

# **Mémoire de Projet de Fin d'Étude**

Présenté en vue d'obtenir :

**Le diplôme d'Ingénieur d'État**

**Spécialité : Génie Informatique**

**Option : Qualité Logiciel**

**N° : 2020/GI**

## **Etude et Conception de deux solutions installables et gérables par Microsoft Dynamics CRM**

Réalisé Par :

**M. BENMIMOUN Ilyass**

Effectué à :

**SQLI – Oujda**

Encadré à l'ENSAO par :

**Prof. BELHASSANI Hanae**

Encadré à SQLI par :

**Mme. LAMZIRA Naoual**

**Mr Boufaddi Mahmoud**

Soutenu le 18 juillet 2020 devant les membres du jury :

**M. BOUCHENTOUF Toumi**

**Mme. BOUGROUN Zineb**

**Mme. BELHASSANI Hanae**

**Année Universitaire 2019-2020**



# Remerciements

Je tiens tout d'abord à remercier les membres du jury pour leur présence, pour leur lecture attentive de ma thèse ainsi que pour les remarques qu'ils m'adresseront lors de cette soutenance afin d'améliorer mon travail. Je voudrais exprimer ma sincère gratitude et remerciements à un ensemble de personnes dont l'aide m'a été très précieuse.

Je tiens avant tout à exprimer ma reconnaissance à **Mr BOUCHENTOUF Toumi**, mon chef de filière qui a toujours su m'orienter, me soutenir et me guider tout au long de ces trois années de mon cycle d'ingénieur.

Je remercie tous ceux sans qui cette soutenance ne serait pas ce qu'elle est, aussi bien par les discussions que j'ai eu la chance d'avoir avec eux, leurs suggestions ou contributions. Je pense ici en particulier à mon manager **Mr NAIMI Youssef**.

Mes remerciements vont aussi à tous mes professeurs qui ont fait de ma formation une réussite, qui ont su m'épauler et ont cru en mes compétences en m'encourageant à mettre la barre toujours un peu plus haut.

Merci à mes encadrants de stage **Mme LAMZIRA Noual** et **Mr BOUFADDI Mahmoud** pour m'avoir transmis leur savoir, pour leur aide et tout le temps qu'ils m'ont consacré.

Je souhaite également remercier mon encadrante **BELHASSANI Hanae** pour tous ses efforts, son suivi et pour ses précieux conseils méthodologiques.

Je tiens aussi à remercier la personne que j'admire le plus et qui m'inspire énormément depuis tout petit, qui n'est qu'autre que mon père, il m'a donné tous les outils de la réussite, m'a toujours soutenue et donné les meilleurs conseils qui restent uniques et sans semblable.

Je remercie tout autant ma mère, source d'amour et de tendresse, qui m'a donné la force de persévérer et de ne jamais abandonner quel que soit les difficultés rencontrées.

Je remercie toute ma petite famille, mon frère et ma sœur, en espérant qu'ils seront fiers de ce que je deviens aujourd'hui.

## Résumé

Le présent document constitue la synthèse du travail accompli dans le cadre de mon projet de fin d'études au sein de la société SQLI-Oujda. Dans mon rapport de stage, vous allez découvrir les diverses étapes d'étude, de conception et de réalisation de deux solutions installables et gérables par Microsoft Dynamics 365.

Les objectifs majeurs de mes solutions sont de faciliter la vie aux utilisateurs de l'organisations CRM, en permettant un gain de temps énorme grâce à un triage de données plus efficaces. Ce d'une part, et de l'autre l'amélioration de la communication entre collaborateurs et permettre une diffusion plus fluide de l'information et aussi un objectif primordial. Mes solutions consistent donc, en première solution du tag qui apportera un meilleur filtrage des entités et des enregistrements et qui part conséquent rendra la recherche et le reporting des tâches faciles et moins complexe, et puis une deuxième solution de notification qui permet aux utilisateurs d'être en permanence au courant des différents changement, des dernières informations et des différentes tâches qu'ils leurs ont été attribuer.

