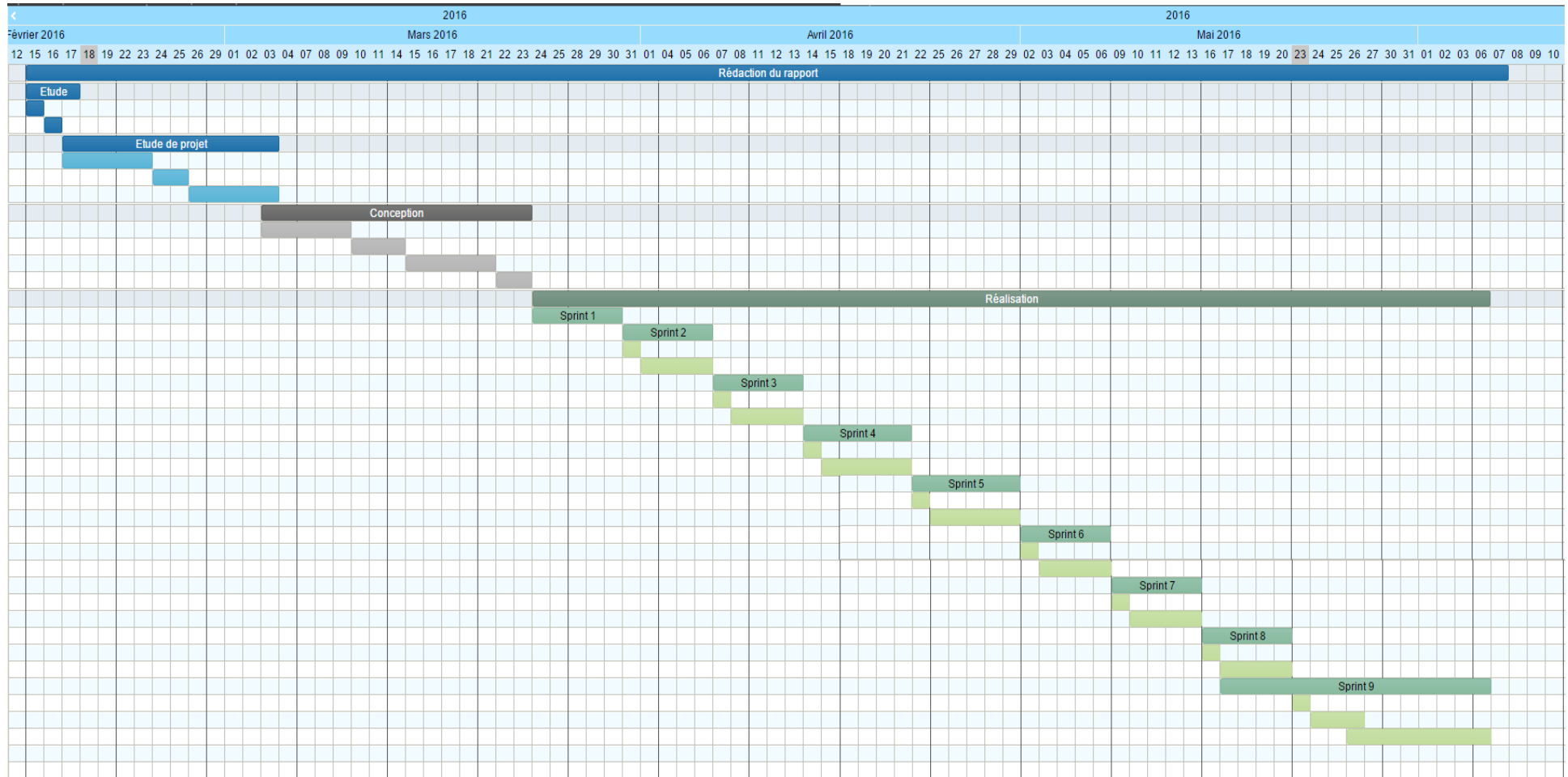


Figure 10: Diagramme de GANTT de notre projet

### Conclusion

Dans ce chapitre avons présenté en détail le projet, sa finalité, ses objectifs et ses différentes caractéristiques. Nous avons ensuite effectué une description de notre cahier de charge fonctionnels, et nous avons présenté la méthode de développement suivie, qui est Scrum et enfin, nous avons donné un aperçu de la planification du projet a l'aide d'un diagramme de Gantt.

# Chapitre 3

---

## Etude Technique du projet

Dans ce chapitre, nous présenterons l'étude technique détaillée du projet. Nous allons parler de l'architecture logicielle de notre projet ainsi que la plateforme utilisée pour le développement de JAVA EE, et les outils et techniques utilisés.

### Introduction

L'étude technique mettra en relief les principales plates-formes utilisées, et parlera également des différentes techniques utilisées pour la sécurisation lors du déploiement.

### 1- Architecture logicielle

Parmi les différentes façons de structurer une architecture, la mieux adaptée et maîtrisée en informatique est l'approche par couches. Une couche (Layer en anglais) est une division logique et horizontale d'un système qui fournit une abstraction particulière du système à couches supérieures. Chaque couche possède des responsabilités spécifiques. Dans une structuration par couches, les couches inférieures prennent en charges des responsabilités qui offrent un socle de base pour les couches supérieures, permettant d'abstraire celles-ci de problématiques qui ne les concernent pas.

Ainsi, nous avons adopté en découpage en quatre couches. Une telle architecture permet également d'obtenir un bon niveau de réutilisabilité, à travers la réplication des solutions aux problèmes rencontrés pour chaque couche, dans d'autres applications utilisant la même architecture.

Ce volet technique (découpage en couches) a été testé par un projet prototype qui nous a permis de valider notre architecture logicielle et que les messages voyagent bien depuis la couche de présentation jusqu'à la couche de stockage.

#### Couche Présentation :

Cette couche correspond à la partie de l'application visible et interactive avec les utilisateurs. On parle d'Interface Homme-Machine. Dans la plupart des cas, il s'agit d'une interface client riche ou d'une interface web. Elle intercepte les événements et fait appel aux services de la couche ViewModel.

#### Couche ViewModel :

Cette couche regroupe les classes java nécessaires au logique métier, elle comporte donc les services offerts par le système à l'utilisateur. Ces services ne sont autres que l'implémentation des scénarios identifiés lors de la phase de conception.

Lors de l'appel d'un service, ce dernier fait appel aux objets métier nécessaires pour répondre à la requête de l'utilisateur.

#### Couche Service :

C'est une couche intermédiaire entre les couches ViewModel et la couche DAO (Data Access Object), son rôle est de gérer les transactions qui seront ouverts à chaque appel de la base de données.

Elle permet aussi de répartir les ressources entre les différents utilisateurs de l'application afin de garantir une meilleure performance du serveur (utilisation des ressources seulement à la demande).

### Couche d'accès aux données :

C'est la couche qui regroupe les classes Java capables de fournir les données nécessaires à la couche métier (ViewModel), et par la suite aux services demandés par l'utilisateur, cette couche est modélisée dans notre système par des composants DAO.

Un DAO est un design pattern (patron de conception) utilisé dans les architectures logicielles objet visant à isoler la logique de persistance dans des classes d'accès aux données.

### Couche stockage :

Cette couche est responsable du stockage des données dans une base de données relationnelles dont le modèle physique est déduit à partir du diagramme de classes du système. Elle est gérée dans notre cas par le SGBD MySQL.

### Couche sécurité :

Il s'agit d'une couche dédiée à la sécurité, chargée de vérifier l'authentification et les autorisations d'accès aux pages et aussi le hachage des mots de passes des utilisateurs.

## 2- Méthodologie de développement : Démarche MVC

Le modèle d'architecture qui cherche à séparer nettement les couches de présentation (UI : User Interface), métier (BLL : Business Logic Layer) et d'accès aux données (DAL : Data Access Layer).

Le but étant d'avoir une dépendance minimale entre les différentes couches de l'application, ainsi les modifications effectuées sur n'importe quelle couche de l'application n'affectent pas les autres couches.

- Modèle : encapsule le cœur fonctionnel de l'application, le domaine logique.
- Vue : les données sont envoyées, par le modèle, à la vue qui les représente à l'utilisateur.
- Contrôleur : reçoit les données et les transmet au modèle ou à la vue.

Une telle architecture est communément appelée architecture 3-tier ou à 3 niveaux.

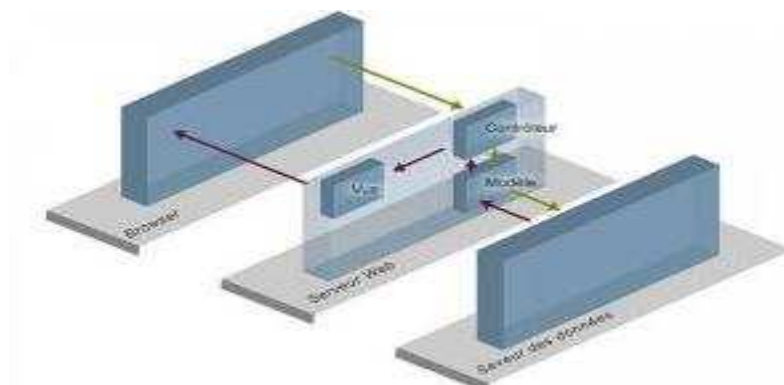


Figure 11: Architecture 3tiers

### Pourquoi MVC ?

Comme nous l'avons déjà cité dans l'étude fonctionnelle, notre application doit être à la disposition de tous ses utilisateurs (Administrateur, modérateur, utilisateur).

C'est pourquoi nous avons décidé qu'elle soit une application WEB, comme ça elle va être accessible à tous les utilisateurs. Donc nous avons été amenés à faire une étude sur les Frameworks permettant de réaliser ce type d'application.

L'objectif de MVC est de faire collaborer deux équipes : une équipe à consonance infographie et une équipe à consonance informatique. Le paradigme MVC est un schéma de programmation qui propose de séparer une application en 3 parties :

Le modèle qui contient la logique et l'état de l'application.

La vue qui représente l'interface utilisateur.

Le contrôleur qui gère la synchronisation entre la vue et le modèle. Le contrôleur réagit aux actions de l'utilisateur en effectuant les actions nécessaires sur le modèle et surveille les modifications du modèle et informe la vue des mises à jour nécessaires.

### 3- La plateforme Java EE

Java EE simplifie le développement des applications d'entreprise en les basant sur des composants standardisés et modulaires, en fournissant un jeu complet de services à ces composants et en prenant automatiquement en charge un grand nombre des détails du comportement de l'application, déchargeant le développeur d'un travail complexe de programmation. Sun et un groupe d'industriels leaders du marché, sous les auspices de la communauté ouverte JCP, ont permis d'unifier la totalité des standards et des API dans le cadre de la plate-forme Java EE.

#### ▪ Etude comparative entre les Frameworks [JAVA/.NET] et [JAVA/LARAVEL] :

Pour réaliser une étude comparative entre les deux Frameworks, nous avons utilisé « Google Trends », cet outil va ne permettre de savoir la fréquence de recherche de ces deux termes sur le moteur de recherche Google. Ainsi les résultats qui se montrent dans les figures suivantes (figure 12 et 13) nous ont donné une idée globale sur le Framework le plus utilisé ou autrement dit le Framework qui a la plus grande quantité des ressources sur Internet.



Figure 12 : Comparaison entre java et .net

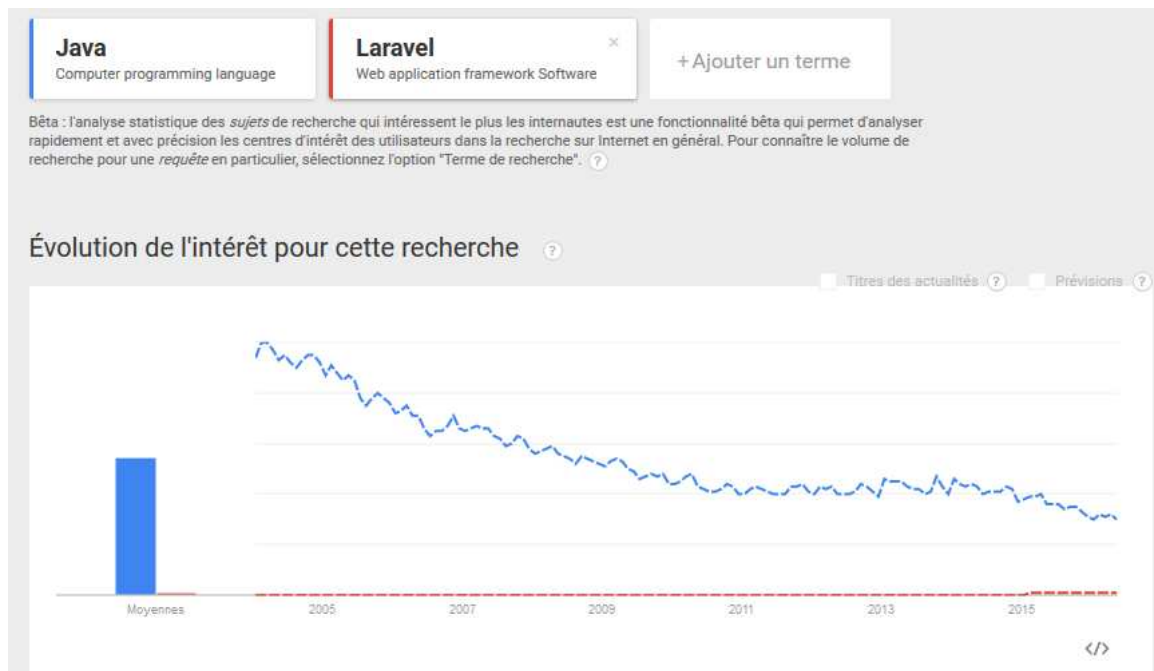


Figure 13: Comparaison entre java et LARAVEL

Donc on peut conclure que le terme JAVA, est le plus demandé sur Google depuis les années 2005 et jusqu'à aujourd'hui. De plus, ce terme a plus de 216 millions résultats archivées [pages web et documents].

### 4- Frameworks

Un Framework est un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel (architecture).

## 4-1 Spring Framework



Figure 14: logo de Spring

Spring est un Framework libre pour construire et définir l'infrastructure d'une application java, dont il facilite le développement et les tests.

<b>Première version</b>	<b>25 Juin 2003</b>
<b>Dernière version</b>	4.2.1 (1 <sup>er</sup> septembre 2015)
<b>Environnement</b>	Multi-plateforme
<b>Écrit-en</b>	Java
<b>Type</b>	Framework
<b>Licence</b>	Licence Apache
<b>Site web</b>	<a href="http://www.spring.io">www.spring.io</a>

Tableau 5: Fiche technique de Spring

### Principe et avantages :

SPRING est effectivement un conteneur dit « léger », c'est-à-dire une infrastructure similaire à un serveur d'application J2EE. Il prend donc en charge la création d'objets et la mise en relation d'objets par l'intermédiaire d'un fichier de configuration qui décrit les objets à fabriquer et les relations de dépendances entre ces objets.

Le gros avantage par rapport aux serveurs d'application est qu'avec SPRING, vos classes n'ont pas besoin d'implémenter une quelconque interface pour être prises en charge par le Framework (au contraire des serveurs d'applications J2EE et des EJBs). C'est en ce sens que SPRING est qualifié de conteneur « léger ».

Spring s'appuie principalement sur l'intégration de trois concepts clés :

- ✓ L'inversion de contrôle est assurée de deux façons différentes : la recherche de dépendances et l'injection de dépendances.
- ✓ La programmation orientée aspect.
- ✓ La couche d'abstraction.

La couche d'abstraction permet d'intégrer d'autres Frameworks et bibliothèques avec une plus grande facilité. Cela se fait par l'apport ou non de couches d'abstraction spécifiques à des Frameworks particuliers. Il est ainsi possible d'intégrer un module d'envoi de mails plus facilement.

### Pourquoi utiliser Spring ?

Spring est un Framework open source et totalement gratuit, ce qui est un plus pour notre projet. En effet, on doit seulement utiliser des Frameworks gratuits. La documentation de Spring est complète et régulièrement mise à jour lors de la diffusion de chaque nouvelle version. C'est un avantage majeur pour nous en tant que débutants dans cette technologie.



Spring est composé de plusieurs modules disponibles sous forme de fichiers jar (Bibliothèques pour tous types de projets et pour tous les besoins de développeurs). Ainsi, on peut n'ajouter au projet que la partie que l'on souhaite utiliser. Cette richesse en fonctionnalités va nous permettre une évolution très rapide du futur système.

Spring fournit des services de type fonctionnel comme par exemple la gestion de transactions (Base de données), ce service est très important, pour une meilleure gestion de ressources. Aussi il y a moins de dépendances entre les différents modules.

## 4-2 Hibernate Framework



Figure 15: logo de hibernate

Hibernate est une solution open source de type ORM (Object Relational Mapping) qui permet de faciliter le développement de la couche persistance d'une application. Hibernate permet donc de représenter une base de données en objets Java et vice versa.

Développeur	Red Hat
Dernière version	5.1.0
Ecrit-on	Java
Environnement	Multiplate-forme (JVM)
Type	Mapping objet-relationnel
Licence	GNU LGPL
Site web	<a href="http://www.hibernate.org">www.hibernate.org</a>

Tableau 6: Fiche Technique de Hibernate

Hibernate facilite la persistance et la recherche de données dans une base de données en réalisant lui-même la création des objets et les traitements de remplissage de ceux-ci en accédant à la base de données. La quantité de code ainsi épargnée est très importante d'autant que ce code est généralement fastidieux et redondant.

Hibernate est très populaire notamment à cause de ses bonnes performances et de son ouverture à de nombreuses bases de données.

Les bases de données supportées sont les principales du marché : DB2, Oracle, MySQL, PostgreSQL, Sybase, SQL Server, Sap DB, Interbase, ...

### Pourquoi utiliser Hibernate ?

Hibernate génère le code SQL nécessaire, ce qui rend l'application plus portable (s'adapte à la base de données). Il évite l'écriture de code répétitif, inintéressant et source d'erreurs difficiles à décoder. Aussi La persistance est transparente et on peut faire de vos classes métiers des classes persistantes sans ajout de code, comme on peut remarquer que la récupération de données est optimisée et on peut interroger la base de données de plusieurs façons (Requête SQL, langage HQL...)

Hibernate présente plusieurs avantages parmi lesquels on trouve :

- Portabilité du code en cas de changement du SGBD.

- Le développeur pense en termes d'objet et pas en termes de lignes de tables.
- Le refactoring du schéma de la base de données ou du modèle objet est facilité.

### 4-3 JSP



Figure 16: logo de JSP

Les JSP (Java Server Pages) sont une technologie Java qui permet la génération de pages web dynamiques.

La technologie JSP permet de séparer la présentation sous forme de code HTML et les traitements écrits en Java sous la forme de JavaBeans ou de servlets. Ceci est d'autant plus facile que les JSP définissent une syntaxe particulière permettant d'appeler un Bean et d'insérer le résultat de son traitement dans la page HTML dynamiquement.

#### Pourquoi utiliser JSP ?

Une très bonne intégration du code Java et du HTML, minimisant par conséquent les problèmes de mise en page. Ceci permet surtout de développer des pages web dynamiques s'appuyant sur un grand volume de données et dont le contenu est adaptable au caractère évolutif de ces données et aux requêtes particulières des utilisateurs.

### 4-4 Spring Security



Figure 17: logo de spring security

Spring Security est un projet du portfolio Spring qui propose une solution de sécurisation complète intégrée aux systèmes utilisant Spring. Très largement utilisée au sein de la communauté Spring, elle peut de facto être considérée comme un standard.

#### Pourquoi utiliser Spring Security ?

Le premier avantage de Spring Security est sa portabilité. Ne dépendant pas d'un serveur d'applications particulier, son utilisation est identique quel que soit le serveur utilisé.

Cette portabilité est particulièrement importante si l'application développée doit pouvoir être vendue à un grand nombre de clients possédant des systèmes hétérogènes.

Il fournit en standard un nombre de fonctionnalités beaucoup plus important qu'un serveur Java EE classique. Parmi les plus simples, et qui manquent cruellement dans la spécification Java EE, citons l'authentification automatique par cookie pour un nombre donné de jours, ainsi que la vérification qu'un utilisateur n'est pas déjà authentifié avec le login demandé.

Enfin il propose en outre des fonctionnalités avancées, telles que le support de solutions de Single Sign-On (une authentification unique pour l'ensemble des applications de l'entreprise) ou la sécurisation des objets de domaine, fournissant ainsi une aide considérable au développement d'applications ayant des besoins complexes en matière de sécurité.

### 4-5 JUnit

JUnit est un Framework open source pour le développement et l'exécution de tests unitaires automatisables. Le principal intérêt est de s'assurer que le code répond toujours aux besoins même après d'éventuelles modifications. Plus généralement, ce type de tests est appelé tests unitaires de non-régression.

JUnit propose:

- Un Framework pour le développement des tests unitaires reposant sur des assertions qui testent les résultats attendus.
- Des applications pour permettre l'exécution des tests et afficher les résultats.

## 5- Environnement de travail

Dans cette partie nous allons présenter l'environnement dans lequel nous avons pu réaliser notre application.

### 5-1 Spring Source Tool Suite (STS)



Figure 18: Interface de STS

SpringSource Tool Suite (STS) est une version modifiée d'Eclipse qui fournit un environnement de développement pour Spring. Il offre plusieurs fonctionnalités dont voici les principales :

- La simplification de la recherche des Beans Spring grâce à la fenêtre Open Bean. Il est possible de les retrouver via leur id, de la même manière qu'on peut retrouver une classe java.
- L'auto-complétion qui apporte une assistance dans les fichiers XML au niveau des noms des classes Java, des propriétés, des Beans Spring références mais aussi des

propriétés possibles (cette option est très pratique surtout quand on débute avec Spring).

