



**HAL**  
open science

## Système informatique de gestion de laboratoire

Jouhayna Sabouneh

► **To cite this version:**

Jouhayna Sabouneh. Système informatique de gestion de laboratoire. Ingénierie assistée par ordinateur. 2015. dumas-01383353

**HAL Id: dumas-01383353**

**<https://dumas.ccsd.cnrs.fr/dumas-01383353>**

Submitted on 18 Oct 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**ISSAE-CNAM**  
**CENTRE REGIONAL ASSOCIE AU**  
**CONSERVATOIRE NATIONAL DES ARTS ET METIERS**

**MEMOIRE**

**Présenté en vue d'obtenir**

**Le DIPLOME d'INGENIEUR CNAM**

**SPECIALITE : INFORMATIQUE**

**Par**

**SABOUNEH JOUHAYNA**

**SYSTEME INFORMATIQUE DE GESTION DE LABORATOIRE**

---

**JURIE**

**PRESIDENT :**

**MEMBRES :**

## RESUME

Le laboratoire est un système complexe, où de nombreuses procédures sont mises en œuvre impliquant un grand nombre de personnes et des acteurs en faveur du service d'analyse offerte. D'où un système informatique automatisant tous les processus métiers opérationnels, de support et de pilotage devient une nécessité pour tout laboratoire afin d'assurer le bon fonctionnement et l'alignement avec les normes de gestion de qualité imposées par l'ISO.

Cette mémoire porte sur l'étude et la réalisation d'un système informatique de gestion de laboratoire (SIGL). L'étude de projet sera menée à travers une cartographie fonctionnelle des processus en utilisant le standard BPMN, complétée par l'approche objet en utilisant le standard UML et guidée par les principes de modélisation agile. Le système suit le paradigme « diviser pour régner », adopte l'architecture web, utilise Java comme langage de programmation et s'appuie essentiellement sur les technologies à source ouverte et les logiciels libres. Une analyse systématique sera dressée pour justifier mes choix techniques comme architecture, langages de programmation et outils de développement etc. De même, on va aborder les différentes stratégies de déploiement, soit sur un serveur local comme un service Windows avec le Wrapper « YAJSW » en utilisant le modèle relationnel de gestion de données, soit sur plateforme en nuage (Cloud Platform) en utilisant le modèle No-SQL avec le leader mondial du Cloud Computing Amazon Web Services (AWS) et Heroku Cloud comme plateforme de service. Enfin, on va justifier notre plan de maintenance et argumenter notre recours à la télé-assistance et la télémaintenance comme stratégie de maintenance.

Ce qui diffère ma mémoire d'autres est l'esprit entrepreneur qui accompagne le développement de ce projet dans toutes ses phases. Cette mémoire va visualiser les étapes clés de la création de notre entreprise « ADHENE » et l'établissement de son environnement de développement tout au long de développement de ce projet.

### MOTS-CLES

Système informatique de gestion de laboratoire (SIGL) - cartographie des processus métiers - Modélisation Agile - source ouverte et logiciel libre - JAVA - Amazon Web Services (AWS) - Cloud Computing - Méthode Agile SCRUM- téléassistance et télémaintenance.

## SUMMARY

The laboratory is a complex system where many procedures are implemented involving many and variant actors to offer the analysis service.

This is why a laboratory information system is a necessity for each laboratory in order to automatize the whole operational process and to control the quality as imposed by ISO.

This report will discuss the study and the realization of a laboratory information management system (LIMS). This project study will be done using the functional cartography of business processes using the BPMN standards, complemented by the object approach using the UML standard and guided by the agile modeling principles.

The System follows the paradigm "divide to conquer", adopts the web architecture, uses Java as programming language and relies mainly on open source technologies and free software. A systematic analysis will be drawn up to justify my choice as technical architecture, programming languages and development tools etc.

Also, we will address the various deployment strategies, either on a local server as a Windows service with the Wrapper "YAJSW" using the relational model of data management, either on platform in cloud (Cloud Platform) using the model No.-SQL with the world leader of Cloud Computing Amazon Web Services (AWS) and Heroku Cloud as service platform.

What differs my report of other is the spirit of entrepreneur who accompanied the development of this project in all its phases. This memory will show the key steps of the creation of our company "ADHENE" and the establishment of its development environment throughout the development of this project.

### KEYWORDS

Laboratory information management system (LIMS) - cartography of business process - Agile modeling - open source and free software - JAVA - Amazon Web Services (AWS) - Cloud Computing - SCRUM Agile Method - remote assistance - remote maintenance.

## REMERCIEMENTS

En préambule à ce mémoire, j'avais envie d'adresser mes sincères remerciements à l'équipe d'ADHENE, pour laquelle j'ai été un membre depuis trois années, pour l'effort, pour l'inspiration, l'aide et le temps qu'elle a bien voulu me consacrer pour m'aider à réaliser ce mémoire. Également, je tiens à exprimer toute ma reconnaissance à Monsieur Bilal MAAZ, mon tuteur de mémoire, pour m'avoir fait partager son expérience et pour s'être porté garant de mon projet jusqu'à la soutenance. Mes remerciements s'adressent également à notre chef de département Monsieur Pascal FARES qui a orienté mes choix professionnels à travers ses enseignements et ses conseils.

Enfin, je remercie ma famille, mes collègues et mes amis de m'avoir encouragé à surmonter les contraintes liées aux cours du soir et à la rédaction de ce mémoire. Je ne doute pas que cette expérience m'assure une meilleure reconnaissance professionnelle suite à l'obtention du diplôme d'ingénieur CNAM, autant qu'il m'aura permis de consolider mes acquis et d'orienter ma carrière professionnelle.

## INTRODUCTION

Cette mémoire marque le point final de ma formation à ISSAE-CNAM, Liban. Mon objectif est d'enrichir ma carrière professionnelle par un diplôme d'ingénieur, en valorisant le travail que j'ai produit pour ADHENE en tant d'entrepreneur et architecte de système. Il s'agit d'une mission d'une année environ, qui a débuté en février 2013 pour s'achever en janvier 2014

Je souhaite présenter ici le contexte du mon projet, sa démarche de développement ainsi que certaines problématiques qui m'ont été confiées en tant que chef de projet tout au long de cette mission. De même, cette mémoire a pour objet de valider ma capacité à être un ingénieur tout en exprimant mes compétences à analyser une problématique, établir l'architecture la plus convenable d'un système informatique et à gérer et diriger son processus de développement.

Dans le cadre de cette mission professionnelle, j'ai conçu, développé puis optimisé une application SIGL ayant comme client direct le centre de contrôle de qualité de la chambre de commerce et d'industrie de tripoli et comme client indirect tout laboratoire et centre de contrôle de qualité dans le monde arabe.

D'abord, ce projet se distingue par sa richesse, comme cahier des charges, technologies utilisées et outils méthodologiques adéquats utilisés lors de développement. En deuxième lieu, l'intérêt du projet vient de l'autonomie, dont j'ai pu bénéficier, qui m'a permis d'effectuer ou de proposer des choix concernant l'environnement de développement comme langages, technologies, techniques et outils.

La situation économique au Liban connaît une dégradation continue sous l'effet de tensions politiques internes et les répercussions du conflit syrien. De même, notre ville Tripoli a connu plusieurs épisodes de violence en 2013 et 2014 ce qui reflète négativement sur notre fonctionnement, surtout que nous sommes une entreprise en démarrage, en minimisant notre productivité, augmentant nos dépenses et rendant notre existence comme une entreprise débutant un défi de survie. Avec ces circonstances, la satisfaction des contraintes de mon projet comme temps et coût sera un challenge réel pour moi comme chef de projet et entrepreneur.

La première partie sera consacrée à la description du projet, présentation de l'entreprise et de mes responsabilités dans le cadre du projet. La deuxième partie détaillera toutes les phases du

cycle de développement du projet dès le recueil et l'analyse de besoins et la disposition du cahier des charges architecturales et techniques du projet, passant par l'implémentation, jusqu'à la phase de déploiement et la maintenance. La troisième partie sera dédiée à la justification des choix technologiques des langages et des outils de développement. Finalement, la quatrième partie s'attachera à la présentation des bonnes pratiques (Best Practices)<sup>1</sup> et des moyens de communication que je tiens à les appliquer dans mon équipe pour atteindre la meilleure performance.

---

<sup>1</sup> Bonnes pratiques (Best Practices) désignent, dans un milieu professionnel donné, un ensemble de comportements qui font consensus et qui sont considérés comme indispensables par la plupart des professionnels du domaine.

# Sommaire

RESUME.....	2
SUMMARY.....	3
REMERCIEMENTS.....	4
INTRODUCTION.....	5
Chapitre 1: A Propos de l'Entreprise.....	12
1.1 Parmi nos services.....	12
1.1.1 Développement des Applications Mobile.....	12
1.1.2 Développement des Systèmes Web.....	13
1.2 Parmi notre Clientèle.....	13
1.3 Parmi nos Produits.....	13
1.4 Etudiant CNAM comme entrepreneur.....	15
1.4.1 Structure hiérarchique.....	16
1.4.2 Culture et Environnement.....	17
1.4.3 Chiffres Optimistes.....	17
Chapitre 2: A Propos du Projet.....	18
2.1 Objectif du projet.....	18
2.2. Infographie du Projet.....	19
2.3 Mes Responsabilités.....	20
2.4 Clientèle du projet.....	21
2.4.1 Client direct.....	21
2.4.2 Client Indirect.....	22
Chapitre 3: Analyse des Besoins.....	23
3.1. Modélisation de Processus Métiers.....	23
3.2. Cartographie Fonctionnelle des Processus.....	24
3.2.1. Les processus opérationnels du laboratoire.....	24



3.2.2. Les processus de support du laboratoire.....	24
3.2.3. Les processus de pilotage du laboratoire .....	25
3.3. Acteurs du Système.....	25
3.4. Outils de Modélisation Utilisé.....	28
3.5. Architecture Basé sur Web .....	28
3.5.1. Personnalisation et Flexibilité .....	29
3.5.2. Importance de Mobilité .....	29
3.5.3. Amélioration de l'Interopérabilité .....	29
3.5.4. Adaptation à l'Augmentation de la Charge de Travail .....	29
3.5.5. Déploiement Facile et Sécurité .....	30
Chapitre 4: Conception et Modélisation.....	31
4.1. Cartographie Applicative du Système .....	31
4.2. Découpage du système en sous-systèmes.....	33
4.2.1. Sous-système de Tests et Types Des échantillons .....	35
4.2.2. Sous-système Clients .....	38
4.2.3. Sous-système de demandes de tests.....	40
4.2.4. Sous-système des Employés .....	45
4.2.5. Sous-système de Stock.....	46
4.2.6. Sous-système de Communication.....	50
4.2.7. Sous-système des Equipements.....	52
4.2.8. Sous-système de Rapports et statistiques .....	54
4.3. Adaptation des Principes de Modélisation Agile .....	54
4.3.1. Comment gagner du temps dans la modélisation .....	54
4.3.2. Les Bonnes Pratiques de modélisation agile.....	55
4.3.3. Lignes Directrices de Style .....	56
4.4. Ergonomie de l'interface utilisateur .....	56

4.5. Outils de Conception et Modélisation .....	57
Chapitre 5: Gestion de Projet.....	59
5.1. Processus de Développement.....	59
5.2. Choix de Méthode Agile .....	59
5.2.1. Cascade .....	60
5.2.2. RUP.....	60
5.2.3. Choix de Méthodologie SCRUM.....	60
5.3. Les rôles dans l'équipe.....	61
5.4 Taille et Membres de l'équipe: .....	61
5.5. Les 3 réunions SCRUM .....	62
5.6. Objectif De Délai .....	63
5.7. Le Plan .....	63
5.8. Objectif de Coût .....	66
5.9. Règles de gestion du projet .....	66
5.10. Plan de communication .....	66
Chapitre 6 : Implémentation.....	67
6.1. Choix de Langage de Programmation .....	67
6.1.1. Pourquoi la source ouverte?.....	67
6.1.2. Pourquoi Java? .....	68
6.1.3. Elimination de choix PHP .....	68
6.1.4. Elimination de .Net .....	69
6.2. Recours à l'utilisation d'un cadre de développement .....	69
6.3. Pourquoi le cadre de développement Play?.....	70
6.3.1. Convention plutôt que configuration .....	70
6.3.2. Modèle-Vue-Contrôleur (MVC).....	71
6.3.3. Architecture de Ne Rien Partager HTTP.....	71

6.4. Avantages de cadre de développement Play.....	71
6.5. Choix de Système de Gestion de Base de Données .....	73
6.5.1 Pourquoi Amazon SimpleDb? .....	73
6.5.2 Pourquoi Siena comme couche de gestion des données (Data Management Layer) .....	74
Chapitre 7 : Installation.....	76
7.1 Déploiement comme Application Autonome Play.....	76
7.2 Installation du Système comme Service .....	76
7.3 Plan de Sauvegarde de Données.....	77
Chapitre 8 : Test et Maintenance .....	79
8.1. Outils de gestion de bogues YouTrack.....	80
8.2. Téléassistance et Télémaintenance avec Teamviewer .....	81
8.2.1. Télémaintenance .....	81
8.2.2. Téléassistance .....	82
8.3. Documentation Logicielle .....	82
8.3.1. Documentation Interne sur l'architecture et la conception .....	83
8.3.2. Manuel de l'Utilisateur .....	83
Chapitre 9 : Outils de Communication.....	85
9.1. Kunagi pour Gestion de Projet.....	85
9.2. Subversion.....	85
9.3. Tortoise SVN.....	86
9.4. Règles de Codage .....	86
9.5. Utilisation de Plugin Checkstyle .....	87
9.6. Erreurs Communs des Bases de Donnés.....	87
9.7. Librairie des Codes Exemples Communs.....	87
9.8. Conseils pour l'Optimisation du code Java .....	88
9.9. Librairie des Services Génériques .....	88

Chapitre 10 : Outils de Développement .....	89
10.1. Éclipse .....	89
10.2. iText PDF Java Library.....	89
10.3. jFree Charts .....	90
10.4. Zxing pour Codes à barres.....	90
10.5. ImgScalr.....	91
Chapitre 11 : Développement de Site Web Comme Outil Commercialisation.....	92
11.1. Développement d'un site web comme outil de commercialisation .....	92
11.2. Interface Utilisateur Basé sur le Concept Widget .....	92
11.3 Plan de Déploiement.....	93
11.3.1. Déploiement en Nuage avec Heroku .....	93
11.3.2. Amazon SimpleDb comme SGBD .....	94
11.3.3. Amazon S3 comme Service de Stockage.....	94
11.4. CSS3 site web multiplateformes .....	94
11.4.1. Réinitialisation du CSS.....	94
11.4.2. Le validateur de balisage W3C.....	95
CONCLUSION.....	96
SUMMARY.....	97
GLOSSAIRE.....	98
RÉFÉRENCES.....	102
TABLE DES FIGURES.....	103

## Chapitre 1: A Propos de l'Entreprise

ADHENE est une entreprise privée de développement de services informatiques et plus précisément de services Internet : développement des applications mobile, systèmes et sites web. Elle a été fondée en janvier 2012 à Tripoli, Liban Nord.

Le nom de l'entreprise ADHENE a pour origine la mythologie nord-est, ADHENE signifie les enfants de l'ambition PRIDE/AMBITION, ils étaient considérés comme des anges déchus du ciel, mais trop bon pour descendre à l'enfer. Ils ont été chassés du ciel parce qu'ils étaient tellement ambitieux, toujours conspirer pour prendre la place d'ODIN, père des dieux d'où il les jetait sur la terre afin qu'ils puissent nourrir l'esprit humain l'ambition et l'orgueil dont ils manquaient.

Notre objectif est de répondre aux besoins des processus opérationnels des métiers en différents secteurs par développer des solutions d'automatisation informatiques adaptées, simples et efficaces sur-mesure tout en garantissant l'intégrité et la fiabilité du système d'information.

Également, ADHENE met à la disposition de sa clientèle une gamme de produits génériques qui lui assure la gestion de ses processus de support comme gestion de stock, ressources humaines, comptabilité, communication, documentation, etc. Pour lui aider à se concentrer sur son activité « métier » qui a évidemment l'impact direct sur la satisfaction du client et la rentabilité. Elle est pionnière dans l'adaptation des techniques sources ouvertes (Open Source) au Liban. Alors qu'elle utilise les nouvelles techniques comme HTML5 pour développer des applications mobiles sur le marché local.

Notre mission c'est de créer des solutions uniques au plus bas coût et maximale efficacité, et d'offrir des services excellents capables d'enrichir l'ambition humaine et l'innovation.

### 1.1 Parmi nos services

#### 1.1.1 Développement des Applications Mobile

ADHENE crée des applications mobiles pour Androïde et iOS. Nous adoptons l'approche web avec HTML5, CSS3 et JavaScript où un seul code noyau est utilisé pour les applications mobiles et web, fondé sur le code web que nous connaissons et expérimentons. En utilisant plusieurs cadres de développement à source ouverte permettant le développement des applications multiplateformes, rapides et impressionnants avec presque une native

expérience d'interface utilisateur comme Sencha Touch2, Adobe PhoneGap qui permettent de déployer une application comme native sur Apple Store et Google App Store.

### 1.1.2 Développement des Systèmes Web

ADHENE est spécialisé en développement des systèmes web sur-mesure automatisant les processus opérationnels d'un métier en garantissant l'intégrité, la fiabilité et la qualité du système d'informations offert.

Nous utilisons Java et les logiciels libres et à source ouvert dans notre environnement de développement. Nous adaptons les processus agiles de développement de logiciels et l'approche pilotée par les tests (test-driven approach) où les exigences et les solutions évoluent grâce à la collaboration entre le client et à l'équipe intérieure de développement.

Pour plus d'informations, consultez-vous notre site web sur <http://www.adhene.com>.

## 1.2 Parmi notre Clientèle

- Le laboratoire de la chambre de commerce, d'industrie et de l'agriculture du Tripoli, Liban;
- le composé éducatif de l'Arz;
- la fédération des municipalités de DANNIEH, Liban Nord;
- la société de développement international ECODIT;
- MARE, projet coordonné par la chambre de commerce, d'industrie et de l'agriculture du Tripoli, Liban soutenu par la commission européenne au titre du septième programme-cadre FP7 et l'union européenne UE;
- l'hôtel Le TOURNANT;
- JA pour les services d'ingénierie mécaniques et électriques;
- plusieurs boutiques, agences de voiture etc.

## 1.3 Parmi nos Produits

### ❑ **Système de gestion de stock ACCUSTOCK;**

Ce système a pour noyau le code du composant de la gestion de stock fournit par notre projet SIGL. Il était développé au service de ma stratégie pour surmonter le faible retour de ce projet accepté dû à notre besoin d'une image de marque et un fort profile comme

une entreprise en démarrage. La chambre de commerce, d'industrie et de l'agriculture du Tripoli peut nous constituer un client de référence sur le marché local. Aujourd'hui, ce produit constitue l'un de nos produits génériques essentiels comme valeur ajoutée et chiffre d'affaires.

Il est divisé en cinq sous-systèmes essentiels qui sont la gestion de produits, la gestion de fournisseurs, la gestion de clients, la gestion de ventes et le générateur des rapports. Il est mené des plusieurs fonctionnalités intéressantes comme la surveillance et l'alarme des dates d'expiration et des limites de stock des produits, le suivi des commandes fournisseurs et leur réception et la gestion des dettes clients et dettes fournisseurs etc.

Pour plus d'informations, consultez-vous notre site web sur <http://www.adhene-accustock.com>.

#### ❑ **AGORA pour gestion des instituts éducationnels**

Ce système est disponible sous trois versions: base, avancée et personnalisée.

La version base couvre les processus opérationnels de l'école seulement, comme l'enregistrement des étudiants, la distribution des étudiants en groupe et des cours, la génération des cahiers des notes, la suivie des frais de scolarité etc. Cette version néglige presque tous les processus de support et de pilotage.

Alors que la version avancée permet la gestion d'un institut éducationnel de bout en bout couvrant tous les processus métiers (finance, ressources humaines, vacances, transports, librairie, clinique, cafétéria, annonces et messagerie, contrôle et suivie parental, préparation et coordination des examens etc.).

ADHENE offre la puissance de personnalisation à ses clients, elle donne toujours l'option de créer une version personnalisée convenable à la situation actuelle et au budget du client en lui donnant la capacité à sélectionner les composants désirés pour les ajouter à la plateforme de base à quel moment voulu. Pour plus d'informations, consultez-vous notre site web sur <http://www.adhene-agera.com>.

#### ❑ **AGARTHIA: un conte magique et éducationnel**

C'est une application mobile téléchargeable sur téléphones intelligents iOS et prochainement sur Android, il s'agit d'un conte amusant et éducatif. Cette application

est le premier épisode d'un certain nombre d'applications à venir sur une histoire du bien contre le mal. Il encouragera la pensée des enfants tout en leur apprenant à propos de la reconnaissance de l'environnement (terre, ville, forêt ...) et les valeurs morales. Il fait apprendre l'enfant des animaux et leurs sons. Notre histoire est un conte de fées qui va informer les enfants à propos des différentes créatures magiques pour enrichir leur créativité et leur imagination. Pour plus d'informations, consultez-vous l'application dans l'Apple Store sur

<https://itunes.apple.com/us/app/agarthia-tale-magic-education/id604869597?mt=8>.

#### ❑ **Système de gestion de pharmacie ACCUPHARM**

C'est un système spécialement conçu pour les pharmacies. Il couvrira tous leurs besoins pharmaceutiques d'une manière amicale, pas comme d'autres systèmes mis sur le marché, qui sont un système de comptabilité pur personnalisé pour usage pharmaceutique.

- ❑ Système de messagerie ACCUSEND;
- ❑ Système de gestion des ressources humaines
- ❑ Système de comptabilité;
- ❑ Système de gestion des clients;
- ❑ Système d'agence des voitures;
- ❑ Système de réservation d'une chambre à l'hôtel en ligne;
- ❑ et autres...

### **1.4 Etudiant CNAM comme entrepreneur**

Avec la crise économique et le chômage de masse qui en découle dans notre pays et surtout dans notre région et la faillite partielle ou totale de cinq entreprises informatiques que j'y travaillais AFAQ, PIOUS, EVISION, AST, INTELLIGILE, il me paraît évident qu'être entrepreneuse est presque ma seule occasion pour attribuer mes objectifs professionnels et être ingénieur CNAM, si je veux un boulot dans mon pays, je dois le créer moi-même.

BIAT, est un centre de développement des entreprises, financé par l'Union européenne et soutenue par le ministère de l'Économie, dont le but d'aider les entrepreneurs à traverser le défi



de démarrage d'une entreprise en offrant la consultation et assurant l'infrastructure nécessaire comme local, réseau et Internet.

BIAT nous a offert un environnement favorable au lancement de notre entreprise. D'autre part, j'avais la chance d'observer et vivre le collapsus de plusieurs entreprises, qui bien diversifient comme mission, structure hiérarchique, structure fonctionnelle, système technique, âge, taille, capital investi, environnement et culture, ce qui me donne l'occasion ultime d'analyser les causes de leur échec, apprendre de leurs défauts et régénérer un modèle consistant qui unit les points forts de toutes ces entreprises et évite leurs faiblesses pour mon entreprise.

#### **1.4.1 Structure hiérarchique**

Durant ma carrière, j'ai expérimenté les différentes structures hiérarchiques, allant de hiérarchie très pyramidale comme AFAAQ, puis pyramidale modérée comme PIOUS jusqu'à la structure plate avec un leader comme INTELLIGILE.

Dans mon entreprise, j'unis les deux structures plate et hiérarchique. J'applique la structure plate "entreprise sans hiérarchie" dans tous les départements reliés directement au processus de développement ce qui rend mon équipe autonome et bien productive loin des décisions hiérarchiques ascendantes et descendantes qui prend beaucoup de temps. L'idée est de prendre toutes les décisions par les opérationnels eux-mêmes qui connaissent bien et vivent la situation et les limites techniques, surtout que les membres du groupe autonome ne sont pas tous exactement au même niveau et éventuellement ils vont suivre le membre le plus professionnel et le considèrent automatiquement comme leader/manager pour les aider, faciliter le travail et résoudre les problèmes à la fois en interne (réunions, gestion des conflits, problèmes techniques...) et en externe avec les clients ou la direction, ce qui me rend le leader aisément par leur choix. De même, l'entreprise sans hiérarchie réduit bien considérablement les coûts de management, c'est ce qui est un paramètre significatif pour le succès d'une entreprise en démarrage.

D'autre part, je trouve l'application de la structure hiérarchique et bien nécessaire dans le département financier dû à la nature de leur travail, où la liberté ultime de décision peut causer

la perte de décision lui-même, la perte de l'alignement avec la stratégie et les objectives définies et le manque de sécurité de l'information financière.

#### 1.4.2 Culture et Environnement

D'après mon expérience, je trouve que dans une petite entreprise, surtout dans un pays ayant la structure sociale et politique du Liban, la diversité politique et culturelle peut être l'origine aux conflits et des incompréhensions qui reflètent directement sur la productivité et même sur l'existence de l'entreprise. Je travaille à créer une équipe presque homogène de point de vue politique et culturelle mais diversifié comme personnage et formation.

#### 1.4.3 Chiffres Optimistes

D'après L'Orient de Jour, Business Monitor International<sup>2</sup> me rend optimiste selon le progrès de mon entreprise. BMI a estimé une augmentation de 7,8 % de la taille du secteur des technologies de l'information en 2014, à 397.6 millions de dollars, en dépit des conséquences de la guerre en Syrie sur l'économie libanaise.

Toujours selon les estimations de BMI, la taille du marché devrait croître d'environ 11 % sur la période 2014-2017 pour atteindre les 560 millions de dollars en 2017. Parmi les forces du secteur au Liban, Business Monitor International a insisté sur les talents de la main-d'œuvre et son niveau de qualification. BMI a ajouté que le marché libanais est bien « positionné pour devenir un hub régional », ce dernier étant encore loin de la phase de saturation et ayant un potentiel important et rapide de croissance.

Notre objectif se résume par surmonter l'instabilité politique, le coût élevé et la mauvaise qualité des télécommunications, ainsi que les taux élevés de piratage, qui selon BMI constituent les faiblesses essentielles du secteur.

Le chapitre suivant sera consacré à la description du projet, sa clientèle tout en signalant mes responsabilités au cycle de développement du logiciel.

---

<sup>2</sup> **Business Monitor** est un important fournisseur indépendant de données exclusives, analyses, notes, classements et des prévisions couvrant 195 pays et 24 secteurs de l'industrie.

## Chapitre 2: A Propos du Projet

### 2.1 Objectif du projet

L'objectif du projet est de développer un système informatique de gestion de laboratoire (SIGL) automatisant tous ses processus opérationnels, de support et de pilotage. Ce système doit offrir une solution efficace et puissante de point de vue fonctionnalité et flexible de point de vue budgétaire, en répondant le mieux aux exigences de qualité et de traçabilité des organismes de contrôle et d'accréditation selon les normes ISO (ISO / IEC 17025, ISO 15189).

La solution doit être essentiellement conviviale et capable d'accommoder ses caractéristiques et fonctionnalités aussi facilement pour s'adapter rapidement au changement des exigences fonctionnelles du laboratoire pour satisfaire à la fois, les besoins immédiats et les besoins futurs comme ce projet possède une large et diverse clientèle.