La mise en œuvre de deux solutions a eu lieu dans une usine logicielle entièrement Microsoft, principalement C# en utilisant le Framework .NET, ainsi que WebSocket/SinglaR qui simplifie le processus d'ajout de fonctionnalités Web en temps réel aux applications. Power Apps est une suite d'applications, de services, de connecteurs et une plateforme de données qui fournissent un environnement de développement applicatif rapide dans le but de concevoir des applications personnalisées et adaptées aux besoins métier, et le progiciel de Microsoft de gestion de la relation client appelé Microsoft Dynamics CRM « Workflow – Plugins – Entity ... ».

Quant au suivi et pilotage du projet, il a été assuré dans un environnement agile en utilisant la méthode SCRUM, avec un cycle de développement itératif incrémental.

**Mots clés: C#, PowerApps, SCRUM, Microsoft Dynamics CRM, .NET, WebSocket/SignalR, Worflow, Plugins, Entity.**

# Abstract

This document is the result of work done as part of my End of studies Project within SQLI-Oujda. In this internship report, will discover the various stages of study, design and realization of two installable and manageable solutions by Microsoft Dynamics 365.

The main objectives of my solutions are to make life easier for users of CRM organizations, allowing enormous time savings thanks to more efficient sorting of data, on the one hand, and on the other the improvement of communication between employees and allow a more fluid dissemination of information and also a primary objective. My solutions therefore consist, in the first solution of the tag which will provide better filtering of entities and records and which therefore will make the search and reporting of tasks easy and less complex, and then in the second notification solution which allows users to be constantly aware of the different changes, the latest information and the different tasks they have been assigned.

The implementation of two solutions took place in an entirely Microsoft software factory, mainly C # using the .NET Framework, thus WebSocket / SinglaR which simplifies the process of adding real-time Web functionalities to applications, Power Apps is a suite of applications, services, connectors and a data platform that provide a rapid application development environment in order to design custom applications adapted to business needs, and the Microsoft software package for customer relationship management called Microsoft Dynamics CRM "Workflow - Plugins - Entity...".

Accounted for the monitoring and project management, it was provided in an agile environment using the agile project management methodology SCRUM coupled with a iterative and incremental cycle of development.

**Keywords: C#, PowerApps, SCRUM, Microsoft Dynamics CRM, .NET, WebSocket/SignalR, Worflow, Plugins, Entity.**

## ملخص

تمثل هذه الوثيقة ملخص العمل المنجز في إطار مشروع التخرج الخاص بي ضمن شركة "Sqli" وجدة، ومن خلال هذا التقرير سوف نسلط الضوء على مختلف المراحل، من دراسة وتصميم وإنجاز لحلين قابلين للتثبيت وتتم ادارتهما بواسطة "Microsoft Dynamics 365".

وتتمثل الأهداف الأساسية لهذا المشروع في تبسيط الحياة لمستخدمي "CRM" من خلال توفير وقت كبير بفضل الفرز الأكثر كفاءة للبيانات. هذا من جهة، ومن جهة أخرى تحسين التواصل بين المستخدمين والسماح بنشر المعلومات بطريقة أكثر مرونة. ويهدف الحل الأول الى انتقاء أفضل للسجلات والمعلومات وبالتالي يجعل من عملية البحث والفرز أمراً هيناً وأقل تعقيداً، في حين ان الحل الثاني المتعلق بالإشعارات يتيح للمستخدمين اطلعا عامّاً على مختلف المعلومات والمهام التي تم تكلفتهم بها.

تم انجاز هذا المشروع من خلال اعتماد كامل على برامج وتطبيقات "Microsoft" وبشكل خاص في إطار عمل ال "Net." بالإضافة الى.... الذي يتيح مجموعة من وظائف الويب بشكل آني وفوري. كما ان "Power Apps" عبارة عن حزمة من الخدمات والتطبيقات بغية توفير بيئة سريعة في تطوير وتصميم تطبيقات مصممة خصيصاً لتلبية احتياجات برامج ادارة الاعمال والعلاقات ويدعى "Microsoft Dynamics CRM".