- Possibilité de navigation entre les Beans Spring références ou vers les méthodes au niveau des fichiers XML (notamment pour l'injection de dépendances), cela présente un réel gain de temps mais aussi une facilité de compréhension du code. C'est très appréciable pour les développeurs surtout pour des gros projets utilisant la surcharge, l'héritage et le polymorphisme.
- La gestion d'erreur et aussi un avantage : STS indique les erreurs ainsi que leurs emplacements. Il est toujours plus facile de corriger des bugs quand ils sont situés dans le code et qu'ils comportent des messages clairs et précis.

### 5-2 vFabric tc Server

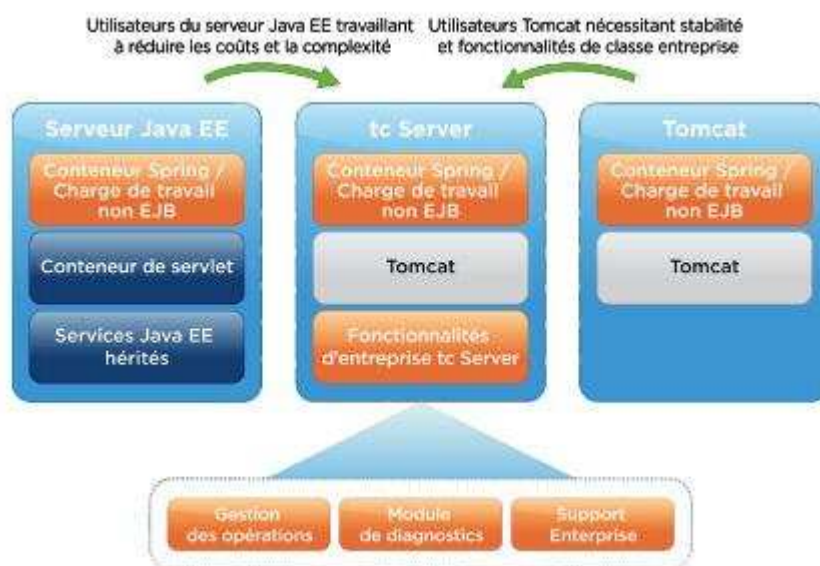


Figure 19: Architecture vFabric tc Server

VMware vFabric tc Server est le serveur d'exécution au cœur de la plate-forme applicative de Cloud VMware vFabric. Disponible en édition Spring, Standard et Developer, tc Server est l'outil de choix pour concevoir et exécuter les applications Spring. En tant que version d'entreprise d'Apache Tomcat, tc Server offre des fonctionnalités principales améliorant l'efficacité des développeurs, le contrôle opérationnel et la flexibilité de déploiement.

Spring remplace la lourdeur du conteneur Enterprise JavaBeans (EJB) pour devenir le standard de fait pour la majorité des applications Java d'entreprise. Spring a également été un facteur important dans l'adoption de Tomcat, aujourd'hui utilisé par plus de 70 % des entreprises, d'après une étude d'Evans Data.

Aujourd'hui, la pile Spring-Tomcat alimente un grand nombre des applications les plus gourmandes au monde, notamment dans le secteur bancaire, de la vente au détail, de la production et de la santé.

Tc Server offre les avantages de Tomcat en termes de puissance et de légèreté, associés aux importantes fonctionnalités de classe entreprise dont ont aujourd'hui besoin les sociétés.

Il est idéal pour les entreprises utilisant Spring avec des serveurs hérités Java Enterprise Edition (EE) mais recherchant une alternative économique et abordable. Il convient également aux utilisateurs Tomcat ayant besoin de fonctionnalités supplémentaires de diagnostic et de gestion opérationnelle, doublées d'un service d'assistance aux entreprises.

### 6- Système de Gestion de Base de Données



*Figure 20: logo MySQL*

MySQL est un serveur de bases de données relationnelles SQL, développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multithread et multi-utilisateur.

C'est un logiciel libre développé sous double licence en fonction de l'utilisation qui en est faite : dans un produit libre ou dans produit propriétaire. Dans ce dernier cas, la licence est payante, sinon c'est la licence publique générale GNU (GPL) qui s'applique.

### 7- Les Outils Utilisés

#### 7-1 XML

XML est en quelque sorte un langage HTML amélioré permettant de définir de nouvelles balises. Il s'agit effectivement d'un langage permettant de mettre en forme des documents grâce à des balises (markup).

Contrairement à HTML, qui est considéré comme un langage défini et figé (avec un nombre de balises limité), XML peut être considéré comme un métalangage permettant de définir d'autres langages, c'est-à-dire définir de nouvelles balises permettant de décrire la présentation d'un texte.

La force de XML réside dans sa capacité à pouvoir décrire n'importe quel domaine de données grâce à son extensibilité. Il va permettre de structurer, poser le vocabulaire et la syntaxe des données qu'il va contenir. En réalité les balises XML décrivent le contenu plutôt que la présentation (contrairement à HTML).

Ainsi, XML permet de séparer le contenu de la présentation, ce qui permet par exemple d'afficher un même document sur des applications ou des périphériques différents sans pour autant nécessiter de créer autant de versions du document que l'on nécessite de représentations. XML a été mis au point par le XML Working Group sous l'égide du World Wide Web Consortium (W3C) dès 1996.

### 7-2 AJAX



Figure 21: logo d'Ajax

Dans notre application, nous avons besoin des interfaces riches avec des comportements dynamiques, en effet nous avons besoin de créer des formulaires de saisie où l'application va proposer les valeurs possibles à saisir dans un champ, après avoir saisi les premières lettres.

Nous avons besoin aussi de créer des formulaires de saisie ; dont le nombre de champs dépend du nombre de données que l'utilisateur veut entrer. Donc c'est celui-ci qui va créer le nombre de champs dont il a besoin, autant plus nous avons besoin de consulter des sources de données et faire des traitements coté serveur sur une partie de la page Web d'une manière instantanée et sans avoir recours à envoyer la page toute entière, comme c'est le cas pour vérifier la validité d'un nom du fichier ou sa version lorsqu'ils sont saisis par un acteur. Donc nous avons choisi d'utiliser Ajax, car il permet de répondre à ces besoins, et il y'a plusieurs Framework permettant d'implémenter ces fonctionnalités.

### 7-3 La bibliothèque JQuery

JQuery est une bibliothèque JavaScript open-source et cross-browser qui permet de traverser et manipuler très facilement l'arbre DOM des pages web à l'aide d'une syntaxe fortement similaire à celle d'XPath (qui est la syntaxe de manipulation d'arbre XML).

JQuery permet par exemple de changer/ajouter une classe CSS, créer des animations, modifier des attributs, etc. mais également de gérer les événements JavaScript ainsi que de faire des requêtes AJAX le plus simplement du monde.

## 8- Outils de dessin



Figure 22: logo de Visio

Microsoft Visio (officiellement Microsoft Office Visio) est un logiciel de diagrammes et de synoptique pour Windows qui fait partie de la suite bureautique Microsoft Office mais se vend séparément. On peut ainsi créer des diagrammes de Gantt, des réseaux de PERT ou encore des diagrammes IDEFO.

Dans Visio, les graphiques utilisés pour créer des diagrammes sont vectoriels. Les versions Standard et Professionnel de l'édition 2013 partagent la même interface, mais cette dernière permet de faire des diagrammes plus avancés, grâce à des modèles supplémentaires.

Cette version offre également une fonctionnalité unique qui permet aux utilisateurs de relier facilement leurs diagrammes à un grand nombre de sources de données et d'afficher les informations recueillies graphiquement.

Dans notre projet nous avons utilisé Microsoft Office Visio pour la création des diagrammes de cas d'utilisation, diagramme de classe, diagramme d'activité et les diagrammes de séquence.

### **Conclusion**

Au terme de ce chapitre, nous avons capturé les besoins techniques du projet, ainsi nous avons présenté l'architecture logicielle de notre système, et ensuite nous avons cité les différents outils et les environnements de développement utilisés et le rapport détaillé des Frameworks choisis.

# Chapitre 4

---

## Conception détaillée du projet

Dans ce chapitre, nous allons présenter la conception détaillée du notre projet. Nous commencerons par spécifier les cas d'utilisation, ensuite les diagrammes d'activité, de séquence et enfin le diagramme de classe et un diagramme de déploiement.



## Introduction

Ce présent chapitre a pour objectif de détailler la conception du système, partant par l'architecture logique de l'application puis la conception détaillée (diagrammes).

## 1-Diagramme de cas d'utilisation

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision stratégique et globale du comportement fonctionnel d'un système logiciel. Ils sont utiles pour des présentations auprès de la direction ou des acteurs d'un projet.

### 1-1 Identification des acteurs

Un acteur représente un élément externe qui interagit avec le système. Cet élément peut être un utilisateur ou un système tiers (autre ordinateur, autre programme, base de données).

En observant les utilisateurs directs du système ainsi que les autres systèmes qui interagissent avec le système que nous tentons de définir, nous trouvons ces acteurs qui opèrent avec le système : Directeur, Chef de projet, Chef d'équipe, Employé, Client.

Acteur	Description
<b>Directeur</b>	Peut créer des comptes pour les clients, il a le droit de gérer et suivre en temps réel les projets et les tâches, ainsi il peut gérer les dépenses, les équipes, les employés et les utilisateurs du système.
<b>Chef de projet</b>	Peut gérer et suivre en temps réel les projets et les tâches.
<b>Chef d'équipe</b>	Peut gérer les équipes
<b>Employé</b>	Peut créer et valider les tâches des projets ainsi que visualiser celles le concernant. Peut également consulter les projets, les dépenses et suivre le temps des projets. Peut également consulter les projets, les dépenses et suivre le temps des projets.
<b>Client</b>	Peut signer des contrats avec l'entreprise, peut également consulter ses projets en temps réel.

Tableau 7: Identification des acteurs

## 1-2 Identification des cas d'utilisation

Un cas d'utilisation est une unité cohérente représentant une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie. Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service.

Cas d'utilisation « Authentification » :

Titre	Authentification
Acteurs	Directeur, Chef de projet, Chef d'équipe, Employé
Pré-conditions	L'utilisateur (Directeur, Employé) saisit login et mot de passe.
Scenarios	Identification : <ul style="list-style-type: none"> <li>• L'utilisateur saisit ses droits d'accès (login et mot de passe).</li> <li>• Le système vérifie si les champs ne sont pas vides, si erreur alors un message d'erreur s'affiche.</li> <li>• Il vérifie ensuite si les informations sont valides, si erreur alors Exception.</li> <li>• Le système redirige l'acteur vers page d'accueil selon son rôle.</li> </ul>
Post-conditions	L'accès au système.

Tableau 8 : Cas d'utilisation « Authentification »

Cas d'utilisation « Créer projet » :

Titre	Créer un projet
Acteurs	Directeur, Chef de projet
Pré-conditions	L'utilisateur doit avoir le droit d'accès L'utilisateur doit s'authentifier
Scenarios	Etapes <ul style="list-style-type: none"> <li>L'utilisateur soumet une demande d'ajouter un nouveau projet.</li> <li>Le système envoie le formulaire d'ajout d'un nouveau projet</li> <li>L'utilisateur remplit le formulaire et clique sur le bouton Ajouter.</li> <li>Le système vérifie si les champs obligatoires sont complets.</li> <li>Le système vérifie si les champs obligatoires et optionnels sont valides.</li> <li>Enregistrer le projet</li> </ul>
Post-conditions	Projet est créé, et l'utilisateur peut y accéder à nouveau et le modifier.

Tableau 9: Cas d'utilisation « Créer projet »

Cas d'utilisation « Supprimer projet » :

<b>Titre</b>	<b>Supprimer un projet</b>
<b>Acteurs</b>	Directeur, Chef de projet
<b>Pré-conditions</b>	L'utilisateur doit avoir le droit d'accès L'utilisateur doit s'authentifier
<b>Scenarios</b>	Etapes
	L'utilisateur consulte la liste des projets existants.
	Le système affiche la liste des projets enregistrés.
	L'usager choisit un projet et demande la suppression.
	Demande de confirmation de l'action.
	Le système effectue la suppression du projet puis une mise à jour de la base de données.
<b>Post-conditions</b>	Projet est supprimé.

Tableau 10: Cas d'utilisation « Supprimer projet »

Cas d'utilisation « Consulter diagramme de Gantt » :

<b>Titre</b>	<b>Consulter diagramme de gantt</b>
<b>Acteurs</b>	Directeur, Chef de projet, Employé
<b>Pré-conditions</b>	L'utilisateur doit avoir le droit d'accès L'utilisateur doit s'authentifier
<b>Scenarios</b>	Etapes
	L'utilisateur consulte la liste des projets existants.
	Le système affiche la liste des projets enregistrés.
	L'usager choisit un projet afin de visualiser son diagramme de Gantt.
	Le système affiche le diagramme de Gantt détaillée pour le projet concerne.

Tableau 11: Cas d'utilisation « Consulter de diagramme de Gantt »

Cas d'utilisation « Consulter les projets » :

<b>Titre</b>	<b>Consulter les projets</b>
<b>Acteurs</b>	Administrateur, Employé
<b>Pré-conditions</b>	L'utilisateur doit avoir le droit d'accès L'utilisateur doit s'authentifier
<b>Scenarios</b>	Etapes
	L'utilisateur consulte la liste des projets existants.
	L'utilisateur choisit un projet.
	Le système envoie les détails concernant le projet.
<b>Post-conditions</b>	Le détail du projet est affiché.

Tableau 12: Cas d'utilisation « Consulter projet »

Cas d'utilisation « Créer une équipe » :

<b>Titre</b>	<b>Créer une équipe</b>
<b>Acteurs</b>	Chef de projet
<b>Pré-conditions</b>	L'utilisateur doit avoir le droit d'accès

	L'utilisateur doit s'authentifier
<b>Scenarios</b>	Etapes
	L'utilisateur soumet une demande d'ajouter une nouvelle équipe.
	Le système envoie le formulaire d'ajout d'une nouvelle équipe.
	L'usager remplit le formulaire et clique sur le bouton Ajouter.
	Le système vérifie les informations fournies par l'utilisateur.
	Enregistrer l'équipe.
<b>Post-conditions</b>	Equipe est créée.

Tableau 13: Cas d'utilisation « Créer une équipe »

Cas d'utilisation « Créer employé » :

<b>Titre</b>	<b>Créer un employé</b>
<b>Acteurs</b>	Directeur
<b>Pré-conditions</b>	L'utilisateur doit avoir le droit d'accès L'utilisateur doit s'authentifier
<b>Scenarios</b>	Etapes
	L'utilisateur soumet une demande d'ajouter un nouvel employé.
	Le système envoie le formulaire de création d'un nouvel employé
	L'utilisateur doit obligatoirement renseigner les informations nécessaires de l'employé.
	Le système vérifie les informations fournies par l'utilisateur.
	Enregistrer l'employé.
<b>Post-conditions</b>	Employé est créer et il peut suivre et gérer ses propres projets et tâches.

Tableau 14: Cas d'utilisation « Créer employé »

Cas d'utilisation « Modifier les informations d'un employé » :

<b>Titre</b>	<b>Modifier les informations d'un employé</b>
<b>Acteurs</b>	Directeur
<b>Pré-conditions</b>	L'utilisateur doit avoir le droit d'accès L'utilisateur doit s'authentifier L'utilisateur doit consulter la liste des employés.
<b>Scenarios</b>	Etapes
	L'utilisateur Consulter la liste des employés.
	L'utilisateur choisit un employé à modifier parmi ceux de la liste.
	L'utilisateur effectue les modifications nécessaires puis il valide.
	Le système sauvegarde les nouvelles informations fournies par l'utilisateur.
<b>Post-conditions</b>	Les nouveaux rôles sont appliqués à l'employé.

Tableau 15: Cas d'utilisation « Modifier employé »

### Cas d'utilisation « Supprimer un employé » :

<b>Titre</b>	<b>Supprimer un employé</b>
<b>Acteurs</b>	Directeur
<b>Pre-conditions</b>	L'utilisateur doit avoir le droit d'accès L'utilisateur doit s'authentifier L'utilisateur doit consulter la liste des employés.
<b>Scenarios</b>	Etapes L'utilisateur Consulter la liste des employés. L'utilisateur choisit un employé à supprimer parmi ceux de la liste puis valide son action. Le système effectue une suppression puis une mise à jour de la base de données.
<b>Post-conditions</b>	L'employé est supprimé.

Tableau 16: Cas d'utilisation « Supprimer employé »

### Cas d'utilisation « Créer tâche » :

<b>Titre</b>	<b>Créer une tâche</b>
<b>Acteurs</b>	Directeur, l'employé
<b>Pre-conditions</b>	L'utilisateur doit avoir le droit d'accès. L'utilisateur doit s'authentifier.
<b>Scenarios</b>	Etapes L'utilisateur soumet une demande d'ajouter une nouvelle tâche. Le système envoie le formulaire de création d'une nouvelle tâche. L'utilisateur définit la date début et la date de fin. L'utilisateur choisit le projet concerné et l'employé responsable sur cette tâche. Insérer la description de la tâche ainsi que les informations nécessaires.
<b>Post-conditions</b>	Tâche est créée.

Tableau 17: Cas d'utilisation « Créer Tâche »

### Cas d'utilisation « Ajouter un client » :

<b>Titre</b>	<b>Ajouter un client</b>
<b>Acteurs</b>	Directeur
<b>Pre-conditions</b>	L'utilisateur doit avoir le droit d'accès. L'utilisateur doit s'authentifier.
<b>Scenarios</b>	Etapes L'utilisateur soumet une demande d'ajouter un client Le système envoie le formulaire de création d'un nouveau client. L'utilisateur saisit les informations nécessaires et envoie le formulaire. Le système vérifie les informations et valide l'insertion des données

<b>Post-conditions</b>	Client est créé.
------------------------	------------------

Tableau 18: Cas d'utilisation « Ajouter Client »

Cas d'utilisation « Modifier un client » :

<b>Titre</b>	<b>Modifier un client</b>
<b>Acteurs</b>	Directeur
<b>Pre-conditions</b>	L'utilisateur doit avoir le droit d'accès. L'utilisateur doit s'authentifier.
<b>Scenarios</b>	<p>Etapes</p> <p>L'utilisateur consulte la liste des clients existants dans la base de données</p> <p>Le système envoie le formulaire de création d'un nouveau client.</p> <p>L'utilisateur saisit les informations nécessaires et envoie le formulaire.</p> <p>Le système vérifie les informations et valide l'insertion des données</p>
<b>Post-conditions</b>	Client est modifié

Tableau 19: Cas d'utilisation « Modifier client »

### 1-3 Diagramme de cas d'utilisation :Gestion des projets , taches et utilisateurs

Le diagramme suivant (figure 23) presente les cas d'utilisation des deux modules gestion des projets et gestion des taches, les deux acteurs principaux sont le directeur et le chef de projet. Le chef de projet peut consulter le detail de chaque projet, créer un nouveau projet, le modifier ou supprimer un projet parmi celles déjà enregistrées.

Egalement il peut visualiser le diagramme de gantt de chaque projet. Le chef de projet peut créer des nouvelles taches pour chaque projet existe, il peut affecter des taches aux equipes, enfin il peut les modifier et valider celles enregistrees ce que necessite la generation des rapports. Le directeur peut créer un utilisateur. Egalement il peut modifier ceux-ci, les consulter et les activer ou les désactiver. Toutes ces activites necessitent l'authentification de l'utilisateur.