Le système va enregistrer, gérer et stocker les données pour le laboratoire, permettant la traçabilité de toute demande et de tout échantillon, l'édition des résultats, la définition et contrôle des tests, la gestion des employés, des machines, des clients, du stock, des approvisionnements fournisseurs, des produits et des équipements utilisés... La solution est menée d'un système de rapports appuyé sur le principe du WYSIWYG (What you see is what you get), un écran de surveillance (monitoring) pour les clients en site et sur web, un système d'archivage en format pdf.

La solution doit assurer un tableau de bord (dashboard) pour l'évaluation de la performance et le pilotage des processus opérationnels du laboratoire dans le but d'aider la direction à orienter leurs décisions pour atteindre de mieux les objectifs stratégiques du laboratoire et intervenir avec des solutions tactiques rapides quand on a besoin.

Consultez-vous dans la page suivante une infographie illustrant le projet et son cycle de production en résumant les difficultés rencontrées, l'architecture adoptée, les technologies de modélisation, de développement et de gestion utilisées etc.

## 2.2. Infographie du Projet

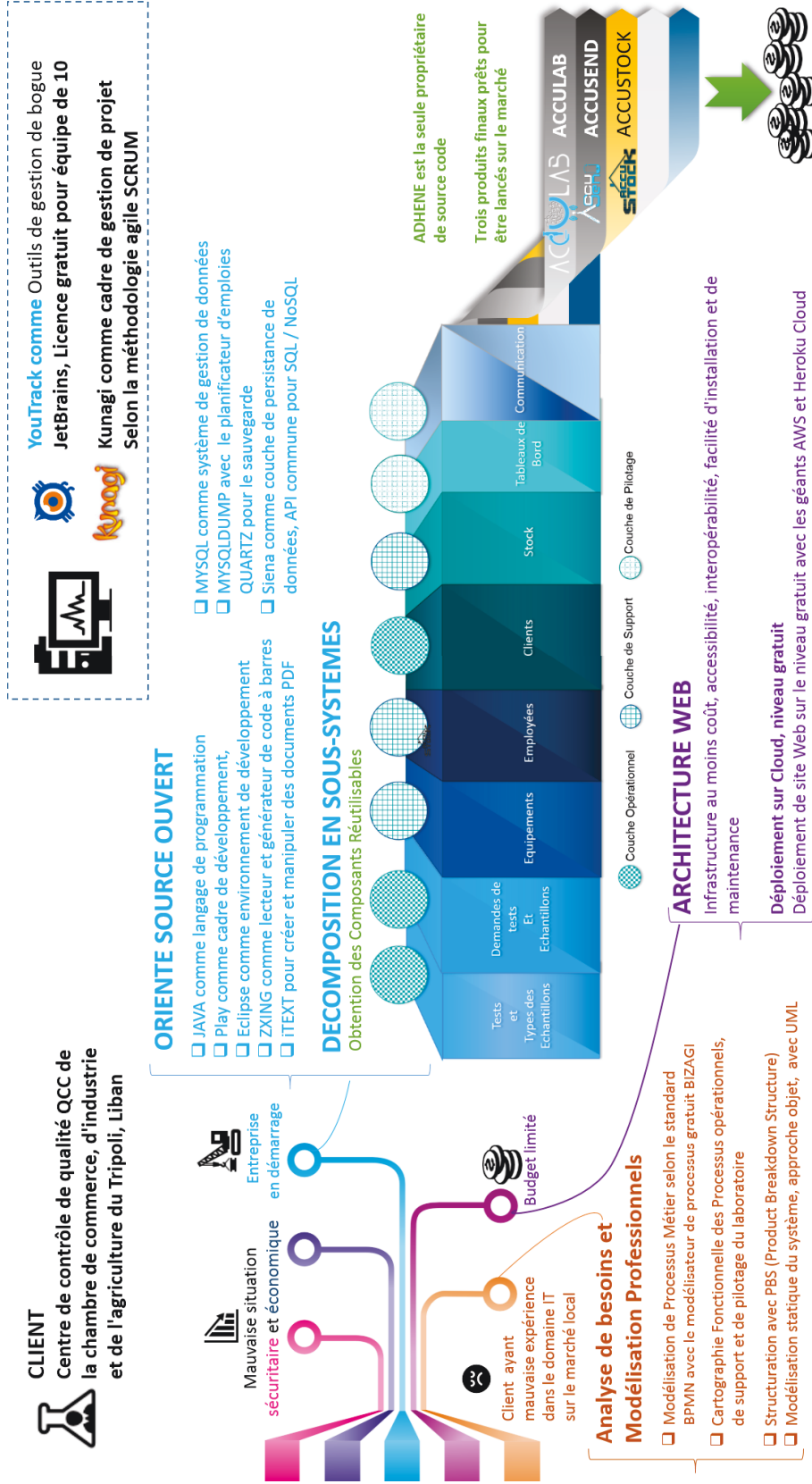


Figure 1. Infographie du projet

## 2.3 Mes Responsabilités

Après mon expérience professionnelle, j'ai conclu que les facteurs clés de succès de l'ingénieur informatique se résument par sa capacité à s'adapter aux changements rapides de la technologie et être autodidacte et par son pouvoir d'abstraction qui lui permet de résoudre un problème complexe par le diviser, extraire l'essentiel de, le modéliser et le communiquer aux autres. D'autre part, l'ingénieur doit exercer une «gestion de soi» efficace avant tout pour avoir l'habilité d'être un leader, diriger et motiver une équipe.

En tant de chefs de projet et comme architecte du système informatique chez ADHENE, mes responsabilités essentielles dans ce projet sont:

- ❑ rédiger le cahier des charges en formalisant les problèmes à résoudre et établissant le cadre du projet;
- ❑ définir la cartographie du système en favorisant la vision métier qui décrit les processus cœur, support et pilotage que le système doit supporter, la vision fonctionnelle et organisationnelle qui permet la structuration de l'ensemble des informations et des fonctions et enfin la vision technique qui définit l'infrastructure de développement et les choix technologiques en utilisant UML et BPM;
- ❑ planifier le projet en matière de ressources humaines tout en respectant le mieux la relation triangulaire temps, coût et qualité, et élaborer le calendrier de production;
- ❑ établir l'environnement de développement comme langages et techniques de programmation, outils de développement, de communication et de collaboration d'équipe et enfin comme stratégie de déploiement et de distribution d'application mobile côté client et serveur;
- ❑ assurer l'ergonomie du système par améliorer la convivialité et l'efficacité des interfaces homme-machine du système pour une meilleure performance; selon le standard HCI (Human Capital Interaction);
- ❑ superviser, supporter, motiver et aider l'équipe de développement à apprendre les nouveaux techniques nécessaires au cycle de développement et identifier ses besoins de formation faire face aux problèmes qu'ils surviennent (le chemin à l'informatique n'a jamais exécuté en douceur surtout pour un nouveau type de développement);

- ❑ orienter les développeurs à effectuer le développement en respectant le cahier des charges et les spécifications;
- ❑ être l'interfaçage de l'équipe de développement avec l'administration et l'équipe d'affaire;
- ❑ superviser l'ensemble du projet, de la phase d'avant-vente au support;
- ❑ assister à la rédaction de la documentation destinée aux utilisateurs du système;
- ❑ collaborer à la phase de programmation quand j'ai quelque temps libre pour accélérer le déroulement de travail.

## 2.4 Clientèle du projet

### 2.4.1 Client direct

Notre client direct de ce projet est le centre de contrôle de qualité QCC de la chambre de commerce, d'industrie et de l'agriculture du Tripoli, Liban.

La sécurité alimentaire représente un obstacle majeur à la compétitivité du secteur de l'agriculture au Liban. Elle empêche des centaines de producteurs agro-alimentaires de petite, moyenne et grande échelle (scale) de fournir des produits qui conforment aux normes internationales exigées pour l'exportation vers d'autres pays. Pour résoudre ce problème, l'USAID a travaillé avec les chambres de Commerce, d'industrie et d'agriculture de Zahlé, Tripoli et Saïda pour mettre à niveau la certification, la capacité et la qualité des trois laboratoires agricoles libanais. Chacun de ces trois laboratoires sont maintenant certifiés ISO 17025 permettant aux agros-producteurs et aux transformateurs d'analyser les nouveaux produits, d'assurer la sécurité des produits et de recevoir un sceau d'approbation internationale et bien reconnu.

Le centre de contrôle qualité (QCC) a été créé pour tester les produits alimentaires et pour améliorer leur qualité, il effectue des analyses microbiologiques, physico-chimiques et de contrôle de la qualité des aliments et de l'eau, pour évaluer la conformité aux normes et aux spécifications telles que celles de LIBNOR<sup>3</sup>, ISO, FDA, UE, SASO etc. Le laboratoire est déjà en

---

<sup>3</sup> LIBNOR : L'institution libanaise de normes LIBNOR est une institution publique liée au ministère de l'industrie. Elle a été établie par une loi le 23/07/1962 comme étant la seule autorité d'établir des normes nationales et de donner le droit d'utiliser la marque de conformité libanaise. Les normes nationales couvrent tous les produits, y compris

conformité avec la norme ISO 9001: 2000: système de gestion de la qualité et à la norme ISO 17025: gestion de l'accréditation des laboratoires.

#### 2.4.2 Client Indirect

Les centres de contrôle de qualité de la chambre de commerce, d'industrie et de l'agriculture du Zahlé et Saida, Liban nous constituent des clients potentiels faciles comme ils partagent les mêmes processus métiers de notre client, QCC de Tripoli sous la même initiative d'USAID.

De plus, tout laboratoire dans l'industrie agroalimentaire, l'industrie pétrolière, ainsi que tous laboratoires d'hôpitaux et des centres médicaux dans le marché local et arabe peut être un potentiel client.

Selon le SWEBOK<sup>4</sup>, les activités-clés du cycle de vie d'un logiciel sont l'analyse fonctionnelle, l'architecture, l'implémentation, la gestion de projet, les tests, la validation et la maintenance. Les chapitres suivants abordent successivement tous les phases du SWEBOK, on va détailler toutes ses activités successivement comme moyen idéal pour décrire un projet. Commençons tout de suite par l'analyse des besoins dans le chapitre suivant.

---

l'agro-alimentaire, les produits chimiques, les méthodes d'essai et des codes de pratique. LIBNOR est un membre de « l'Organisation Internationale de Normalisation (ISO) », de « Développement Industriel Arabe ».

<sup>4</sup> Le dispositif **SWEBOK** (Software Engineering Body of Knowledge) est une initiative de l'IEEE-Computer-Society (Institute of electrical and electronics engineers) piloté par le laboratoire de recherche en génie logiciel de l'université du Québec à Montréal et l'École de technologie supérieure. Le SWEBOK est le document de base de l'IEEE-Computer-Society pour la normalisation en ingénierie du logiciel.

## Chapitre 3: Analyse des Besoins

Cette phase consiste à récolter les informations nécessaires, définir les besoins et les fonctionnalités correspondant au système prévu et s'assurer qu'ils sont à l'attend de l'ensemble des utilisateurs et de clients cibles tout en respectant les diverses contraintes légales et budgétaires. Le but de cette étape est également de concevoir un système de gestion de laboratoire de qualité aligné avec la norme ISO/CEI 9126.

### 3.1. Modélisation de Processus Métiers

D'après mon expérience et après ma collaboration à un nombre bien considérable des projets lors de ma carrière professionnelle, j'ai trouvé que parmi les causes primaires d'échec d'un projet est que le maître d'œuvre est incapable à bien définir ses besoins et les communiquer. De même nous considérons toujours les déclarations du client comme quelques choses sacrées et plutôt ce n'est pas le cas. Je crois que notre responsabilité doit dépasser la réception passive des informations à la recherche d'information même à offrir l'aide nécessaire à notre client pour réviser, rectifier et optimiser ses processus métiers, ceux de processus opérationnels et de support, parce qu'un processus non performant ou souffrant d'une défaillance sur terre ne peut jamais aboutir à une solution informatique d'automatisation réussie.

D'où et d'après les impacts majeurs de la phase d'analyse fonctionnelle sur tout le cycle de développement du projet informatique, je recours aux modélisations de processus métier avec le standard BPMN. Ce standard propose une notation facilement compréhensible par la clientèle, les développeurs techniques et les directeurs commerciaux qui suivent et gèrent les processus. BPMN est prévu pour servir comme langage visant à combler un déficit de communication qui survient souvent entre la conception des processus métier et l'implémentation. De même, il fournit aux entreprises la capacité de comprendre leurs procédures internes dans une notation graphique et donnera aux entreprises la possibilité de communiquer ces procédures d'une manière standard avec le maître d'ouvrage, toute autre entreprise ou prestataire de services d'où notre client a souvent l'enthousiasme d'appliquer notre approche en analyse de besoins.



D'après ce qui précède, on recourt à l'élaboration d'une cartographie du système, car elle assure au lecteur le maximum de clarté et de rapidité de compréhension comme elle utilise le pouvoir du langage visuel.

### 3.2. Cartographie Fonctionnelle des Processus

L'utilisation d'une approche processus peut conduire à élaborer une cartographie des processus qui permet de représenter l'entreprise à travers la modélisation de ses processus opérationnels, de support et de pilotage.

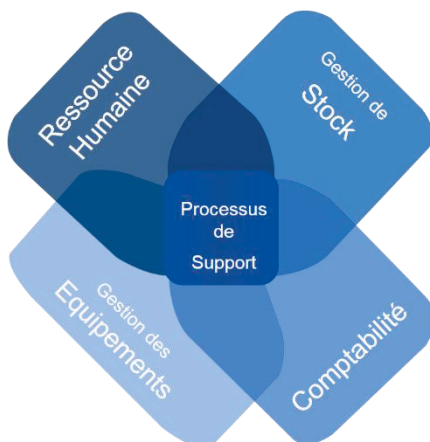
#### 3.2.1. Les processus opérationnels du laboratoire



La gestion des tests et des types d'échantillons, le management des demandes de test et des clients représentent ensemble l'activité cœur de tout laboratoire. Donc nous avons concentré principalement sur l'identification et l'étude de ces activités clés pour rendre responsable de l'efficacité et la bonne performance du laboratoire

Figure 2. Les processus opérationnels du laboratoire

#### 3.2.2. Les processus de support du laboratoire



Les processus de support mettre à la disposition du laboratoire les ressources nécessaires à la réalisation des processus opérationnels. Pour tout laboratoire, les processus de support couvrent principalement la gestion de stock, de ressource humaine, des équipements et la Comptabilité, etc.

Figure 3. Les processus de support du laboratoire

### 3.2.3. Les processus de pilotage du laboratoire

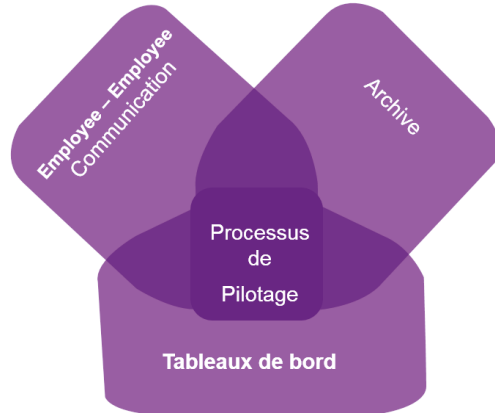


Figure 4. Les processus de pilotage du laboratoire

Les processus de pilotage représentent l'activité d'élaboration des informations internes permettant le pilotage de l'activité de l'entreprise. Pour tout laboratoire, les processus de pilotage sont basés principalement sur les tableaux de bord, l'archive des documents, et système de messagerie et communication interne employé – employé etc.

### 3.3. Acteurs du Système

La modélisation des processus aboutit à clarifier les rôles et les responsabilités exercées dans l'entreprise et par conséquent, à définir les acteurs du notre système.

Les acteurs du système sont :

- administrateur ;
- technicien ;
- auditeur ;
- réceptionniste ;
- client

Dans la page suivante, on va schématiser le processus opérationnel principal du laboratoire, le processus de demande de test (figure 5). On constate que ce processus est un macro-processus comportant plusieurs sous-processus (figure 6) collaborant à leur exécution qui sont :

- saisie de l'information du client demandeur du test;
- saisie de l'information de l'échantillon;
- saisie les détails de transaction

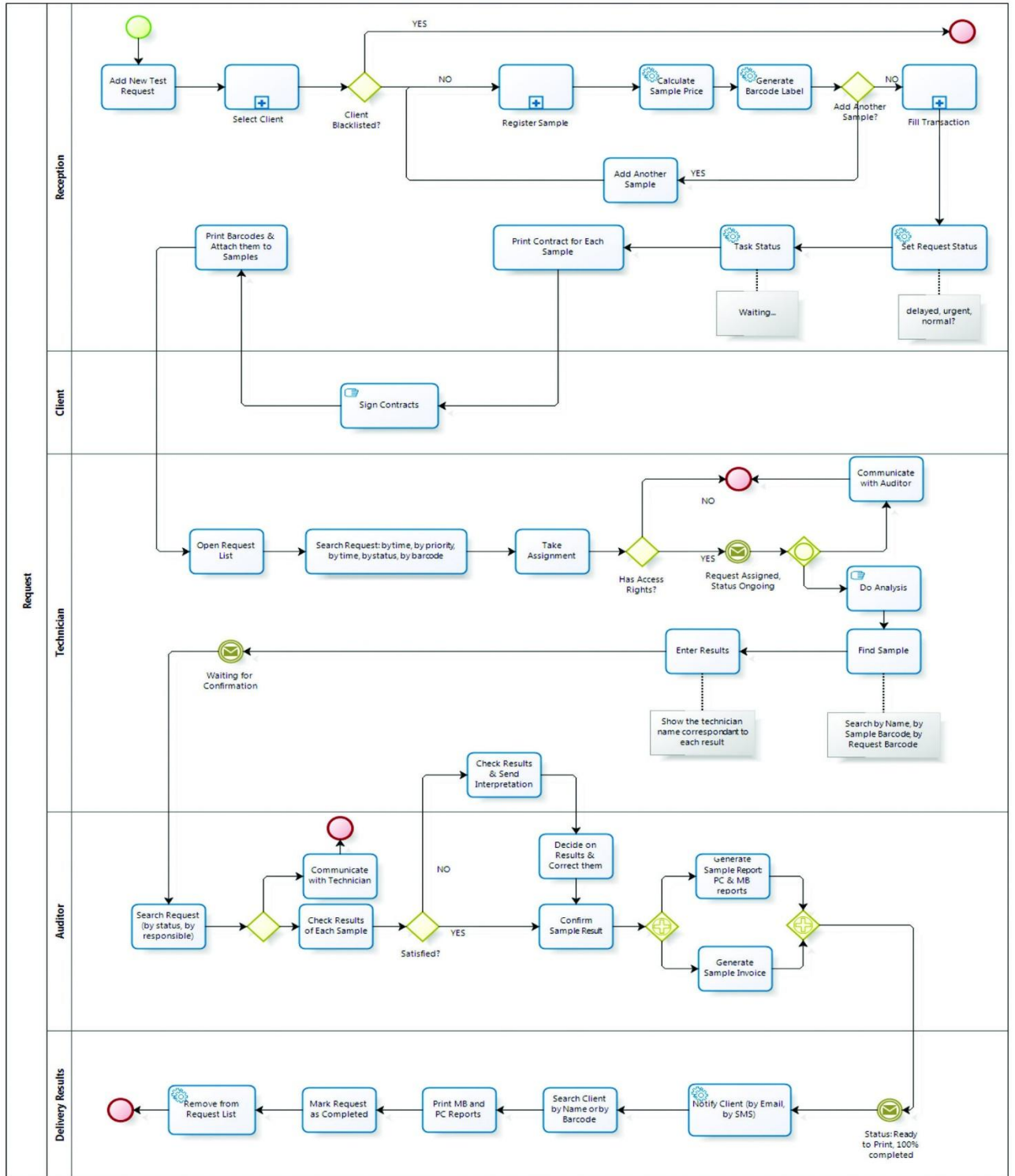
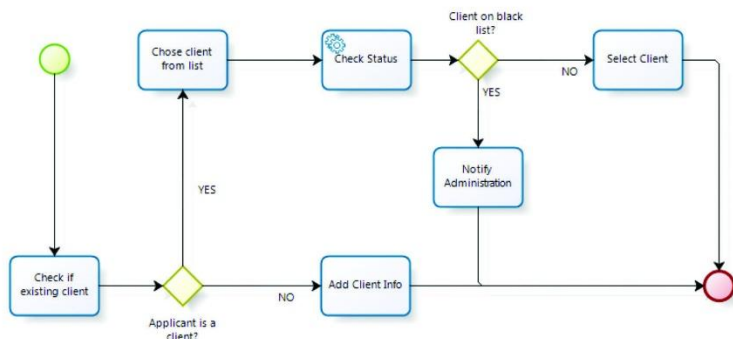
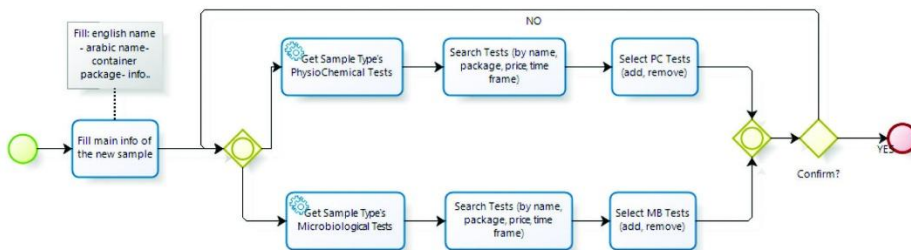


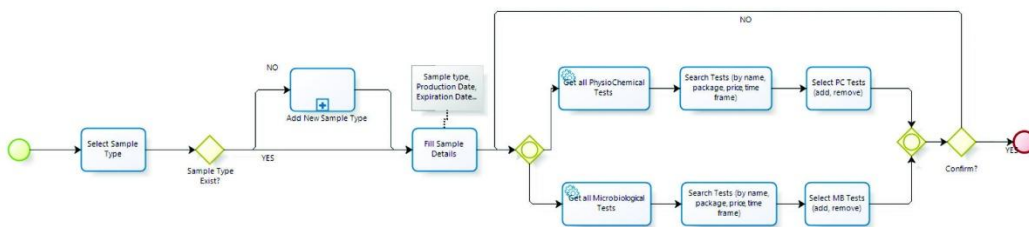
Figure 5. Processus de saisir d'une demande de test



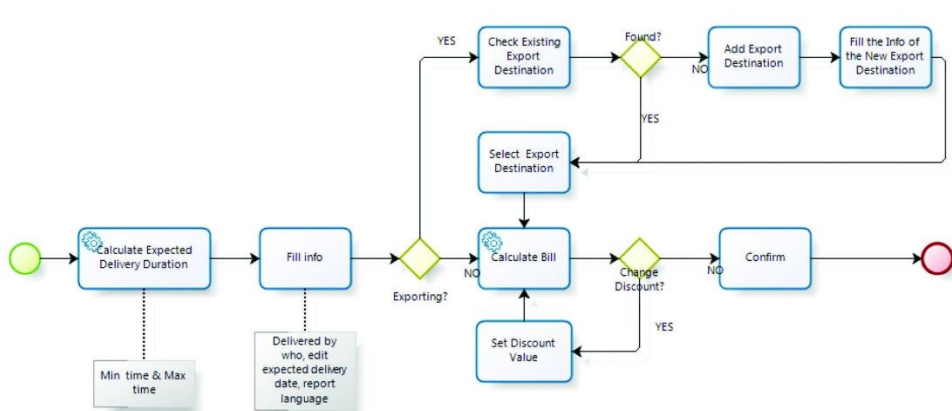
Saisie de l'information du client demandeur du test



Saisie de l'information de l'échantillon



Sélection du type d'échantillon



Saisir les détails de transaction

Figure 6. Les sous-processus du processus de saisir d'une demande de test

### 3.4. Outils de Modélisation Utilisé

Après recherche, j'ai choisi BIZAGI comme outil de modélisation. C'est un modélisateur de processus gratuit. Il sert à schématiser et à documenter les processus d'une entreprise d'une manière rapide et facile. Il est aligné avec le standard BPMN. Le BIZAGI est un membre actif du groupe en charge de la définition de la norme dans l'OMG<sup>5</sup>. Il est mené d'une documentation riche, d'un tutoriel vidéo, des puissantes options de formatage et d'un mécanisme de simulation.

### 3.5. Architecture Basé sur Web

Choisir l'architecture appropriée pour l'application est la clé de son succès. L'étude des projets en difficulté a montré que l'équipe de développement n'utilise pas de meilleures pratiques bien connues et l'architecture la plus appropriée.

Une application Web est une application qui utilise un site comme interface utilisateur. Les utilisateurs accèdent à l'application depuis n'importe quel ordinateur connecté au réseau Internet ou intranet à l'aide d'un navigateur standard, au lieu d'utiliser une application qui a été installée sur leur ordinateur local.

J'ai choisi cette architecture car les applications Web offrent une série d'avantages d'affaires sur les applications de bureaux traditionnels de point de vue coût, accessibilité, mobilité, personnalisation, interopérabilité et facilité d'installation et de maintenance.

Le budget était consacré à mon projet représente un obstacle essentiel devant son succès d'où je dois étudier tout moyen m'aidant à diminuer les dépenses et le coût de développement. C'est pourquoi cette architecture me paraît la plus convenable de point de vue coût parce que les utilisateurs accèdent au système via un environnement uniforme qui est le navigateur, alors l'application a un code unifié sur toutes les versions et configurations de systèmes d'exploitation possibles. D'autre part l'installation et la maintenance avec cette architecture deviennent moins

---

<sup>5</sup> OMG Acronyme de "Object Management Group", c'est une adhésion ouverte internationale, ayant des normes technologiques non lucratives. Fondée en 1989, les normes OMG sont entraînées par les fournisseurs, les utilisateurs finaux, les établissements universitaires et les organismes gouvernementaux. Les normes de modélisation d'OMG, y compris l'Unified Modeling Language (UML), Business Process Modeling BPM et Model Driven Architecture (MDA) etc.

compliquées et coûteuses, car une fois la nouvelle version est installée sur le serveur hôte, tous les utilisateurs peuvent accéder immédiatement. Il n'est pas nécessaire de mettre à jour chaque client, un professionnel expérimenté travaillant sur le serveur hôte et bien suffisant, ce qui minimise le coût de maintenance côté client.

### **3.5.1. Personnalisation et Flexibilité**

L'interface utilisateur des applications basées sur le Web est plus facile à personnaliser que dans les applications de bureau. Cela rend plus facile la personnalisation de l'information à différents rôles d'utilisateurs selon l'évolution des processus d'affaires (Business Processes) appliquées et le changement des règles d'affaires (Business Rules) avec l'évolution de la maturité du client.

### **3.5.2. Importance de Mobilité**

Nous sommes dans l'ère de mobilité, les systèmes web doivent être accessibles à tout moment, n'importe où, donc le contenu doit également être personnalisé sur n'importe quel appareil connecté au système, à travers l'internet ou/et l'intranet, y compris PDA, téléphones mobiles, pc, etc. Ce qui devient aujourd'hui facile avec l'HTML5 et CSS3 pour une application basée sur web.

### **3.5.3. Amélioration de l'Interopérabilité**

En utilisant les techniques web, il est possible de parvenir à un niveau beaucoup plus élevé d'interopérabilité entre les applications que les systèmes de bureau isolés. Cette architecture nous permet d'intégrer rapidement les systèmes déjà utilisés par notre client ou ajouter au futur nos produits génériques à l'application de base, en faveur des processus du support (ressource humaine, comptabilité, surveillance des caméras, documentation...) et renforce notre possibilité à recourir aux services tertiaires pour attribuer notre vision d'un produit complet automatisant les processus d'un laboratoire de A à Z.