بغية إنزال هذا المشروع على أرض الواقع تم استخدام طريقة "Scrum" في إتباع دورة تدريجية و تكرارية مع احترام المنهج المنصوص عليه من قبل الشركة.

الكلمات الدالة:

**C#, PowerApps, SCRUM, Microsoft Dynamics CRM, .NET, WebSocket/SignalR, Worflow, Plugins, Entity.**

## Liste des figures

Figure 1 : Logo du groupe SQLI .....	14
Figure 2 : Les agences du groupe SQLI .....	15
Figure 3 : Quelques clients du groupe SQLI.....	16
Figure 4 : Chiffre d'affaires 2019 .....	16
Figure 5 : Effectif de SQLI .....	16
Figure 6 : Répartition des agences SQLI .....	16
Figure 7 : l'évaluation de qualité chez SQLI.....	18
Figure 8 : conduite du stage .....	23
Figure 9 : Graphe de traitement d'une demande client sous Jira. ....	28
Figure 10 : Interface Jira.....	29
Figure 11 : Diagramme de Gantt .....	31
Figure 12 : cycle de vie de chaque solution.....	32
Figure 13 : Diagramme de cas d'utilisation pour la solution de tag .....	38
Figure 14 : Diagramme de séquence pour le rôle d'administrateur.....	38
Figure 15 : Diagramme de séquence pour le rôle d'utilisateur .....	39
Figure 16 : Diagramme d'activité de la configuration.....	40
Figure 17 : Diagramme d'activité d'ajout.....	41
Figure 18 : Diagramme d'activité de la suppression.....	42
Figure 19 : Diagramme de cas d'utilisation pour la solution de tag .....	43
Figure 20 : Diagramme de séquence pour l'administrateur global .....	44
Figure 21 : Diagramme de séquence pour l'administrateur de centre notification .....	45
Figure 22 : Diagramme d'activité de la gestion des notifications.....	46
Figure 23 : Logo de Microsoft Dynamics.....	48
Figure 24 : Architecture de la solution CRM .....	48
Figure 25 : Microsoft Dynamics CRM Interface. ....	53
Figure 26 : Entité de tag .....	54
Figure 27 : Création d'un nouveau tag.....	54
Figure 28 : Entité de devis contient le web ressource avant l'enregistrement.....	55
Figure 29 : Entité de devis contient le web ressource après l'enregistrement.....	55
Figure 30 : Système de Tag.....	56
Figure 31 : Associe d'un tag a un enregistrement .....	56
Figure 32 : Information sur le tag .....	57
Figure 33 : Système de traduction .....	57
Figure 34 : Interface des notifications .....	58
Figure 35 : Alerte de type annonce simple .....	58
Figure 36 : Alerte de type annonce basée sur des règles .....	59
Figure 37 : Alerte de type annonce basée sur des règles simple .....	59
Figure 38 : Alerte de type annonce basée sur des règles complexes .....	59

## Liste des abréviations

Abréviation	Désignation
API	<b>Application Programming Interface</b>
CRM	<b>Costumer Relationship Management</b>
TFS	<b>Team Foundation Server</b>
VS	<b>Visual Studio</b>
WF	<b>Workflow Foundation</b>
ORM	<b>Object Relational Mapping</b>
DAL	<b>Data Acces Layer</b>
DTO	<b>Data Transfer Object</b>
EF	<b>Entity Framework</b>
IIS	<b>Internet Information Service</b>
CMMI	<b>Capability Maturity Model Integration</b>
IDE	<b>Integrated Developpement Environment</b>
C#	<b>C Sharp</b>
JS	<b>JavaScript</b>
VSTS	<b>Visual Studio Team Server</b>
UML	<b>Unified Modeling Language</b>
PL	<b>Plugins</b>
PA	<b>Power Apps</b>
HTML	<b>Hypertext Markup Language</b>
CSS	<b>Cascading Style Sheets</b>