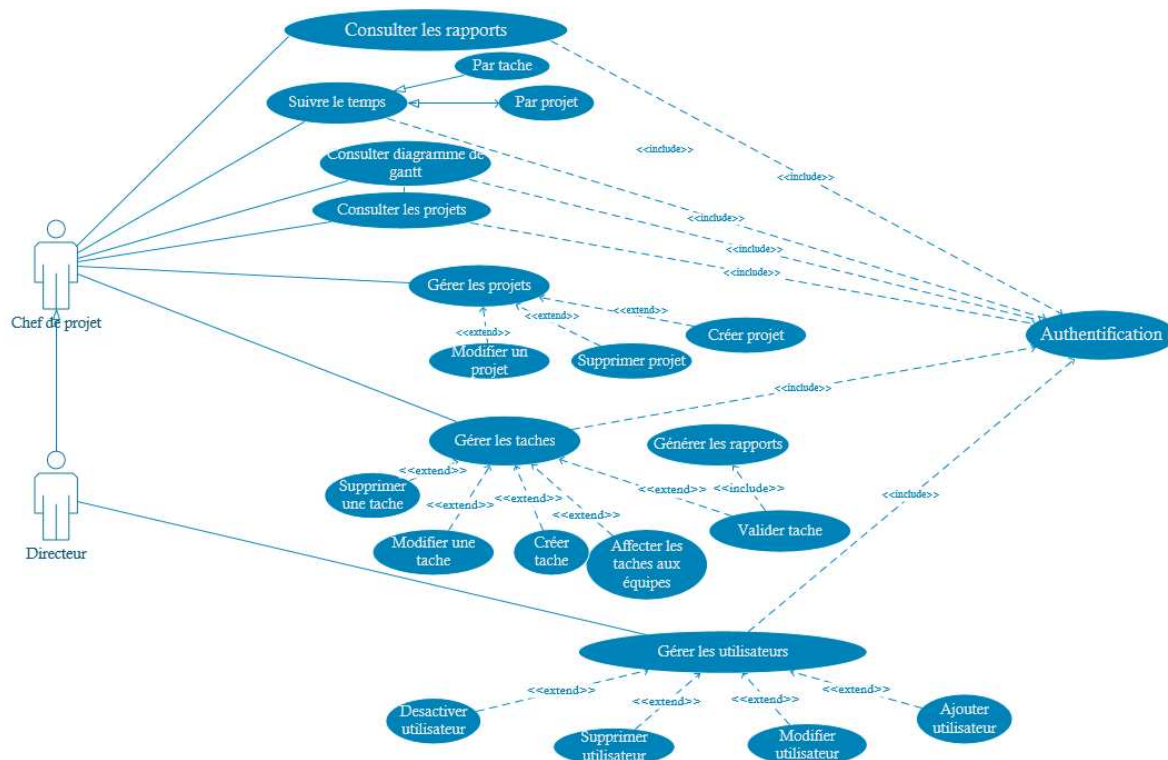


Figure 23: Diagramme cas d'utilisation pour la gestion des projets, taches et utilisateur

#### 1-4 Diagramme de cas d'utilisation : Pour le directeur

Diagramme de cas d'utilisation suivant (figure 24) montre que le directeur peut tout d'abord les clients c'est-à-dire il peut ajouter, modifier ou supprimer un client. Également il a le droit d'effectuer d'appliquer la totalité de ces opérations pour un employé. Pour que le directeur puisse réaliser ses activités, il doit s'authentifier



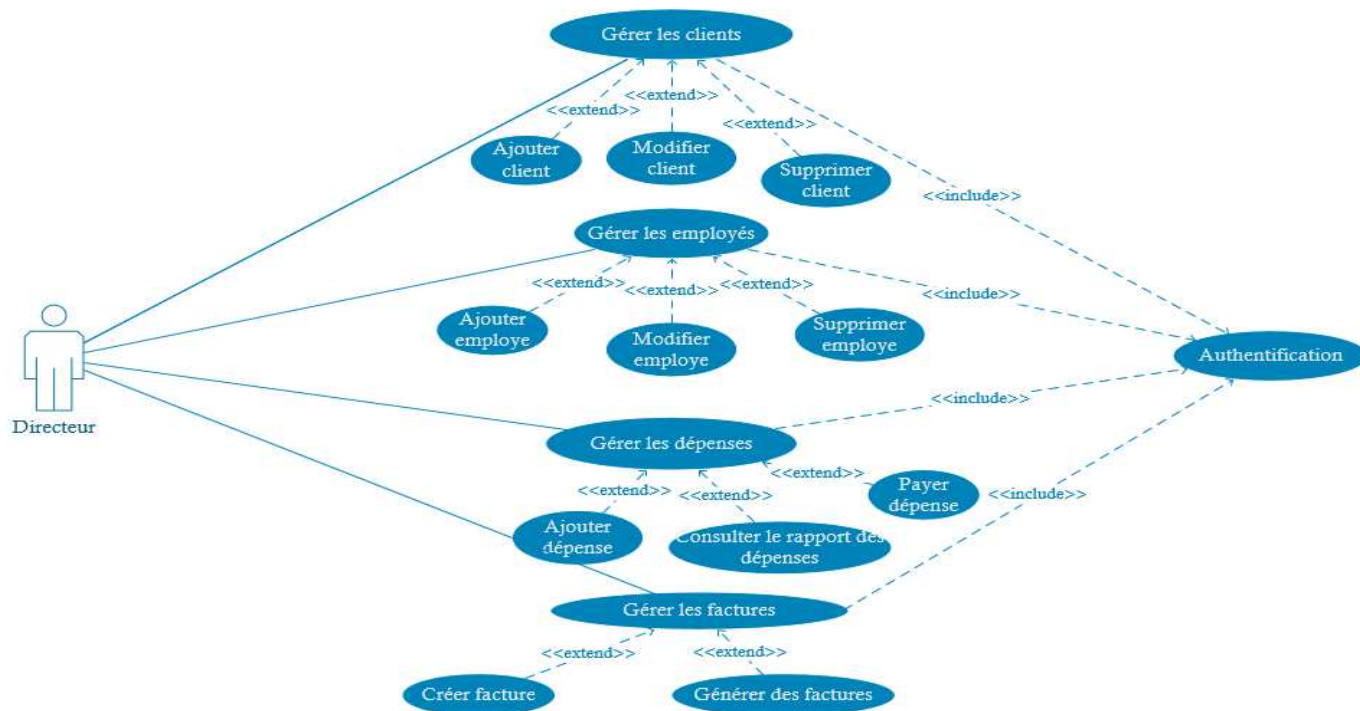


Figure 24: Diagramme cas d'utilisation pour le directeur

## 1-5 Diagramme de cas d'utilisation : pour un employé

Pour ce qui est du diagramme suivant : il présente les cas d'utilisation pour un employé. Il peut gérer ses tâches, modifier, valider une tâche et il consulte l'état d'avancement de chaque tâche. Il peut également consulter ses projets et les modifier. Enfin il a le droit de consulter le tableau de bord.

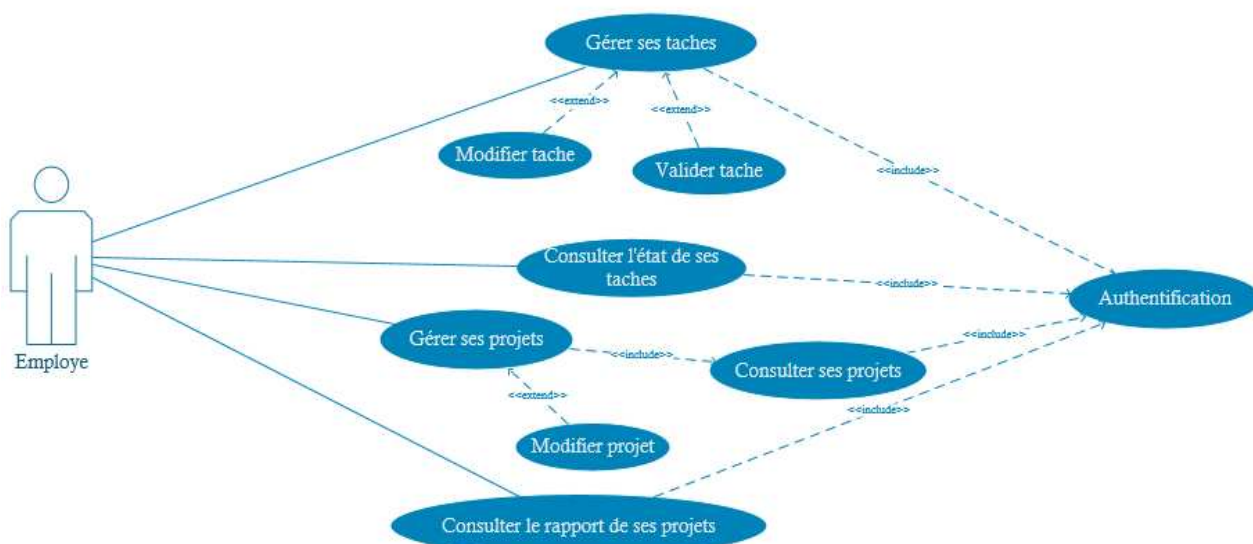


Figure 25: Diagramme de cas d'utilisation pour l'employé



## **2-Diagramme d'Activité**

Dans le langage UML, un diagramme d'activité fournit une vue du comportement d'un système en décrivant la séquence d'actions d'un processus.

Les diagrammes d'activité sont similaires aux organigrammes de traitement de l'information, car ils montrent les flux entre les actions dans une activité. Les diagrammes d'activité peuvent, cependant, aussi montrer les flux parallèles simultanés et les flux de remplacement.

### **i. Diagramme d'activité du module gestion des tâches**

Le diagramme suivant (figure 26) est la représentatif de ce qui se passe au sein du module de gestion des tâches.

Tout d'abord, le directeur s'authentifie si les informations de l'authentification sont correctes alors il a le droit d'accéder à la gestion des tâches. Ainsi il se trouve face à deux opérations :

On commence par la création d'une nouvelle tâche, il remplit le formulaire par les informations nécessaires, le système valide le formulaire si les données entrées sont correctes alors la tâche est créée.

Ensuite le directeur consulte la liste des tâches, il a le choix entre quatre opérations :

- Supprimer une tâche : le directeur choisit une tâche parmi ceux que le système retourne et il valide la suppression ainsi le système procède à une mise à jour de la base de données.
- Le directeur peut modifier les informations d'une tâche, il choisit la tâche concernée et effectue les modifications nécessaires puis valide l'opération.
- Ensuite l'utilisateur peut valider une tâche, il choisit la tâche qui à l'état terminée il passe la tâche à l'état validée, ainsi le système modifie la tâche concernée.
- L'utilisateur peut affecter la tâche à une équipe : il choisit la tâche il demande l'affectation de celle-ci puis le système retourne la liste des équipes. Ici l'utilisateur assigne la tâche à l'équipe et il valide l'opération.

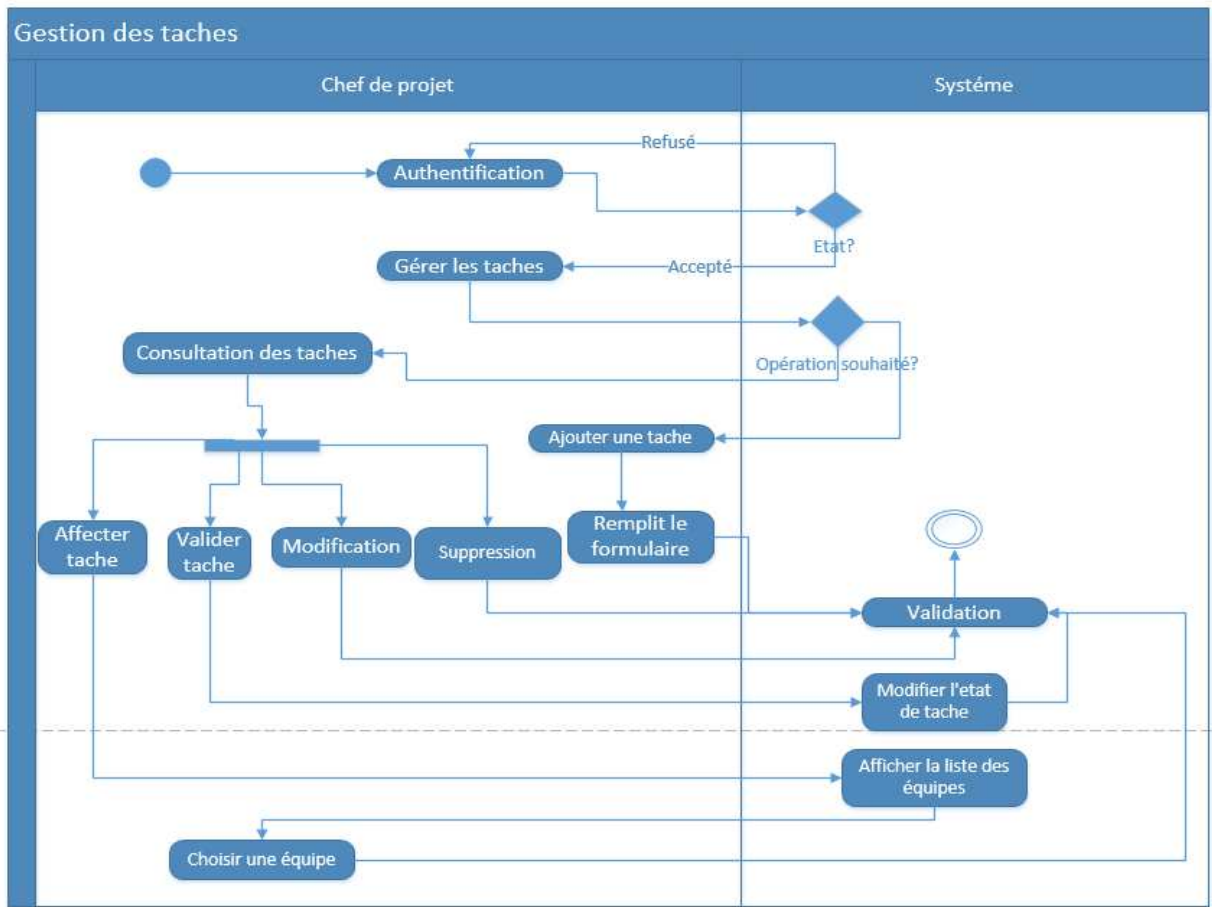


Figure 26: Diagramme d'activité

### 3- Diagramme de séquence

Les diagrammes de séquence présentent la coopération entre différents objets. Les objets sont définis et leur coopération est représentée par une séquence de messages entre eux.

Le diagramme de séquence permet de cacher les interactions d'objets dans le cadre d'un scénario d'un diagramme des cas d'utilisation. Dans un souci de simplification. Dont le but est de décrire comment se déroulent les actions entre les acteurs ou objets.

Après la description des cas d'utilisation, nous allons élaborer le modèle dynamique dans lequel nous allons décrire les scénarios de quelques cas d'utilisation, sous forme de diagrammes de séquence.

### i. Diagramme de séquence du scénario : « Authentification »

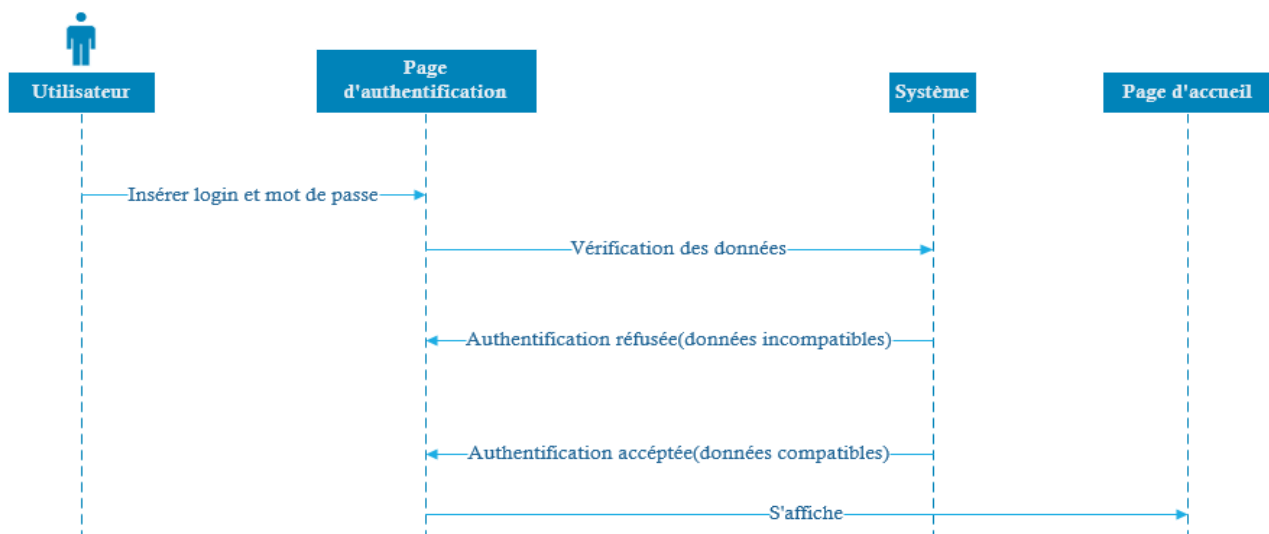


Figure 27: Diagramme de séquence du scénario : « Authentification »

#### Détails du scénario :

Tout d’abord l’utilisateur insère son login et mot de passe, puis le système procède à la vérification de compatibilité des données saisies avec ceux existant dans la base de données. Si les données entrées par l’utilisateur sont correctes alors il se redirige vers la page d’accueil, sinon l’utilisateur reçoit un message d’erreur.

### ii. Diagramme de séquence du scénario : « Consulter diagramme de Gantt »

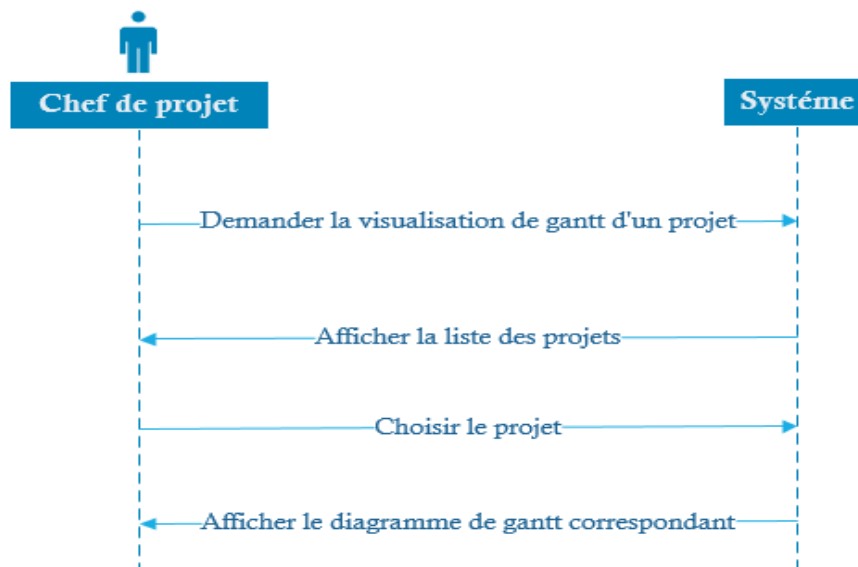


Figure 28: Diagramme de séquence : « Consulter diagramme de Gantt »

#### Détails du scénario :

L’utilisateur effectue une demande de consulter le diagramme de gantt d’un projet, le système affiche la liste des projets existants dans la base de données.

L'utilisateur choisit un projet voulu afin de visualiser son diagramme de gantt , ainsi le système procède à l'affichage du diagramme correspondant.

### iii. Diagramme de séquence du scénario : « Gestion des utilisateurs »

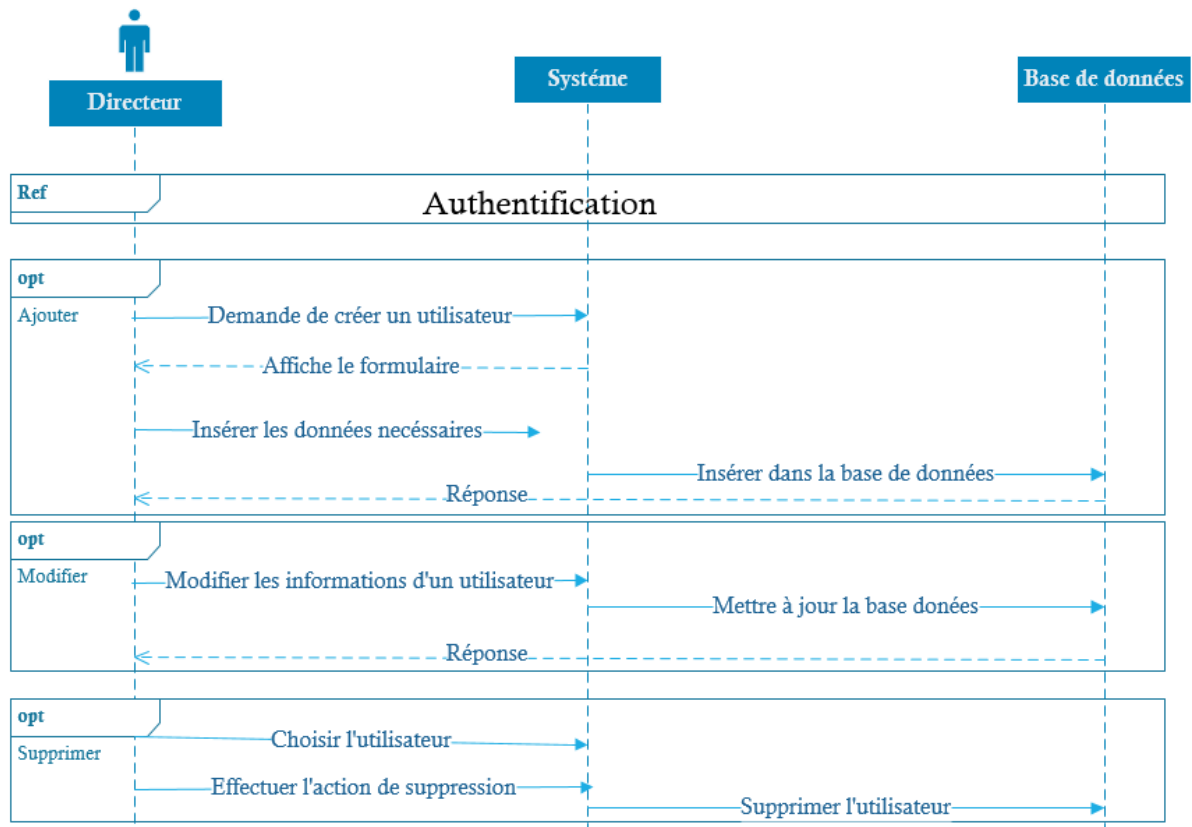


Figure 29: Diagramme de séquence pour la gestion des utilisateurs

#### Détails du scénario :

Le directeur doit tout d'abors s'authentifier afin d'accéder au système, puis l'utilisateur se trouve face entre trois choix. Le directeur peut ajouter un utilisateur, ainsi ildemande de créer un nouvel utilisateur puis le système affiche le formulaire pour l'utilisateur a ajouter.