### **3.5.4. Adaptation à l'Augmentation de la Charge de Travail**

Aujourd'hui le laboratoire cible, le client direct du projet, est en démarrage et mené par une petite équipe de dix employés permanentes, mais je dois prendre en considération son expansion rapide. Avec cette architecture, l'augmentation de la capacité peut avoir lieu par

l'augmentation de puissance de serveur hôte, "regroupement" des serveurs, ou exécution du logiciel sur plusieurs serveurs simultanément quelle que soit la capacité des machines client.

### **3.5.5. Déploiement Facile et Sécurité**

Notre client utilise le réseau de la chambre de commerce donc mon application (basé web) sera déployé sur des serveurs dédiés, qui sont surveillés et entretenus par les administrateurs systèmes, réseaux et sécurité de la chambre. C'est beaucoup plus efficace et moins coûteux.

Cette architecture nous ouvre l'horizon vers «logiciel en tant que service» SaaS<sup>6</sup> en cassant tous les limites devant l'effort de commercialisation du projet pour attribuer un maximum profit au futur.

Le chapitre suivant sera consacré à la modélisation et conception des exigences fonctionnelles accumulées dans cette phase.

---

<sup>6</sup> Acronyme de Software as a Service (SaaS) est un modèle d'exploitation commerciale des logiciels dans lequel ceux-ci sont installés sur des serveurs distants plutôt que sur la machine de l'utilisateur.

## Chapitre 4: Conception et Modélisation

Je crois que les concepteurs devraient également coder. Ils ne doivent jamais s'isoler et supposer que leur conception fonctionne, ils doivent également être en mesure de le prouver avec le code. Les concepteurs devraient chercher activement des informations sur le projet et être prêts à examiner et à agir en conséquence.

Les deux objectifs principaux de toute conception logicielle sont la réduction des coûts et l'augmentation de la qualité du système; la réduction des coûts est principalement réalisée par la réutilisation et par la diminution du temps de maintenance d'où je recours à la stratégie de décomposition « diviser pour régner ».

Dans notre découpage en modules, on a pris en considération les deux bonnes conceptions en génie logiciel, le couplage et la cohésion, pour avoir les meilleurs liens de dépendance entre les modules en assurant le plus faible couplage et la plus forte cohésion possibles.

### 4.1. Cartographie Applicative du Système

La cartographie applicative représente le niveau du « comment », elle succède la cartographie fonctionnelle comme on ne peut bien comprendre le niveau du « comment » que lorsque le niveau du « quoi » est bien établi.

Cette étape consiste principalement à décliner les processus modélisés en modules et composants du système.

Les modules résultant de cette décomposition sont schématisés par le diagramme de la structure de décomposition d'un produit PBS (Product Breakdown structure) détaillé ci-dessous.

#### **PBS (Product Breakdown Structure)**

Le PBS me représente l'outil principal de structuration puisqu'il constitue les livrables du projet et me permet d'identifier toutes les fonctions qui composent le produit à livrer et d'identifier avec le client, sous forme graphique, les différents livrables du projet. Ce diagramme aide à maîtriser les itérations du projet et à orienter le déroulement de développement.



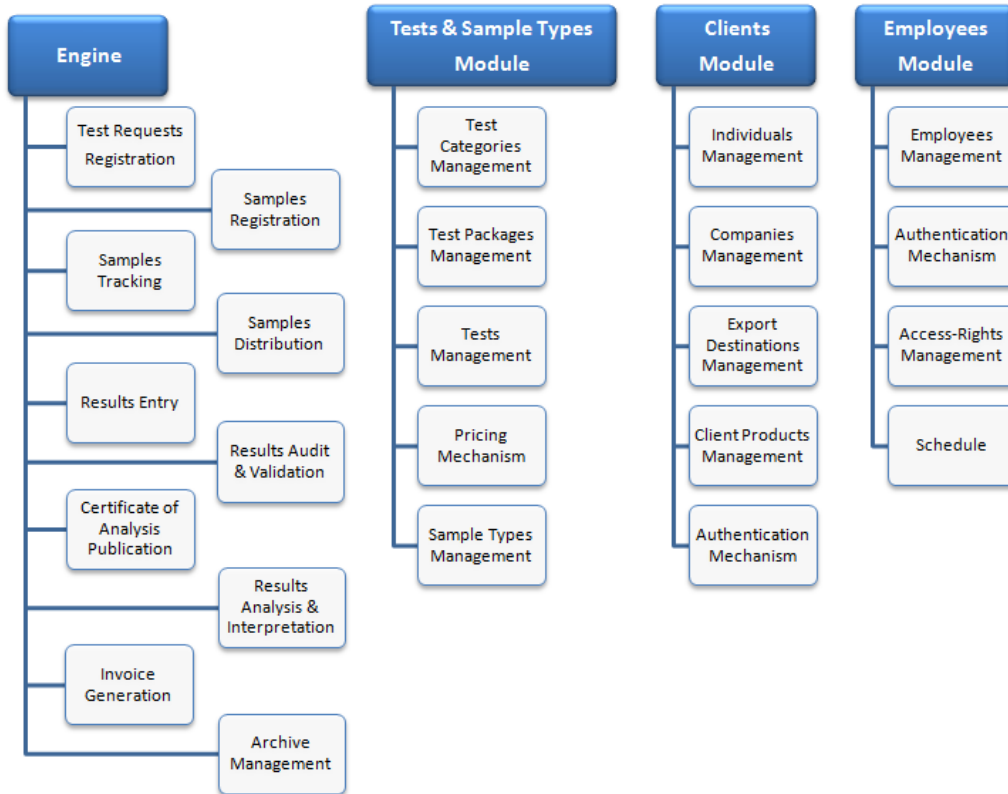


Figure 7. PBS diagramme du système - partie 1

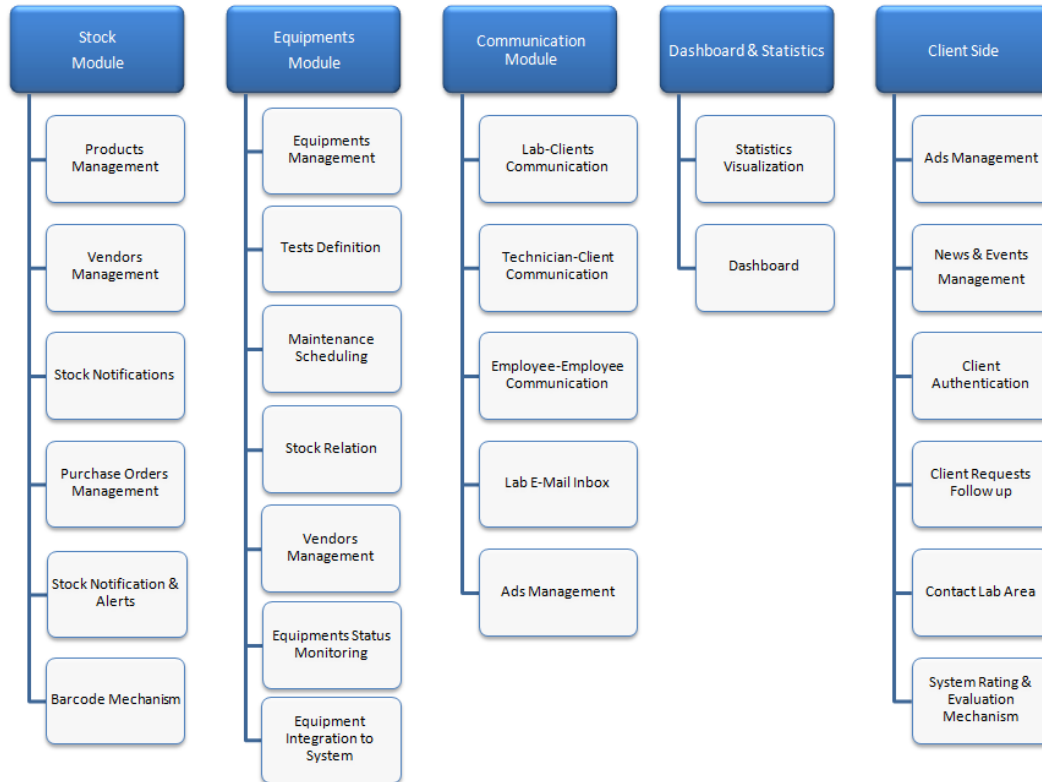


Figure 8. PBS diagramme du système - partie 2

## 4.2. Découpage du système en sous-systèmes

Due à la complexité et à la taille de notre projet, il me paraît nécessaire de découper le système en sous-systèmes, selon le paradigme « diviser pour régner », pour garantir la récolte des besoins et des exigences fonctionnelles du maître d'ouvrage, faciliter et simplifier l'étude, la conception, l'implémentation et la maintenance du projet. Également, ce paradigme supporte le cycle de développement itératif et la méthodologie SCRUM de gestion que nous adaptons.

Les différents sous-systèmes seront schématisés par la figure informative (figure 9) ci-dessous :

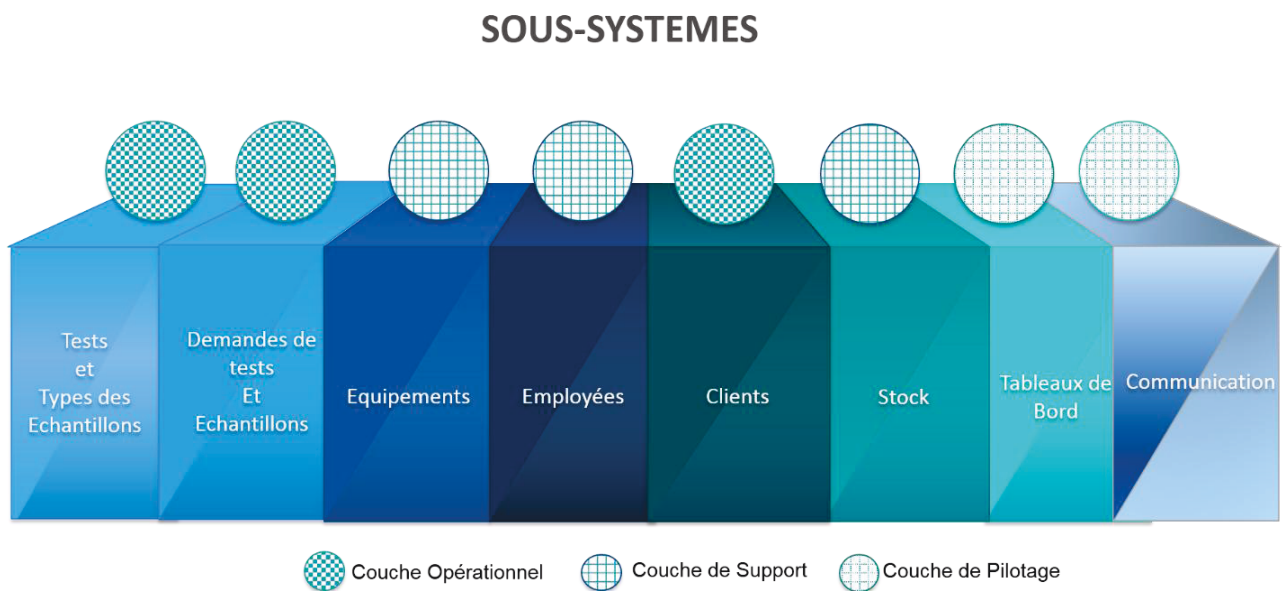


Figure 9. La division en sous-systèmes

Pour plus de détails à propos de subdivision du système, consultez le diagramme UML de division aux sous-systèmes dans la page suivante :

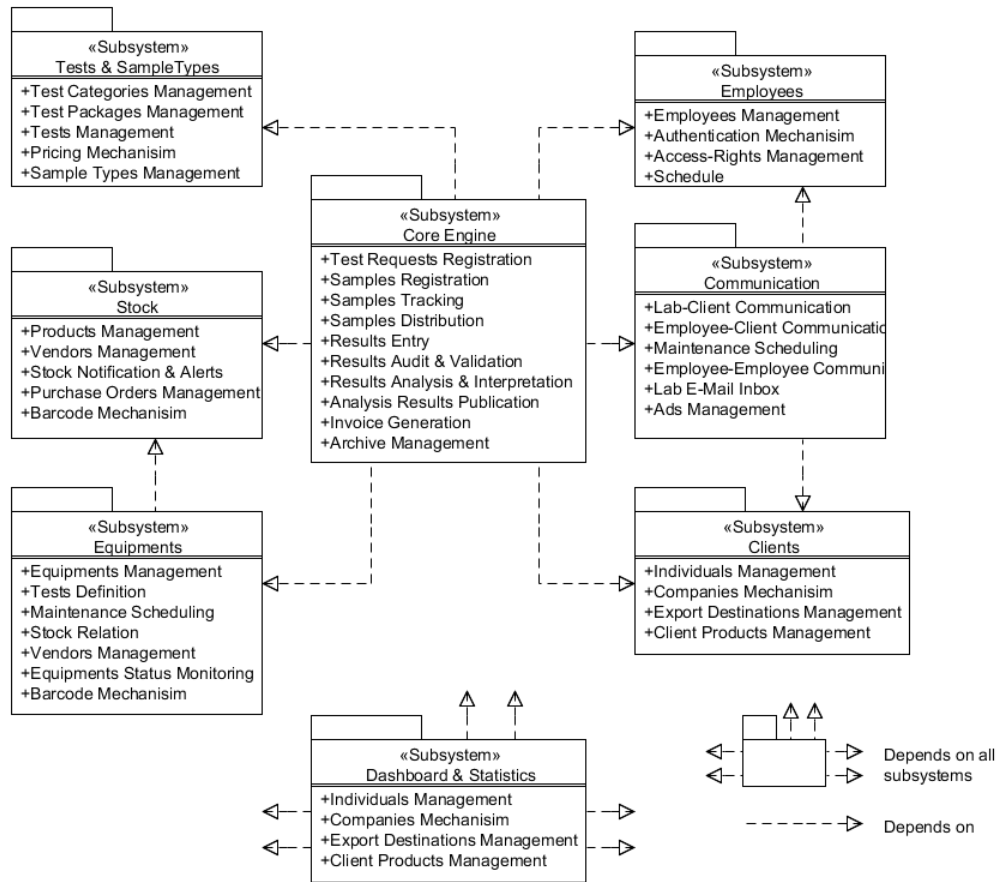


Figure 10. Diagramme de division aux sous-systèmes

Pour une étude cohérente et complète, on doit accomplir l’approche processus fonctionnelle par l’approche objet, assuré par la modélisation statique du système en UML. Cette approche consiste à traiter les données et les informations nécessaires aux fonctionnalités et les encapsuler sous formes des objets bien définis, décrites par des classes, partageant les responsabilités du système à travers un ensemble bien déterminé de données et fonctions.

Le diagramme de classes est considéré comme le diagramme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation. Il présente les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Les classes sont utilisées dans la programmation orientée objet. Elles permettent de modéliser un programme et ainsi de découper une tâche complexe en plusieurs petits travaux simples.

Maintenant, on va détailler les sous-systèmes résultants de la subdivision comme fonctionnalités, données et structure. Les sous-systèmes seront évoqués selon la relation d'interdépendance et le facteur priorité pour attribuer une meilleure compréhension de flux du système.

#### 4.2.1. Sous-système de Tests et Types Des échantillons

Ce module permet la définition et la gestion des tests que le laboratoire peut réaliser et la définition et la gestion des types d'échantillons (produits) qu'il a la capacité à les tester comme infrastructure, équipements et ressources humaines.

Les fonctions de ce module seront schématisées par la figure ci-dessous :

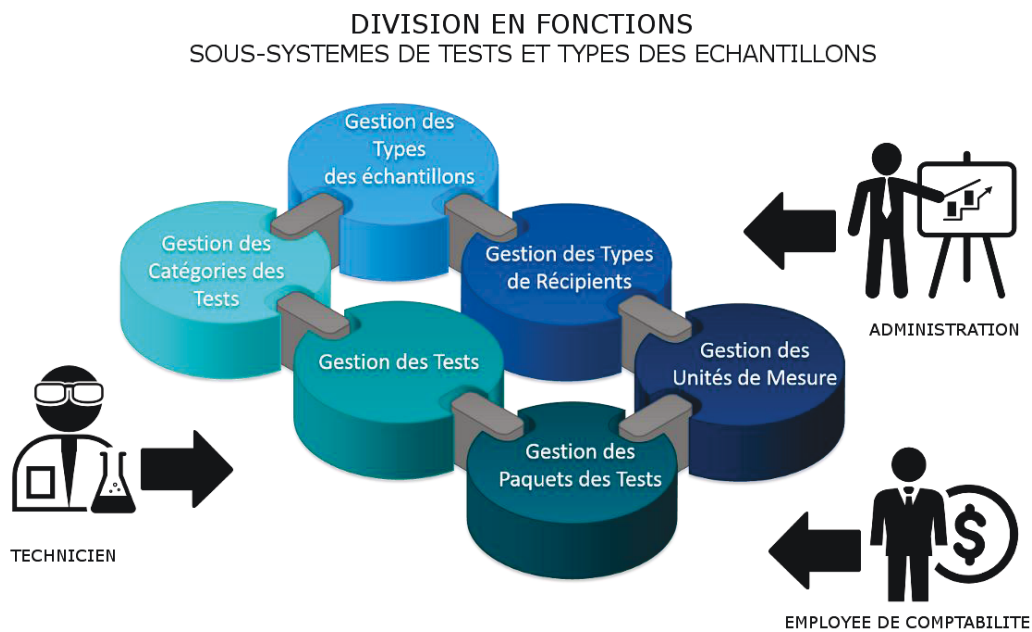


Figure 11. Division en fonctions de sous-système de tests et de types des échantillons

L'identification des couches de données dans ce module est obligatoire pour avoir un scénario robuste et correct dans la manipulation des tests définis et un flux non-interruptif.

La figure 12 dans la page suivante schématise clairement ces couches pour vous assurer une meilleure compréhension du flux de données de ce sous-système.

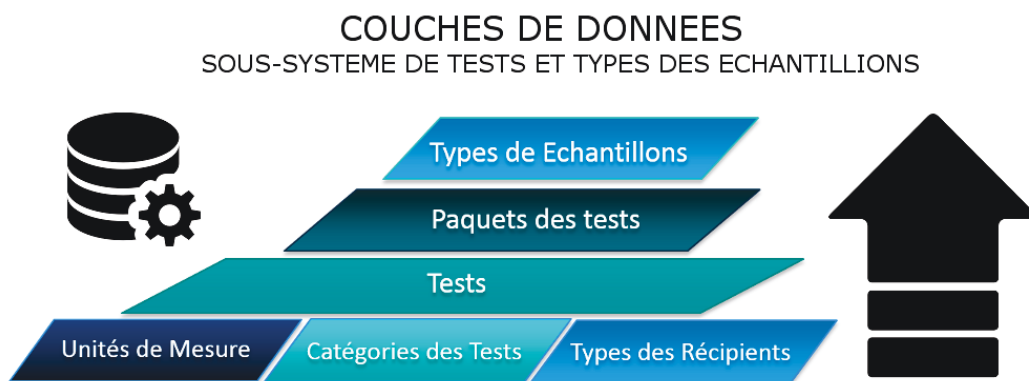


Figure 12. Couches de données du sous-système de tests et de types d'échantillons

Avec ce module, on peut définir un nouveau test, modifier, activer ou désactiver un. Les tests peuvent être regroupés sous des paquets et des catégories pour faciliter leur manipulation de point de vue technique, administratif et commercial. Les différentes fonctions de ce sous-système seront détaillées brièvement ci-dessous :

#### **Fonction 1 : Gestion des catégories des tests**

- ajouter, modifier et supprimer une catégorie;
- spécifier le escompte y appliqué;

#### **Fonction 2 : Gestion des unités de mesure**

- ajouter, modifier et supprimer une unité de mesure en faveur de test et échantillons (g, g/L ml...);

#### **Fonction 3: Gestion des tests**

- ajouter, modifier et supprimer un test;
- activer ou désactiver un test;
- spécifier la catégorie et le paquet à qui il correspond;
- spécifier le coût d'un test;
- chercher un test par nom, catégorie et/ou paquet;

#### **Fonction 4 : Gestion des paquets**

- ❑ ajouter, modifier et supprimer un paquet;
- ❑ activer ou désactiver un paquet;
- ❑ spécifier le coût et l'escompte du paquet;
- ❑ gérer les tests composants, ajouter et enlever un tes du paquet;

#### **Fonction 5 : Gestion des types des récipients**

- ❑ ajouter, modifier et supprimer un récipient;
- ❑ activer ou désactiver un paquet;
- ❑ spécifier le coût et l'escompte du paquet;
- ❑ gérer les tests composants le paquet;

#### **Fonction 6 : Gestion des types des échantillons**

- ❑ ajouter, modifier et supprimer un type d'échantillon;
- ❑ activer ou désactiver un type;
- ❑ gérer les tests nécessaires au test de ce type d'échantillon;
- ❑ identifier les tests obligatoires et les tests optionnels, et spécifier pour chaque test l'unité de mesure, les limites, minimale et maximale, d'acceptation correspondant à ce type;
- ❑ chercher un type par nom;

Dans la page suivante, Consultez-vous le diagramme de classe (figure 13) présentant les classes et les interfaces de ce sous-système ainsi que les différentes relations entre celles-ci en faveur de la programmation orientée objet et le découpage des fonctions en tâches lors de l'implémentation.

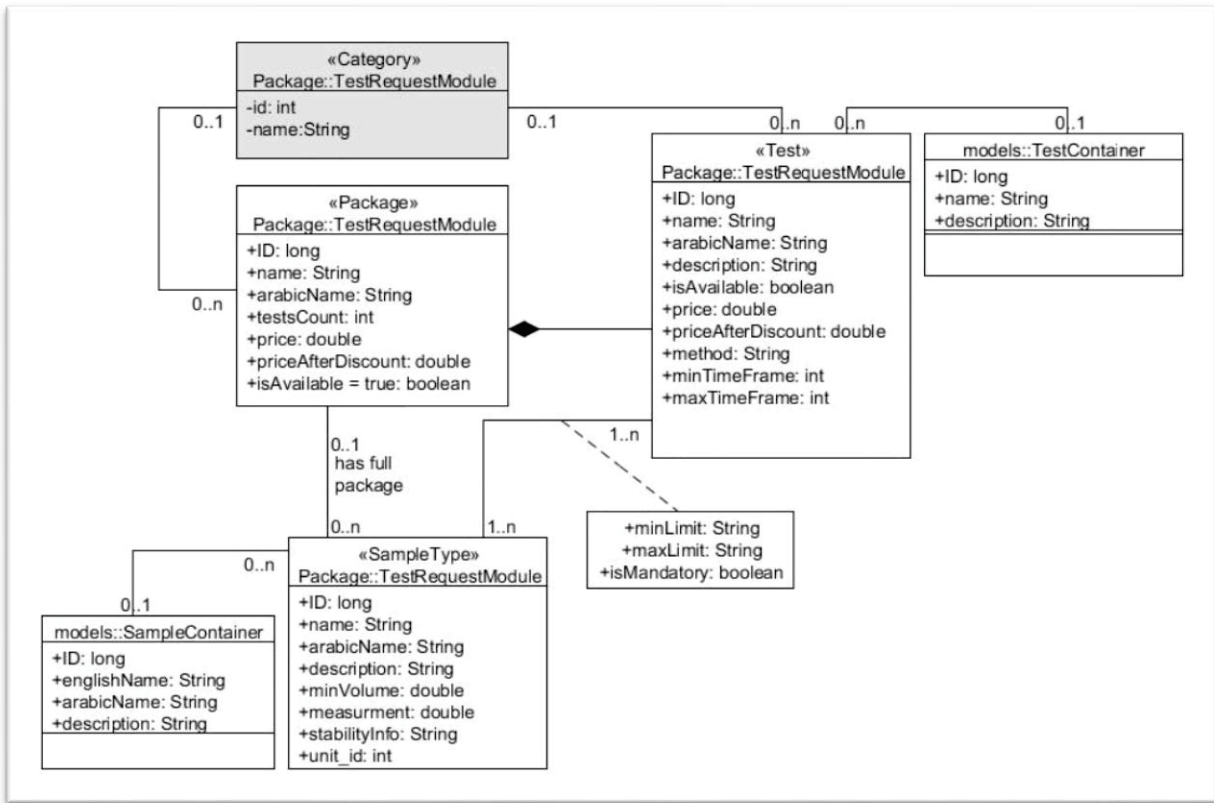


Figure 13. Diagramme de classe de Sous-système de Tests et Types Des échantillons

#### 4.2.2. Sous-système Clients

Le laboratoire a de nombreux clients et il doit répondre soigneusement aux besoins de chacun. Le but essentiel de ce module est de personnaliser toute demande de test selon la recommandation des organismes de contrôle et d'accréditation. Cette personnalisation aide à optimiser le processus de demande par la réduction des pertes de temps dues à la saisie répétée de l'information de client et à la réduction de marge d'erreur due à la saisie erronée de ses coordonnées critiques. On a deux types de client, personne et entreprise, chacun ayant ses propriétés reliées à son contexte de demande. Ce module permet la gestion des dossiers clients. Chaque client à un dossier contenant ses coordonnées personnelles, ses informations de contact, l'historique de ses transactions pour assurer au laboratoire une traçabilité complète et une meilleure performance commerciale. Sûrement un système de recherche avancée est mené. Les fonctions de ce module seront schématisées par la figure (14) à la page suivante :

## DIVISION EN FONCTIONS SOUS-SYSTEME DES CLIENTS

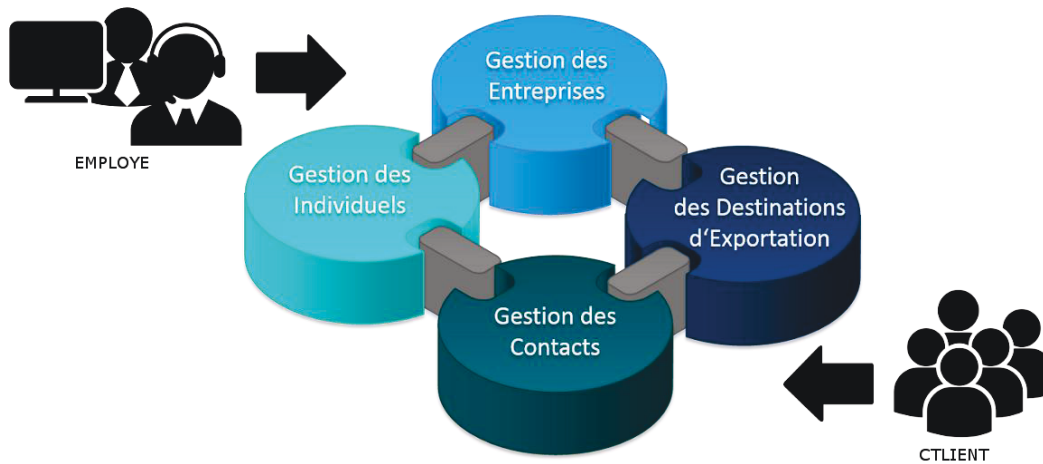


Figure 14. Division en fonctions de sous-système de tests et de types des échantillons

Les fonctions de ce sous-système sont brièvement:

### Fonction 1 : Gestion des individus

- ajouter, modifier et supprimer un individu;
- chercher un individu par nom ou numéro de téléphone;

### Fonction 2 : Gestion des Entreprises

- ajouter, modifier et supprimer une entreprise;
- chercher une entreprise par nom ou numéro de téléphone;

### Fonction 1 : Gestion des contacts

- ajouter, modifier et supprimer un contact pour un client;
- lister les contacts d'une entreprise spécifique;
- avoir une liste de tous les contacts du laboratoire exportée sous forme de fiche Excel;

### Fonction 1 : Gestion des destinations d'exportation d'une entreprise

- ajouter, modifier et supprimer un contact pour une destination;
- chercher un client par nom ou numéro de téléphone;
- lister les destinations d'exportation d'une entreprise spécifique;



Consultez-vous ci-dessous le diagramme de classe présentant ce sous-système.