# Table des matières

Liste des figures .....	7
Liste des abréviations .....	8
Introduction générale .....	11
Chapitre I : Le contexte général du projet .....	13
1. Présentation de l'organisme d'accueil .....	14
1.2 Le groupe SQLI .....	14
1.3 Domaine d'activité SQLI .....	15
1.4 Le groupe SQLI en quelque chiffre.....	16
1.5 L'agence SQLI Oujda.....	17
1.6 Le Skill Centre Microsoft .....	17
1.7 La qualité au sien de SQLI.....	18
2. Présentation du projet.....	19
2.1 Introduction.....	19
2.2 Problématique .....	20
2.3 Solution proposée.....	20
2.4 Objectifs et fonctionnalités attendus .....	20
Chapitre II : Le déroulement du stage .....	22
1. Déroulement du stage .....	23
2. Conduite du projet .....	25
2.1 La méthodologie choisie .....	25
2.2 Déroulement du projet .....	27
2.3 Planification provisionnelle .....	30
3. Choix du cycle de développement du projet .....	32
Chapitre III : La spécification des besoins et conception du système .....	34
1. Méthodologie utilisée.....	35
1.1 Introduction.....	35
1.2 Choix de méthodologie .....	35
2. Besoins fonctionnels.....	35
2.1 Solution Tag .....	35
2.2 Solution Notification .....	36
3. Besoins non fonctionnels.....	36

<b>4. Etude conceptuelle .....</b>	<b>37</b>
4.1 Solution Tag .....	37
4.2 Solution Notification .....	42
<b>Chapitre IV : Etude technique et Réalisation .....</b>	<b>47</b>
1. Etude technique .....	48
2. Outils et technologies utilisés .....	50
3. Environnement de travail .....	52
4. Réalisation .....	54
<b>Conclusion générale.....</b>	<b>61</b>
<b>Bibliographie et Webographie.....</b>	<b>62</b>
<b>Annexes .....</b>	<b>62</b>

## Introduction Générale

Dans le cadre des nombreux projets proposés par notre Manager afin de permettre aux stagiaires de développer leurs connaissances et mettre en situation les différentes facettes des technologies Microsoft ; les projets « Dynamics CRM » sont l'occasion idéale de mettre en pratique tout ce qui a été assimilé durant la formation et de découvrir de nouvelles notions. Ainsi, dans les projets Dynamics CRM, On se retrouve toujours face aux demandes spécifiques et aux exigences particulières, d'où vient la nécessité de développer des fonctionnalités qui n'existent pas par défaut et qu'on ne trouve pas aussi sur le marché, et cela revient au fait que chaque projet est unique en son genre.

Toujours à la recherche de faciliter la tâche à l'utilisateur, notre rôle en tant que développeurs au sien de ce projet consiste à développer deux solutions installables pour Dynamics CRM, l'un va permettre aux utilisateurs de procéder par utilisation de tags dans une organisation CRM et l'autre va le notifier se basant sur son statut d'Agent Customer Service ainsi qu'un ensemble d'autres critères.

Une bonne étude des besoins, et une expression de qualité de ces derniers n'est qu'autre que la clé d'un projet réussi. Nous avons donc effectué une étude générale qui nous a mené vers une détection des besoins fonctionnels et leurs différents traitements, une fois cette étape achevée, l'objectif de la solution est devenu clair, ce qui m'a facilité le travail par la suite. Le but est donc pour la première solution, la représentation thématique des données pour faciliter le repérage et la recherche. En ce qui concerne la deuxième solution, le but est de notifier l'utilisateur Dynamics CRM et de lui permettre de définir des critères pour que les Agents Customer Service soient notifiés. Ensuite nous nous sommes focalisés sur la conception de ses solutions et la présentation d'un ensemble de propositions dans le but d'augmenter la performance de ces dernières solutions. La partie conception terminée, on s'est concentré sur l'implémentation des solutions et des résultats obtenus sur une organisation réelle pour la mise en production en faisant en sorte que ces solutions soient installables sur CRM et en veillant à bien respecter les critères spécifiques à chaque solution. En ce qui concerne les technologies utilisées j'ai opté pour un développement en Dynamics 365, C#, WebSocket/SignalR.