Le directeur saisit les informations necessaires et valide la creation ainsi le système insere l'utilisateur dans la base de donnees.

#### iv. Diagramme de séquence du scénario : « Gestion des projets »

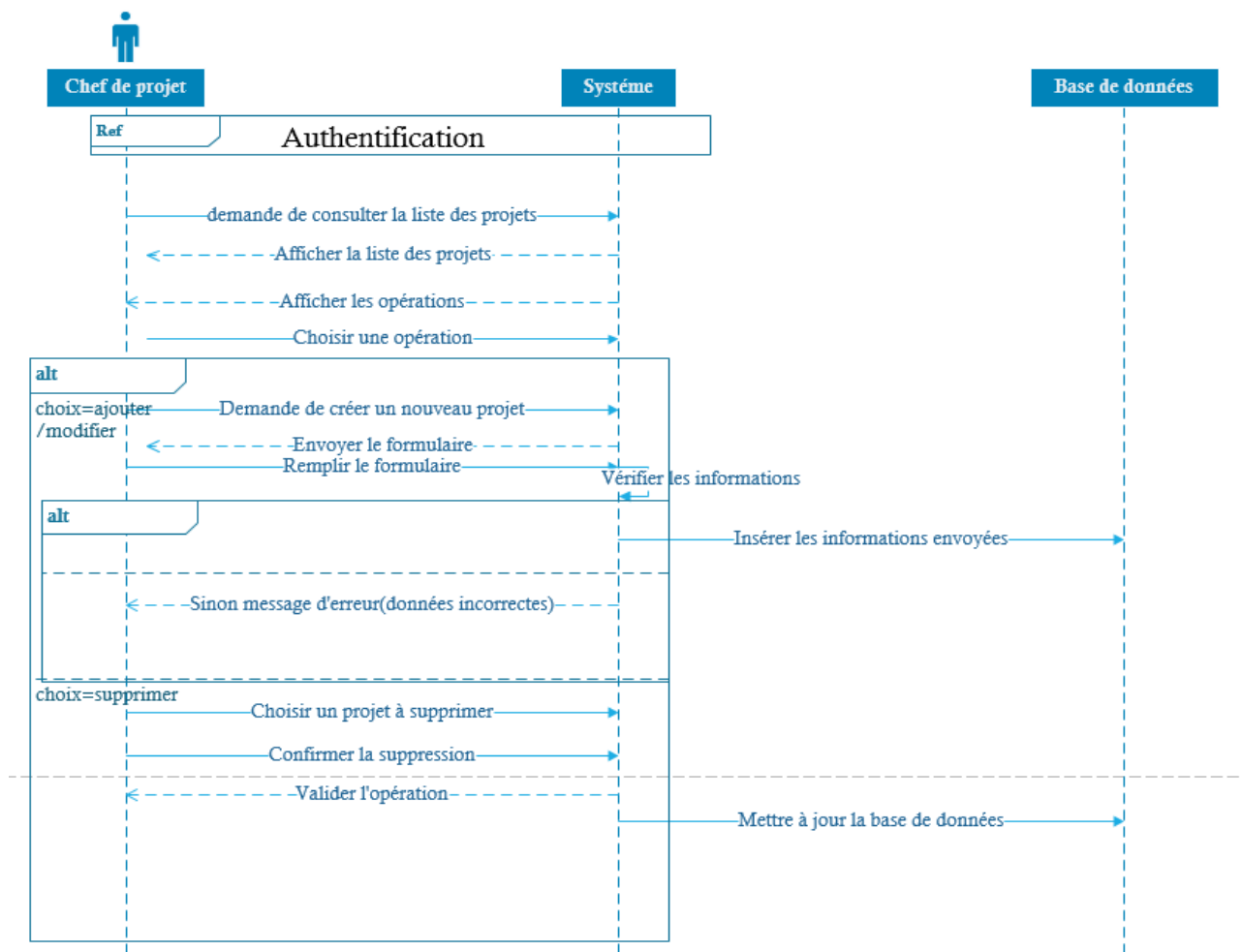


Figure 30: Diagramme de séquence pour la gestion des projets

##### Détails de scénario :

L'utilisateur doit premièrement s'authentifier, puis il effectue une demande de consulter la liste des projets. Le système affiche la liste concernée ainsi que la liste des opérations (ajouter, modifier et supprimer).

##### ■ Le premier choix : Ajouter ou modifier un projet.

L'utilisateur demande de créer/modifier un projet, le système affiche le formulaire, après que l'utilisateur remplit et soumet le formulaire des données.

Ainsi le système procède à la validation des données si les informations nécessaires sont correctes alors les données sont insérées dans le serveur BD sinon un message d'erreur s'affiche.

##### ■ Le deuxième choix : Supprimer un projet.

L'utilisateur choisit les projets à supprimées et il effectue l'opération de suppression. Ensuite il reçoit un message de confirmation, il valide l'opération ensuite la modification se fait dans la base de données.

## 4- Diagramme de classe

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation. Le diagramme de classes en montre la structure interne. Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d'utilisation.

A partir de l'analyse lors de la partie analyse du projet et des diagrammes de séquence présente, nous avons extrait un ensemble d'entités et de dépendances, cela a été traduit par UML en un diagramme de classes (figure 31) qui sera ensuite décrit par un tableau détaillant le rôle de chaque entité.

Classe	Description
<b>Employé</b>	Classe contenant les informations sur l'employé.
<b>Projet</b>	La classe qui contient les informations nécessaires d'un projet notamment la date début et la date de fin.
<b>Tâche</b>	C'est une classe qui modélise la tâche d'un projet, elle contient toutes les attributs d'une tâche à titre d'exemple le statut et la date début et la date de fin.
<b>Utilisateur</b>	Classe contenant les informations sur l'utilisateur, elle contient notamment les attributs suivant : l'adresse email et si l'utilisateur est actif.
<b>Rôle</b>	La classe qui représente le rôle d'un utilisateur.
<b>Client</b>	C'est une classe contenant les informations d'un client notamment les attributs : nom de client et l'adresse email.
<b>Dépense</b>	Classe qui permet de déterminer les dépenses d'une tâche.
<b>Domaine</b>	Représente le domaine d'un projet précis.
<b>Equipe</b>	Classe qui permet de déterminer l'équipe de l'ensemble des employés.
<b>Etat</b>	La classe qui modélise l'état d'une tâche précis en temps réel.
<b>Compétence</b>	La classe qui contient les informations sur les compétences d'un employé.
<b>Facture</b>	Classe contenant les informations sur les factures des projets.
<b>Priorité</b>	La classe qui modélise la priorité d'une tâche.

Tableau 20 : Dictionnaire de données

### i. Les règles de gestion :

- Chaque client peut être le propriétaire de plusieurs projets.
- Un projet est d'un seul client
- Un projet concerne un seul domaine.
- Un domaine peut être concerné par plusieurs projets
- Un projet se divise en plusieurs tâches
- Chaque projet a une facture
- Une facture concerne un seul projet
- Chaque tâche concerne un seul projet
- Chaque tâche peut avoir une seule priorité et état

- Un état et une priorité peut être concernés par plusieurs tâches
- Chaque tâche a une dépense
- Chaque dépense s'inscrit dans une catégorie
- Chaque dépense concerne une seule tâche
- Chaque catégorie peut être affectée à plusieurs tâches
- Un employé peut travailler sur plusieurs tâches
- Une tâche est affectée à un seul employé
- Un employé peut travailler dans plusieurs équipes
- Un équipe contient plusieurs employés
- Chaque employé peut avoir plusieurs compétences
- Un employé est un utilisateur, un employé peut être un directeur, un chef de projet ou un chef d'équipe.
- Chaque Utilisateur a plusieurs rôles.
- Un rôle peut être affecté à plusieurs utilisateurs

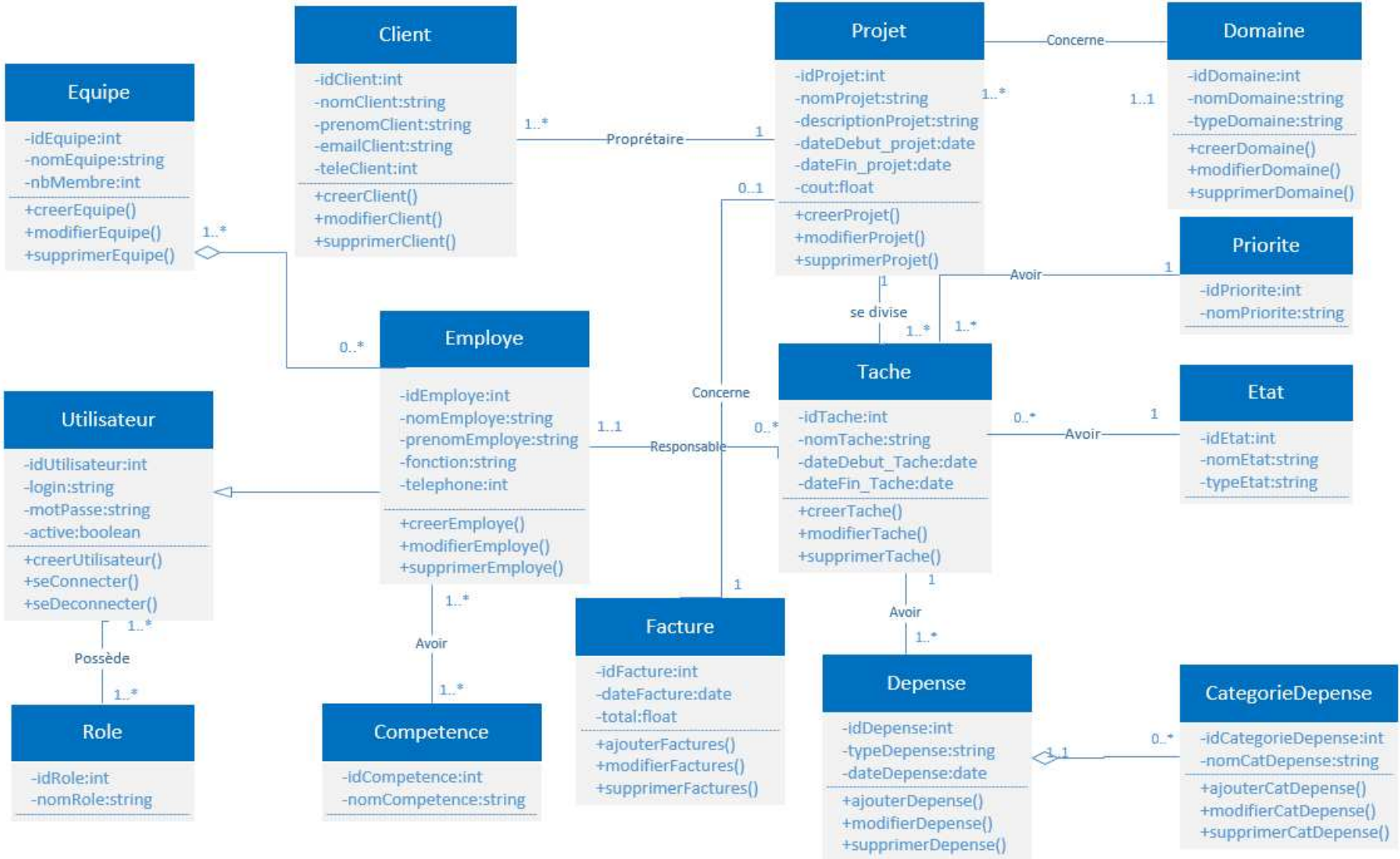


Figure 31: Diagramme de classe de notre système



## 5- Diagramme de déploiement

Dans UML, les diagrammes de déploiement modélisent l'architecture physique d'un système. Les diagrammes de déploiement affichent les relations entre les composants logiciels et matériels du système, d'une part, et la distribution physique du traitement, d'autre part.

Les diagrammes de déploiement, que vous préparez généralement pendant la phase d'implémentation du développement, présentent la disposition physique nœud et les composants et autres éléments que les artefacts implémentent. Les nœuds représentent des périphériques matériels tels que des ordinateurs, des détecteurs et des imprimantes, ainsi que d'autres périphériques qui prennent en charge l'environnement d'exécution d'un système. Les chemins de communication et les relations de déploiement modélisent les connexions dans le système.

Le diagramme de déploiement suivant met en évidence les différents niveaux de l'application.

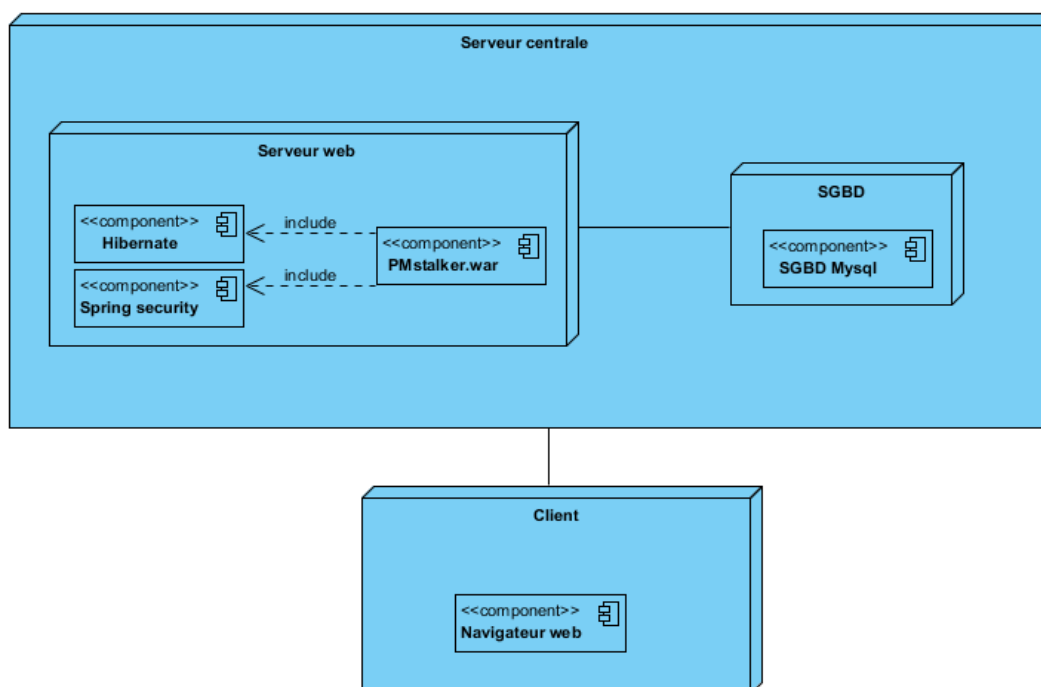


Figure 32: Diagramme de déploiement

## Conclusion

Ce chapitre a permis de donner une image claire et nette du système. Nous avons spécifié les cas d'utilisation du système, ses diagrammes d'activité et de séquence, un diagramme de classe et un diagramme de déploiement spécifiant l'infrastructure physique du système.

# Chapitre 5

---

## Mise en œuvre du projet

Dans ce chapitre, nous allons présenter le déroulement des tests que nous avons effectué tout au long du développement du projet, l'architecture du système, ensuite nous passerons à la description, accompagnées des interfaces homme-machine de l'application.

## **Introduction**

Dans ce chapitre nous allons présenter la phase de réalisation du projet, ses interfaces graphiques, et les tests d'intégration globaux de la solution PMSTalker.

### **1- Déploiement de PMSTalker**

Le déploiement appelé aussi « livraison » ou « mise en exploitation ». Il regroupe toutes les activités qui mènent à l'installation et mise en marche de l'application développée.

Pour déployer l'application dans le serveur local de la société nous avons suivi l'ensemble des étapes suivantes :

- Configuration réseau du serveur dans lequel PMSTalker sera déployé.
- Installation du conteneur de servlets « vFabric tc Server ».
- Création des services de démarrage de « vFabric tc Server » et du SGBD MySQL.
- Importation de la base de données de l'application.
- Remplissage de la base de données par les données de base de la société.
- Déploiement d'application.
- Tests d'intégration qui seront détaillés dans la section suivante.

### **2- Les Tests d'intégration**

En informatique, un test désigne une procédure de vérification partielle d'un système. Son objectif principal est d'identifier un nombre maximum de comportements problématiques du logiciel afin d'en augmenter la qualité (si les problèmes identifiés lors des tests sont corrigés). Néanmoins, le test peut aussi avoir pour objectif d'apporter des informations quant à cette qualité afin de permettre la prise en décisions.

Les tests de vérification ou de validation visent ainsi à vérifier que ce système réagit de la façon prévue par ses développeurs (spécifications) ou est conforme aux besoins du client, respectivement.

Le but de cette activité est de tendre vers un produit « zéro défaut »

#### **2-1 Tests unitaires**

Le test unitaire est une procédure permettant de vérifier le bon fonctionnement d'une partie précise d'un logiciel ou d'une portion d'un programme. Et voici dans la figure suivante (figure 33) le processus des tests unitaires

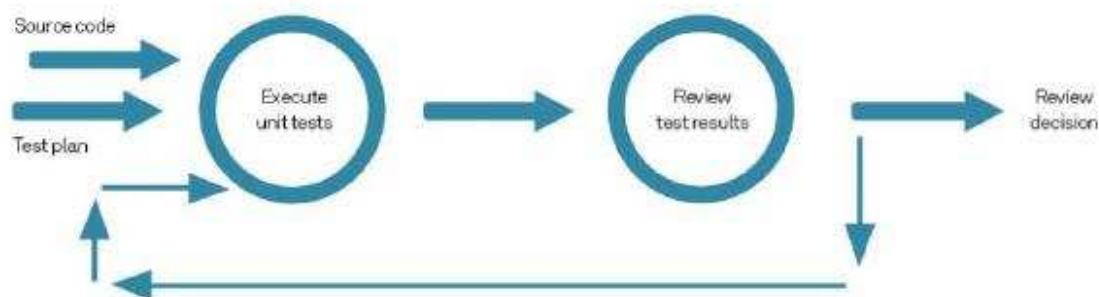


Figure 33: Processus des Test unitaires

Après avoir clôturé l'étape de la conception détaillée dans le cycle de développement Scrum, nous avons rédigé un plan de test. En se basant sur ce dernier et sur le framework TestNG, nous avons pu tester chaque fonctionnalité du système à part.

Pour tester chaque module ou fonctionnalité de l'application il faut implémenter le test unitaire approprié, et chaque test comporte toutes les méthodes testant chaque fonction du module objet de test.

L'environnement de développement « Spring STS » facilite l'implémentation de ces tests en intégrant le framework TestNG. Il suffit de cliquer avec le bouton droit sur la classe à tester puis choisir « Create TestNG » Afin de sélectionner les fonctions à tester, dans ce cas TestNG va créer une classe de test Java contenant l'ensemble des fonctions test de la classe à tester comme on peut remarquer dans la figure suivante :

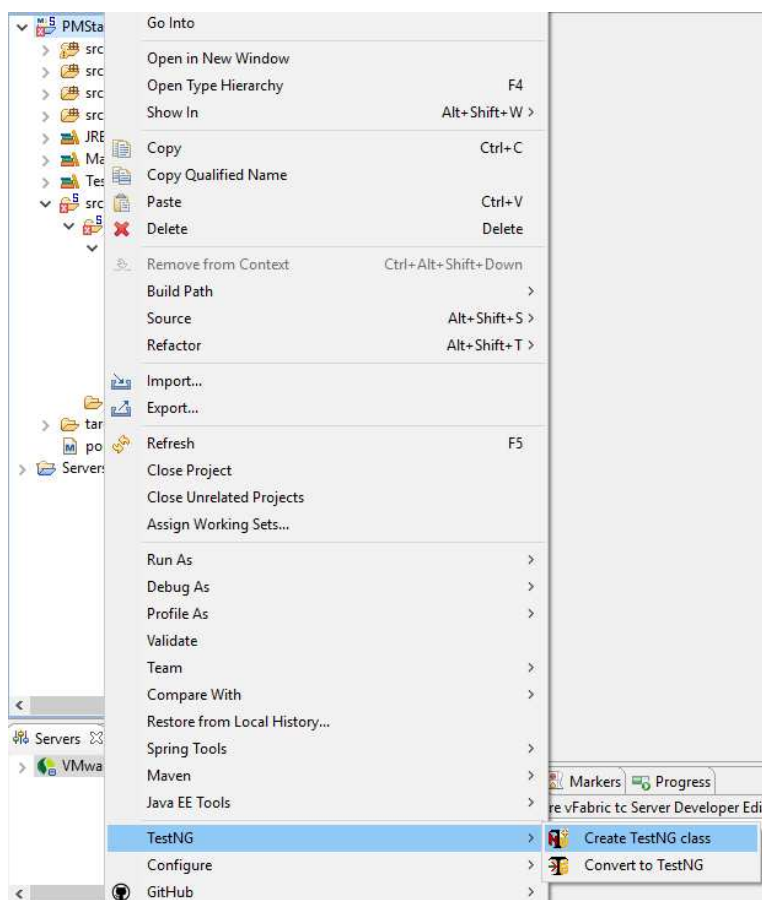


Figure 34: Création d'un test unitaire

Nous avons réalisé ensemble des tests sur différents classes de la couche métier :

On a réalisé la classe ClientMetierImplTest qui contient le test des méthodes pour gérer les clients.