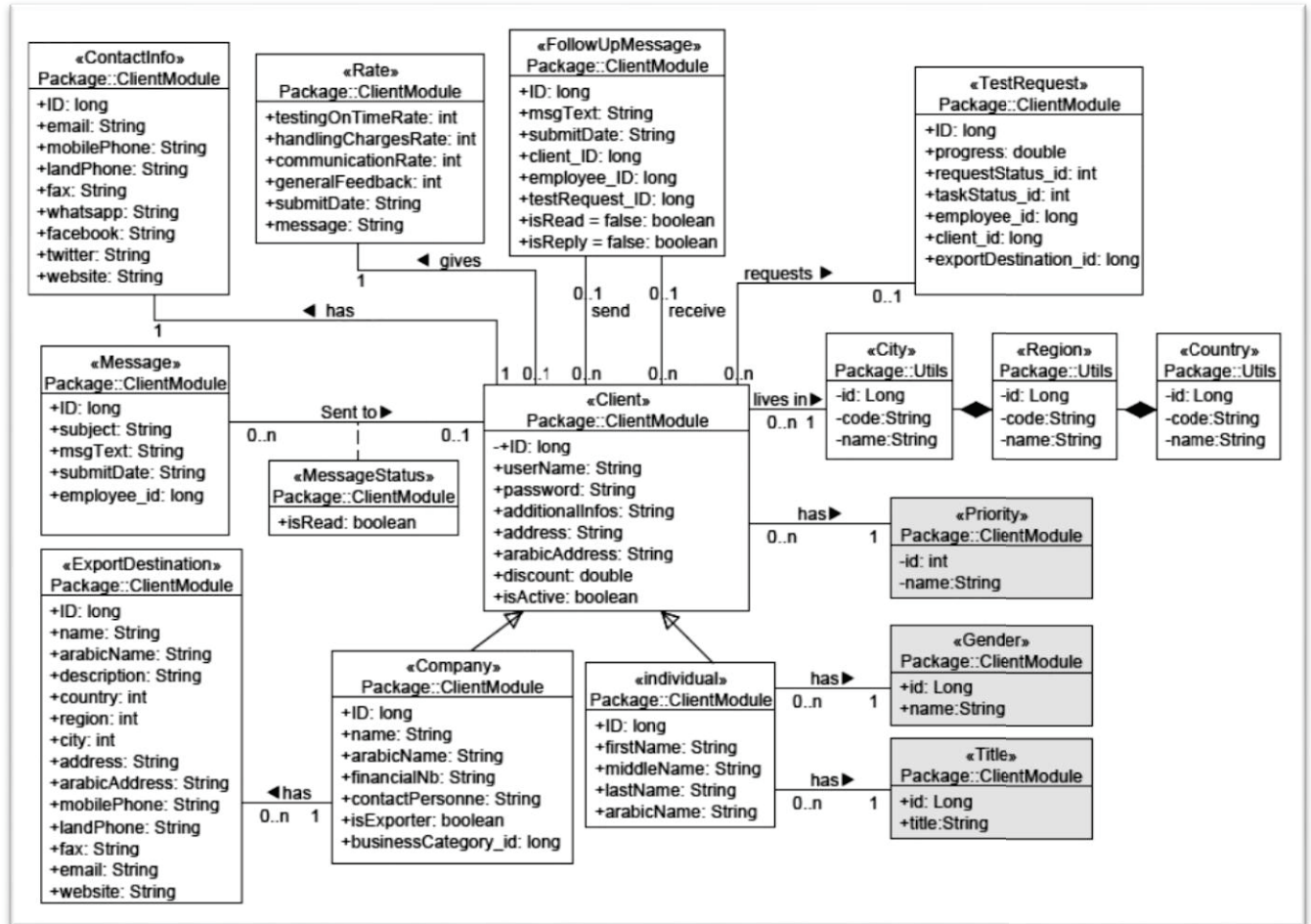


Figure 15. Diagramme de classe de sous-système Client

#### 4.2.3. Sous-système de demandes de tests

Ce module représente le cœur du système, il possède cinq acteurs principaux qui sont le réceptionniste, le technicien, l'auditeur, l'administrateur et le client. Chaque acteur interagit avec ce module de son point de vue, utilisant les outils et les fonctionnalités qui le rend capable de performer son rôle d'une manière effective et efficace dans le processus essentiel de la valeur de chaîne du laboratoire « la réception de demandes de Tests ».

Consultez-vous sur la page suivante la figure informative (16) illustrant les différentes fonctions de ce module :

DIVISION EN FONCTIONS  
SOUS-SYSTEME DES DEMANDES DES TESTS

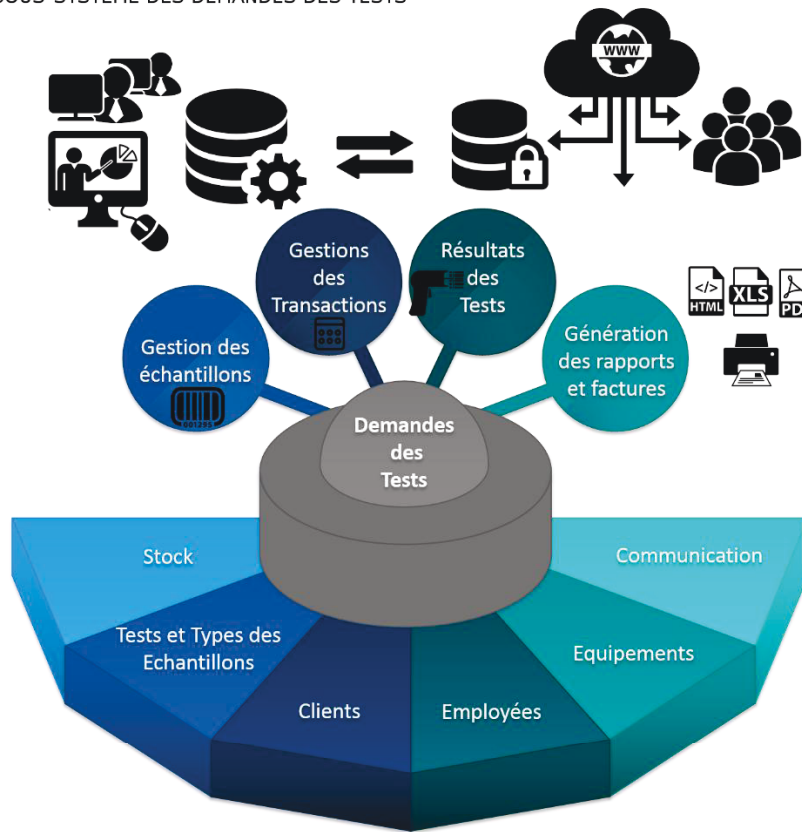


Figure 16. Division en fonctions de sous-système de demandes de tests

Parmi les fonctionnalités essentielles de ce module on cite ci-dessous:

- ❑ la gestion du flux de travail et d'affectation des tâches par les techniciens et l'administration;

ID	Client	Priority	Task Status	Responsible	Due Total (LBP)	Submit	Due Date	Progress	Chat
76	Social welfare Institution مؤسسة رفاه الإجتماعية	Low	readyToPrint	Khaled El Omari	4800.0	08/09/2014 09:46 AM	11/09/2014	100.0 %	
78	Al Amina المؤسسة الأمينة	High	ongoing	Khaled El Omari	0.0	08/09/2014 11:47 AM		0.0 %	
79	Al Amina المؤسسة الأمينة	High	ongoing	Khaled El Omari	45000.0	08/09/2014 01:14 PM	09/09/2014	0.0 %	
80	Social welfare Institution مؤسسة رفاه الإجتماعية	Low	waiting	Khaled El Omari	4800.0	08/09/2014 01:57 PM	11/09/2014	0.0 %	
81	test	High	readyToPrint	Khaled El Omari	56025.0	09/09/2014 10:46 AM	10/09/2014	100.0 %	
82	Social welfare Institution مؤسسة رفاه الإجتماعية	Low	waiting	Khaled El Omari	44250.0	11/09/2014 11:08 PM	12/09/2014	0.0 %	

Figure 17. Liste de demandes de test

- ❑ le suivi et traçabilité d'un échantillon utilisant la technique de code à barres (barcode labeling);
- ❑ la saisie des informations des clients par le réceptionniste;
  - la recherche d'un ancien client soit par son nom soit par son numéro de téléphone;

Figure 18. Recherche d'un ancien client

- la saisie des informations d'un nouveau client:

Figure 19. Saisir les informations d'un nouvel client

- ❑ la saisie des informations de chaque échantillon

Figure 20. Définition des échantillons à tester

- ❑ la sélection des tests à performer pour chaque échantillon. Certains clients n'ont pas les connaissances ou l'expertise suffisante pour demander l'analyse appropriée ou interpréter les résultats. Par conséquent le personnel du laboratoire devra les assister pour sélectionner le bon test. Pour un taux minimal d'erreur, les tests, soient obligatoire soient optionnels nécessaires à l'analyse de chaque type d'échantillon est déjà défini par l'administration supérieure pour rendre la mission de réceptionniste plus systématique. Cette étape sera schématisée clairement avec la capture d'écran (figure 21) ci-dessous :

Select All	Name	Method	Price (LBP)	Package
<input type="checkbox"/>	Acidity (lactone and total) of honey	AOAC 962.19/ Titrimetric	18000	NA
<input type="checkbox"/>	Bromide	Ion Chromatography	22500	Anions
<input type="checkbox"/>	Chloride	Ion Chromatography	22500	Anions
<input type="checkbox"/>	Fluoride	Ion Chromatography	22500	Anions
<input type="checkbox"/>	Nitrate	Ion Chromatography	31500	Anions
<input type="checkbox"/>	Nitrite	Ion Chromatography	27000	Anions
<input type="checkbox"/>	Phosphate	Ion Chromatography	31500	Anions
<input type="checkbox"/>	Sulfate	Ion Chromatography	31500	Anions
<input type="checkbox"/>	Tetracycline	ELISA method	0	Antibiotic
<input type="checkbox"/>	Oxytetracycline	ELISA method	0	Antibiotic
<input type="checkbox"/>	Ash	Muffle Furnace	4500	tests
<input type="checkbox"/>	Ammonium	Ion Chromatography	36000	Cations
<input type="checkbox"/>	Calcium	Ion Chromatography	27000	Cations

Figure 21. Définition les tests à réaliser de l'échantillon

- la saisie les détails de transaction;

The screenshot shows the 'Transaction Info' form in the ADULAS system. The form is for 'Test Request ID #85' from 'Client Name : Social welfare Institution'. It contains several input fields: 'Delivered by \*', 'Ar Delivered by \*', 'Expected Delivery Duration' (set to 11/01/2015), 'ExportDestination' (a dropdown menu), and 'Reporting & Communication language \*' (set to Arabic). There is also a text area for 'Additional Delivery Info'. A summary section at the bottom shows 'Total before Discount: 206250.0 (LBP)', 'Applied Discount: 0.0%', 'Total After Discount: 206250.0 (LBP)', and a 'Discount' field set to 0%. The final total is 'Total after discount is : 206250.0 (LBP)'. A 'Save' button is located at the bottom left of the form.

Figure 22. Saisir des détails de la transaction

- la saisie de résultats par techniciens;
- l'analyse et l'interprétation du résultat par le système sous une interaction complète avec le technicien, l'auditeur et l'administrateur, validation de résultats par auditeurs;
- la publication des rapports et de contrat d'échantillon. Dans notre système, les documents générés sont regroupés, selon les standards ISO, en deux catégories «pré-examination», et «post-examination»;

The screenshot shows the 'Request's Samples List' for 'Test Request #85'. The table has the following data:

Confirmed	#ID	Type	Additional Info	Time Frame	Price ( LBP )	Progress	Action
	115	Akkawi Cheese		1 - 3 days	206250.0	0.0 %	Edit Delete

Below the table, there are buttons for 'Add New Sample' and 'Go To Reporting Page'. The footer of the page includes 'View Your Client Web Site' and 'ADHENE © 2014 | www.adhene.com'.

Figure 23. La publication des rapports d'un échantillon

Consultez-vous ci-dessous le diagramme de classe présentant le sous-système de demandes de tests.

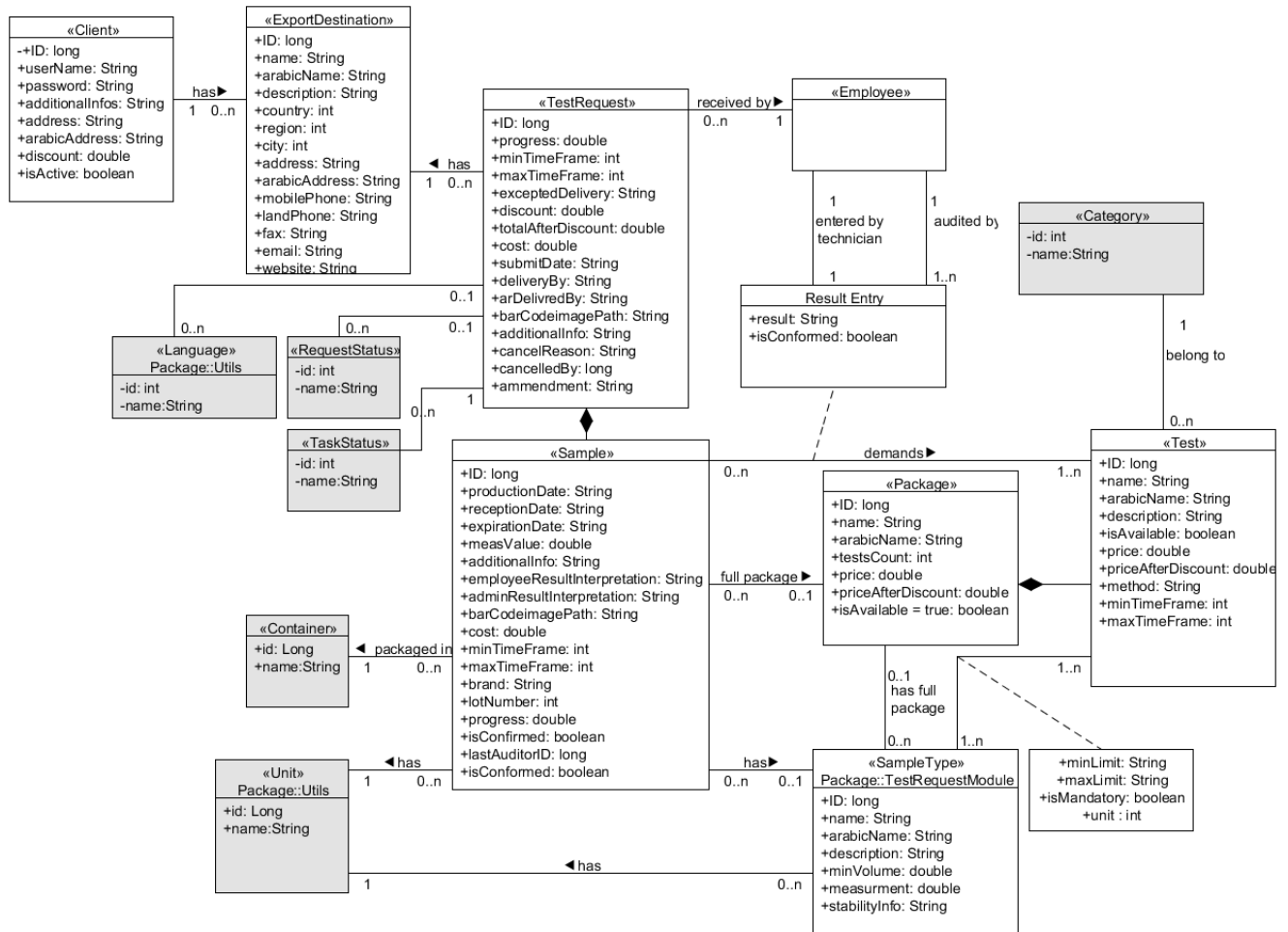


Figure 24. Diagramme de classe de Moteur du système

#### 4.2.4. Sous-système des Employés

Ce module représente un outil de gestion de ressources humaines, il permet la gestion de profils des employés, leur rémunération comme salaire, avantages sociaux, récompenses et déductions. De même, il gère l'horaire de travail, le calendrier de vacances, les demandes de congés, l'absence et la présence pour simplifier la préparation des bulletins de paye. D'autre part, ce module assure la définition des départements, des postes et des politiques pour avoir un meilleur contrôle des ressources humaines.

Ce module est réutilisé dans le projet AGORA ayant le composé éducatif de l'Arz comme client direct. De plus, nous sommes en train d'achever ce module comme un produit générique autonome sous le nom « TalentCare » pour être lancé sur le marché local au début de février 2015 tout en confirmant la réussite de notre projet SIGL comme retour financière et chiffre d'affaires.

Consultez-vous ci-dessous le diagramme de classe présentant ce sous-système :

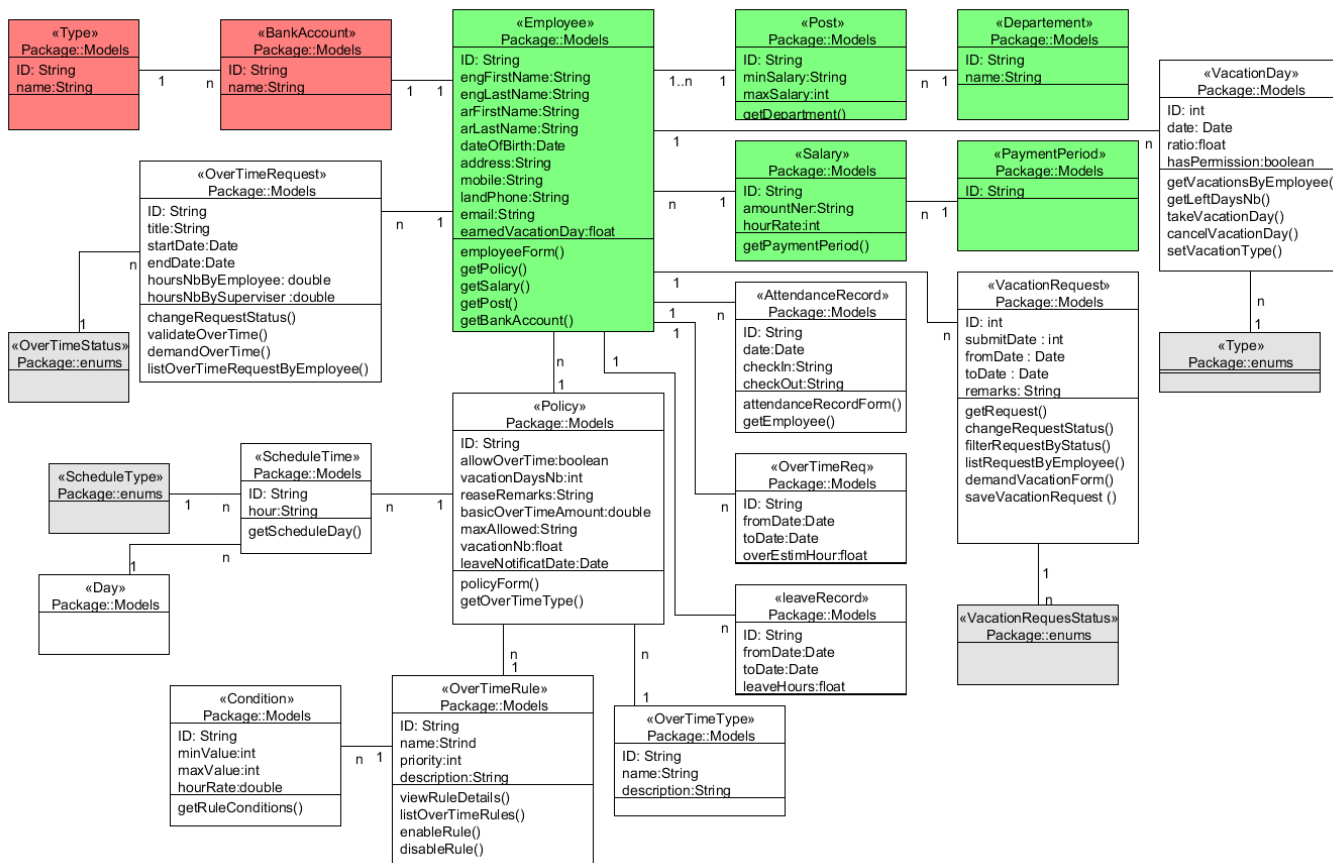


Figure 25. Diagramme de classe de Sous-système de gestion de ressources humaines

#### 4.2.5. Sous-système de Stock

Le but essentiel de ce sous-système est de s'assurer que les matières et les outils nécessaires au fonctionnement du laboratoire et les analyses seront toujours disponibles.

De même, ce module est réutilisé dans le projet AGORA ayant le composé éducatif de l'Arz comme client direct. Par ailleurs, ce module est achevé et lancé en avril 2014 comme un

produit générique autonome sous le nom « ACCUSTOCK » pour être aujourd’hui un facteur essentiel dans notre succès et survie comme il a une large et diverse clientèle.

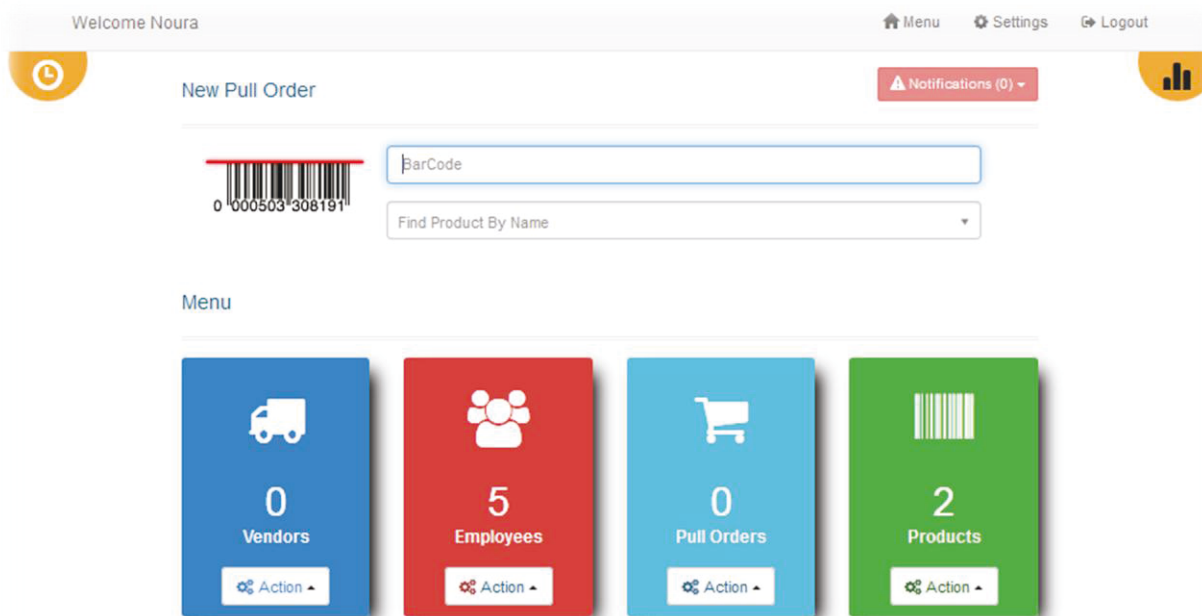


Figure 26. La page principale du module de stock

Parmi les fonctions diverses de ce module, on va citer plusieurs ci-dessous :

#### **Fonction 1 : Gestion des produits**

- ajouter, modifier et supprimer un produit ;
- générer un code à barres pour le produit s’il n’a pas un ;
- imprimer des étiquettes du code à barres pour chaque produit ;
- consulter les détails d’un produit ;
- chercher un produit par nom ou code à barres ;
- gérer les catégories de produits, ajouter, modifier et supprimer une catégorie ;

#### **Fonction 2 : Gestion des fournisseurs**

- ajouter, modifier et supprimer un fournisseur ;
- surveiller la performance des fournisseurs (délais de livraison, niveaux de qualité, respect des conditions négociées par l'entreprise) ;
- définir les produits qu’un vendeur peut assurer ;



- gérer les dettes ;

### **Fonction 3 : Gestion des commandes d'achats**

- créer une commande d'achats;
- annuler une commande;
- retourner une commande, en précisant la cause, pour aider à l'évaluation des fournisseurs;
- mettre à jour les quantités d'articles dès la réception de commandes;
- chercher une commande par son identificateur ID;
- filtrer les commandes par fournisseurs, statut de livraison, et par date;
- suivre la livraison des commandes d'achats et leur réception ;

### **Fonction 4 : Contrôler les limites de stock**

- définir des avertissements de bas niveau de stocks pour orienter le processus de commande d'achats ;
- recevoir des alarmes à propos des produits qui s'approchent de se terminer ou sont terminés ;
- recevoir des alarmes à propos des produits qui s'approchent de sa date d'expiration ou la dépassent ;

Veillez-vous regarder le diagramme de classe (figures 27 - 28) présentant ce sous-système dans la page suivante.