En somme, ce projet consiste à l'implémentation de deux nouvelles solutions innovatrices, et mon rôle a été d'atteindre les objectifs attendus et respecter les normes et les

bonnes pratiques de développement Framework .NET afin de fournir un travail efficient et performant qui fera de ce projet une réelle réussite.

Au long de ce rapport, je vais résumer mon stage en 4 chapitres principaux :

Chapitre 1 : Dans ce chapitre, je présente le cadre général du stage en découvrant l'organisme d'accueil (SQLI-OUJDA) et Microsoft Dynamics CRM, ainsi que le contexte général du stage.

Chapitre 2 : Ce chapitre est consacré au déroulement du stage, la conduite du projet et les démarches adoptées pour le mener à bien.

Chapitre 3 : Dans cette partie, je vais détailler les besoins fonctionnels et non fonctionnels du projet ainsi que les cas d'utilisation du système présentant la conception et l'architecture du projet.

Chapitre 4 : Ce chapitre est dédié à la décortication de la phase de mise en œuvre du projet, sous tous ses détails, à savoir, l'architecture logicielle de la solution, les outils et les environnements de développement. Ainsi qu'un aperçu sur le travail réalisé.

Le rapport va être conclus par un résumé qui englobe toutes les phases achevées dans mon projet chez SQLI.

# Chapitre I :

## Le Contexte Général Du Projet

*Dans ce premier chapitre, On expose le cadre général du stage. Tout d'abord vous auriez droit à la présentation de l'organisme d'accueil, ses secteurs d'activités, sa méthodologie. Par la suite, On se focalisera sur la présentation du projet, dans lequel on précise le contexte du stage, la problématique ainsi que les objectifs visés, et finalement on éclaire le planning suivi lors de la réalisation du projet.*

## 1.1. Présentation de l'organisme d'accueil :

La première partie de ce chapitre donne quelques informations sur le groupe SQLI et décrit ses secteurs d'activités ainsi que ses différents pôles.

### 1.1.1. Le groupe SQLI :

Créé en 1990, le Groupe SQLI est le leader français des sociétés de services en ingénierie informatique, ses 2400 collaborateurs sont spécialisés sur 3 segments de compétence :



Figure 1 : Logo du groupe SQLI

- ✓ Les technologies & usages Internet (Microsoft, Java/J2EE, Open Source...)
- ✓ Les nouvelles offres SAP (Net Weaver, CRM...)
- ✓ La Business Intelligence (BO, Cognos, Open Source...)

Son positionnement de "Grand Spécialiste" unique en France lui permet de proposer une offre globale alliant la capacité de production industrielle d'un groupe international à l'expertise et la souplesse d'un spécialiste, en mesure de donner de la valeur aux innovations des entreprises qu'il accompagne.

Parce que le succès des projets est une priorité, le groupe s'est investi très tôt dans la qualité, la maîtrise du pilotage des projets, les méthodologies CMMI et Agile. SQLI a été la toute première société de services européenne à avoir intégré le modèle de qualité CMMI pour sa production en 2002, et à être certifiée CMMI de niveau 3 dès 2006. Le Groupe déploie actuellement l'ensemble des pratiques de niveaux 4 et 5 à l'ensemble de ses agences.

SQLI s'est spécialisée dans l'accompagnement des entreprises dans l'utilisation des nouvelles technologies et la réalisation des systèmes d'information de la nouvelle génération. Elle est organisée en agences de proximité afin de conserver le maximum de réactivité face aux besoins de ses clients. En prenant compte, au sein même de son organisation, des préoccupations du tissu économique régional, SQLI offre une approche sur mesure aux enjeux spécifiques des entreprises. En plus de ses multiples implantations en France, SQLI a ouvert 32 agences en Europe et à l'internationale : Londres, Lausanne, Genève, Bruxelles, Luxembourg, Amsterdam, Rabat, Oujda ...