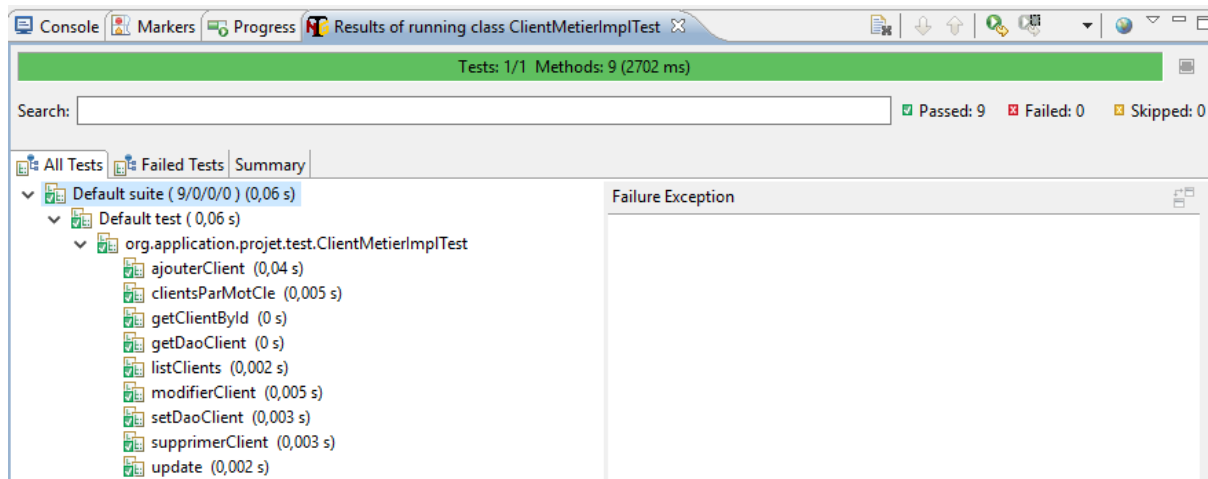


Figure 35: Test MetierClient

La classe DepenseMetierImplTest est la classe de test pour les méthodes qui définissent les dépenses.

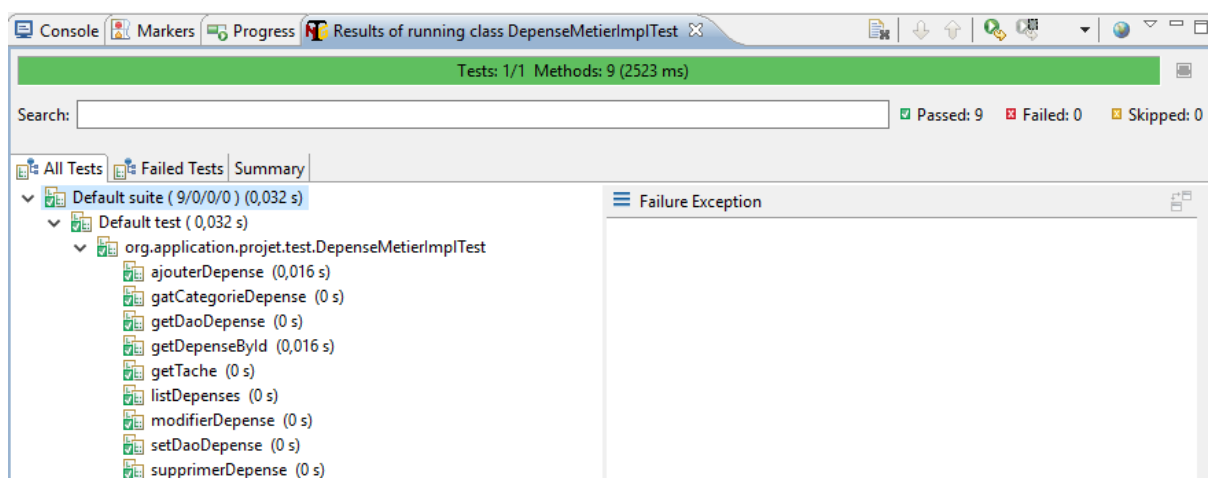


Figure 36: Test MetierDepense

La classe EmployeMetierImplTest c'est la classe de test des méthodes pour gérer les employés (figure 37).

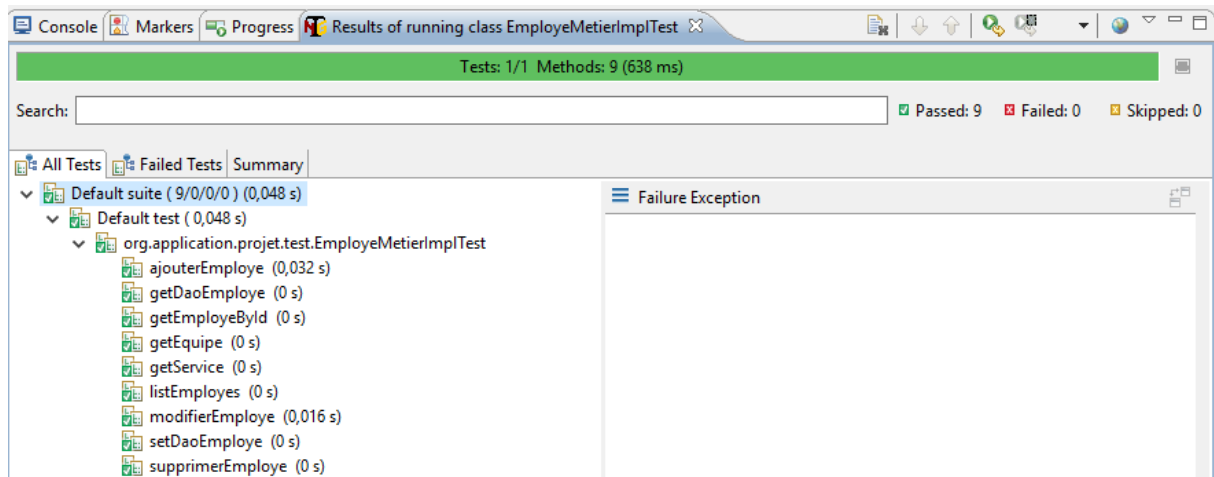


Figure 37: Test MetierEmployee

## 2-2 Architecture du système

Comme le design pattern choisi est MVC, L'application est découpée en 3 couches distinctes, présentation, Métier et DAO.

### ▪ Couche présentation :

Cette couche correspond à la partie de l'application visible et interactive avec les utilisateurs. Ici c'est l'interface Homme-Machine, il s'agit d'une interface web elle intercepte les évènements et fait appel aux services de la couche View.

### ▪ Couche Service :

C'est une couche intermédiaire entre les couches View et la couche DAO (Data Access Object), son rôle est de gérer les transactions qui seront ouvertes à chaque appel de la base de données.

Cette couche est mise en œuvre à l'aide du framework Spring MVC (module de gestion des transactions).

### ▪ Couche d'accès aux données :

C'est la couche qui regroupe les classes Java capable de fournir les données nécessaires à la couche métier, cette couche est modélisée dans notre système par les composants DAO. Pour l'implémentation nous avons utilisé les frameworks JPA et Hibernate qui permettent de mapper les données.

## 3- Les interfaces graphiques

Dans cette section, on va voir les fonctionnalités de PMStalker, en donnant une description de chaque interface.

La première page qui apparaît à l'utilisateur est la page d'authentification, l'utilisateur doit remplir le login et le mot de passe pour accéder à l'application et profiter de différentes fonctionnalités de PMStalker selon son rôle.

The login page has a white background with a dark teal header bar at the top. The text 'Bienvenue à PMStalker' is centered in a dark teal font. Below it are two input fields: 'Nom d'utilisateur' and 'Mot de passe', both with light gray placeholder text. A light gray button with the text 'S'identifier' is centered below the fields. A dark teal footer bar is at the bottom.

Figure 38: Page d'authentification

### 3-1 Partie administrateur

Une fois l'administrateur saisit le login et le mot de passe, il peut accéder directement à l'espace d'administration, cette partie permet la gestion complète du système à savoir la gestion des utilisateurs, et la gestion des clients, la gestion des employés, la gestion des projets, la gestion des tâches, la gestion des dépenses, la gestion des factures.

- Rapport :

Après l'authentification l'utilisateur est rediriger directement vers la page de rapport de gestion des projets où on peut trouver tous les graphes et les statistiques à propos l'avancement des projets que l'entreprise exécute.

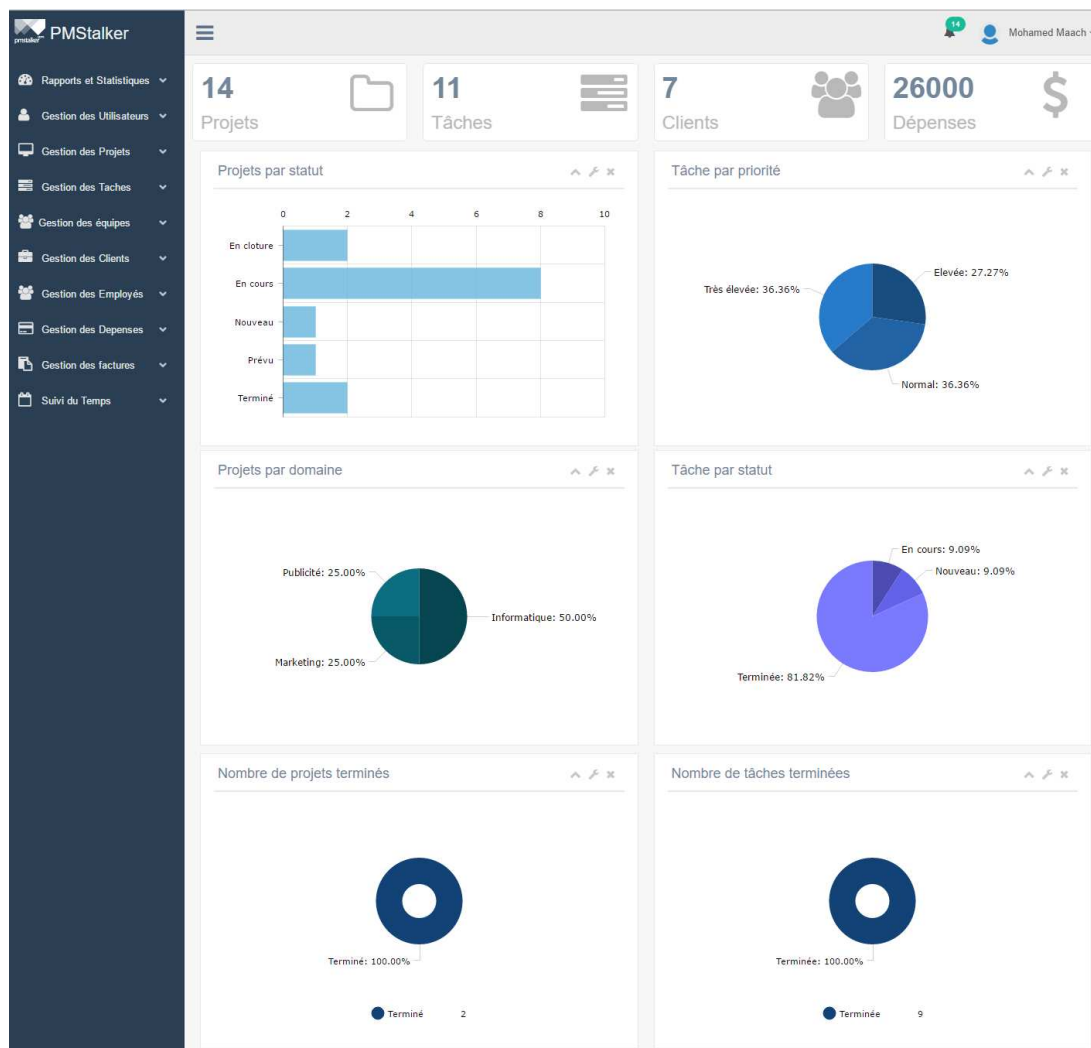


Figure 39: Page d'accueil

### ■ Gestion des clients :

L'interface relative aux clients se présente comme suit (figure 40 ) où on trouve une liste des clients de l'entreprise avec qui elle travaille ses projets :



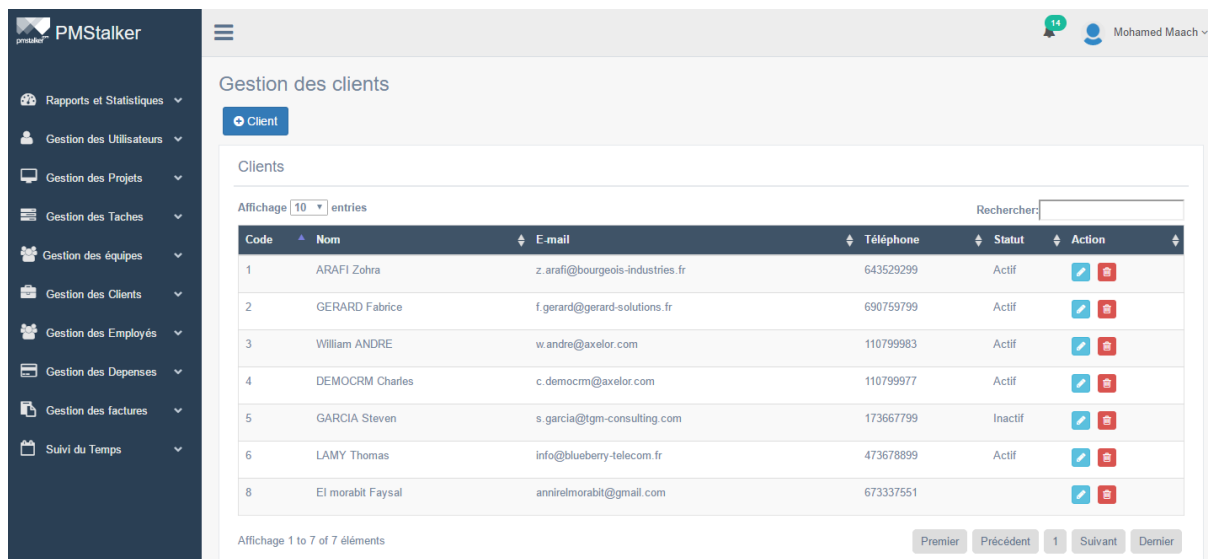


Figure 40: Interface gestion des clients

La figure montre la liste des clients existant dans la base de données, ici l'administrateur peut modifier ou supprimer un client, aussi il peut ajouter un nouveau client. Passons maintenant à la création d'un nouveau client. L'utilisateur remplit le formulaire suivant :

**Information Général**

Nom:

Prénom:

Statut:

Photo:

**Coordonnées**

Email:

Téléphone:

Adresse:

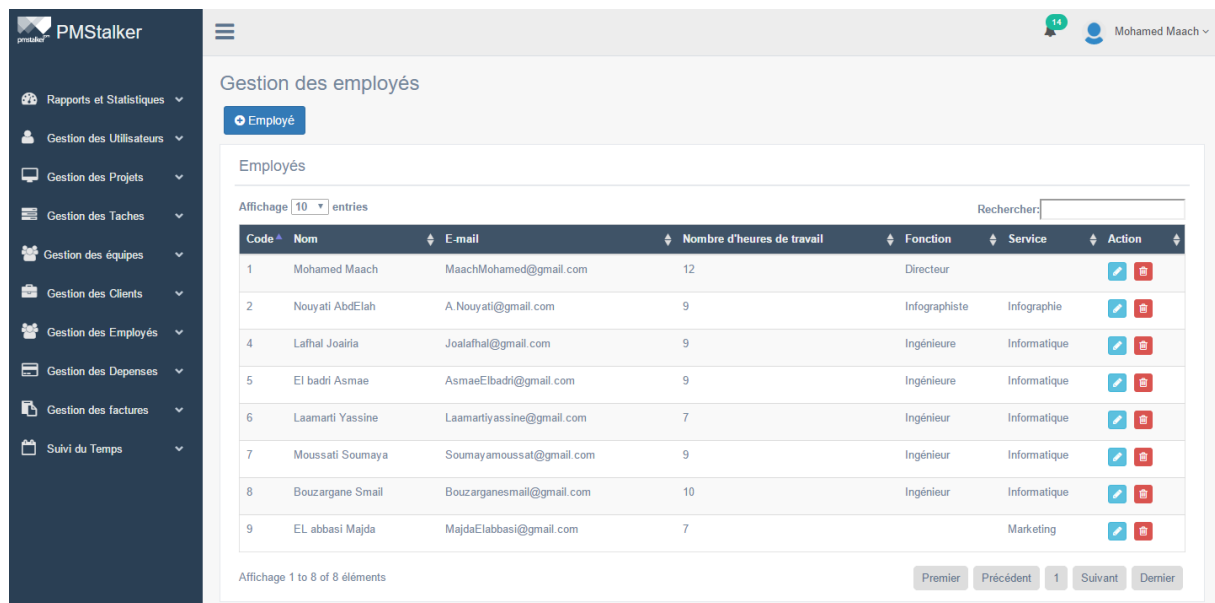
Pays:  Ville:  Code postal:

Figure 41: Ajouter client

Ici il insère les informations nécessaires par exemple l'adresse email de client et son numéro de téléphone et il doit renseigner le statut du client s'il est actif ou inactif.

### ■ Gestion des employés :

La vue générale de la gestion des employés est représentée dans la figure qui suit ( figure 42): L'interface regroupe les employés travaillant chez l'entreprise, l'utilisateur peut effectuer les opérations usuelles (modification, suppression et l'insertion).



**PMStalker**

**Gestion des employés**

**Employé**

Affichage 10 entries

Rechercher:

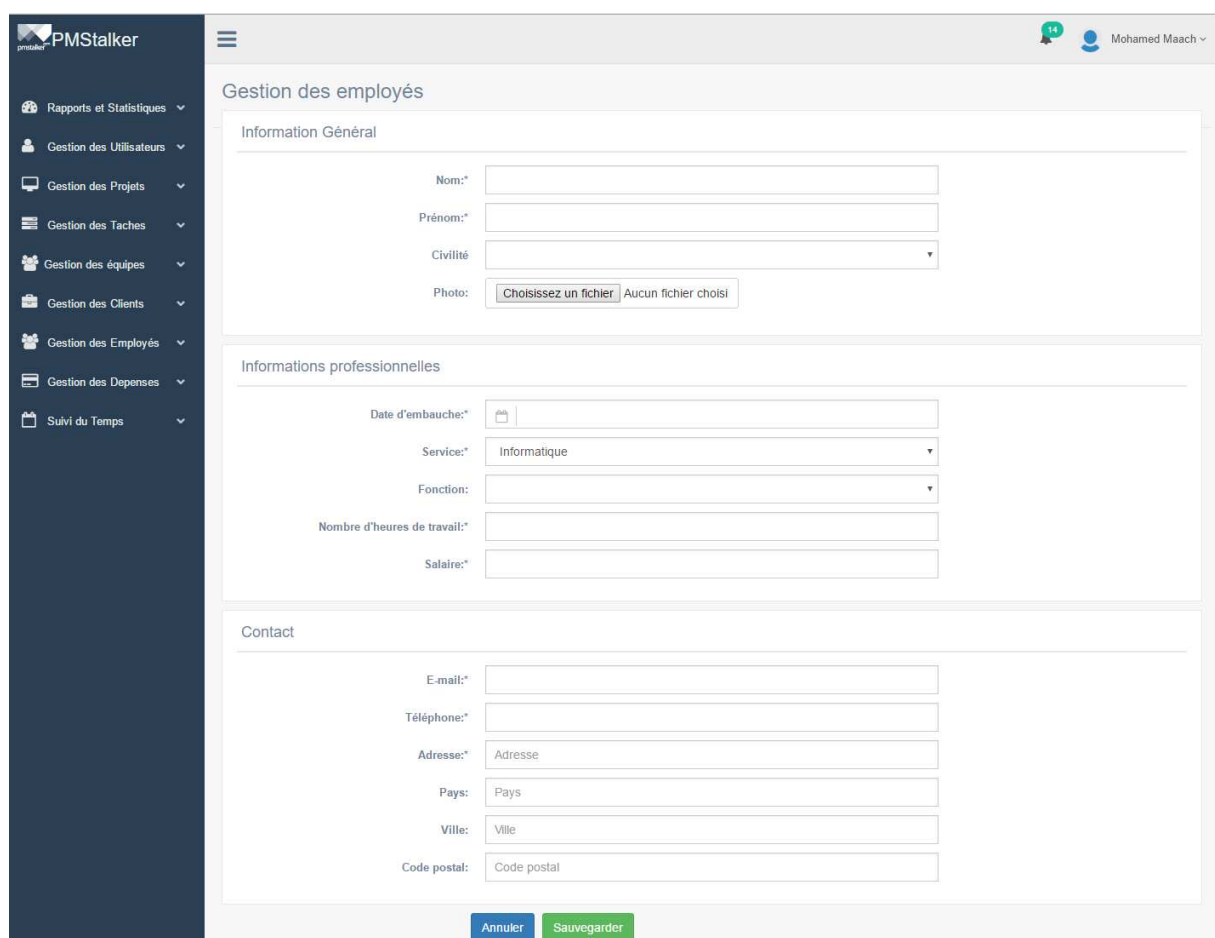
Code	Nom	E-mail	Nombre d'heures de travail	Fonction	Service	Action
1	Mohamed Maach	MaachMohamed@gmail.com	12	Directeur		
2	Nouyati Abdelah	A.Nouyati@gmail.com	9	Infographiste	Infographie	
4	Lafhal Joalria	Joalafhal@gmail.com	9	Ingénieure	Informatique	
5	El badri Asmae	AsmaeElbadri@gmail.com	9	Ingénieure	Informatique	
6	Laamarti Yassine	LaamartiYassine@gmail.com	7	Ingénieur	Informatique	
7	Moussati Soumaya	Soumayamoussati@gmail.com	9	Ingénieur	Informatique	
8	Bouzargane Smail	Bouzarganesmail@gmail.com	10	Ingénieur	Informatique	
9	EL abbasi Majda	MajdaElabbasi@gmail.com	7		Marketing	

Affichage 1 to 8 of 8 éléments

Premier Précédent 1 Suivant Dernier

Figure 42: Interface gestion des employés

L'utilisateur peut créer un nouvel employé via l'interface suivante en entrant les informations à propos l'employé dans chaque case du formulaire :



**PMStalker**

**Gestion des employés**

**Information Général**

Nom:\*

Prénom:\*

Civilité

Photo:  Aucun fichier choisi

**Informations professionnelles**

Date d'embauche:\*

Service:\*

Fonction:\*

Nombre d'heures de travail:\*

Salaire:\*

**Contact**

E-mail:\*

Téléphone:\*

Adresse:\*

Pays:\*

Ville:\*

Code postal:\*

Figure 43: Ajouter employé

L'interface contenant les informations générales d'un employé, ainsi que les informations professionnelles par exemple le salaire et la date d'embauche puis il valide l'opération via le bouton sauvegarder.