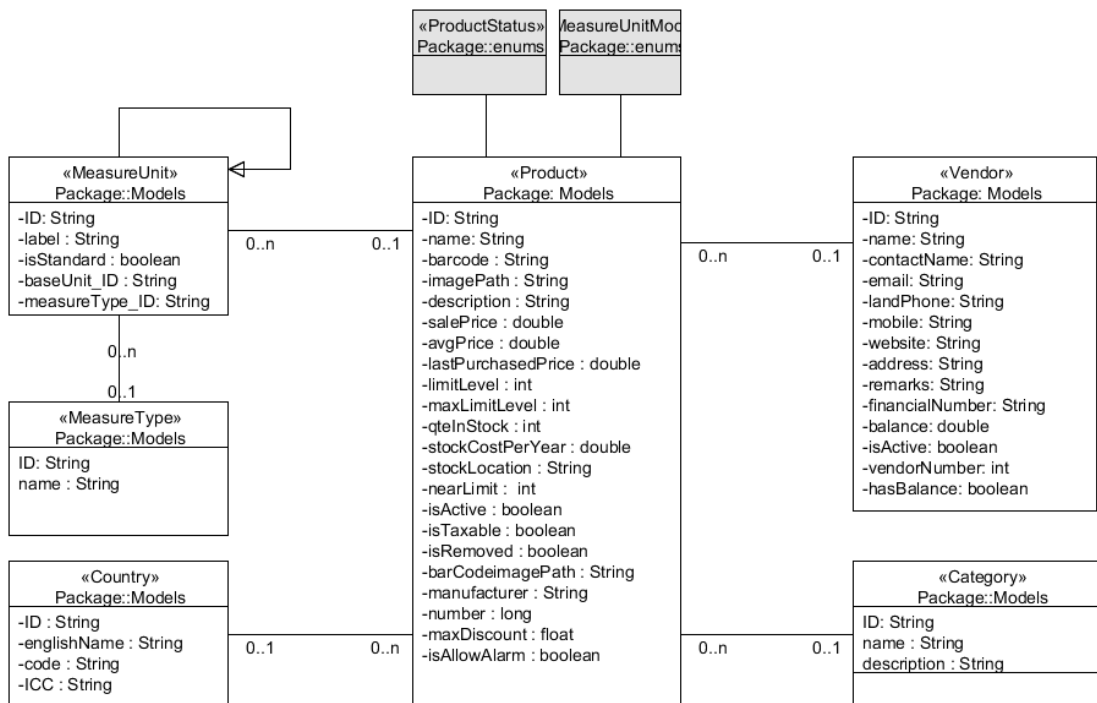


Figure 27. Diagramme de classe de stock - partie 1

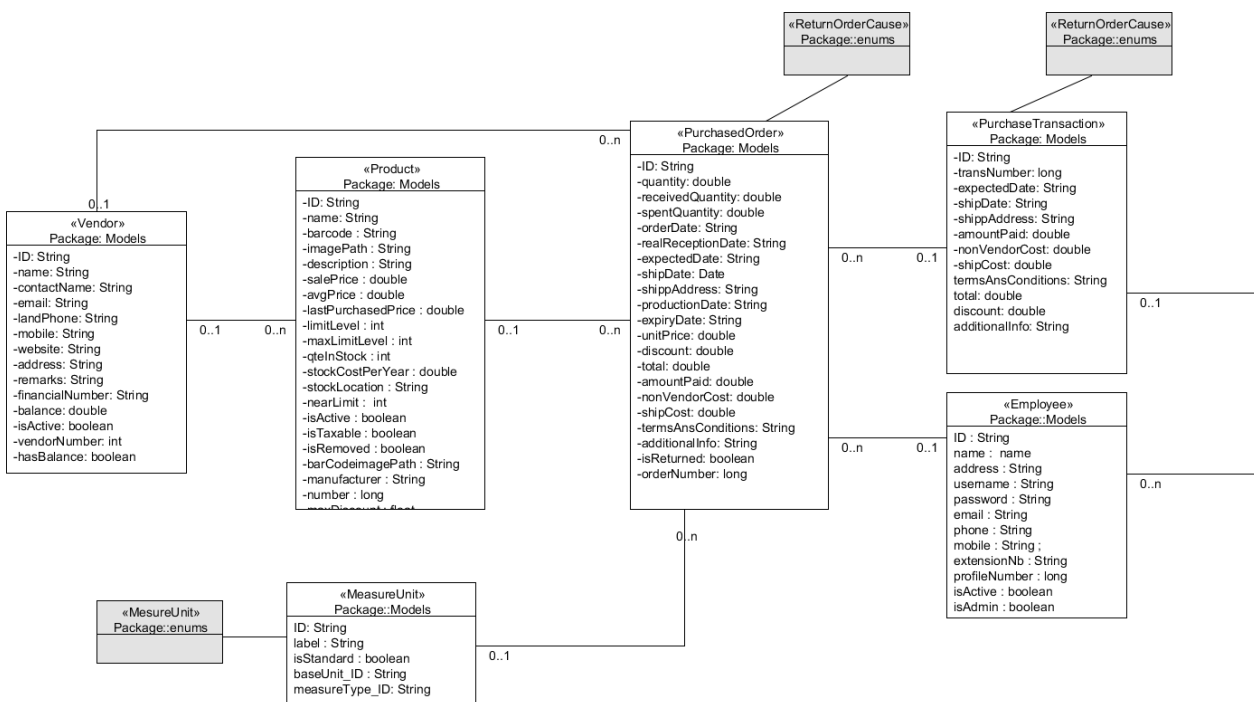


Figure 28. Diagramme de classe de Sous-système de Stock – partie 2

#### 4.2.6. Sous-système de Communication

Ce sous-système est divisé en deux modules:

##### **Module de communication interne**

Ce module supporte les processus du pilotage pour avoir un meilleur contrôle sur les processus opérationnels. Il est constitué de:

- ❑ courriel entre les employés ;
- ❑ diffusion des messages de l'administration à tous les employés ;
- ❑ messages directes sur le flux de travail et dans le contexte des données représentés par un commentaire sur un échantillon spécifique, sur les résultats élaborés d'une analyse, sur les détails de transaction d'une demande de test et sur le dossier d'un client etc. ;

##### **Module de communication externe**

Le concept de service client a souvent été négligé au laboratoire qui est cependant une entreprise de service, donc on croit qu'il est primordial que les clients du laboratoire collaborent directement aux flux de notre logiciel comme la satisfaction de clientèle est une composante majeure de tout système de gestion de la qualité et l'un des axes développés dans les normes ISO pour tout laboratoire.

Ce module permet au client de suivre la progression de l'analyse de ses échantillons en cours et consulter l'historique de ses demandes de tests. L'accès à ces données exige l'authentification du client au système par un nom d'utilisateur et un mot de passe.

De même, à travers l'interface web, le client peut se communiquer avec le personnel du laboratoire à propos une demande de test ou une analyse. Il peut passer une suggestion ou réclamation à l'administration.

D'autre part et suivant la norme ISO, ce module assure l'évaluation du service du point de vue client, la collection et la suivie de ses commentaires. Donc, il aide à supporter le mécanisme d'amélioration des processus du laboratoire qui est un facteur clé de succès de tout système de gestion de qualité.

Consultez-vous dans la page suivante quelques captures d'écran permettant la visualisation rapide de ce module.

ID	Submit date	Delivered By	samples number	Task Status	Due Total (LBP)	Due Date	Progress	Chat
#85	10/01/2015 10:36 AM	Ahmad El-Mir	1	waiting	206250.0	11/01/2015	0.0 %	

Figure 29. Interface du client

## Publicité et annonces

D'ailleurs, le système est mené d'un site web pour aider le laboratoire à refléter son image de marque sur le web, accéder rapidement ses clients, communiquer ses offres et échanger ses idées et ses informations à l'aide des annonces bien classées, à bas coût comparant avec les autres outils de publicités.

Figure 30. Le curseur de nouvelles dans le site web

D'autre part, le site web aide les clients du laboratoire à poursuivre et surveiller aisément leurs demandes de test et le statut de leurs échantillons sans la peine d'arriver au laboratoire ou le contacter que pour la réception de résultat officiel. L'accès du client et l'autorité lui offert peuvent être contrôlés selon la politique du laboratoire.

Les fonctionnalités de ce module peuvent être résumées à travers les diagrammes de cas d'utilisation ci-joints :

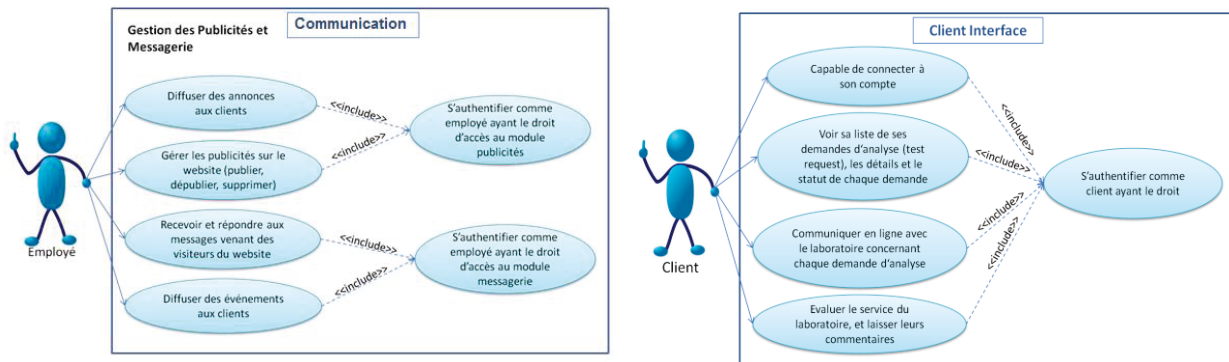


Figure 31. Diagramme de cas d'utilisation du sous-système de communication

#### 4.2.7. Sous-système des Equipements

La gestion des équipements est l'un des facteurs essentiels du système de gestion de la qualité pour tout laboratoire. Une bonne gestion des équipements est nécessaire pour assurer la l'intégrité, la fiabilité et la pertinence des analyses.

Ce sous-système permet d'assurer un plan de maintenance préventive des équipements pour répondre aux exigences de qualité par la FDA et ISO.

De même, il aide à maintenir la productivité du laboratoire en réduisant les interruptions de fonctionnement dues à des pannes et des défauts en équipements ce qui reflète directement par une meilleure satisfaction du client. Il diminue les frais de réparation et augmente la durée de vie des équipements par l'organisation du calendrier de maintenance pour tout équipement.

- ❑ saisir le contrat de maintenance et de réparation d'un équipement;
- ❑ arrêter ou éliminer un équipement;
- ❑ définir les utilisateurs qui peuvent accéder et utiliser cet équipement ;
- ❑ superviser tous les systèmes de gestion de l'équipement au sein du laboratoire.

Consultez-vous dans la page suivante le diagramme de classe présentant ce sous-système.

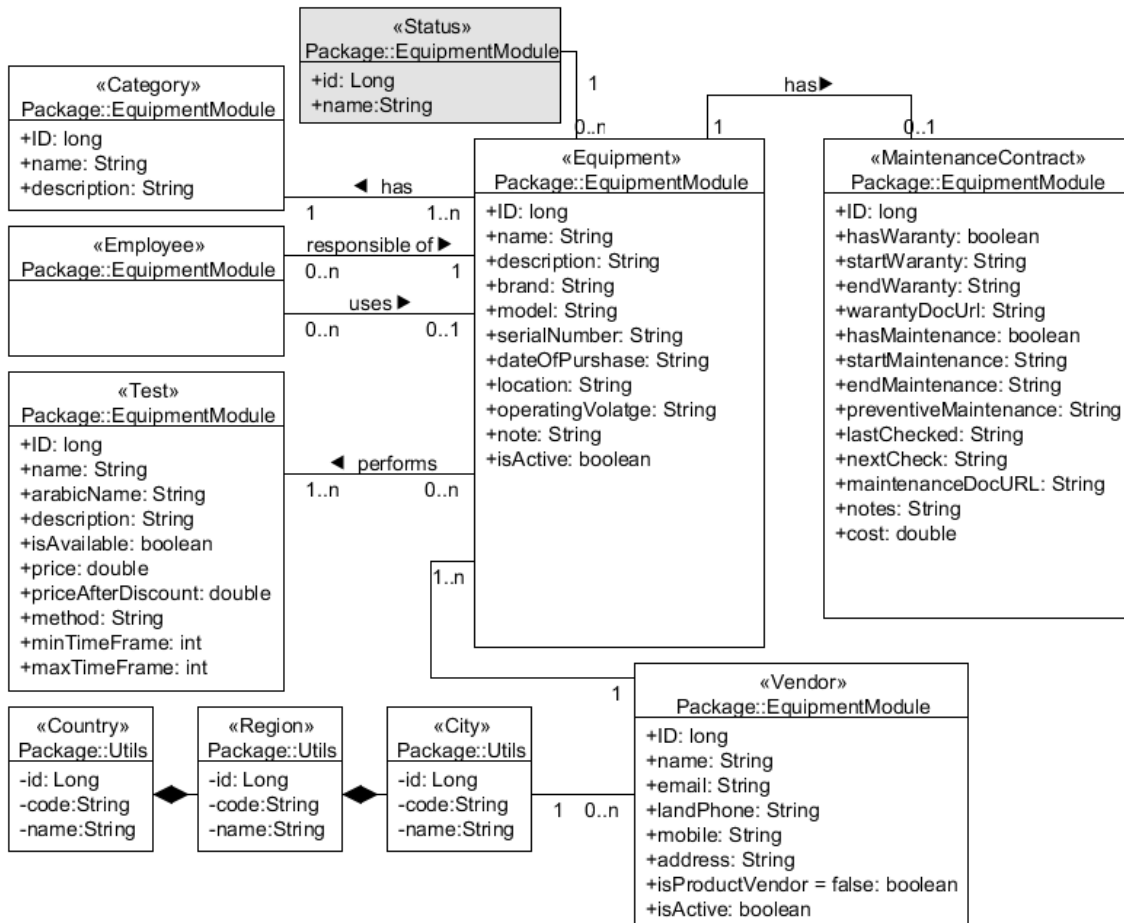


Figure 32 : Diagramme de classe de sous-système des Equipements

Selon notre plan, au futur et sous le contrat de maintenance de 2015, une relation directe et puissante entre le module de définition des tests, le module de gestion des équipements, le module de gestion de ressources humaines et le module de stock sera établie dans le but d'avoir un contrôle ultime sur le processus opérationnel principal qui est la demande de tests d'un échantillon. Cette relation permet au réceptionniste, lors de la réception des échantillons, de savoir instantanément la disponibilité d'un test demandé en vérifiant le statut des équipements qu'ils l'exécutent, la présence des outils et des matières nécessaires dans le stock et la présence des employés qu'ils ont la compétence nécessaire pour réaliser ce test.

#### **4.2.8. Sous-système de Rapports et statistiques**

C'est principalement un tableau de bord (dashboard) qui fournit des présentations et des diagrammes graphiques résumant la performance globale du laboratoire et sa productivité en chiffres à l'aide de plusieurs indicateurs comme le nombre des clients, le chiffre d'affaires signalé, le nombre de demandes de test etc.

Ce module permet à l'administration de détecter rapidement les obstacles, les erreurs et les évaluer correctement pour l'aider à planifier des mesures correctives efficaces et rapides gardant la performance et la productivité du laboratoire. De plus, le but essentiel de ce sous-système doit être l'amélioration et l'optimisation des processus métiers soient opérationnels, de support et de pilotage.

### **4.3. Adaptation des Principes de Modélisation Agile**

Je crois que la nécessité de l'agilité doit commencer dès la phase de conception et de modélisation pour avoir un projet vraiment agile. La modélisation pour nous est une activité que doit avoir lieu tout au long du projet et pas seulement au début du projet avant d'arriver à l'écriture du code source. Nous sommes, les membres de l'équipe SCRUM, les concepteurs et les développeurs simultanément.

#### **4.3.1. Comment gagner du temps dans la modélisation**

Guidé par la modélisation agile, on croit que les modèles de conception doivent être juste assez bons. Nous n'avons pas besoin d'obtenir un ensemble bien documenté des modèles en place avant que nous pouvions commencer à coder. Puis on itère, itère, itère... en travaillant un peu sur les exigences, un peu d'analyse, un peu de la conception, du codage, des tests, et itérer entre ces activités en fonction des besoins.

Pour un modèle donné, nous utilisons l'outil simple qui répondra à l'exigence de modélisation. Plutôt, les papiers et les tableaux blancs nous paraissent les meilleurs outils.

Le standard UML offre plusieurs types de diagrammes, le devoir du concepteur se résume par choisir le diagramme le plus efficace et le plus rapide pour communiquer ses idées et les exigences de la fonction traitée aux autres membres de l'équipe.

Basant sur mon expérience professionnelle, je pense que l'acquisition des capacités de modélisation en circulation est à l'origine de notre succès en ADHENE.

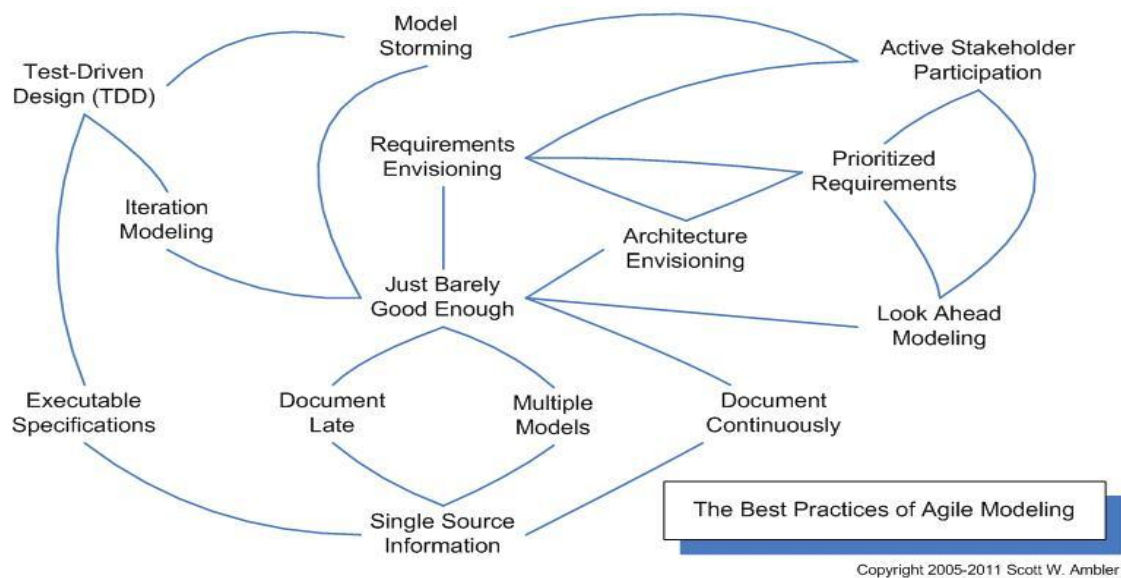


Figure 33 : Les Bonnes Pratiques de modélisation agile (Agile Modeling AM)

#### 4.3.2. Les Bonnes Pratiques de modélisation agile

Parmi les bonnes pratiques de modélisation agile, j'ai l'envie de citer les pratiques qui me paraissent les plus importantes et que nous appliquons en ADHENE:

- ❑ modéliser ensemble, il faut créer l'ensemble des modèles critiques à notre projet de base ensemble comme une seule équipe ;
- ❑ utiliser le diagramme le plus convenable, on doit connaître les forces et les faiblesses de chaque type de diagramme et quand on doit ou pas l'utiliser en faveur de la phase de développement ;
- ❑ itérer avec un autre type de diagramme si on a seulement si on était incapable de communiquer les idées, la structure de données ou les exigences fonctionnelles à travers le premier diagramme ;
- ❑ prouver le modèle obtenu avec le code si c'est nécessaire ;
- ❑ utiliser les plus simples outils, dessiner sur un tableau blanc, sur papier pour réfléchir à une question ou une problématique.



- ❑ considérer le projet comme propriété collective, on doit accepter que tout le monde peut travailler sur n'importe quel modèle et idée concernant le projet, s'il le souhaite et sent sa capacité à le performer ;
- ❑ avoir un seul endroit pour conserver les informations et modèles résultants de la phase de conception ;
- ❑ avoir un concept par un seul modèle ;
- ❑ modéliser en parallèle si on a le temps et la capacité surtout au cas d'interdépendance entre plusieurs concepts ;
- ❑ créer et représenter un modèle de la manière la plus simple et la plus claire possible ;
- ❑ avoir l'humilité, on doit reconnaître que les meilleurs développeurs ne savent pas tout, que leurs collègues, leurs clients. En fait, toutes les parties prenantes du projet ont aussi leurs propres domaines d'expertise et ont une valeur à ajouter à un projet et donc ils doivent être traités avec respect.

#### 4.3.3. Lignes Directrices de Style

- ❑ Éviter les lignes croix le plus possible et lorsque on a besoin d'avoir deux lignes qui se croisent l'un d'eux on doit visualiser un saut de l'un sur l'autre ;
- ❑ appliquer pour les objets similaires d'un modèle une taille constante à l'exception de celui qui représente l'idée essentielle du diagramme ;
- ❑ afficher uniquement ce qui est vraiment nécessaire et utile et inclure les espaces blancs dans les diagrammes;
- ❑ réorganiser un grand diagramme en plusieurs autres plus petits ;
- ❑ se concentrer sur le contenu d'abord et puis sur l'apparence ;
- ❑ organiser finalement les diagrammes de gauche à droite, de haut en bas;
- ❑ définir et suivre une convention de nommage;
- ❑ ajouter les couleurs aux diagrammes pour être plus significatifs;

#### 4.4. Ergonomie de l'interface utilisateur

L'ergonomie de l'interface utilisateur devient de plus en plus importante et la clé essentielle de succès de toute application informatique. Il est clair que les utilisateurs sont souvent perdus dans

les applications informatiques surtout si le système automatise des processus complexes et possède une vaste gamme des fonctionnalités. L'ergonomie offre des importants principes qu'on doit les respecter pour avoir un système facile à utiliser et attractif. Selon la définition, l'ergonomie est un attribut de la qualité de l'interface utilisateur couvre si le système est facile à apprendre, efficace à utiliser, agréable, et ainsi de suite...

Les lignes directrices respectées et appliquées dans nos systèmes développés sont :

- ❑ Fournir uniquement les informations dont on a besoin à la tâche courante, l'utilisateur ne peut regarder aussi d'informations ou de lire autant de texte sur un seul écran sans perdre intérêt ;
- ❑ rendre l'information facile à comprendre et à analyser ;
- ❑ utiliser des courtes descriptions et textes;
- ❑ supposer que les utilisateurs font des erreurs, anticiper les tentent à les empêcher;
- ❑ utiliser une confirmation avant d'agir sur l'action de l'utilisateur qui peut conduire d'une erreur grave;
- ❑ prévenir les erreurs de se produire par l'utilisateur. Le meilleur message d'erreur n'est pas de message du tout ;
- ❑ diviser la tâche complexe et la tâche qui peut provoquer des erreurs en plusieurs étapes;
- ❑ respecter la mémoire fragile chez l'homme, les gens ne peuvent se rappeler qu'environ 3-4 actions à la fois ;
- ❑ noter bien que les choses qui sont proches les unes aux autres sont considérées à «être» ensemble ;
- ❑ assurer que les polices (fonts) soient assez grandes. En utilisant des polices qui ne sont pas trop décoratives afin qu'ils soient faciles à lire.

## 4.5. Outils de Conception et Modélisation

### UMLet

Après recherche, j'ai choisi « UMLet » comme outil de modélisation en UML. C'est un outil à source ouverte avec une interface utilisateur simple. Il nous permet de dessiner des diagrammes

UML d'une manière simple et rapide et les exporter à eps, pdf, jpg, svg, et presse-papiers. UMLet fonctionne autonome ou comme plug-in Eclipse sous Windows, OS X et Linux.

Le chapitre suivant sera dédié à la gestion du projet, parlant du cycle de développement et détaillant le plan comme délais et personnel.

## Chapitre 5: Gestion de Projet

Cette étape consiste à préciser les changements pour optimiser les processus de développement et les rendre plus efficaces. Ces changements peuvent comprendre une simplification des processus actuels, des modifications en profondeur de certains processus, l'ajout de nouveaux processus ou activités et l'élimination des doublons etc. Tout cela conduit à une meilleure utilisation des ressources et répartition des charges de travail, réduction des pertes de temps, amélioration des méthodes de travail, meilleure définition des responsabilités et à l'amélioration des pratiques de gestion etc.

### 5.1. Processus de Développement

Il est important de choisir le processus approprié de cycle de développement pour le projet à accomplir, puisque toute la chaîne des activités de développement sera affectée fortement par ce processus. Pour la plupart des projets modernes de développement de logiciels, une sorte de méthodologie spirale est utilisée sur un processus en cascade (waterfall). L'utilisation de ce processus est quasi importante que la façon dont il soit exécuté.

### 5.2. Choix de Méthode Agile

Après la consultation de tous les processus de développement interprétés par le génie logiciel, cascade, spirale, itératif, unifié et méthode agile, j'ai choisi la méthode agile, car elle met l'accent sur les changements constants de cahier des charges et de code source des logiciels en favorisant une collaboration serrée et une forte implication de l'utilisateur client (on doit différencier entre les utilisateurs du logiciel et celui qui nous signe le chèque). Ce modèle assure un cycle de développement en spirale avec de nombreuses et courtes itérations, ce qui me paraît très nécessaire à mon client qui avait perdu la confiance dans le secteur informatique au Liban après plusieurs expériences d'échec avec le développement de ce système et qui était incapable de définir ses besoins et exigences clairement dès le début sans intervention de notre équipe. La dérivation rapide selon des itérations courtes était le meilleur moyen pour prouver nos compétences et capacités, éclaircir ses besoins, cibler les attentes tout en nous permettant une grande réactivité à ses demandes.

### 5.2.1. Cascade

J'élimine la programmation en "cascade" des choix comme c'est un processus fortement structuré qui compte lourdement sur la planification initiale et un ensemble d'étapes séquentielles, prescrites. Chaque étape a typiquement son équipe propre d'experts et des jalons soigneusement préparés d'avance et aucune étape ne peut commencer tant que l'étape précédente n'a pas été terminée ce qui contredit entièrement avec notre besoin et structure interne de l'équipe et reflète un manque de la flexibilité et de l'agilité nécessaire dans notre cas.

### 5.2.2. RUP

RUP faire face à plusieurs faiblesses du développement en "cascade" et constitue une bonne méthode pour des agences gouvernementales et institutions éducatives qui apprécient un cycle de développement de logiciel stable et répétable et ce n'est sûrement le cas de notre projet.

### 5.2.3. Choix de Méthodologie SCRUM

Inspirée par Patrick Grégoire [2] et basée sur mon expérience professionnelle, j'avais tendance à appliquer la méthode SCRUM parmi les méthodes agiles SCRUM, Extreme Programming et Rational Unified Process. J'ai expérimenté SCRUM dans mon dernier travail au PIOUS où j'ai prouvé son efficacité et m'a rendue familière avec ses bons pratiques. En plus, son processus s'articule autour d'une équipe soudée, qui cherche à atteindre un but ensemble renforçant ainsi notre modèle de gestion de ressources humaines.

Le principe de base de SCRUM est de focaliser l'équipe de façon régulière sur un ensemble de fonctionnalités à réaliser, dans des délais de 30 jours, appelés Sprints (dans notre cas, nous avons choisi 15 jours). Chaque Sprint possède des buts à atteindre, définis par le Directeur de produit (Product Owner), à partir desquels sont choisies les fonctionnalités à implémenter dans ce sprint et les tâches à accomplir. Ces fonctionnalités ou ces buts à atteindre sont définis sous forme de «User Story». Un sprint aboutit toujours à la livraison d'un produit partiellement fonctionnel. Pendant ce temps, le ScrumMaster a la charge de réduire au minimum les perturbations extérieures et de résoudre les problèmes non techniques de l'équipe afin de s'assurer que tout va pour le mieux et que chacun sait sur quoi il doit travailler.

### 5.3. Les rôles dans l'équipe

**Directeur de produit (Product Owner): Ingénieure biomédicale Jinan El-Mir**

- ❑ définir les priorités et les tâches, savoir ce qui est le plus important;
- ❑ représenter le client (l'utilisateur);
- ❑ réduire les embûches et les problèmes qui surviennent;
- ❑ être responsable de l'évolution du product backlog;
- ❑ être disponible pour répondre aux questions de l'équipe SCRUM.

**SCRUM Master**

L'équipe de développement est pilotée par mes efforts comme SCRUM master, mes responsabilités fondamentales est de résoudre les obstacles, participer au développement en cas de besoin et mettre en œuvre des solutions rapides pour que les objectifs soient réalisés durant le Sprint.

- ❑ assurer que les tâches soient accomplies et le bon déroulement de Sprint ;
- ❑ protéger l'équipe afin qu'ils restent focalisés sur leurs tâches et l'objectif du Sprint.

### 5.4 Taille et Membres de l'équipe:

- ❑ une ingénieure biomédicale comme Product Owner ;
- ❑ 6 développeurs Java ;
- ❑ un UI designer ;
- ❑ support complet de l'équipe de management.