Figure 2 : Les agences du groupe SQLI

### 1.1.2. Domaine d'activité SQLI :

Tout en développant son activité, SQLI veille à maintenir une grande diversité de sa clientèle et des secteurs d'activités auxquels elle s'adresse, de façon à contenir le risque de concentration sur un nombre restreint de clients.

Ainsi, SQLI compte plus de 1800 clients, issus de tous les secteurs d'activité à savoir:

- ✓ Les services.
- ✓ L'industrie.
- ✓ Les banques et assurances.
- ✓ Les administrations et services publics.
- ✓ La distribution.
- ✓ L'immobilier.
- ✓ Les transports.
- ✓ Les télécoms.

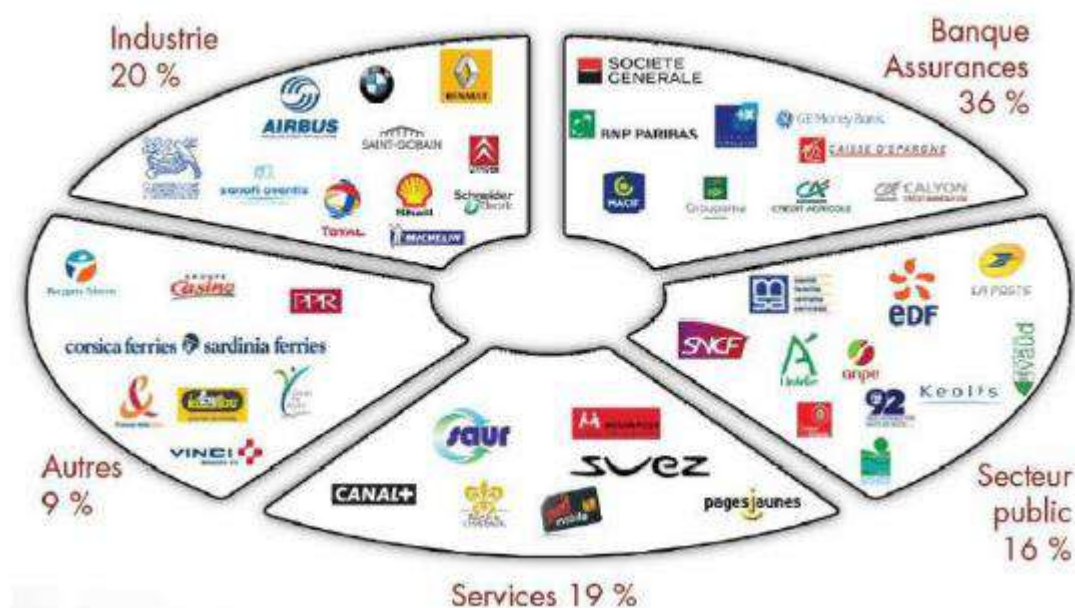


Figure 3 : Quelques clients du groupe SQLI

### 1.1.3. Le groupe SQLI en quelques chiffres

Quelques chiffres clés de SQLI :

- ✓ 28 années d'existence (créée en 1990) ;
- ✓ 20 ans d'expérience des projets e-business ;
- ✓ 50% du chiffre d'affaires au forfait ;
- ✓ Plus de 2400 collaborateurs ;
- ✓ 32 agences dans 12 pays ;
- ✓ En 2019, SQLI a réalisé un chiffre d'affaire de 239 M€.



Figure 4 : Chiffre d'affaires 2019



Figure 5 : Effectif de SQLI



Figure 6 : Répartition des agences SQLI



#### 1.1.4. L'agence SQLI Oujda :

L'Offshore constitue une véritable opportunité de réduction des coûts de développement informatique et permet aux entreprises de répondre à leurs nouvelles contraintes de maîtrise budgétaire.

SQLI propose à ses clients des solutions très compétitives, réalisées au Maroc, tout en garantissant une totale maîtrise qualitative des développements. Adoptant une approche unique fondée sur l'appartenance et la cohésion, SQLI dispose de ses propres centres de développement au Maroc ; à Rabat et Oujda, soit près de 500 collaborateurs travaillant en étroite synergie avec l'ensemble des agences. À l'instar des autres agences, ces deux centres obéissent à la démarche qualité CMMI et à toutes les méthodes et processus communs à la société.