La vue suivante donne à l'administrateur une vision statistique sur les employés de l'entreprise, les figures représentent les nombres des employés pour chaque service ainsi la totalité des salariées.

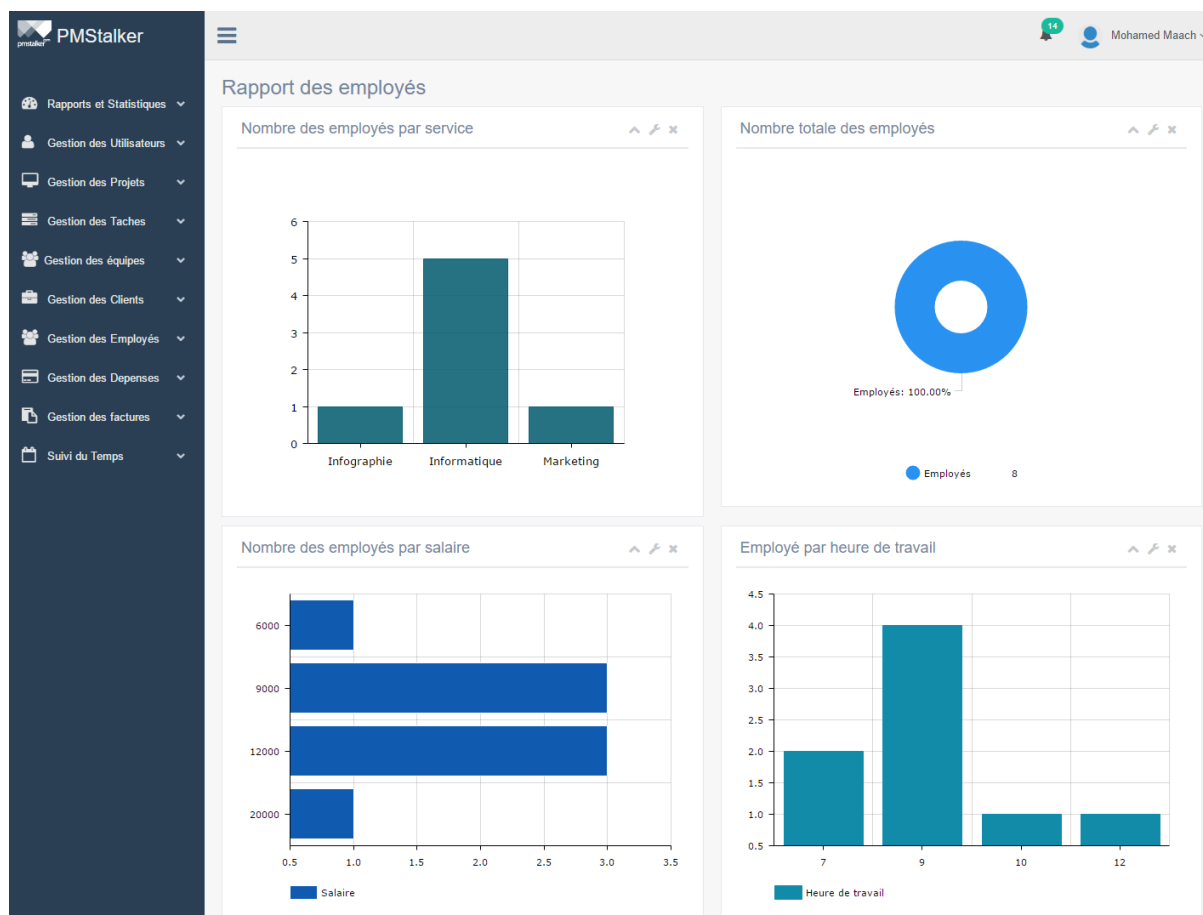


Figure 44: Rapport des employés

### ■ Gestion des utilisateurs :

Pour la gestion des utilisateurs, l'administrateur peut effectuer les différentes options de gestion (modifier, ajouter, supprimer) comme ça se montre dans la figure qui suit (figure 45).

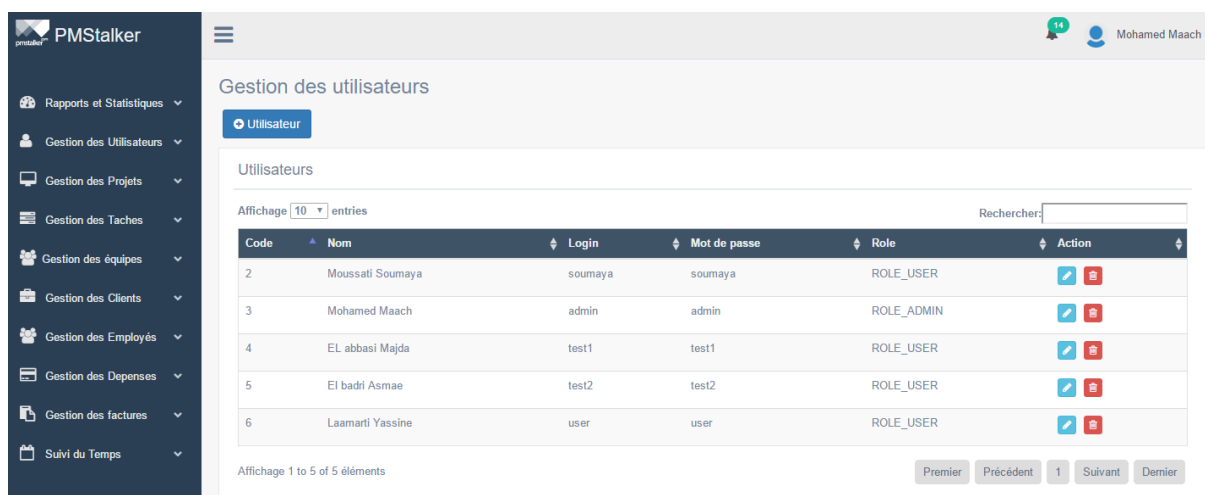


Figure 45: Interface gestion des utilisateurs

L'administrateur du système peut créer un utilisateur via l'interface suivant, il remplit le formulaire par les informations nécessaires login et le mot de passe puis il affecte un rôle à l'utilisateur créé.

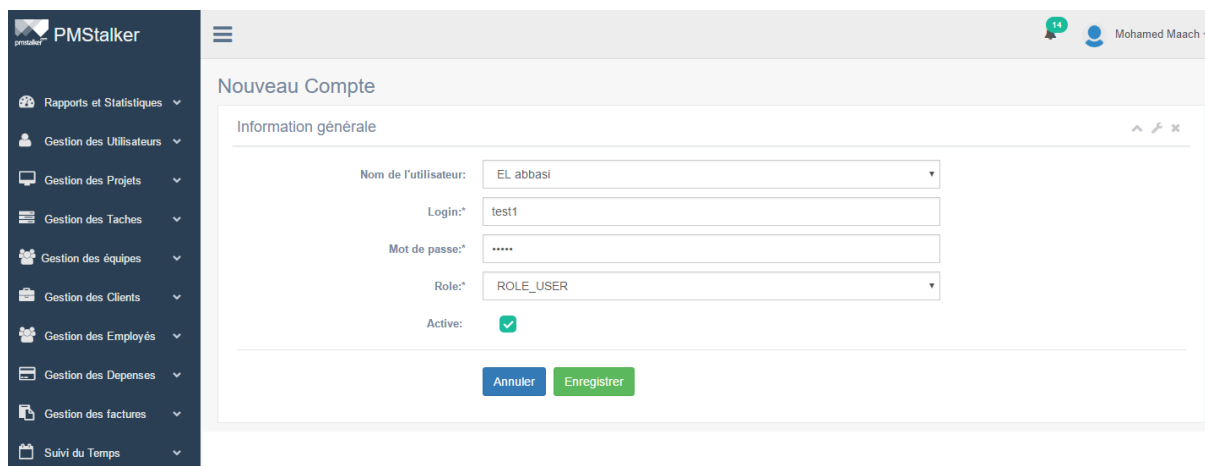


Figure 46: Ajouter utilisateur

### ■ Gestion des équipes :

La vue suivante (figure 47) permet à l'administrateur de consulter la liste des équipes, et il peut créer des nouvelles équipes, ainsi modifier ou supprimer ceux existents. Egalement l'administrateur peut affecter des membres à une équipe.

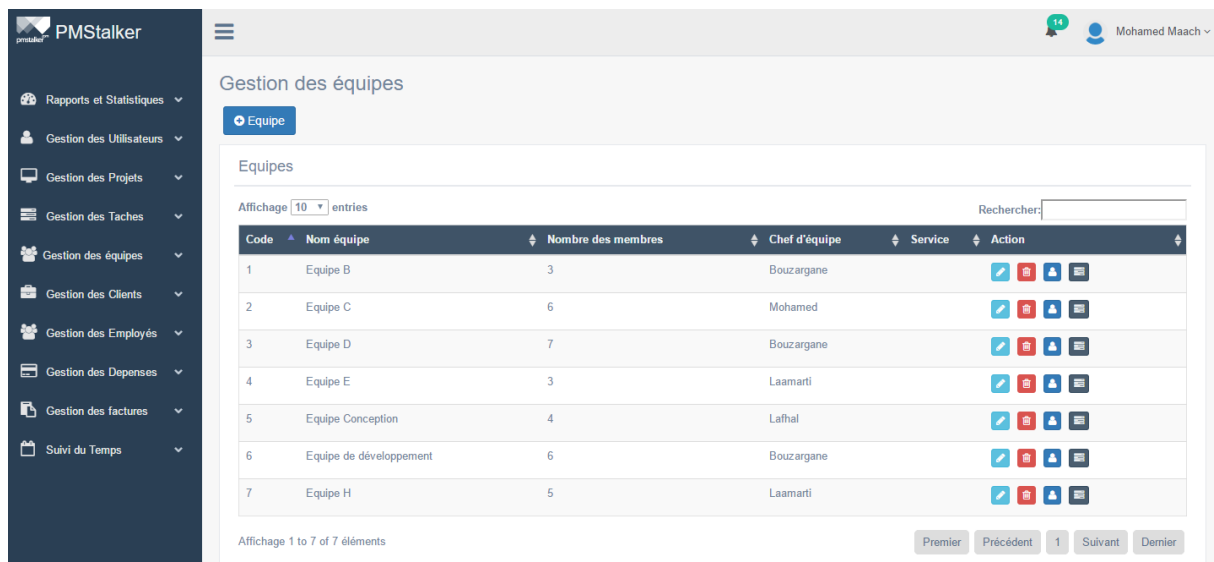


Figure 47: Interface gestion des équipes

L'administrateur remplit le formulaire suivant pour créer une équipe, il doit obligatoire renseigner un chef pour cette équipe.

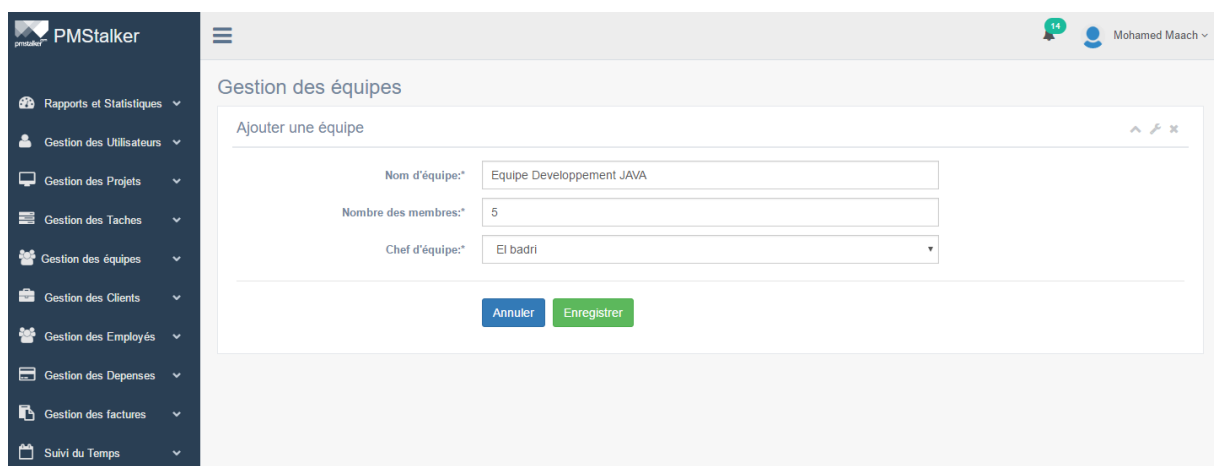


Figure 48: Ajouter équipe

L'affectation des employés aux équipes est très importante. Tout d'abord l'administrateur consulter la liste des équipes et choisit l'opération ajouter membre ainsi il se redirige vers l'interface suivante (figure 49) il choisit les employés concernés et il valide l'affectation via le bouton affecter.

**Gestion des équipes**

Ajouter des membres à l'équipe

Affichage 10 entries Rechercher:

Numéro	Nom	Fonction	Service
1	Mohamed Maach	Directeur	
2	Nouyati AbdElah	Infographiste	Infographie
4	Lafhal Joairia	Ingénieure	Informatique
5	El badri Asmae	Ingénieure	Informatique
6	Laamarti Yassine	Ingénieur	Informatique
7	Moussati Soumaya	Ingénieur	Informatique
8	Bouzargane Smail	Ingénieur	Informatique
9	EL abbasi Majda		Marketing

Affichage 1 to 8 of 8 éléments

Premier Précédent 1 Suivant Dernier

Affecter

Figure 49: Affectation des membres aux équipes

### ■ Gestion des projets :

Le module de gestion des projets parmi les modules les plus intéressants de notre système, l'interface suivante permet à l'administrateur de consulter la liste des projets de l'entreprise. On remarque le champ progression qui représente la progression de projet en temps réel.

**Gestion des projets**

Projet

-- Statut -- Filtre

Liste des projets

Affichage 10 entries Rechercher:

Numéro	Nom projet	Date début	Date fin	Progression	Statut	Actions
2	Système gestion des projets	2016-02-01 00:00:00.0	2016-05-31 00:00:00.0	40%	En cours	[Icons]
3	Publicité agence de voyage	2016-04-12 00:00:00.0	2016-06-10 00:00:00.0	80%	En cloture	[Icons]
4	Evaluation du marché des ERP	2016-04-01 00:00:00.0	2016-09-30 00:00:00.0	95%	Terminé	[Icons]
5	Plateforme gestion commerciale	2016-04-01 00:00:00.0	2016-08-10 00:00:00.0	55%	En cours	[Icons]
9	Site E-commerce	2016-06-09 00:00:00.0	2016-08-31 00:00:00.0	30%	En cours	[Icons]
10	Projet sécurité réseau	2016-07-11 00:00:00.0	2016-08-31 00:00:00.0	60%	En cours	[Icons]
11	Projet CRM odoo	2016-06-29 00:00:00.0	2016-08-31 00:00:00.0	9%	Nouveau	[Icons]
12	Site E-voyage	2016-06-08 00:00:00.0	2016-08-18 00:00:00.0	10%	Prévu	[Icons]
17	Plateforme gestion des équipes	2016-06-08 00:00:00.0	2016-09-30 00:00:00.0	20%	En cours	[Icons]
18	Plateforme e-voyage	2016-07-29 00:00:00.0	2016-08-31 00:00:00.0	11%	En cours	[Icons]
19	Application gestion RH	2016-04-04 00:00:00.0	2016-07-31 00:00:00.0	40%	En cours	[Icons]
20	Solution de gestion et suivie de production	2016-04-01 00:00:00.0	2016-09-30 00:00:00.0	80%	En cloture	[Icons]

Affichage 1 to 12 of 14 éléments

Premier Précédent 1 2 Suivant Dernier

Figure 50: Interface gestion des projets

Pour pouvoir créer un projet, l'administrateur remplit le formulaire suivant par les informations d'un projet comme la date de début et la date de fin. Suite à une validation de l'administrateur, le système vérifie les données entrées et valide l'action.

**Gestion des projets**

Information Général

Nom du Projet: SID suivi des équipes

Propriétaire: William

Responsable du Projet: El badri

Statut: En cours

Progression: 50

Détails du Projet

Domaine: Informatique

Date Démarrage: 06/01/2016

Date Fin: 08/31/2016

Coût estimé: 10000

Nombre des tâches: 6

Information Supplémentaires

Description:

Annuler Sauvegarder

Figure 51: Ajouter un projet

La figure suivante illustre les documents dépendants à un projet, à partir de cette interface l'administrateur peut consulter les documents d'un projet, les télécharger ou les supprimer.

**Gestion des documents**

Document

Liste des documents

Affichage 10 entrées Rechercher:

Code	Titre	Description	Contenu	Date de création	Téléchargement	Action
1	Cahier des charges	cahier des charges de projet	application/pdf			

Affichage 1 to 1 of 1 éléments

Premier Précédent 1 Suivant Dernier

Figure 52: Liste des documents de projets

L'administrateur peut choisir l'opération d'ajouter un document, il clique sur document ensuite il obtient l'interface suivante (figure 53) il insère les données convenables et il valide l'insertion.

Ajouter un nouveau document

Nom de document:\*

Fichier:\*  Aucun fichier choisi

Description:

Affichage 1 to 1 of 1 éléments

Figure 53: Ajouter document d'un projet

L'administrateur peut visualiser chaque projet avec ses taches à l'aide de diagramme de Gantt. Pour ce faire, il se redirige vers l'interface de consultation des projets et sélectionne un projet auquel il est généré le diagramme de Gantt.

Le diagramme illustre de façon détaillée les taches avec ses dates de début et dates de fin.

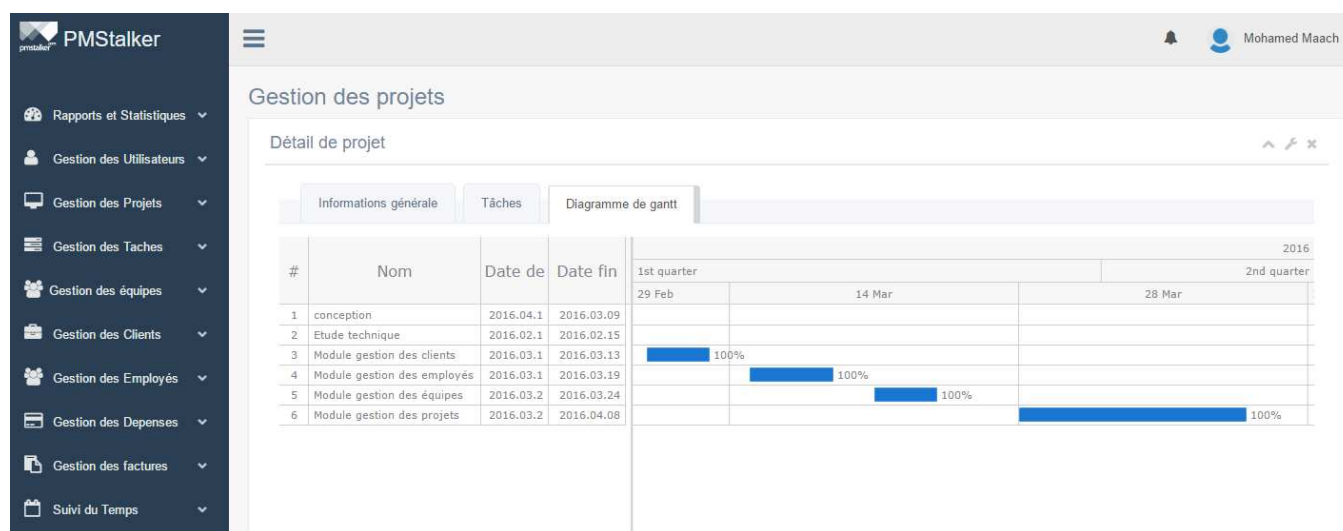
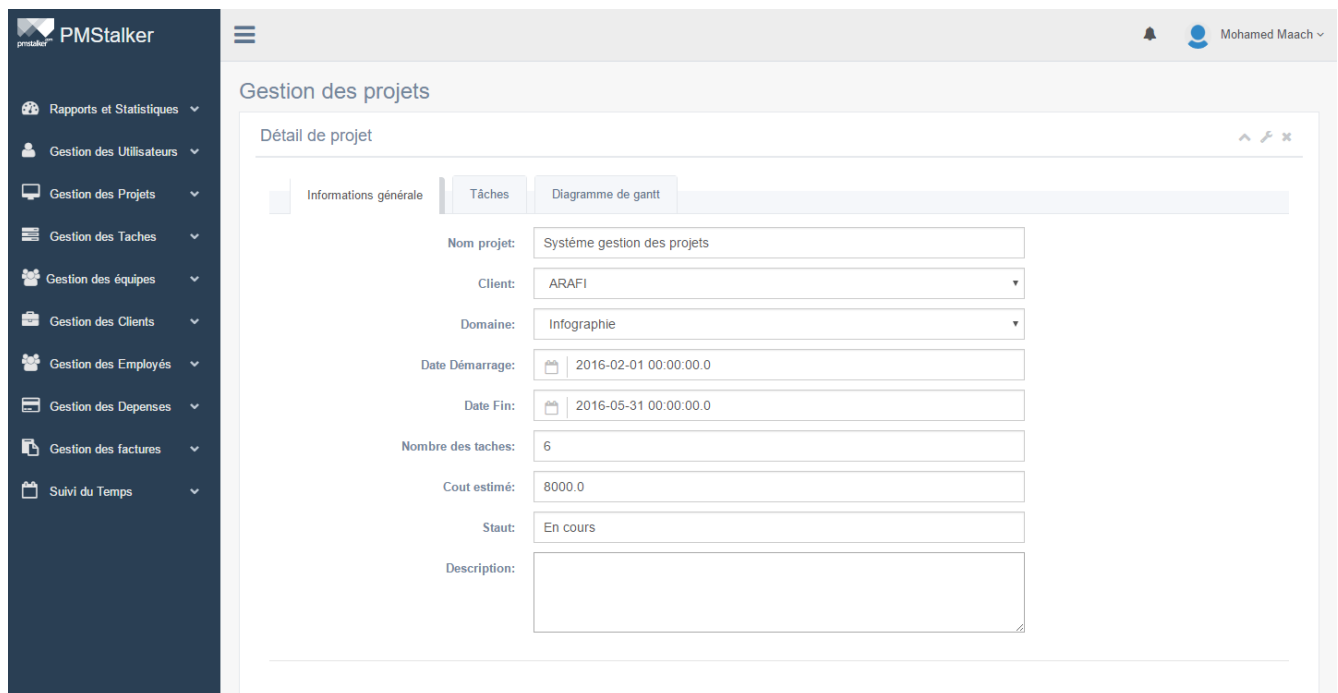


Figure 54: Diagramme de Gantt d'un projet

A partir de la liste de consultation des projets, l'administrateur peut consulter les détails d'un projet, les informations générales d'un projet, ses taches associées et le diagramme de Gantt. La figure suivante (figure 55) montre les informations générales d'un projet, par exemple sa date de début et sa date de fin.