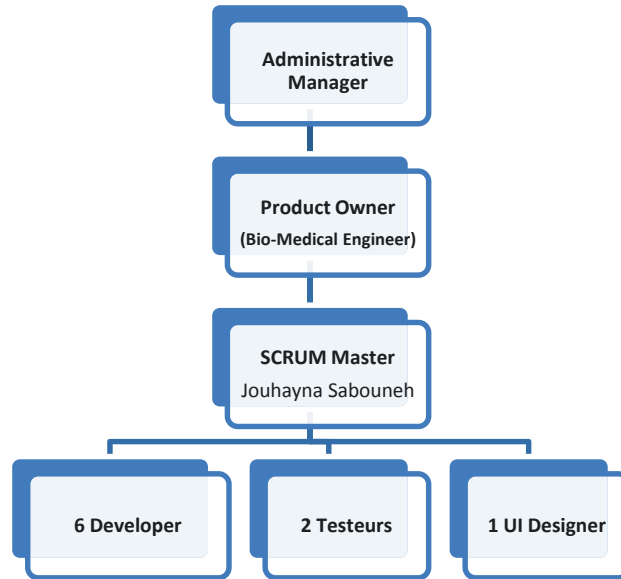


Figure 34 : Organigramme de l'équipe de développement du projet

## 5.5. Les 3 réunions SCRUM

### La Réunion Quotidienne sur les Pieds (Daily Stand-up Meeting)

Nous avons une réunion quotidienne où chaque membre d'équipe doit répondre à trois questions simples : qu'avait fait hier, que va faire aujourd'hui et a-t-il des problèmes pouvant l'empêcher d'accomplir ses tâches ? Cette réunion a lieu très rapidement, même que nous devrions rester debout. Le but premier était de dire où nous sommes dans notre travail afin que tous coéquipiers sachent vers où le projet avance. Cette réunion dure au maximum cinq à dix minutes.

### Réunion de Révision (Sprint Review Meeting)

Cette réunion a lieu à la fin de chaque sprint, que ce soit aux 14 jours. Le but de cette réunion est de montrer au directeur de produit et notre client ce que nous avons accompli et livré une nouvelle itération de notre projet prêt à être testée.

### Réunion de Planification (Sprint Planning Meeting)

On fait la planification généralement ensemble où chacun planifie ses tâches et le nombre d'heures qu'il pense que cela va lui prendre pour les accomplir. Notre sprint équivaut à 100 heures de travail, on travaille à réduire nos exigences à 80 heures planifiées de travail afin de prévoir le temps passé aux réunions et le temps pris pour les améliorations de dernières minutes.

À cette étape, le directeur de produit précise les buts à atteindre (User Stories) et les place en ordre de priorité. Il est alors important pour toute l'équipe de savoir qui doit faire quoi et combien de temps cela prendra afin de réduire au minimum le temps perdu de l'attente due à la mauvaise répartition de travail et au mal coordination.

SCRUM focalise sur la communication et l'importance de l'équipe, qui doit régler ses propres problèmes, trouver lui-même les solutions et avoir les meilleures connaissances. Chaque membre apporte ce qu'il a de mieux à l'équipe et c'est ensemble que nous réussirons. L'agilité que nous adaptons favorise la motivation et l'auto-organisation de l'équipe de développement.

## 5.6. Objectif De Délai

**Délai de développement:** 12 mois.

**Date de début:** 1 Février 2013.

**Date de livraison:** 24 Janvier 2014.

## 5.7. Le Plan

Consultez le diagramme de Gant (figure 35) et le plan chronologique (figure 36) du projet pour avoir une vision détaillée sur le plan de travail et les différentes itérations de développement:



# Plan Chronologique

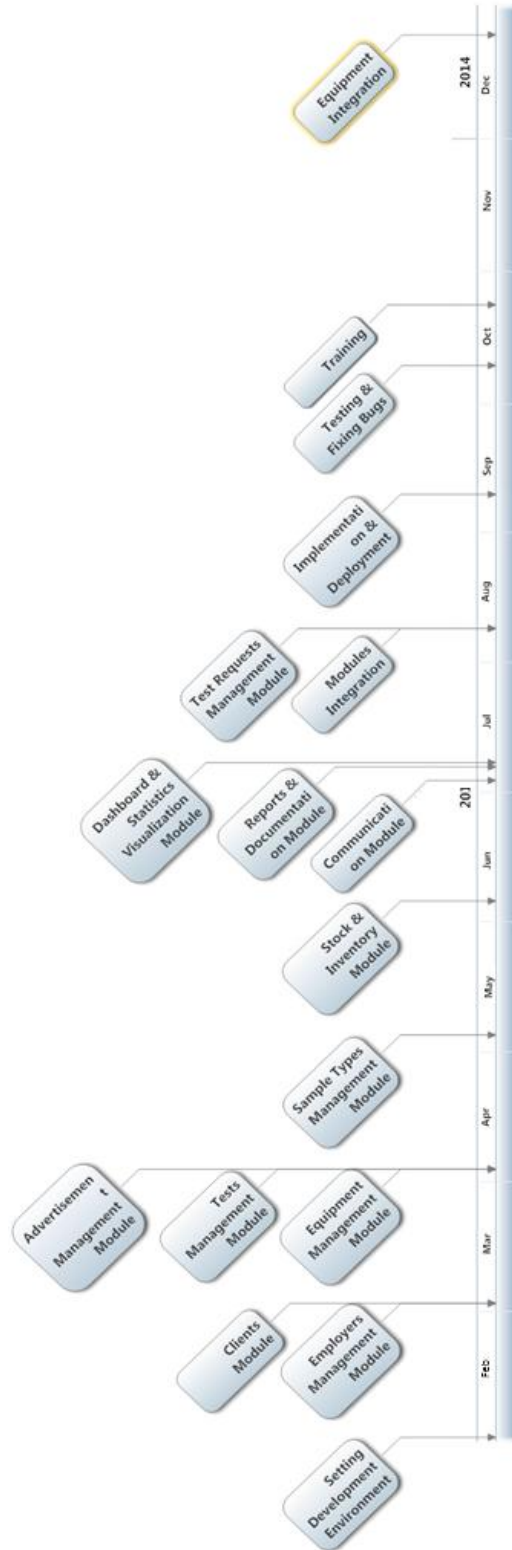


Figure 35. Le plan chronologie du projet

## Diagramme de Gant

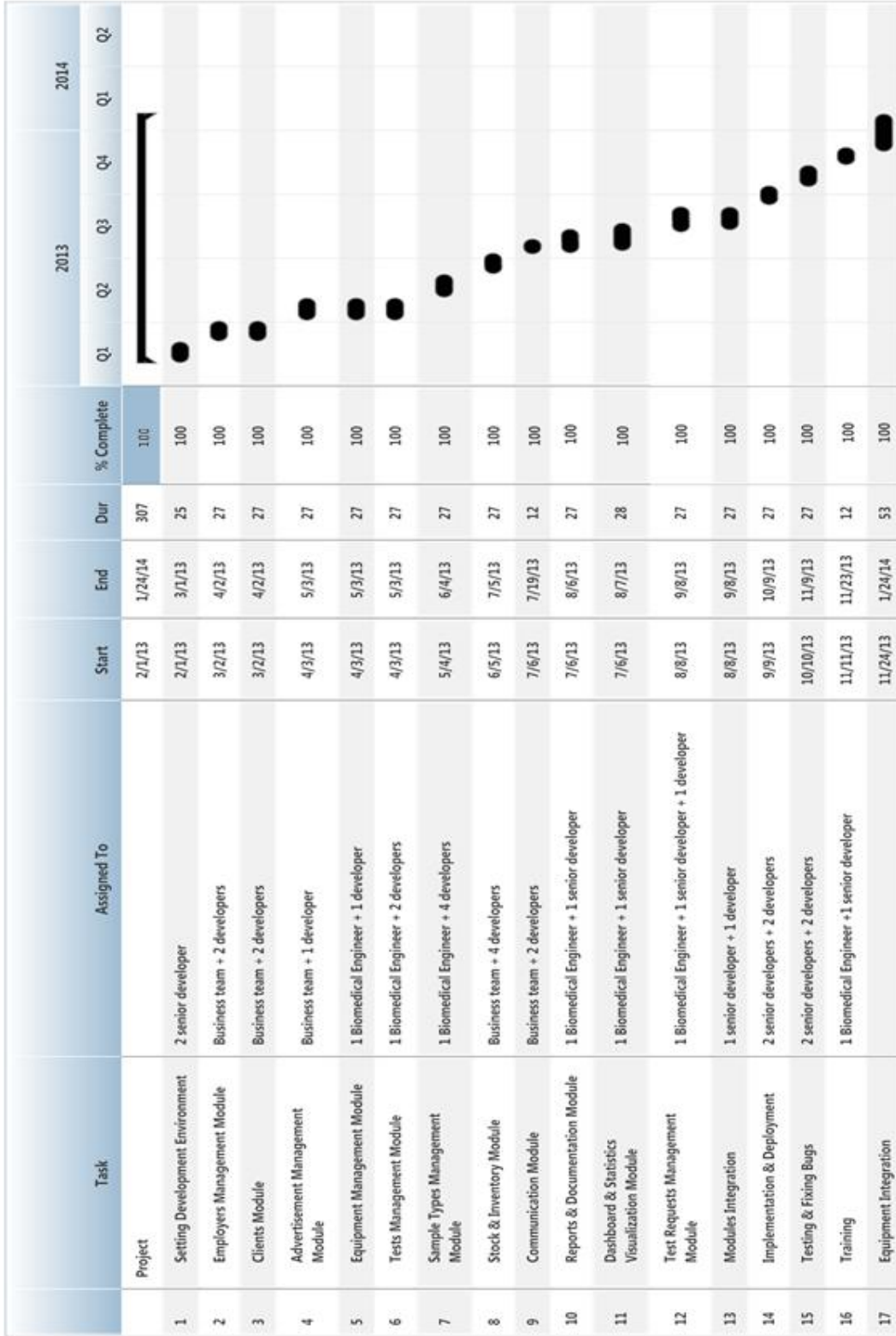


Figure 36. Le diagramme de Gant

## **5.8. Objectif de Coût**

Nous sommes une entreprise en démarrage donc la minimisation de coût de développement est l'un de nos objectifs essentiels d'où je recours à l'utilisation des techniques sources ouvertes détaillés ci-dessous dans le chapitre suivant.

## **5.9. Règles de gestion du projet**

Je fais mon mieux pour gérer la progression simultanément sur les axes : production des livrables, calendrier et budget. Je recours à plusieurs outils de pilotage et de gestion détaillés ci-dessous dans le chapitre suivant.

## **5.10. Plan de communication**

Le plan de communication servira à présenter régulièrement les avancées du projet et les bénéfices déjà obtenus à l'ensemble des employés. De même, il sert à clarifier les nouvelles tâches et responsabilités des employés concernés. La communication aide à préparer le changement. SCRUM nous favorise les bases de notre plan de communication appuyé sur plusieurs outils et bonnes pratiques détaillés dans le chapitre suivant.

## Chapitre 6 : Implémentation

Cette phase consiste à la rédaction du code source, les instructions qui offriront les fonctions attendues et qui sont le corps du logiciel. La programmation est effectuée en suivant les plans initialement établis lors de la conception et sera détaillée ci-après.

### 6.1. Choix de Langage de Programmation

#### 6.1.1. Pourquoi la source ouverte?

Nous sommes une entreprise en démarrage donc la minimisation de coût de développement est l'un de nos objectifs fondamentaux pour survivre à la situation économique contemporaine à notre pays et maintenir un aspect compétitif simultanément, d'où et basant aussi sur ma croyance professionnelle, justifié ci-dessous, je recours à l'utilisation des techniques du logiciel libre (source ouverte).

Le logiciel libre<sup>7</sup> nous permet de rejoindre une communauté mondiale en ligne avec excellents forums pour obtenir l'aide et échanger l'expérience et il y a même des forums dédiés spécifiquement aux nouveaux arrivants qui ont besoin d'explication supplémentaire.

Et si on a expérimenté un problème avec l'utilisation d'un logiciel libre et était incapable à le résoudre avec nos compétences et le soutien de la communauté, on peut toujours recourir au support technique commercial offert par tout logiciel libre choisi à un coût bien considérable.

Le concept « source ouverte » est protégé par l'OSI<sup>8</sup>, qui constate et assure, sur son site web officiel, qu'une solution à source ouverte doit inclure le code source et autoriser sa distribution sous une forme convenable aidant tout programmeur à le comprendre et le modifier. Toutes ces hypothèses nous rendent le maître de la situation et non plus son esclave comme le cas avec les

---

<sup>7</sup> **Logiciel libre** est un logiciel dont l'utilisation, l'étude, la modification et la duplication en vue de sa diffusion sont permises techniquement et légalement.

<sup>8</sup> **OSI** est un organisme à vocation communautaire qui examine et approuve les licences d'utilisation d'un logiciel, pour s'assurer qu'elles sont conformes à la définition établie. L'OSI détermine une série de critères que l'on doit respecter pour qu'un logiciel soit approuvé « code à source ouvert ».

grands éditeurs des logiciels fermés et de mon point de vue, c'est la première étape vers l'excellence dans notre domaine.

### 6.1.2. Pourquoi Java?

Je ne peux pas nier qu'un temps bien considérable soit consacré à la recherche et le choix de la technique appropriée et les outils de développement et leur apprentissage au monde de Java, mais quand cela est fait, notre cycle de développement sera bien accéléré et notre processus de développement sera vraiment optimisé...

De point de vue coût, Java est essentiellement disponible gratuitement et il a beaucoup d'outils à source ouverte disponibles en haute qualité.

Avec java, il y a littéralement une bibliothèque source ouverte gratuite pour tout ce qu'on peut penser grâce à son vaste écosystème de bibliothèques telles que:

- ❑ les bibliothèques et outils Apache ;
- ❑ les bibliothèques dans le vaste écosystème Eclipse ;
- ❑ les bibliothèques contribués/ entretenus par Google ;
- ❑ Jboss et tous les outils d'entreprise connexes tenus par Red Hat.

Java est l'un des langages de programmation les plus populaires dans le monde. Cela signifie qu'il y a tout simplement beaucoup de ressources disponibles pour le programmeur Java comme forums et groupes. Il y a beaucoup de livres disponibles, de nombreuses conférences pour aller, de nombreux articles à lire...

D'autre part, Java est la technique la plus familière pour moi et pour la plupart de mes collègues d'après nos cursus à ISSAE-CNAM, Liban.

D'ailleurs, Java a une syntaxe simple qui est relativement facile à apprendre et il possède une bibliothèque standard incroyablement riche et facile à utiliser. Il contient beaucoup de choses pour lesquelles la plupart des autres langages doivent recourir à des librairies tierces ce qui reflète positivement sur l'expansion de l'équipe de développement quand c'est nécessaire.

### 6.1.3. Elimination de choix PHP

PHP est un langage faiblement typé contrairement au Java, d'où on ne peut jamais garantir que les types de données employées décrivent correctement les données manipulées. Ce qui aboutit

à compliquer certaines tâches et à augmenter le taux d'erreurs surtout pour un système complexe comme le notre.

Ce qui implique que Java nous constitue le bon choix comme le développement de notre projet est complexe. Plutôt que les normes définies précisément à Java facilitent la compréhension d'un code et d'accroître l'efficacité et la communication pour tout équipe.

#### **6.1.4. Elimination de .Net**

.Net n'était jamais une option pour moi grâce à deux raisons fondamentales qui sont le coût et le concept du système fermé.

Les licences Visual Studio (non-Express) sont très chères et malgré la présence d'autres IDEs gratuits comme Mono, à notre région les développeurs ne sont jamais familiarisés avec ces outils qui sont considérablement différentes de l'interface graphique du Visual Studio. D'où le coût nécessaire pour que les membres d'équipe deviennent tout aussi productifs, soit avec un IDE gratuit soit avec Visual studio 2010 Ultimate, sont bien élevés.

## **6.2. Recours à l'utilisation d'un cadre de développement**

Approfondir le concept de réutilisation dans le processus de développement est un facteur essentiel de son succès, "on ne doit jamais réinventer le roue", d'où j'ai utilisé un cadre de développement.

On connaît que le développement d'applications peut être facilité grâce à l'utilisation de la programmation orientée objet et les modèles de conception, les boîtes à outils prêts et les bibliothèques proposant des utilitaires basiques et services lors de codage.

L'objectif d'un cadre de développement est de capitaliser le savoir-faire et faciliter la mise en œuvre des fonctionnalités de l'ouvrage permettant au développeur de se concentrer sur les tâches spécifiques à l'application en cours de développement plutôt qu'à des tâches techniques récurrentes telles que l'architecture de base de l'application, l'accès aux données, la journalisation des événements, la sécurité (authentification et gestion des rôles), la compilation, et le paramétrage de l'application... Ce qui aboutit à accroître sa productivité dramatiquement.

De même, l'utilisation de cadre de développement facilite et organise l'intégration d'un nouveau membre dans l'équipe, même s'il a une expérience bien limitée, ce qui me donne quelque tranquillité de point de vue ressources humaines.

Malgré tous les avantages d'utilisation d'un cadre de développement, le temps de prise en main du cadre de développement par les développeurs peut se heurter à certaines difficultés. Mais ces difficultés peuvent être confrontées par le bon choix du cadre de développement qu'il doit avoir, en plus de sa richesse et performance. La minimale complexité, la riche documentation et les projets exemples de démarrage peuvent simplifier dramatiquement la phase d'apprentissage.

### **6.3. Pourquoi le cadre de développement Play?**

Play est un cadre de développement à source ouverte écrit en Java et Scala permettant la réalisation d'applications web. Elle vise à optimiser la productivité des développeurs en utilisant le paradigme convention plutôt que configuration et en supportant complètement le modèle-vue-contrôleur (MVC) comme modèle architectural et "Share Nothing" comme architecture HTTP.

#### **6.3.1. Convention plutôt que configuration**

Je pense que ce principe est à l'origine de la simplicité du Play et il me présente un motif essentiel pour mon choix de cadre. Convention plutôt que configuration (connu aussi sous le nom de codage par convention) signifie essentiellement qu'un développeur doit seulement indiquer les aspects non conventionnels de la demande. Par défaut, les systèmes informatiques, les bibliothèques et les cadres de développement devraient avoir un comportement raisonnable. Cette convention vise à réduire le nombre de décisions dont les développeurs ont besoin pour faire gagner la simplicité mais ne pas nécessairement perdre la flexibilité.

Une des conséquences des systèmes respectant le principe de « convention plutôt que configuration » est que leurs utilisateurs peuvent se sentir obligés à suivre une certaine méthodologie pour attribuer une lisibilité maximale de code et par suite une collaboration ultime les différents développeurs du projet.

### 6.3.2. Modèle-Vue-Contrôleur (MVC)

Play suit le modèle architectural MVC avec tous ses avantages et points forts. Il nous offre une grande souplesse pour l'introduction de toutes nouvelles technologies. Cette séparation qui existe entre le client, le serveur et le SGBD permet une distribution efficace des tâches entre les différents membres de l'équipe guidée par leur spécialisation et compétences montrés sur chaque tiers de l'architecture, présentation associée au client (vues), fonctionnelle liée au serveur (contrôleurs) ou données liée au serveur de base de données (modèle) .

### 6.3.3. Architecture de Ne Rien Partager HTTP

L'architecture « Ne Rien Partager » est une architecture distribuée dans laquelle chaque nœud est indépendant et autonome et il n'y a pas un seul point de discorde au sein du système. Plus particulièrement, aucun des nœuds ne partage aucune mémoire ou aucun disque de stockage ce qui reflète positivement sur la performance de notre application.

## 6.4. Avantages de cadre de développement Play

Le cycle de développement du Play représente l'un de ses essentiels points forts. Avec Play, on corrige juste le code, on retourne à notre navigateur et recharge la page web tout simplement pour un résultat instantané sans écrire les commandes typiques de développement web J2EE, compiler, déployer et redémarrez le serveur...

- ❑ Play supporte plusieurs IDE (Éclipse, IntelliJ, NetBeans), il supporte notre préférable IDE Eclipse ;
- ❑ Play a une collection infinie de bibliothèques à faire tout ce qu'on veut, comme il est juste java et en plus il a une vaste gamme bien organisée des modules fournis par sa communauté sur son site <http://www.playframework.com> ;
- ❑ fonctionne sur la JVM, qui a des années d'expérience et d'optimisation ;
- ❑ il est vraiment un paquet complet: il est livré avec tout ce qu'on a besoin pour développer une application web, compilateur, serveur web, base de données en mémoire, documentation, gestionnaire de base de données intégrée pour SQL et



NoSQL<sup>9</sup> SGBD, gestion de cache local et distribué, évolutions, gestion des dépendances, apache mail, tests unitaires, soutien de l'IDE...

- ❑ développement à zéro configuration de l'environnement, il suffit de décompresser Play Zip pour commencer à développer ;
- ❑ il a une excellente documentation, grande et rapide communauté grandissante et beaucoup d'exemples d'applications pour se démarrer ;
- ❑ il a un système de modèles efficaces et propres basés sur Groovy comme langage d'expression. Il fournit un modèle d'héritage supportant le bon pratique Master Page<sup>10</sup>,
- ❑ Play peut être déployé sur n'importe quel conteneur de servlet, également il peut être livré avec son propre serveur léger. De même il est amical avec les technologies Cloud, il travaille sur gae, Heroku, playas, cloudbee, dotCloud, etc.
- ❑ il possède une collection croissante sans cesse des modules ;
- ❑ l'un de principaux avantages de Play, est que son code source est assez lisible, compréhensible et non plus conçu. Même sans beaucoup de connaissances Java, il est possible d'avoir un œil sur le code, de le comprendre, et commencer à contribuer au projet ;
- ❑ il est beaucoup plus facile à apprendre en comparant avec autres cadres de développement java comme Spring et Ruby on Rails... Il a les meilleurs paramètres par défaut et une très simple configuration ;
- ❑ ses messages d'erreur sont très compréhensibles, on n'a pas besoin de Google chaque fois que quelque chose ne fonctionne pas. Play nous montre le code source et la ligne exacte contenant le problème, même dans les vues.

---

<sup>9</sup> NoSQL signifie "Not Only SQL", littéralement "pas seulement SQL". Ce terme désigne l'ensemble des bases de données qui s'opposent à la notion relationnelle des SGBDR. À l'origine, servant à manipuler des bases de données géantes pour des sites web de très grande audience tels que Google, Amazon.com, Facebook ou eBay, le NoSQL s'est aussi étendu par le bas après 2010

<sup>10</sup> Le Technique du Master Page est de créer une disposition cohérente et unique pour tous les pages web d'une application. Il contient le header, le footer, le menu et tous les css et les JavaScripts commun.

## 6.5. Choix de Système de Gestion de Base de Données

Due aux limites budgétaires et aux restrictions sécuritaires imposés par l'administration réseau du client, MySQL me parait la meilleure option pour l'application côté serveur et clients internes du logiciel comme il est le système de base de données à source ouverte le plus populaire du monde, il a une licence générale publique et soutenu par une énorme communauté active. Il est multi plate-forme (Linux, Windows, Solaris) ce qui implique une flexibilité de déploiement. De point de vue Sécurité, l'installation MySQL est très sûre que tous les mots de passe des bases de données sont stockés sous forme cryptée et la sécurité de serveur MySQL est reliée directement à la sécurité du réseau interne de la chambre, donc elle est parmi leurs responsabilités.

D'autre part, pour la clientèle externe du logiciel, je recours à l'utilisation de duplication des données justes nécessaires à suivre la progression de leurs demandes de tests et la consultation des annonces du laboratoire dans le but d'éliminer tout risque d'attaque externe sur le serveur hôte menaçant l'arrêt du service interne du laboratoire.

Pour cette base de données d'une part, et pour répondre aux futures exigences imposées par notre logiciel comme «logiciel en tant que service» (Saas) d'autre part, je recours à l'utilisation d'Amazon SimpleDb.

### 6.5.1 Pourquoi Amazon SimpleDb?

#### Amazon SimpleDB

C'est un Cloud plate-forme, partie de l'Amazon Web Services, hautement disponible, sécurisée, flexible et NO-SQL qu'on peut utiliser sans se soucier de l'infrastructure d'approvisionnement, de disponibilité, de la performance, des schémas et de la gestion des index. Il assure une couche gratuite pour une limite déterminée. De même, SimpleDb offre une capacité pratiquement illimitée à la demande, à un coût très bas et acceptable tout en prenant soin de toutes les tâches d'administration de base de données nécessaires. Due à sa présence, on n'a plus besoin de construire des plates-formes coûteuses complexes.

#### La Base de Données NO-SQL

SimpleDb me représente la solution stratégique de gestion de données pour mon logiciel comme il utilise le No-SQL.

Les bases de données relationnelles exigent que les schémas doivent être définis avant d'ajouter des données. Une base de données SQL a besoin de savoir ce qu'on stocke à l'avance ce qui contredit et correspond mal à la méthodologie agile que nous adaptons parce que chaque fois qu'on termine des nouvelles fonctionnalités, le schéma de notre base de données doit souvent changer.

Les bases de données NoSQL sont construites pour permettre l'insertion de données sans schéma prédéfini. C'est pourquoi il est facile de faire des changements importants en temps réel, sans se soucier des interruptions de service, ce qui signifie que le développement est plus rapide, l'intégration de code est plus fiable et le temps d'administration de base de données est moins faible.

### **6.5.2 Pourquoi Siena comme couche de gestion des données (Data Management Layer)**

Siena est une couche de gestion de données à source ouverte sous la licence Apache 2.0. Il fournit une API<sup>11</sup> commune pour SQL / NoSQL. Il justifie la transition lors du changement de base de données de SQL à NoSQL et vice-versa en étant capable de réutiliser (presque) le même code sans presque aucune modification ce qui m'offre une flexibilité ultime dans la phase de déploiement m'aidant à soutenir mes différents clients dans l'échelle temporelle et économique tout en supportant l'aspect générique du mon projet.

Siena est conçu suivant le modèle ActiveRecord qui apporte une approche simple et intuitive pour gérer les objets Java par rapport aux entités de base de données. Consultez-vous dans la page suivante le schéma informatif (figure 31) résumant notre utilisation du SQL et NO-SQL dans notre système.

---

<sup>11</sup> API est un acronyme pour Application Programming Interface. C'est un ensemble de fonctions, procédures ou classes qui servent de façade par laquelle un logiciel offre des services à autres logiciels. Il est plus souvent accompagné d'une description qui spécifie comment des programmes consommateurs peuvent se servir des fonctionnalités du programme fournisseur.

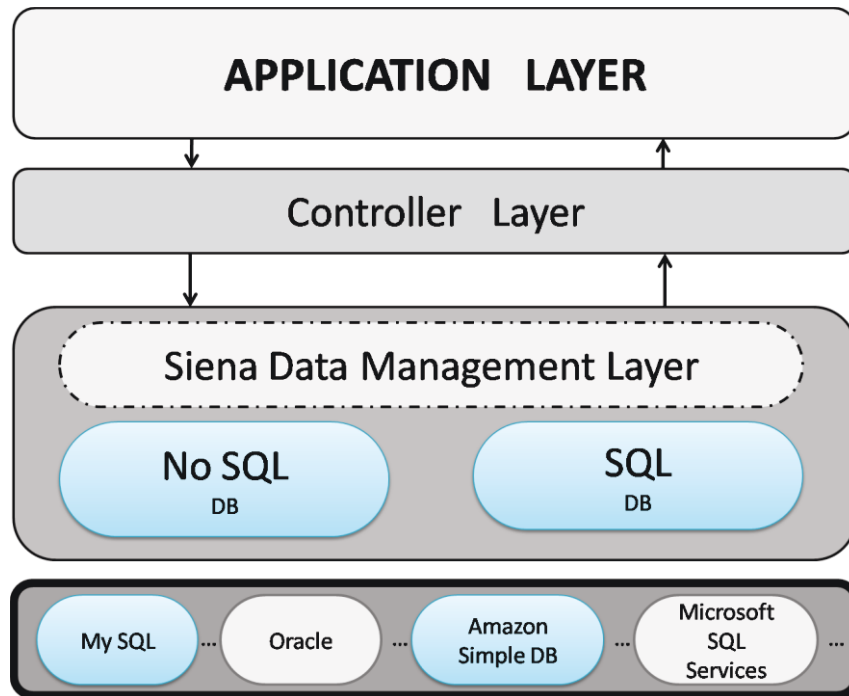


Figure 37 : Siena data management layer

Siena abstrait la logique nécessaire pour accéder aux données quel que soit le système de base de données cible en offrant des fonctionnalités d'accès aux données communes afin de rendre l'application plus facile à configurer, à évoluer et à entretenir. C'est un seul JAR, ne demande aucune dépendance externe;

Dans le chapitre suivant, on va détailler la phase d'installation du système et notre plan de sauvegarde de données.