En particulier, le projet de construction de la plate-forme offshore sur le campus technologique SQLI d'Oujda, qui est mené en partenariat avec la région de l'Oriental et l'Université Mohammed 1er d'Oujda, est considéré comme un véritable choix stratégique pour le Groupe. Le campus est d'une grande valeur ajoutée, nous citons par exemple :

- ✓ Une parfaite cohésion dans la conduite des projets, sans rupture dans la chaîne de réalisation.
- ✓ Une culture et des méthodes de travail identiques : CMMI.
- ✓ Des équipes intégrées et impliquées.
- ✓ Des ingénieurs de haut-niveau spécialisés par domaine d'activité

#### 1.1.5. Le SKILL CENTER MICROSOFT :

Dès notre arrivée à SQLI, nous avons été affectés au SKILLS Center Microsoft dont le manager est M. NAIMI Youssef.

Le SKILL Center Microsoft (Anciennement appelé Business Unit Microsoft) a été créé en 2015, et est spécialisée dans le développement des applications .NET, WPF, Web, CRM... Les collaborateurs ont accumulé une expérience considérable dans les traitements métiers de ce genre d'applications au fil du temps, ce qui leur permet de mettre rapidement en place des solutions très efficaces, plus rapidement, et de garantir ainsi la satisfaction des clients.

### 1.1.6. La qualité au sein de SQLI

- **CMMI chez SQLI**

Dans les années 1999 à 2002, SQLI a grandi avec la bulle internet et la croissance fut très importante. Elle a entraîné une dilution du savoir-faire des experts, et progressivement une perte de la maîtrise des projets de la société.

En 2002, face à la crise due à l'explosion de la bulle internet, la nouvelle équipe dirigeante a décidé de faire une rupture stratégique dans la réalisation des projets. L'ambition fut de faire de SQLI la référence en France de réalisation des projets. Après étude, SQLI a décidé de s'appuyer sur CMMI pour mener cette démarche.

Le schéma ci-dessous illustre les principales étapes depuis 2002 :