**PMStalker**

Gestion des projets

Détail de projet

Informations générale | Tâches | Diagramme de gantt

Nom projet: Système gestion des projets

Client: ARAFI

Domaine: Infographie

Date Démarrage: 2016-02-01 00:00:00.0

Date Fin: 2016-05-31 00:00:00.0

Nombre des taches: 6

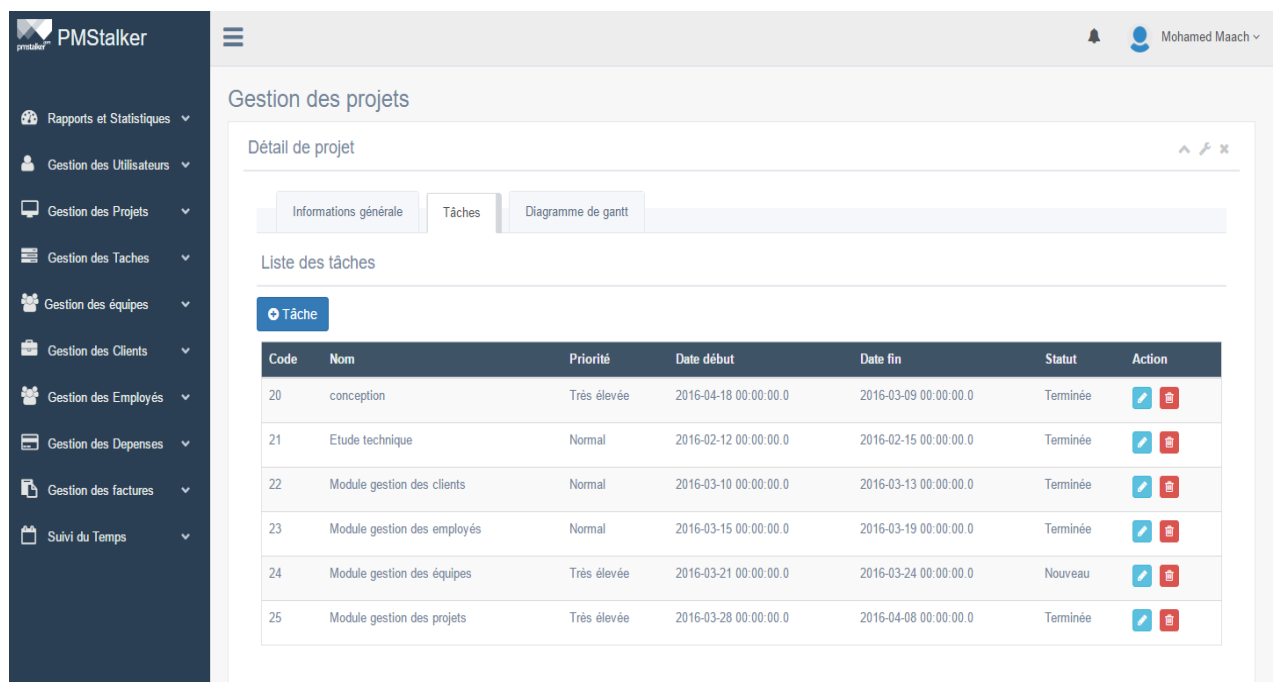
Cout estimé: 8000.0

Statut: En cours

Description:

Figure 55: Détail de projet-informations générales-

L'interface suivante présente la liste des taches associées à un projet selon ses priorités, ses dates de début et ses dates de fin.



**PMStalker**

Gestion des projets

Détail de projet

Informations générale | Tâches | Diagramme de gantt

Liste des tâches

+ Tâche













Code	Nom	Priorité	Date début	Date fin	Statut	Action
20	conception	Très élevée	2016-04-18 00:00:00.0	2016-03-09 00:00:00.0	Terminée	 
21	Etude technique	Normal	2016-02-12 00:00:00.0	2016-02-15 00:00:00.0	Terminée	 
22	Module gestion des clients	Normal	2016-03-10 00:00:00.0	2016-03-13 00:00:00.0	Terminée	 
23	Module gestion des employés	Normal	2016-03-15 00:00:00.0	2016-03-19 00:00:00.0	Terminée	 
24	Module gestion des équipes	Très élevée	2016-03-21 00:00:00.0	2016-03-24 00:00:00.0	Nouveau	 
25	Module gestion des projets	Très élevée	2016-03-28 00:00:00.0	2016-04-08 00:00:00.0	Terminée	 

Figure 56: Détail de projet-taches de projet-

Comme on peut à partir de l'onglet suivante (figure 57) de visualiser le diagramme de Gantt du projet :

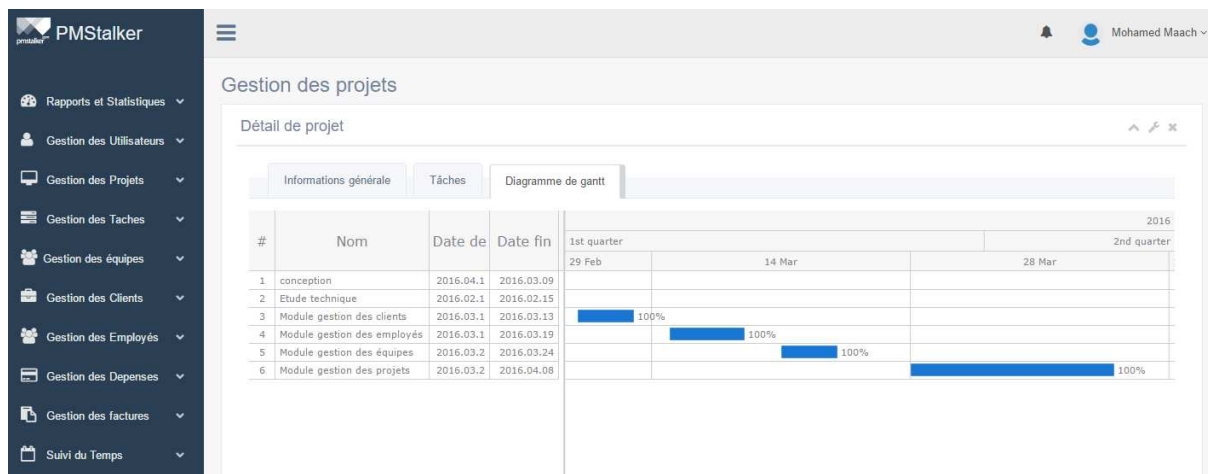


Figure 57: Détail de projet-Gantt de projet-

### ■ Gestion des tâches :

L'administrateur peut consulter la liste des tâches à travers l'interface suivante, cette interface propose les fonctionnalités CRUD. L'administrateur peut modifier et supprimer une tâche. Egalement il peut ajouter une nouvelle tâche.

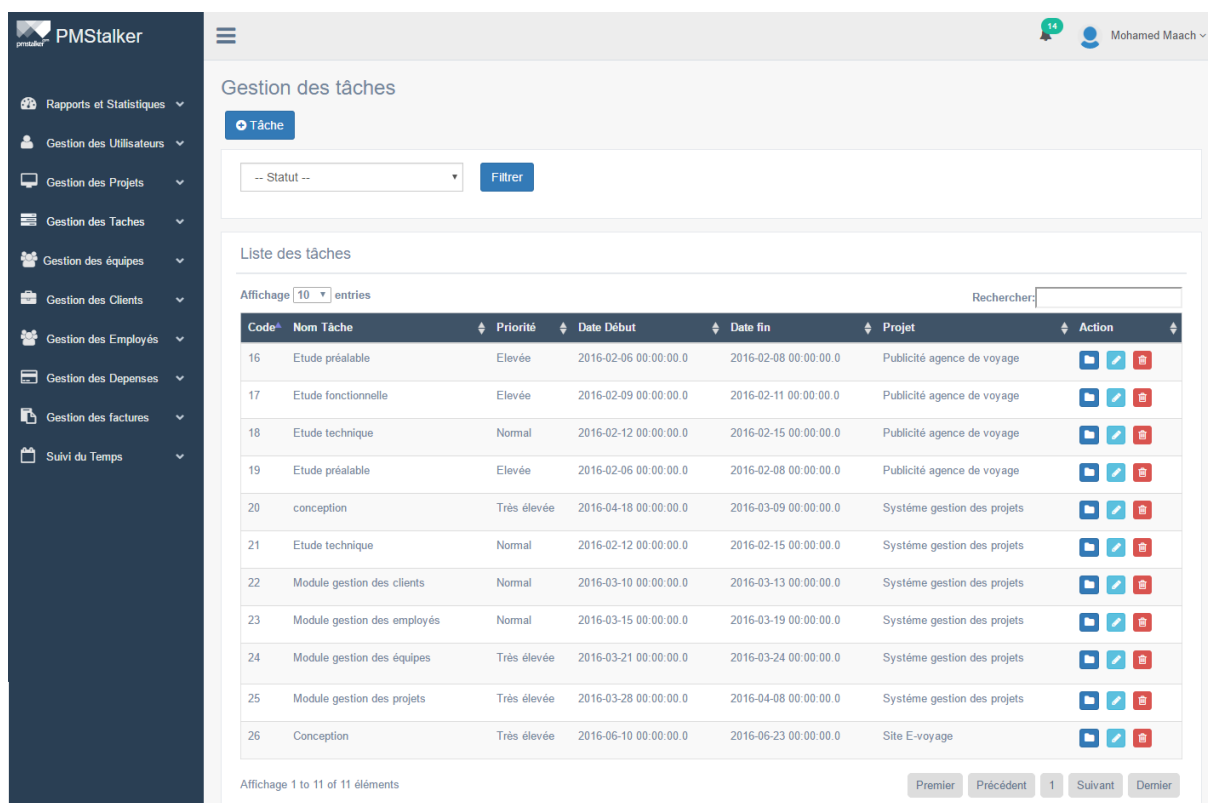


Figure 58: Liste des tâches

L'utilisateur peut ajouter une nouvelle tâche via l'interface de nouvelle tâche (figure 59). Il remplit le formulaire par les informations de la tâche que l'on souhaite ajoutée.

**Gestion des tâches**

Information Général

Nom:\* Conception de projet

Projet:\* Systeme gestion des factures

Responsable de tâche:\* Lafhal

Priorité:\* Très élevée

Détails du Tâche

Date Début: 06/01/2016

Date Fin: 08/04/2016

Etat:\* Nouveau

Progression:\* 60

Equipe: Equipe Conception

Information Supplémentaires

Description:

Annuler Sauvegarder

Figure 59: Ajouter une tâche

### ■ Gestion des dépenses :

L'administrateur a la possibilité de consulter la liste des catégories des dépenses et il peut effectuer les opérations de gestion (ajouter, modifier et supprimer) une catégorie des dépenses.

**Gestion des dépenses**

+ Catégorie des dépenses

Types de dépense

Affichage 10 entrées Rechercher:

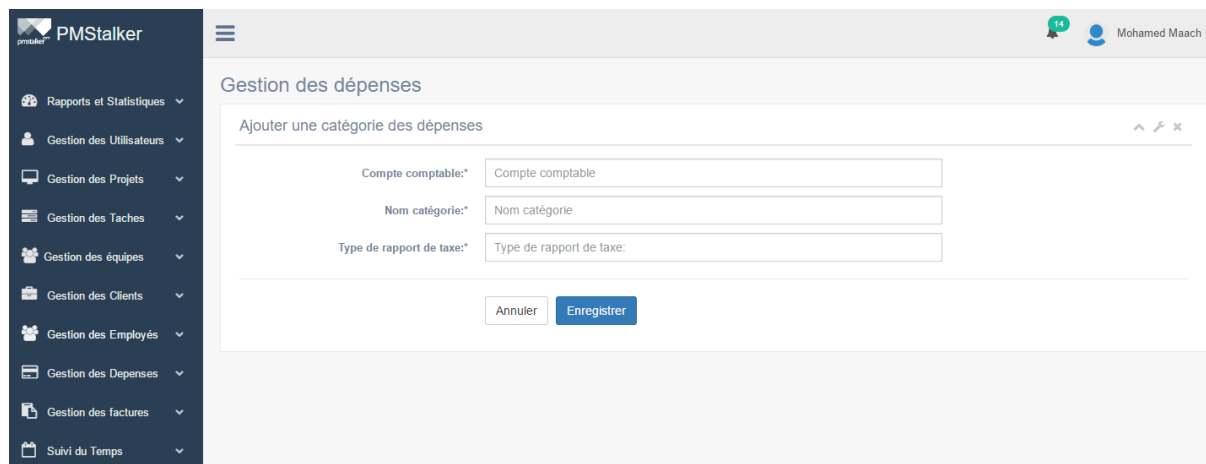
Code	Compte comptable	Nom	Type de rapport de taxe	Action
1		Assurance		
2		Transport		
3		Abonnements		
4		Frais bancaire		
5		Hébergement		
6		Frais de port		
7		Dépenses de bureau		
8		Formation du personnel		
9		Publicité		
10		Promotion		
11	Compte BB1	Frais bancaire et finance	Annuel	
12		Logiciels		

Affichage 1 to 12 of 12 éléments

Premier Précédent 1 Suivant Dernier

Figure 60: Gestion des catégories des dépenses

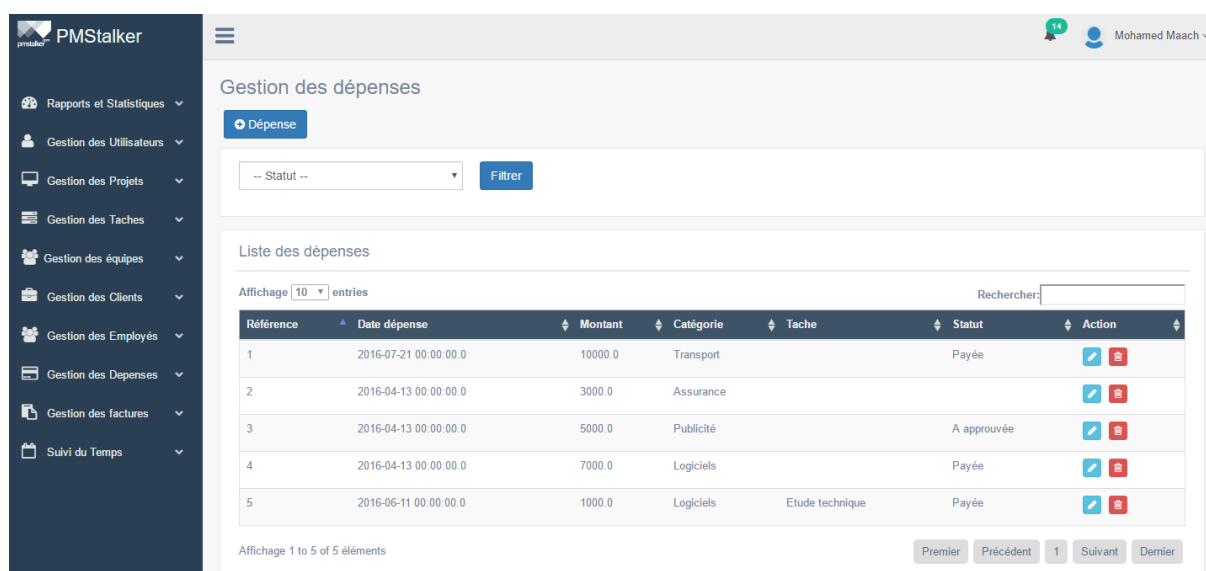
Pour l'ajout d'une nouvelle catégorie des dépenses, on obtient un formulaire contenant le compte comptable, nom de catégorie et le type de rapport de taxe.



The screenshot shows the 'Gestion des dépenses' section of the PMStalker application. A form titled 'Ajouter une catégorie des dépenses' is displayed. It contains three input fields: 'Compte comptable:', 'Nom catégorie:', and 'Type de rapport de taxe:'. Below the fields are two buttons: 'Annuler' and 'Enregistrer'.

Figure 61: Ajouter une catégorie des dépenses

L'administrateur a la possibilité de consulter la liste des dépenses, afin de les gérer (ajouter, modifier, supprimer) dépense.



The screenshot shows the 'Gestion des dépenses' section of the PMStalker application. A table titled 'Liste des dépenses' is displayed. The table has columns: Référence, Date dépense, Montant, Catégorie, Tache, Statut, and Action. There are 5 rows of data. Below the table, there is a pagination bar showing 'Affichage 1 to 5 of 5 éléments' and buttons for 'Premier', 'Précédent', '1', 'Suivant', and 'Dernier'.











Référence	Date dépense	Montant	Catégorie	Tache	Statut	Action
1	2016-07-21 00:00:00.0	10000.0	Transport		Payée	 
2	2016-04-13 00:00:00.0	3000.0	Assurance			 
3	2016-04-13 00:00:00.0	5000.0	Publicité		A approuvée	 
4	2016-04-13 00:00:00.0	7000.0	Logiciels		Payée	 
5	2016-06-11 00:00:00.0	1000.0	Logiciels	Etude technique	Payée	 

Figure 62: liste des dépenses

Pour ajouter les dépenses, l'administrateur doit remplir le formulaire suivant (figure 63) par la date de dépense et il doit choisir la tache concernée et la catégorie de dépense.

**PMStalker**

Gestion des dépenses

Ajouter nouvelle dépense

Date Dépense:

Tache:

Catégorie:

Description:

Montant:

Devise:

Statut:

Méthode de paiement:

Facturable: ☐

Figure 63: Ajouter dépense

L'administrateur peut consulter le rapport des dépenses via l'interface suivante, elle représente les statistiques concernant les dépenses par exemple, le nombre de dépenses par statut et selon la catégorie.