## Chapitre 7 : Installation

### 7.1 Déploiement comme Application Autonome Play

Le cadre Play est conçu pour être réactif, léger, apatride (stateless), et consomme des ressources (CPU, mémoire, threads) minimales comme il est pleinement RESTFULL et ne maintient pas une session Java EE par connexion.

Après recherche et consultation de plusieurs forums d'expertises en ligne, j'ai décidé qu'on n'a pas besoin d'un serveur récipient d'application. La façon la plus simple et la plus robuste est de déployer simplement mon système est de déployer chaque composant (module) comme une application autonome. Le serveur HTTP intégré peut servir des milliers de requêtes HTTP par seconde de sorte qu'il ne sera jamais le goulot d'étranglement des performances. En outre, il utilise un modèle de thread plus efficace (où un conteneur de servlet utilise un thread par demande).

Je pense que de cette façon, en utilisant le même environnement que nous avons utilisé au cours du processus de développement, un grand nombre de bogues peut être découverte. Je pense que de cette façon, en utilisant le même environnement que nous avons utilisé au cours du processus de développement, un grand nombre de bugs peut être découverte.

### 7.2 Installation du Système comme Service

Au Liban, Windows gagne la popularité du marché initial sur Linux comme système d'exploitation soit sur la machine client soit sur la machine serveur. Le serveur imposé par la chambre a un système d'exploitation Windows. De même, son équipe IT est spécialisée en Microsoft donc elle a une pauvre expérience en java et les techniques à source ouverte. D'où on doit faciliter la phase de déploiement pour assurer la maximale stabilité possible à notre système.

Sur les systèmes Windows, les applications Java sont exécutées dans une fenêtre de commande. Cela nécessite que la fenêtre de commande soit ouverte tout le temps en cours d'exécution sur le bureau. Avoir tout simplement le risque d'appuyer sur la mauvaise touche sur l'invite de commande et de tuer l'application Java.

Windows a la capacité d'exécuter des applications en arrière-plan en tant que service pour résoudre ces problèmes. Les services sont lancés au démarrage du système ce qui augmente

considérablement la sécurité et la stabilité globale du système, car il n'est pas possible pour un technicien d'arrêter, démarrer ou jeter sauf s'il est un administrateur sur la machine.

Le problème est que Java tout seul ne peut pas être exécutée en tant que service. Après recherche, je recourus dans la phase de production au logiciel gratuit YAJSW, sous licence d'Apache, qui assure l'enveloppement de mon application java et son exécution comme service Windows ou Linux démon indépendamment de plate-forme Windows/Linux, ce qui supporte mon projet d'être générique et adaptable à l'infrastructure pour tout client potentiel.

Cependant, on peut rencontrer un seul et petit problème avec YAJSW, le service est lancé mais l'application pourrait être encore en démarrage et n'est pas prête. Mais dans notre situation, ce problème mineur est résolu comme mon application est la seule application Java sur la machine hôte, profitant seule de la performance de la machine Java virtuelle JVM et comme elle est lancée par défaut et rarement arrêtée.

### 7.3 Plan de Sauvegarde de Données

Pour tout système informatique, la sauvegarde (backup en anglais) des données est une opération primordiale surtout pour notre type de client. La sauvegarde physique et matérielle des données est la responsabilité de la chambre et son équipe IT. Il doit protéger les données des risques de vol ou d'incendie, couvrir toute défaillance et dysfonctionnement du matériel.

Notre responsabilité se résume par l'aspect logiciel du plan de sauvegarde de données. MySQL, notre système de gestion de données adapté, est mené d'un outil efficace de sauvegarde « mysqldump » qui peut être utilisé pour sauvegarder ou transférer une base de données ou une collection de bases de données sous forme des fichiers au format sql, xml ou csv. Mais c'est un programme sur la ligne de commande ce qui nécessite l'intervention d'un technicien. De même nous pouvons facilement faire ce travail en utilisant un outil graphique de MySQL comme « MySQL Workbench » qui assure l'administration et la maintenance de base de données à travers une interface graphique dans un environnement de développement unique. Mais je crois que ces deux méthodes ont un point de faiblesse majeure, c'est la nécessité d'intervention humaine, surtout que les statistiques montrent que l'erreur humaine est l'une des principales causes de perte de données avec 32%. D'où je recourus à l'automatisation de la sauvegarde

comme il empêche tout oubli des sauvegardes et assure la réplication instantanée des données et la capacité à spécifier la fréquence de sauvegarde assez facilement à partir de quelques paramètres en code java. On exécute des commandes sur « mysqldump » en utilisant « java.lang.Runtime » qui permettent de stocker les bases de données sous forme des fichiers « sql ».

L'exécution de ces commandes est planifiée à l'aide du planificateur d'emplois Quartz. On a développé des Jobs avec « play.jobs.Job » et on les exécute périodiquement en dehors de toute requête http selon une fréquence bien déterminée spécifié selon l'importance de données et leur rythme de changement en utilisant l'annotation « Every » avec les expressions flexibles et puissants de CRON<sup>12</sup> de la bibliothèque puissante à source-ouverte Quartz intégrée par le cadre Play.

On a développé une interface graphique avec lequel l'IT de la chambre peut spécifier et changer facilement les dates et la fréquence de sauvegarde sans la nécessité de nous contacter. Les paramètres nécessaires à cette opération sont sauvegardés dans un fichier de configuration dans le répertoire du projet. Profitant de l'architecture de composition adaptée par le projet, on peut planifier la sauvegarde de chaque module du système à part en donnant à l'IT sur terre, ayant la capacité à une meilleure décision, un outil simple à utiliser avec une grande marge de flexibilité.

---

<sup>12</sup> C'est un outil UNIX qui permet aux utilisateurs des systèmes d'exécuter automatiquement des scripts, des commandes ou des logiciels à une date et une heure spécifiées ou selon un cycle défini à l'avance.

## Chapitre 8 : Test et Maintenance

Le test est une étape importante pour arriver à un produit qui répond au cahier de charges prévue et pour augmenter la qualité du système développé. Cette phase, nécessite la présence effective des différents acteurs du projet, maîtrise d'œuvre et maîtrise d'ouvrage.

Notre stratégie consiste d'intégrer la phase de test dans le cycle de développement, dès la première phase où on recourt à la validation de processus métiers saisies après la récolte des besoins clients. Pour nous, la fonction test constitue un système de suivi et de pilotage de la qualité du projet. Elle respecte le modèle itératif de développement adapté. Elle aide à évaluer la situation réelle du projet à tout moment, la comparer à la situation prévue au plan d'exécution et à prendre les décisions nécessaires pour corriger la situation, si des écarts sont observés ou prévus.

Notre processus de test sera schématisé dans la figure ci-dessous :

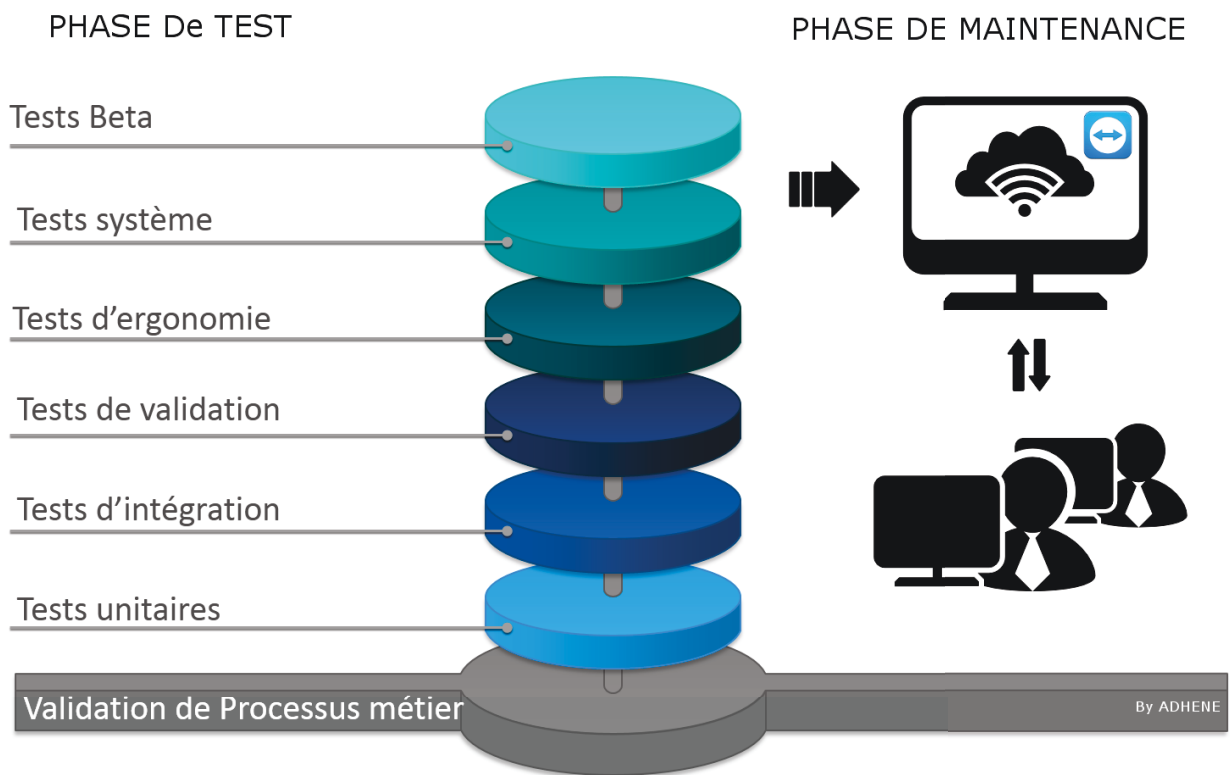


Figure 38: Processus de test



**Elle est formée de 7 étapes successives de test détaillées ci-dessous :**

**Tests de code source :** tester le code par son développeur plusieurs fois avant l'ajouter au répertoire du projet et signaler comme prêt en utilisant surtout la technique de boîte blanche<sup>13</sup>. A cette étape, les erreurs et les problèmes identifiés doivent être corrigés le plus vite possible et durant le temps dédié à la tâche correspondante sur KUNAGI. Si le développeur responsable de ce code est incapable à résoudre ce bogue ou problème il doit l'ajouter comme « obstacle » de la tâche correspondante sur KUNAGI.

**Tests unitaires :** faire tester les parties du logiciel par leurs développeurs

**Tests d'intégration :** tester pendant l'intégration

**Tests de validation/recette :** permet de vérifier si le système répond aux exigences fonctionnelles client décrites dans le cahier de charges.

**Test d'ergonomie:** permet d'évaluer l'utilisabilité du système selon les principes d'ergonomie de l'interface utilisateur détaillées précédemment dans la chapitre quatre au paragraphe 4.4.

**Tests système:** tester dans un environnement proche de l'environnement de production

**Tests Bêta:** faire tester par le client sur le site de production

## 8.1. Outils de gestion de bogues YouTrack

YouTrack est un logiciel commercial basée sur web. Il permet la suivie des bogues et des problèmes d'un système informatique dans la phase de développement et de maintenance. JetBrains est un leader en matière de conception et gestion de bogues des logiciels. Il a 75 000 clients dans le monde, parmi lesquels Apple, LinkedIn, Siemens, et La Banque de l'Amérique. Il offre une licence gratuite pour une équipe de 11 personnes et déduction de 50% pour toute entreprise en démarrage quel que soit la taille de son équipe ce qui le rend une option.

---

<sup>13</sup> **Test « Boite blanche »**

Ce test consiste à analyser la structure interne du programme en déterminant les chemins minimaux. Afin d'assurer que: Toutes les conditions d'arrêt de boucle ont été vérifiées; Toutes les branches d'une instruction conditionnelle ont été testées; Les structures de donne interne ont été testées (pour assurer la validité).

Parmi les avantages multiples de ce logiciel, j'aime éclaircir sa capacité à créer une nouvelle problématique par courrier électronique ce qui nous offre un contrôle robuste sur le système dans la phase de maintenance surtout avec notre plan de la télémaintenance via Teamviewer. L'application est menée d'un code Java, utilisant la bibliothèque « Apache Commons Email », intégrée dans le cadre Play, qui envoie un e-mail à notre IT sur « it@adhene.com » relié à YouTrack en cas d'erreur. Ce qui permet l'enregistrement instantané d'un bogue quand il aurait lieu et par suite l'interaction rapide de notre équipe. De même, cet outil est assez puissant dû à son mécanisme de recherche basée sur des requêtes utilisant une syntaxe simple et très similaire à la langue naturelle et son mécanisme de rapport.

## **8.2. Téléassistance et Télémaintenance avec Teamviewer**

On commence la phase de maintenance par cinq visites, suffisant pour rendre notre client confiant de notre support, et puisque rien n'est comme le contact direct et face-à-face pour fournir l'information et comprendre les réactions de notre client.

ADHENE est une entreprise débutant, la taille de l'équipe de développement nous obligeait à trouver une solution qui nous coûte moins chers de point de vue ressource humaine, temps et coût. Également on doit maintenir le niveau du service après-vente comme maintenance et assistance à l'utilisation du système d'où on a adopté la technique de la télémaintenance et téléassistance avec le logiciel pionnière de partage de bureau et de collaboration sur Internet « TeamViewer »<sup>14</sup>.

### **8.2.1. Télémaintenance**

Dès la première visite chez le laboratoire, on installe le système sur une machine serveur dédiée à notre projet durant la phase de test et les premiers trois mois de maintenance pour avoir le droit et l'autorité de contrôler entièrement et d'effectuer des opérations de maintenance

---

<sup>14</sup> C'est un logiciel propriétaire de bureau à distance, disposant de fonctions de téléassistance, de télémaintenance, de télé-administration, de rencontres en ligne, de conférences Web et de transfert de fichiers.

(sauvegarde, analyse antivirus, défragmentation, mise à jour, etc.) même plutôt sans avertir l'utilisateur pour une meilleure efficacité.

### **8.2.2. Téléassistance**

Pour simplifier la compréhension des conseils par téléphone, renseigner et à assister notre client sur l'utilisation du système le plus vite possible, on recourt à adapter le télé-assistant dans notre phase de maintenance qui visualise l'écran de l'interlocuteur afin de le guider plus aisément. Un logiciel d'accès à distance comme un bureau distant plutôt destiné à la télémaintenance est souvent utilisé pour visualiser l'écran de la personne assistée.

Le protocole de la téléassistance diffère souvent du celui de la télémaintenance. En effet, pour des raisons évidentes de sécurité et de confidentialité et en accord avec la personne demandant assistance, l'outil ne sera donc pas paramétré de la même manière que pour une télémaintenance qui consiste à contrôler entièrement le poste distant pour effectuer des opérations de maintenance sans spécialement en avertir l'utilisateur. De cette façon, la personne assistée sera avertie de l'appel distant et devra confirmer la connexion du télé-assistant à son poste informatique. Aucune intervention directe ne devra être effectuée par le télé-assistant au risque de dépasser les termes du contrat de téléassistance.

### **Contrat de maintenance**

Après les trois mois de maintenance gratuite, notre client peut signer un contrat de maintenance et/ou de téléassistance s'il sent le besoin de pour un coût supplémentaire. Un plan de maintenance basé sur nombre de visite a lieu par défaut selon le contrat du projet.

TeamViewer permet un accès facile, rapide et sécurisé à distance aux systèmes Windows, Mac et Linux. Il utilise une clé RSA publique/privée de 1024 bits et une clé AES 256 bits pour le chiffrement de session.

### **8.3. Documentation Logicielle**

La documentation d'un projet a une importance primordiale: c'est l'outil de communication et de dialogue entre les membres de l'équipe projet et les intervenants extérieurs (membre des

instances de pilotage, chef de projet, utilisateurs, etc...). Elle assure aussi la pérennité des informations au sein du projet.

C'est pourquoi deux documentations étaient nécessaires:

1. documentation interne sur l'architecture et la conception ;
2. manuelle utilisateur: Explique comment le logiciel fonctionne, et/ou comment on doit l'employer.

### **8.3.1. Documentation Interne sur l'architecture et la conception**

Afin d'organiser la gestion de la documentation produite par projet, il convient au préalable d'identifier tous les types de documents relatifs aux diverses étapes d'un projet, de les référencer de manière homogène pour ensuite définir un mode de gestion commun à tous les projets;

Ces documents ne décrivent pas comment programmer une fonction (routine) particulière, ou même pourquoi cette fonction particulière existe sous cette forme, mais expose les exigences générales qui motivent l'existence d'une telle fonction. Un bon document d'architecture est court sur les détails mais dense sur l'explication. Il peut suggérer des approches pour des conceptions de plus bas niveau, mais laisse les études d'exploration effectives à d'autres documents.

Alors que pour la documentation de niveau technique, c'était accompli par nos programmeurs qui emploient l'expression « documentation logicielle ». Lorsqu'ils développent du logiciel, le code source est insuffisant à lui seul. Il doit y avoir du texte qui l'accompagne pour décrire les différents aspects du fonctionnement attendu. Cette documentation est habituellement incluse dans le code source lui-même de telle sorte qu'elle soit facilement accessible à qui serait amené à le traverser.

### **8.3.2. Manuel de l'Utilisateur**

À la différence de la documentation sur le code, les documents utilisateurs sont généralement assez éloignés du code source du programme, et décrivent simplement comment il est employé.

Il y a trois grandes manières d'organiser la documentation utilisateur :

- ❑ Tutoriel : On considère qu'une approche par tutoriel est la plus utile pour un nouvel utilisateur. Dans cette méthode l'utilisateur est guidé à chaque étape d'accomplissement des tâches particulières.

- ❑ Thématique: Pour un utilisateur intermédiaire, on emploie généralement une approche thématique, dans laquelle les chapitres ou sections se concentrent sur un domaine d'intérêt particulier.
- ❑ Liste : Le type final de principe d'organisation est celui dans lequel les commandes ou les tâches sont simplement listées par ordre alphabétique, souvent via des indices croisés. Cette dernière approche est d'un intérêt très élevé pour des utilisateurs avancés qui connaissent exactement quelle sorte d'information ils recherchent. Un grief universellement exprimé par les utilisateurs au sujet de la documentation logicielle est qu'elle n'adopte que l'une de ces trois approches à l'exclusion des deux autres.

Un des obstacles souvent affronté dans un manuel utilisateur c'est le syndrome du bricoleur du dimanche. Le syndrome du bricoleur du dimanche est illustré par la boutade suivante : Comment faire pour qu'un utilisateur d'applications n'ouvre pas un document ? Il suffit d'écrire « mode d'emploi » dessus. C'est là un des malheurs des modes d'emploi : souvent, ils ne sont pas lus.

À partir de là, certains en sont venus à considérer que le seul intérêt d'un manuel d'utilisation était de donner du contrôle à l'utilisateur: une série de questions et réponses qui met au point les préoccupations de l'utilisateur et répondent à elles. Similaire au centre aide du FACEBOOK.

Le manuel d'utilisation était créé donc de la manière qu'il soit:

- ❑ utile : une information exacte, mais inutile, ne fait que renforcer le sentiment d'inutilité et gêne la recherche de l'information pertinente;
- ❑ agréable : quant à la forme, la clarté (présentation, style), mais aussi en mettant en avant les préoccupations de l'utilisateur et non pas les caractéristiques techniques du logiciel.

## Chapitre 9 : Outils de Communication

Construire le code est vraiment juste une petite partie de l'effort total du projet, même si c'est ce que la plupart des gens associent l'ensemble du processus, car il est le plus visible. D'autres pièces sont aussi de valeur plus importante que nous avons déjà dépassé ci-dessus à savoir les exigences, l'architecture, l'analyse, la conception et les tests. Une bonne pratique pour le code de la construction implique constructions quotidiennes et tests.

Je crois qu'un codage peut être efficace et évolutif seulement s'il suit les normes et les lignes directrices de codage appropriées. Ce chapitre détaillé notre parcours à appliquer ces normes fondamentales pour toute entreprise informatique, surtout si elle appuie sur le paradigme de réutilisabilité comme facteur clé de son succès comme la nôtre.

### 9.1. Kunagi pour Gestion de Projet

Comme on a évoqué précédemment dans le chapitre cinq au paragraphe 5.2, nous adaptons SCRUM comme méthodologie agile ce qui rend Kunagi l'outil convenable de gestion de projet pour notre équipe. C'est un service gratuit fondé sur le Web qui offre un cadre de développement de gestion de projets agiles, complétant SCRUM par une sélection des meilleures pratiques afin de couvrir tous les besoins de gestion de projets. En outre, il favorise la collaboration avec une simple interface utilisateur tout en s'exécutant dans un navigateur et donc être accessible aisément de toute station.

### 9.2. Subversion

Pour toute équipe de développement, le logiciel de gestion de sources et de contrôle de versions est une nécessité. Subversion est le logiciel leader à source ouverte, il permet l'accès souple aux fichiers d'un système, en local ou via un réseau.

Je recours à l'utilisation de CollabNet Subversion Edge pour l'entreprise comme il est une source ouverte sous le certificat de la géante Apache Subversion. Il est mené d'une interface Web qui simplifie l'administration et la configuration du serveur, ce qui minimise la perte de temps nécessaire à cette étape gênante et perpétuelle. Il se connecte de façon transparente à tous les produits de CollabNet et la CollabNet Cloud pour une expansion future.

### 9.3. Tortoise SVN

C'est un client Windows à source ouverte pour le système de contrôle de versions Apache™ subversion, facile à utiliser.

C'est un client Windows à source ouverte pour le système de contrôle de versions Apache™ subversion, facile à utiliser. En s'intégrant dans l'explorateur de Windows, il offre aux utilisateurs de Windows une interface graphique permettant de réaliser la plupart des tâches offertes par SVN en ligne de commande, et de visualiser instantanément l'état (à jour, modifié, en conflit...).

### 9.4. Règles de Codage

Les règles de codage sont un ensemble de règles à suivre pour uniformiser les pratiques de développement logiciel, diffuser les bonnes pratiques de développement et éviter les erreurs de développement "classiques" au sein d'un groupe de développeurs.

Les règles de codage s'articulent autour de plusieurs thèmes, les plus courants étant:

- ❑ l'organisation des fichiers du code source,
- ❑ les conventions de nommage, ou règles de nommage,
- ❑ les commentaires et la documentation du code source,
- ❑ recommandations sur la déclaration des variables,
- ❑ recommandations sur l'écriture des instructions, des structures de contrôle et l'usage des parenthèses dans les expressions.

Les règles de codage permettent d'assurer une meilleure lisibilité du code en utilisant le même style de codage par tout membre de l'équipe en évitent les constructions qui rendent le code difficile à lire ou à modifier. Elles permettent également d'éviter les erreurs liées au langage pouvant donner des résultats incorrects, pouvant entraîner l'effondrement des systèmes ou des failles de sécurité. Certaines règles sont également liées aux buts poursuivis par le projet logiciel: portabilité, contraintes mémoires, etc.

Les règles de codage participent à la qualité logicielle. Ainsi, plus l'importance des développements est élevée, le plan de maintenance est long, plus les besoins en règle de codages sont nécessaires.

### **9.5. Utilisation de Plugin Checkstyle**

Checkstyle pour l'éclipse est un outil qui automatise le processus de vérification de style du code Java pour aider le développeur à épargner le temps nécessaire à cette tâche ennuyeuse mais importante. Ce plugin est idéal pour une entreprise qui souhaite appliquer une norme de codage dans son équipe pour augmenter au maximum la lisibilité de code et, par conséquent, l'interopérabilité entre les différents membres de l'équipe qui représente la clé de succès essentielle pour toute équipe surtout une SCRUM équipe. Checkstyle offre par défaut deux configurations SUN Check et SUN Check (Eclipse) et en plus il est hautement configurable pour créer une configuration altérée lors du besoin.

### **9.6. Erreurs Communs des Bases de Données**

Comme une équipe cohérente de développement, on croit qu'une perte de temps arrive quand un membre prend du temps pour résoudre un même problème technique ou erreur de code déjà confronté par un autre membre. Ça sera un gaspillage de temps et un manque de l'esprit d'équipe. D'où on a créé une base de données partagée sur la Subversion où tout membre de l'équipe y ajoute une erreur ou problème confronté, muni d'une description (trace d'erreur, capture d'écran...), précision du couche (modèle, vue, contrôleur, Configuration...), et de la solution qui l'a trouvée. Tout écrit d'une manière simple, claire et précise, avec des conseils ou recommandations aidant à éviter l'occurrence de ce problème ou erreur.

### **9.7. Librairie des Codes Exemples Communs**

Pour avoir un code lisible et facile à maintenir, nous développons une librairie contenant des codes exemples, testés et optimisés le plus possible, pour les fonctionnalités les plus fréquentes



dans notre cycle de développement, concernant essentiellement les fonctions CRUD<sup>15</sup> d'une entité et le mécanisme de pagination, ce qui aboutit à une diminution bien considérable de taux d'erreurs dans nos applications développées.

## 9.8. Conseils pour l'Optimisation du code Java

Inspirant par le célèbre livre "Effective Java: Programming Language Guide"<sup>16</sup>, on a résumé une liste des règles efficaces qui peuvent nous aider à écrire un code clair, performant, correct, lisible, robuste et réutilisable. Ces règles sont bien résumées sur un fichier partagé sur Subversion pour être facilement accessible par tout membre de l'équipe.

## 9.9. Librairie des Services Génériques

Cette librairie contient toutes les méthodes générales qui peuvent servir toute application comme les méthodes de calcul, de format, d'IO<sup>17</sup>...

Ce module constitue une librairie cumulative totalement réutilisable avec toute application ayant lieu au futur en profitant de l'effort de programmation qui a lieu et en assurant un gain de temps bien considérable dans le développement d'une application prochaine. Ce module s'ajoute à la librairie de code de l'entreprise après un test bien efficace. Bien sûr, ces méthodes sont utilisées d'une manière complètement abstraite par toute application, par un simple appel.

Dans le chapitre suivant, on va résumer rapidement quelques outils de développement et bibliothèques utilisés lors de l'implémentation.

---

<sup>15</sup> Acronyme informatique anglais CRUD pour Create (créer), Read (lire), Update (mettre à jour), Delete (supprimer). CRUD désigne les quatre opérations de base pour la persistance des données, en particulier le stockage d'informations en base de données.

<sup>16</sup> "Effective Java: Programming Language Guide" par Joshua Bloch qui a passé des années en développant, implémentant, et en utilisant le langage de programmation Java chez Sun Microsystems

<sup>17</sup> Acronyme de Input/Output, c'est toute activité qui implique que des données entrent et sortent de programme : saisie utilisateur, print sur un terminal, lecture ou écriture sur une socket, sur un fichier, sur cookies...

## Chapitre 10 : Outils de Développement

### 10.1. Éclipse

Je constate qu'il n'y a pas une caractéristique qui rend l'éclipse bien meilleure qu'autre IDE libre (comme Netbeans, IntelliJ etc....) mais, il a l'avantage d'être le plus utilisé donc il a une très vaste communauté et gamme des plugins. Je l'ai choisi essentiellement, car j'avais une forte expérience avec et je connais beaucoup de ses trucs et ses points forts.

C'est un environnement de développement (IDE<sup>18</sup>) historiquement destiné au langage Java, même si grâce à un système de plugins, il peut également être utilisé avec d'autres langages de programmation.

Figurant parmi les grandes réussites de l'Open source, Éclipse est devenue un standard du marché des logiciels de développement, intégré par de grands éditeurs logiciels et sociétés de services comme IBM, Aptan etc. Techniquement, l'IDE Eclipse est composé de plugins, ce qui le rend extrêmement personnalisable.

### 10.2. iText PDF Java Library

Pour la facturation, la publication des résultats des tests et les contrats ISO des échantillons, soit pour l'impression soit pour la déposition dans l'archive du laboratoire, on a besoin d'une librairie flexible et puissante de manipulation des documents. Après recherche, j'ai choisi iText PDF. C'est une bibliothèque à source ouverte permettant de créer et de manipuler des documents PDF<sup>19</sup> en favorisant la génération dynamique à partir du code java de tout document nécessaire au processus de demandes de test.

---

<sup>18</sup> Acronyme pour integrated development environment. C'est un ensemble d'outils pour augmenter la productivité des programmeurs. Il comporte un éditeur de texte destiné à la programmation, des fonctions qui permettent, par pression sur un bouton, de démarrer le compilateur ou l'éditeur de liens ainsi qu'un débogueur en ligne, qui permet d'exécuter ligne par ligne le programme en cours de construction...

<sup>19</sup> Acronyme de Portable Document Format. PDF est devenu un standard ouvert d'échange de documents électroniques géré par l'ISO (International Organization for Standardization). Lorsque vous convertissez des documents, formulaires, images et pages web au format PDF, ils se présentent exactement comme s'ils étaient imprimés.

Cette librairie nous permet de:

- ❑ servir PDF à un navigateur ;
- ❑ générer des documents dynamiques à partir de fichiers XML ou des bases de données ;
- ❑ utilisez de nombreuses fonctions interactives de PDF ;
- ❑ ajouter des signets, des numéros de page ;
- ❑ diviser et concaténer des pages PDF ;
- ❑ automatiser le remplissage de formulaires PDF ;
- ❑ ajouter des signatures numériques pour un fichier PDF.

### 10.3. jFree Charts

jFreeChart est une bibliothèque graphique Java, source ouverte, distribuée sous les termes de la licence GNU, 100% gratuits ce qui rend facilement pour les développeurs d'afficher des graphiques statistiques de qualité professionnelle avec différentes modes de représentations typiques tels que des histogrammes, des camemberts, des courbes de répartitions... Dans notre projet, il nous sert au développement du module des statistiques (Dashboard) et le mécanisme de l'évaluation du service de laboratoire par sa clientèle. JFreeChart est sans doute, dans le monde Java, l'outil de génération de graphiques le plus connu et le plus réputé, comme il est facile à utiliser et à intégrer dans une application, qu'elle soit installable, soit utilisée par un navigateur web. Ses résultats fournis sont d'une qualité remarquable.

jFreeChart comprend:

- ❑ API cohérente et bien documentée, elle soutient un large éventail de types de graphiques ;
- ❑ conception souple facile à étendre qui peut être exploitée à la fois au côté serveur et au côté client de l'application ;
- ❑ support de nombreux types de sortie, y compris les composants Swing, des fichiers d'image (y compris PNG et JPEG), et les formats vecteurs de fichiers graphiques (y compris PDF, EPS et SVG)...

### 10.4. Zxing pour Codes à barres

Pour la génération des étiquettes à code-barres responsables de la poursuite des échantillons à

la phase d'analyse, saisie des résultats et leur livraison au client, je recours à l'utilisation de la librairie Zxing à source ouverte. C'est une librairie en java qui prend en charge le décodage et la génération de codes-barres (comme QR Code, PDF 417, EAN, UPC, Aztec, Datamatrix, Codabar) dans des images. Il manipule des multi-formats 1D/2D barres de traitement d'images à Java. Il sert à lire et décoder des codes à barres, sans communiquer avec un serveur. Toutefois, le projet peut être utilisé pour coder et décoder des codes à barres sur les ordinateurs de bureau et serveurs ainsi sur les appareils mobiles.

De même, cette librairie est utilisée extensivement dans le module de stock.

### **10.5. ImgScalr**

« Imgscalr » est une efficace (accélération matérielle) librairie, source ouverte à java, aussi simple et facile à utiliser ce qui sert au redimensionnement des images, reflétant positivement sur la performance et l'ergonomie des pages web.

# Chapitre 11 : Développement de Site Web Comme Outil Commercialisation

## 11.1. Développement d'un site web comme outil de commercialisation

De nos jours, une présence web est l'outil le plus simple et facile qui nous permet de diffuser notre produit et montrer la qualité de notre travail au Liban et en dehors de nos limites géographiques surtout avec une intégration complète du contenu de site web avec les médias sociaux dans le cadre de la campagne de commercialisation.

Le but essentiel de site web est de refléter que notre solution est conçue pour répondre aux besoins de n'importe quel laboratoire (petite, de taille moyenne ou grande) que pour les exigences ou budgets. Il est construit sur des composants autonomes qui peuvent fonctionner indépendamment où des composants supplémentaires peuvent être facilement attachés à la base du système LIMS. Donc, notre client peut ajuster uniquement les composants qui répondent à ses besoins et son budget tout en maintenant ses besoins futurs.

**Le système va avoir 3 versions:**

- ❑ **Version de base:** Le système contient seulement les composants bases de LIMS qui couvrent uniquement les processus métier opérationnel de la chaîne de valeur de n'importe quel laboratoire.
- ❑ **Version complète:** Le système comprend les composants déjà cités du processus métier opérationnel, mais offre aussi du support, et un site interactif avec les clients, où les clients seront capables de consulter ses résultats et de donner une évaluation sur le laboratoire.
- ❑ **Créez votre propre version:** Ici, le client peut choisir les composants qu'il souhaite ajouter à son LIMS pour adapter le système spécifiquement à ses besoins.

## 11.2. Interface Utilisateur Basé sur le Concept Widget

En tant que développeurs, nous apprécions la réutilisation des composants visuels et d'interaction prêts à intégrer « widget <sup>20</sup>» dans l'interface utilisateur surtout que notre formation nous jamais permet de jouer ce rôle au niveau demandé, même qu'il est bien significatif dans le

---

<sup>20</sup> Le terme «widget » désigne tout élément de l'interface utilisateur qu'il sert à l'affichage de l'information.

développement des applications informatiques surtout web. De même, l'interface d'une application doit être fondée sur des modèles communs que les utilisateurs comprennent déjà intimement.

D'où on a commencé dès le début à accumuler notre propre et riche librairie des Widgets et nous travaillons perpétuellement à l'enrichir.

La structuration de notre librairie est schématisée dans l'image suivant:

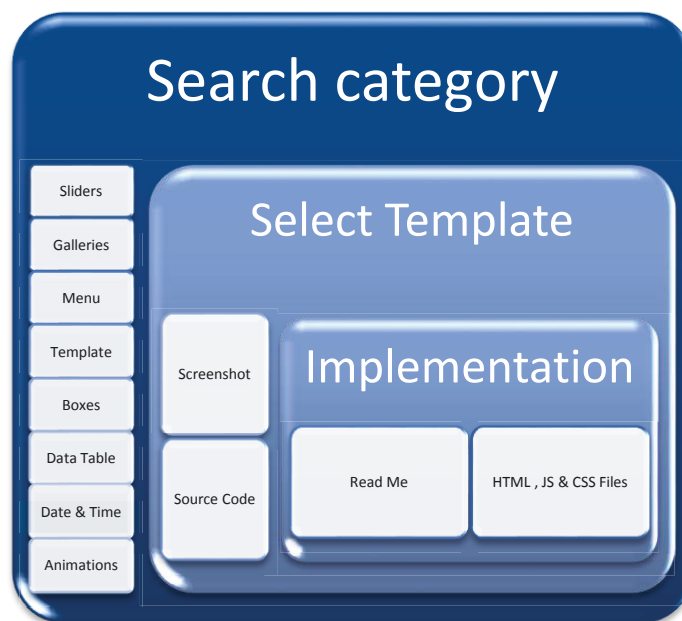


Figure 39: Schéma de structure de librairie

## 11.3 Plan de Déploiement

### 11.3.1. Déploiement en Nuage avec Heroku

Heroku est un service de l'informatique en nuage de type plate-forme en tant que service (PaaS) supportant le déploiement des applications web. L'infrastructure utilisée par Heroku repose sur le géant AWS<sup>21</sup>. Il est un des premiers CLOUD plateformes qui supportent Java, Scala et Play Framework. Le prix d'Heroku est fondé sur les ressources que le client utilise mené d'un plan de base gratuit. Avec Heroku, on n'a pas besoin de penser à des serveurs à tous.

<sup>21</sup> **Amazon Web Services (AWS)** est une collection de services informatiques distants (aussi appelés Services Web) fournis via internet par le groupe américain de commerce électronique Amazon.com.

Nous gérons notre application en utilisant l'outil de ligne de commande Heroku et nous déployons notre code en utilisant le système de contrôle de révision populaire Git.

### **11.3.2. Amazon SimpleDb comme SGBD**

On utilise Amazon SimpleDb détaillé précédemment comme SGBD pour notre site web.

### **11.3.3. Amazon S3 comme Service de Stockage**

Le service de stockage AWS S3 est un « magasin très résistant et disponible » et peut-être utilisé pour sauvegarder de manière fiable le contenu de l'application telle que les fichiers médias, les images etc., dynamiques et statiques. Il nous permet de décharger notre infrastructure de stockage entier et offre une meilleure évolutivité, fiabilité et vitesse de stockage des fichiers.

S3 est un autre type de service de fichiers et a une sémantique différente des autres services fondés sur des fichiers. Tous les fichiers sont stockés dans S3 seaux qui agissent comme un récipient de haut niveau tout comme un répertoire. Tous les fichiers envoyés à S3 appartiennent à un seau dont le nom de seau doit être unique sur l'ensemble du système d'Amazon.

AWS S3 est un complément idéal pour le système de fichiers éphémère d'Heroku.

## **11.4. CSS3 site web multiplateformes**

Le plus grand problème dans notre métier est sans aucun doute le parc navigateurs. Trop de versions de navigateurs, trop de moteurs différents à prendre en compte. Comment faire en sorte que l'affichage soit identique sur Chrome, Internet Explorer, Opera, Safari et Firefox?

Pour essayer de répondre à cette problématique on a recours à utiliser les techniques et les outils suivants:

### **11.4.1. Réinitialisation du CSS**

La réinitialisation du CSS (ou «Reset CSS») est souvent un ensemble de règles CSS comprimé (minified) qui remette le style de tous les éléments HTML à une base cohérente.

Chaque navigateur possède son propre défaut "feuille de style", qu'il utilise pour faire des sites Web. Par exemple, la plupart des navigateurs font par défaut les liens bleus et les liens visités pourpres, donnent aux tableaux une certaine quantité de frontière et le rembourrage, s'appliquent des tailles de police variables à H1, H2, H3, etc. C'est pourquoi les boutons diffèrent

d'un navigateur à l'autre. Évidemment, cela crée un problème pour les auteurs de CSS, qui ne peuvent pas travailler sur la façon de rendre leurs sites se ressemblent dans tous les navigateurs. L'utilisation d'un reset CSS peut forcer tous les navigateurs d'avoir tous ses modèles remis à zéro, évitant ainsi les différences autant que possible.

#### 11.4.2. Le validateur de balisage W3C

Le validateur de balisage est un service gratuit fourni par le W3C, permettant de vérifier la validité des documents Web.

Le validateur W3C<sup>22</sup> compte sur le soutien de la communauté pour l'hébergement et le développement.

Le service de validation de balisage est un validateur par le World Wide Web Consortium (W3C) qui permet aux utilisateurs d'Internet de vérifier les documents HTML et XHTML pour balisage. La validation par balisage est une étape importante en vue d'assurer la qualité technique des pages web. Cependant, n'est pas une mesure complète des standards du Web conformité. Bien que la validation W3C est importante pour la compatibilité du navigateur et la convivialité du site, il n'a pas été confirmée quel effet cela a sur l'optimisation des moteurs de recherche.

Alors que le W3C et les autres validateurs HTML et XHTML évalueront les pages codées dans ces formats, un validateur indépendant comme le validateur CSS<sup>23</sup> du W3C est nécessaire pour confirmer qu'il n'y a aucune erreur dans le « Cascading Style Sheet» associé. Les validateurs CSS fonctionnent d'une manière similaire aux validateurs HTML et XHTML en ce qu'ils appliquent les normes CSS actuels aux documents CSS référencés.

---

<sup>22</sup> Le W3C est un sigle utilisé pour définir le World Wide Web Consortium qui est une organisation non lucrative permettant définir des standards pour les technologies liées aux web.

<sup>23</sup> CSS "Cascading Style Sheet" : Il s'agit des feuilles de style adjointes aux fichiers HTML, permettant de séparer les données de leur présentation.



## CONCLUSION

En conclusion de ce mémoire, j'aimerais signaler la livraison du projet selon le calendrier prévue dans le cahier de charges et l'acceptation du logiciel par le maître d'ouvrage. Due au changement de location de notre client, le processus de formation sur le logiciel était lent et son lancement a connu un retard. Une conférence sera organisée par la chambre de commerce dédiée au lancement du projet va bientôt avoir lieu.

Due à la dégradation de la situation économique continue au Liban sous l'effet de tensions politiques internes et les répercussions du conflit syrien et régionale, notre produit connaît un écart de calendrier de commercialisation surtout due au retard de conférence dédiée au lancement du projet au niveau du Liban par la chambre de commerce comme notre ville Tripoli a connu plusieurs épisodes de violence en 2013 et 2014.

Maintenant on travaille à renouveler nos mécanismes de commercialisation et accéder à de nouveaux marchés soit à travers notre site web [www.adhene-acculab.com](http://www.adhene-acculab.com), les médias sociaux et à travers de firmes de marketing en lignes.

Malgré la taille énorme de ce projet, un budget limité lui était dédié, donc pour révéler ce problème on a concentré notre effort de développement sur les processus opérationnels du laboratoire et on a développé seulement une base pour tous les processus de support et pilotage permettant le bon fonctionnement de l'activité cœur du laboratoire. Simultanément, on a établi avec notre client une relation à long termes exprimée par une chaîne successive des contrats où notre client peut ajouter à tout moment une de nos produits génériques qui lui assure la gestion de ses processus de support d'une manière ultime comme gestion de stock, ressources humaines, comptabilité, communication, commercialisation, documentation, etc. ou demander l'ajout d'un module supplémentaire sur mesure ayant son propre cahier de charges.

## SUMMARY

At the end of this project, I would like to focus on the fact that this software has been delivered on time as planned, and accepted by the laboratory administration. But due to the relocation of our client's lab, the training on the software part was relatively slow, and caused a delay of software launching. A conference, scheduled by the trading room to launch the system, is decided to take place soon.

The series of violence between 2013 and 2014 in Tripoli, as well as the war in Syria have had a great impact on the economics in Lebanon. As a result launching the system by the chamber of Trading had been rescheduled several times. This caused a great delay in the marketing process. At the present time we are working on renewing our marketing techniques, by targeting new markets, using our website [www.adhene-acculab.com](http://www.adhene-acculab.com) as a marketing tool, social media and various online marketing firms.

We faced some challenges regarding the large-scale of the project and the tight budget, so we concentrated on the operational process of the laboratory, while only covering the basis of the support processes. By doing so, on one hand we didn't compromise the main flow of the laboratory, on the other hand we've established a long term relationship with the client through a chain of contracts, and enabled him to add one of our generic product or ask for a custom made product. By doing so we will give him the ultimate control of the support process, like stock management, human resources management, finance, communication, ads, documentation, etc.

## GLOSSAIRE

<b>LIMS (SIGL)</b>	Acronyme de Laboratory Information Management System signifiant <i>système informatique de gestion de laboratoire</i> , SGL) est un progiciel de gestion intégré utilisé dans les laboratoires et permettant la traçabilité des échantillons, la gestion des utilisateurs, des instruments, des stocks, des approvisionnements fournisseurs, le suivi des produits et équipements utilisés, la définition des tournées de prélèvement, la saisie des prélèvements sur le terrain, ...
<b>Développement logiciels agiles</b>	<b>de</b> Groupe de méthodes de développement logiciel basé sur le développement itératif et incrémental, où les exigences et les solutions évoluent grâce à la collaboration entre l'auto-organisation, les équipes inter-fonctionnelles et le client. Il favorise la planification de l'adaptation, le développement de l'évolution et de la livraison selon une approche itérative.
<b>UML</b>	Unified Modeling Langage : Langage mondialement connu permettant de visualiser, spécifier, construire et documenter les artefacts d'un système. Il ne s'agit pas d'une méthode mais d'un langage permettant d'utiliser toute méthode orientée objet.
<b>BPMN</b>	Acronyme de Business Process Modeling Notation est un standard pour la modélisation de processus métier qui fournit une notation graphique permettant de définir des processus métier dans un diagramme de processus métier (BPM), basé sur une technique d'organigrammes très proche de celle utilisée par les diagrammes d'activité UML.
<b>AMDD</b>	Acronyme de Agile Modeling Driven Development, AMDD est que, s'agit de créer des modèles agiles qui sont à assez bon et animent et intensifient les efforts de développement. Sa stratégie se résume par commencer un projet agile par une modélisation initiale et que la modélisation est encore présente dans chaque itération de la construction.
<b>Règle de gestion (Business Rule)</b>	Un règle de gestion définit ou contraint certains aspects de l'entreprise en décidant de vraie ou fausse. Les règles de gestion peuvent s'appliquer aux personnes, les processus, le comportement des entreprises et des systèmes informatiques d'une organisation, et sont mis en place pour aider l'organisation à atteindre ses objectifs.

<b>OMG</b>	Acronyme de "Object Management Group", c'est une adhésion ouverte internationale, ayant des normes technologiques non lucratif. Fondée en 1989, les normes OMG sont entraînés par les fournisseurs, les utilisateurs finaux, les établissements universitaires et les organismes gouvernementaux. Les normes de modélisation d'OMG, y compris l'Unified Modeling Language (UML), Business Process Modeling BPM et Model Driven Architecture (MDA)...
<b>USAID</b>	USAID is the lead U.S. Government agency that works to end extreme global poverty and enable resilient, democratic societies to realize their potential.
<b>SaaS</b>	Acronyme de <i>Software as a Service</i> (SaaS) est un modèle d'exploitation commerciale des logiciels dans lequel ceux-ci sont installés sur des serveurs distants plutôt que sur la machine de l'utilisateur. Les clients ne paient pas de licence d'utilisation pour une version, mais utilisent généralement gratuitement le service en ligne ou payent un abonnement récurrent.
<b>PBS</b>	Acronyme de Product Breakdown Structure, c'est une décomposition hiérarchique et organisée du produit. c'est l'outil principal de structuration du chef de projet puisqu'il constitue les livrables du projet.
<b>SGBD</b>	Acronyme de Système de Gestion de bases de données : Outil assurant la gestion des relations et des requêtes sur les tables d'une base de données, comme par exemple Sybase. Dans le langage courant, on utilise souvent ce terme tout simplement comme synonyme de « base de données ».
<b>Master Page</b>	Le Technique du Master Page est de créer une disposition cohérente et unique pour tous les pages web d'une application. Il contient le header, le footer, le menu et tous css et javascript commun.
<b>API</b>	API est un acronyme pour Application Programming Interface. C' est un ensemble de fonctions, procédures ou classes qui sert de façade par laquelle un logiciel offre des services à autres logiciels. Il est plus souvent accompagnée d'une description qui spécifie comment des programmes consommateurs peuvent se servir des fonctionnalités du programme fournisseur.
<b>SQL</b>	Structured query language : Langage commun à toutes les bases de données relationnelles, servant à effectuer des requêtes sur une base de données pour en extraire des informations.

<b>NO-SQL</b>	NoSQL signifie “Not Only SQL”, littéralement “pas seulement SQL”. Ce terme désigne l’ensemble des bases de données qui s’opposent à la notion relationnelle des SGBDR. À l'origine, servant à manipuler des bases de données géantes pour des sites web de très grande audience tels que Google, Amazon.com, Facebook ou eBay <sup>1</sup> , le NoSQL s'est aussi étendu par le bas après 2010
<b>Object-DB</b>	Une base de données à objets (anglais object database) est un stock d'informations groupées sous formes de collections d'objets persistants. Dans une base de données à objets les informations sont regroupées sous forme d'objets.
<b>SGBD</b>	Système de Gestion de bases de données : Outil assurant la gestion des relations et des requêtes sur les tables d’une base de données, comme par exemple Sybase. Dans le langage courant, on utilise souvent ce terme tout simplement comme synonyme de « base de données ».
<b>Active-Record</b>	En génie logiciel, le patron de conception (design pattern) active record est une approche pour lire les données d'une base de données. Les attributs d'une table ou d'une vue sont encapsulés dans une classe.
<b>CRUD</b>	Acronyme informatique anglais CRUD pour <i>Create</i> (créer), <i>Read</i> (lire), <i>Update</i> (mettre à jour), <i>Delete</i> (supprimer). CRUD désigne les quatre opérations de base pour la persistance des données, en particulier le stockage d'informations en base de données.
<b>IO</b>	Acronyme de Input/Output, c’est toute activité qui implique que des données entrent et sortent de programme : saisie utilisateur, print sur un terminal, lecture ou écriture sur une socket, sur un fichier, sur cookies...
<b>IDE</b>	Acronyme pour <i>integrated development environment</i> ). C'est un ensemble d'outils pour augmenter la productivité des programmeurs. Il comporte un éditeur de texte destiné à la programmation, des fonctions qui permettent, par pression sur un bouton, de démarrer le compilateur ou l'éditeur de liens ainsi qu'un débogueur en ligne, qui permet d'exécuter ligne par ligne le programme en cours de construction...
<b>PDF</b>	Acronyme de Portable Document Format. PDF est devenu un standard ouvert d'échange de documents électroniques géré par l'ISO (International Organization for Standardization). Lorsque vous convertissez des documents, formulaires, images et pages

web au format PDF, ils se présentent exactement comme s'ils étaient imprimés.

---

<b>widget</b>	Le terme «widget » désigne tout élément de l'interface utilisateur qu'il sert à l'affichage de l'information.
<b>CSS</b>	CSS "Cascading Style Sheet" : Il s'agit des feuilles de style adjointes aux fichiers HTML, permettant de séparer les données de leur présentation.
<b>W3C</b>	Le W3C est un sigle utilisé pour définir le World Wide Web Consortium qui est une organisation non lucrative permettant définir des standards pour les technologies liées aux web.

---

## RÉFÉRENCES

- [1] L'Orient Le Jour, 2014. Le marché des technologies de l'information en hausse de 5 % en 2013 au Liban, <http://www.lorientlejour.com/article/852232/le-marche-des-technologies-de-linformation-en-hausse-de-5-en-2013.html>
- [2] Patrick Grégoire, 2011, Faire le grand saut vers SCRUM, pourquoi?, <http://www.patrickgregoire.com/index.php/2011/08/faire-le-grand-saut-vers-scrum-pourquoi/>
- [3] Fueldeck website, 2013, 25 Resources for Best Practices in Client Side Web Development, <http://www.fueldeck.com/blog/25-resources-for-best-practices-in-web-development-2013/>
- [4] Susan Weinschenk, Ph.D., 2010, The Psychologist's View of UX Design, <http://uxmag.com/articles/the-psychologists-view-of-ux-design>
- [5] Jan Van den Nieuwenhof, 2011, How to Convince Your CFO to Use Scrum, <http://leblogdumanagementdeprojet.com/2011/02/02/comment-convaincre-votre-directeur-financier-dutiliser-scrum/>
- [6] Scott Ambler + Associates, 2011-2012, Agile Modeling (AM) - Effective Practices for Modeling and Documentation, <http://www.agilemodeling.com>
- [7] Amazon Web Services (AWS) Website, SimpleDb documentation, <http://aws.amazon.com/simplied>
- [8] Amazon Web Services (AWS) Website, S3 ocumentation, <http://aws.amazon.com/s3/>
- [9] Joshua Bloch, 2001, Effective Java: Programming Language Guide, Addison Wesley
- [10] Audrey Nemeth, Vladimir Borisov, 2006, Agile Modelling in Software Engineering
- [11] Site web <http://opensource.org/osd>

## TABLE DES FIGURES

Figure 1. Infographie du projet .....	19
Figure 2. Les processus opérationnels du laboratoire .....	24
Figure 3. Les processus de support du laboratoire .....	24
Figure 4. Les processus de pilotage du laboratoire .....	25
Figure 5. Processus de saisir d'une demande de test .....	26
Figure 6. Les sous-processus du processus de saisir d'une demande de test .....	27
Figure 7. PBS diagramme du système - partie 1 .....	32
Figure 8. PBS diagramme du système - partie 2 .....	32
Figure 9. La division en sous-systèmes .....	33
Figure 10. Diagramme de division aux sous-systèmes .....	34
Figure 11. Division en fonctions de sous-système de tests et de types des échantillons .....	35
Figure 12. Couches de données du sous-système de tests et de types d'échantillons .....	36
Figure 13. Diagramme de classe de Sous-système de Tests et Types Des échantillons .....	38
Figure 14. Division en fonctions de sous-système de tests et de types des échantillons .....	39
Figure 15. Diagramme de classe de sous-système Client .....	40
Figure 16. Division en fonctions de sous-système de demandes de tests.....	41
Figure 17. Liste de demandes de test .....	41
Figure 18. Recherche d'un ancien client.....	42
Figure 19. Saisir les informations d'un nouvel client .....	42
Figure 20. Définition des échantillons à tester .....	43
Figure 21. Définition les tests à réaliser de l'échantillon .....	43
Figure 22. Saisir des détails de la transaction .....	44
Figure 23. La publication des rapports d'un échantillon .....	44
Figure 24. Diagramme de classe de Moteur du système.....	45
Figure 25. Diagramme de classe de Sous-système de gestion de ressources humaines .....	46
Figure 26. La page principale du module de stock .....	47
Figure 27. Diagramme de classe de stock - partie 1 .....	49
Figure 28. Diagramme de classe de Sous-système de Stock – partie 2 .....	49
Figure 29. Interface du client .....	51



Figure 30. Le curseur de nouvelles dans le site web.....	51
Figure 31. Diagramme de cas d'utilisation du sous-système de communication.....	52
Figure 32 : Diagramme de classe de sous-système des Equipements.....	53
Figure 33 : Les Bonnes Pratiques de modélisation agile (Agile Modeling AM) .....	55
Figure 34 : Organigramme de l'équipe de développement du projet.....	62
Figure 35. Le plan chronologie du projet .....	64
Figure 36. Le diagramme de Gant.....	65
Figure 37 : Siena data management layer.....	75
Figure 38: Processus de test .....	79
Figure 39: Schéma de structure de librairie.....	93