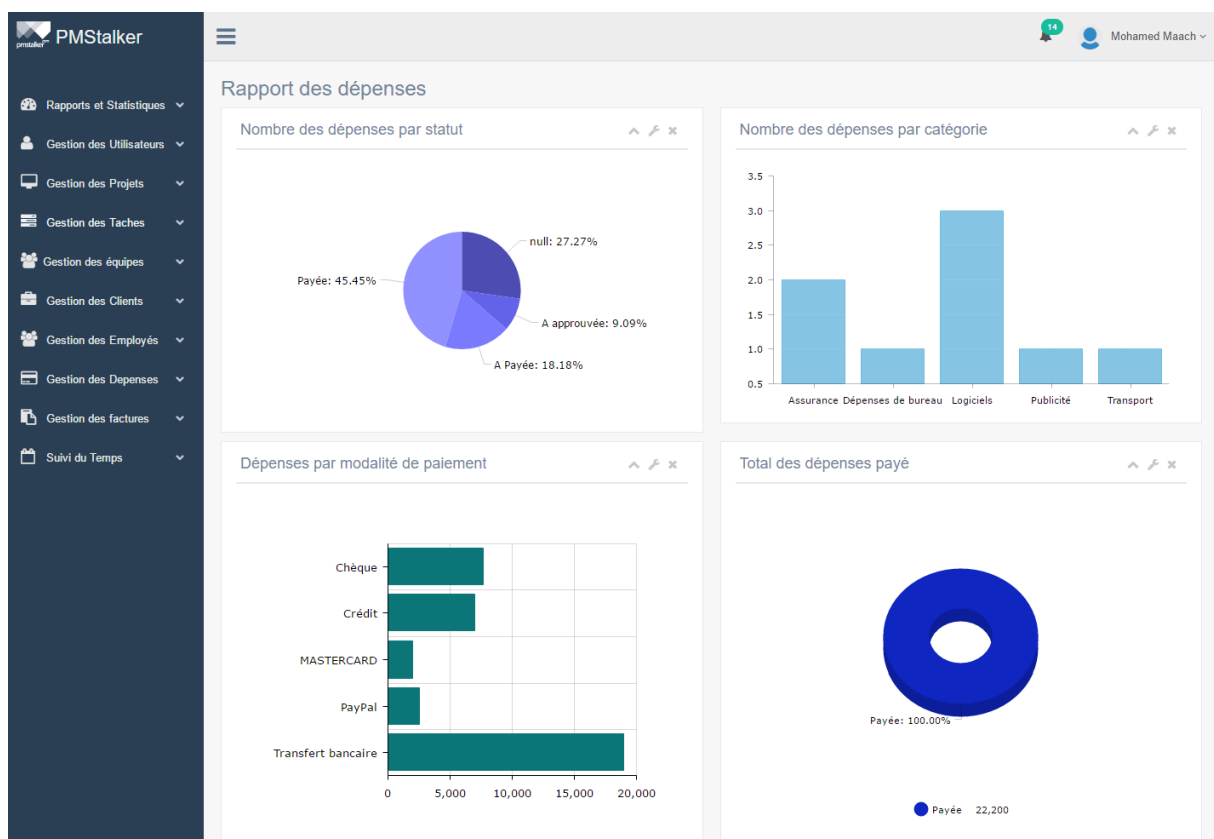
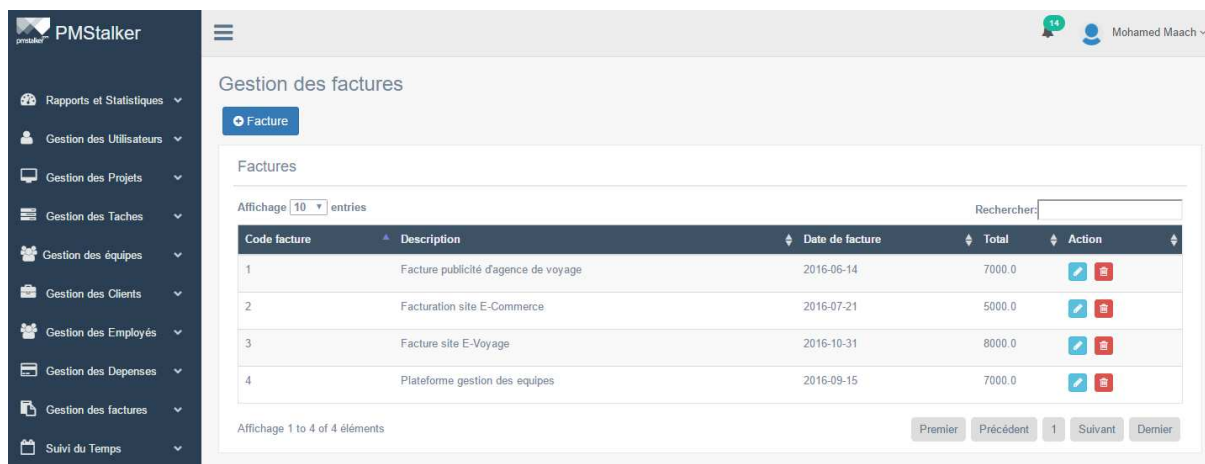


Figure 64: Rapport des dépenses

### ■ Gestion des factures :

La gestion des factures est une partie principale dans le rôle de directeur, et dans la page concernée, s'affiche la liste des factures ajoutées avec les deux boutons de suppression et modification.



Gestion des factures

Factures

Affichage 10 entrées

Rechercher:

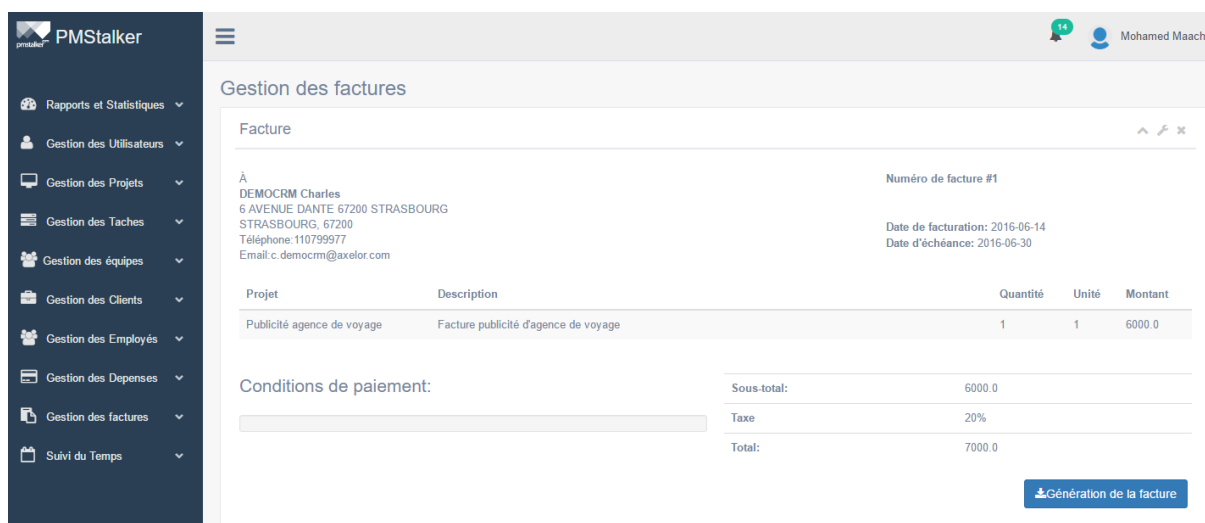
Code facture	Description	Date de facture	Total	Action
1	Facture publicité d'agence de voyage	2016-06-14	7000.0	
2	Facturation site E-Commerce	2016-07-21	5000.0	
3	Facture site E-Voyage	2016-10-31	8000.0	
4	Plateforme gestion des équipes	2016-09-15	7000.0	

Affichage 1 to 4 of 4 éléments

Premier Précédent 1 Suivant Dernier

Figure 65: Liste des factures

La figure suivante montre les détails d'une facture, contient tous les informations nécessaires d'une facture, en cliquant sur le bouton Génération de la facture pour extraction sous format PDF.



Gestion des factures

Facture

À DEMOCRM Charles  
6 AVENUE DANTE 67200 STRASBOURG  
STRASBOURG, 67200  
Téléphone: 110799977  
Email: c.democrm@axelor.com

Numéro de facture #1

Date de facturation: 2016-06-14  
Date d'échéance: 2016-06-30

Projet	Description	Quantité	Unité	Montant
Publicité agence de voyage	Facture publicité d'agence de voyage	1	1	6000.0

Conditions de paiement:

Sous-total:	6000.0
Taxe	20%
Total:	7000.0

Génération de la facture

Figure 66: Détails d'une facture

### ■ Suivi du temps :

Le directeur peut consulter le calendrier de ses projets, l'interface suivante (figure 67) illustre le calendrier qui contient les noms des projets à réaliser selon les dates de début.

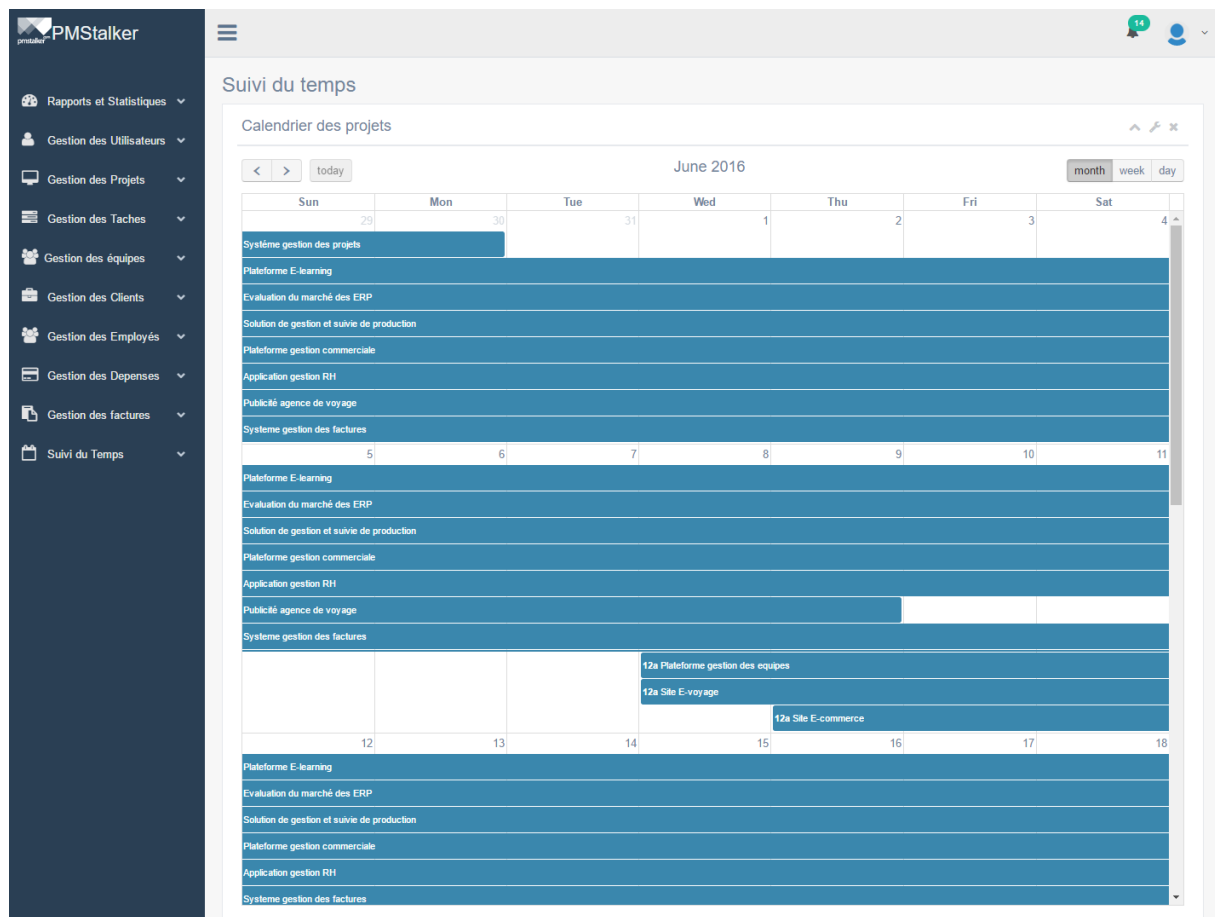


Figure 67: Planning des projets

Chaque tâche créée sera affichée dans le calendrier de planning des tâches suivantes comme elle représente la figure suivante (figure 68) :

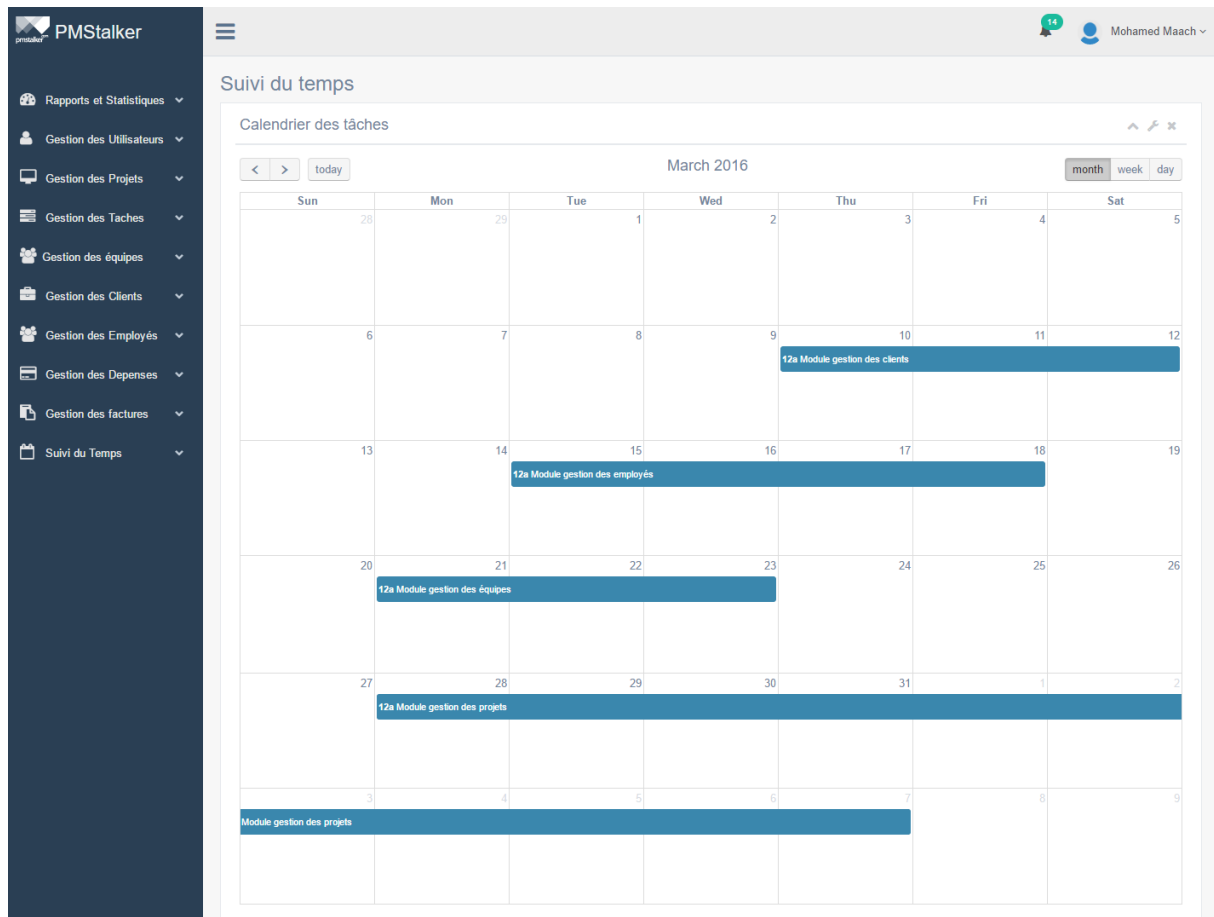


Figure 68: Planning des tâches

### 3-2 Partie Employé

A son tour L'employé peut se connecter à notre système pour voir l'avancement des projets sur lesquels il travaille et aussi voir les tâches qui lui sont affectés. Il peut aussi modifier l'état de la tâche qu'il travaille lorsqu'il la termine.

Dans cette figure l'employé peut consulter la liste de ses projets :

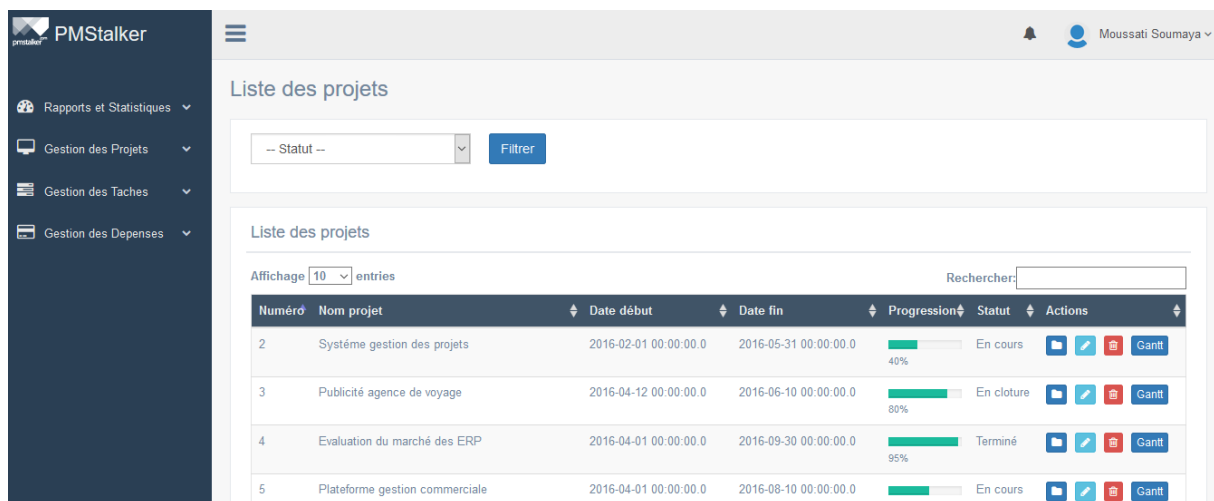


Figure 69: les projets de l'employé



Dans la figure suivante l'employé peut consulter seulement ses tâches et valider la tâche qu'il a terminé comme suit :

Code	Nom Tâche	Date Début	Date fin	Progression	Action
17	Etude fonctionnelle	2016-02-09 00:00:00.0	2016-02-11 00:00:00.0	100%	Valider
18	Etude technique	2016-02-12 00:00:00.0	2016-02-15 00:00:00.0	100%	Valider
19	Etude préalable	2016-02-06 00:00:00.0	2016-02-08 00:00:00.0	100%	Valider
20	conception	2016-04-18 00:00:00.0	2016-03-09 00:00:00.0	100%	Valider
21	Etude technique	2016-02-12 00:00:00.0	2016-02-15 00:00:00.0	100%	Valider
22	Module gestion des clients	2016-03-10 00:00:00.0	2016-03-13 00:00:00.0	100%	Valider
23	Module gestion des employés	2016-03-15 00:00:00.0	2016-03-19 00:00:00.0	100%	Valider
24	Module gestion des équipes	2016-03-21 00:00:00.0	2016-03-24 00:00:00.0	100%	Valider

Figure 70: liste des tâches de l'employé

## Conclusion

Au cours de ce chapitre, nous avons effectué une brève présentation du processus de test, nous avons donné un aperçu de notre démarche de développement afin de garantir la mieux organisation de cette phase. Après avoir illustrée quelques vues et aspects de notre applications nous avons exposée des tests effectués tout au long du codage qui ont pour but de tester le système fabriqué.

# Conclusion Général et Perspectives

Notre projet de fin d'étude a été réalisé au profit de la société Adrénaline Holding. Durant ce projet la mission suivante nous a été confiée : la conception et la réalisation d'une application de gestion et suivie des projets de l'entreprise afin d'aider à la décision. A ce stade de réalisation, on a pu développer une solution informatique (PMStalker) couvrant totalement les besoins métiers fonctionnels et techniques qui répondent totalement aux exigences des utilisateurs finaux.

Notre choix de la plateforme a porté ses fruits en poussant les performances de l'application à des records surprenants. D'autre part, la décomposition en trois couches permet de bénéficier d'un maximum d'ouverture et d'extensibilité propice à un développement facile pouvant être effectuée par des tiers.

Pour ce faire nous avons commencé dans un premier lieu par comprendre le contexte général de notre application et identifier les différentes exigences de notre futur système. Ensuite une étude détaillée du cahier de charges afin de comprendre les besoins des futurs utilisateurs du système, et par la suite nous avons commencé à développer les différents vues et objets, en respectant les différentes étapes du cycle de développement Scrum.

Les perspectives possibles à la suite du présent projet sont multiples et couvrent plusieurs aspects, tels que l'émigration des données à une base de données NoSQL comme Cassandra et la création d'une version mobile.

Finalement, cette expérience était opportunité que nous avons apprise en travaillant avec des technologies récentes et sollicitées sur le marché, plus particulièrement, le développement des applications avec la spécification Java EE et en utilisant des Framework très connus à titre exemple Spring MVC, Spring Security, Hibernate ORM, comme elle nous a offert l'opportunité d'intégrer dans l'environnement de l'entreprise et d'améliorer nos capacités dans notre vie professionnelle tout en enrichissant nos connaissances et notre formation théorique et pratique acquise durant nos études à l'ENSATE.

# Référence

**[Ref 1]** : <http://www.blog-gestion-de-projet.com/henry-laurence-gantt-et-son-diagramme/>

**[Ref 2]** : <http://www.clipindustrie.com/fr/pages/gestion-de-projet>

**[Ref 3]** : [https://fr.wikipedia.org/wiki/Scrum\\_\(m%C3%A9thode\)](https://fr.wikipedia.org/wiki/Scrum_(m%C3%A9thode))

**[Ref 4]** : <http://mastercomputersystem.com/reseaux.php?page=res>

**[Ref 5]** : <https://www.jtips.info/index.php?title=Spring/Test>

# Bibliographie

- **Spring par exemple** – Gary Mak, édition Pearson
- **Spring par la pratique** - Thierry Templier, édition Eyrolles
- **Hibernate in action**- Christian Bauer, édition Hanning
- **UML2 pratique de la modélisation**- Yann THIERRY-MIEG, édition Pearson

# Webographie

- **Spring Pour Nous :**  
<http://projects.spring.io/spring-security/>, 05/03/2016
- **Mkyong :**  
<http://www.mkyong.com/spring-mvc/>, 06/03/2016
- **Websystique :**  
<http://websystique.com>, 15/03/2016
- **Mkyong tutorials**  
<http://www.mkyong.com/tutorials/spring-mvc-tutorials/>, 18/03/2016
- **JUnit Tests**  
<http://www.junit.fr/>, 24/04/2016