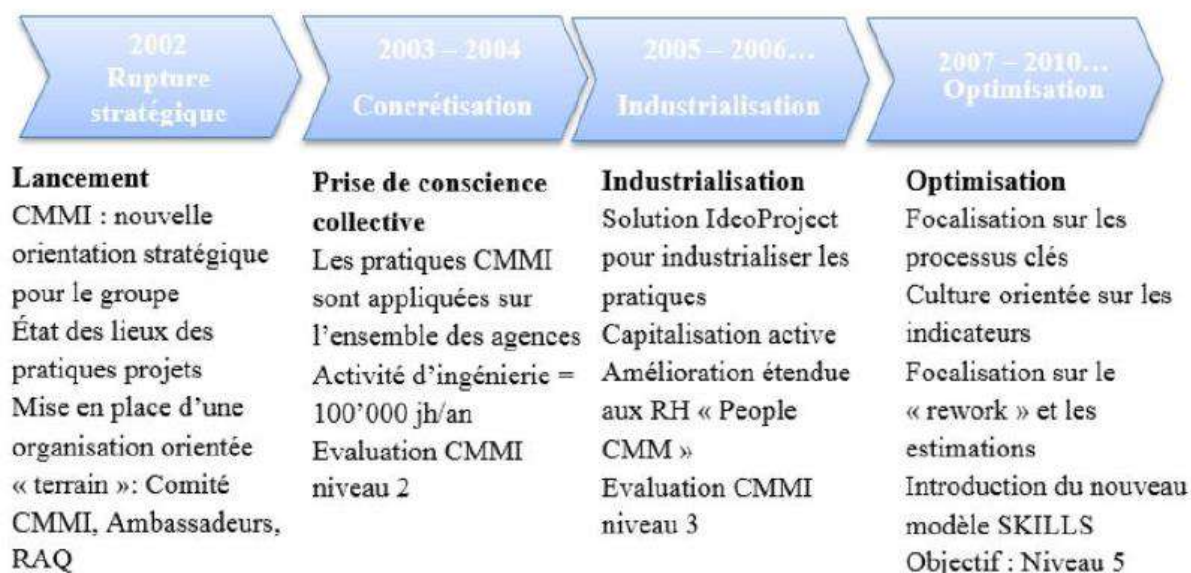


Figure 4 : l'évaluation de qualité chez SQLI

- **SKILLS : un couplage Agilité/Qualité**

Les méthodes SCRUM et CMMI semblent souvent de deux mondes opposés, le premier évoquant la souplesse maximale avec le minimum de documents et le second la rigidité et la documentation de tous les processus. Ce sentiment est dû au fait que l'on confonde CMMI avec les méthodes rigides souvent mise en place. En effet, le fondement de CMMI est de préconiser la mise en place de méthodes, sans pour autant indiquer lesquelles.

Il est donc tout à fait possible de s'inscrire dans une démarche de qualité CMMI en mettant en place des méthodes agiles, ceci en appliquant leurs préconisations à objectifs similaire et éviter les préconisations contradictoires.

## 1.2. Présentation du projet :

### 1.2.1. Introduction

D'une part, toujours à la recherche de faciliter la tâche à l'utilisateur, notre rôle en tant que développeur au sien de ce projet consiste à développer deux solutions installables pour Dynamics CRM, l'un va permettre aux utilisateurs de procéder par utilisation de tags dans une organisation CRM et l'autre va le notifier se basant sur son statut d'Agent Customer Service ainsi qu'un ensemble d'autre critères.

D'autre part, Customer Relationship Management (CRM), ou gestion de la relation client en Français, est la mise en place d'outils et de techniques dans le but de conquérir de nouveaux clients, et de fidéliser les clients existants, en captant leurs informations, puis en les traitants afin de les restituer de manière organisée, dans le but d'avoir une vision globale du potentiel marketing de l'entreprise.

Microsoft Dynamics CRM va ainsi par exemple permettre la gestion d'un processus de vente, la mise en place de campagnes marketing, et ce dans un même endroit. Elle va également permettre la gestion d'accès aux enregistrements contenus dans la base de données, via la gestion de rôles de sécurité.

Par conséquent, ce projet consiste à l'implémentation de deux nouvelles solutions innovatrices, en première solution du tag qui apportera un meilleur filtrage des entités et des enregistrements et qui par conséquent rendra la recherche et le reporting des tâches faciles et moins complexe, et puis en deuxième solution de notification qui permet aux utilisateurs d'être en permanence au courant des différents changements, des dernières informations et des différentes tâches qu'ils leur ont été attribuer.

En somme, ce projet consiste à l'implémentation de deux nouvelles solutions innovatrices, et mon rôle a été d'atteindre les objectifs attendus et respecter les normes et les bonnes pratiques de développement Framework .NET afin de fournir un travail efficient et performant qui fera de ce projet une réelle réussite.

### 1.2.2. Problématique

Une organisation est toujours sujette à un énorme flux de données à analyser et interpréter ou bien trier pour pouvoir les utiliser par la suite. Plus ce flux augmente plus la recherche y devient compliquée, et ce abstraction faite de l'énorme problème de diffusion des informations et communication entre collaborateurs qui accompagne la croissance de l'entreprise, la gestion devient alors une affaire complexe et difficile dont la faisabilité ne fait que décroître, ce qui impacte le travail de tous les collaborateurs sur tous les différents points hiérarchiques de la société, ainsi les délais de production sont prolongés et le rendement est à son tour négativement influencé. Trouver une solution pour gagner du temps au niveau de la recherche et du filtrage des données ainsi qu'une solution pour veiller à la bonne coordination et communication entre tous les membres de l'organisation dans un temps minime devient donc une affaire cruciale.

### 1.2.3. Solution Proposée

L'apport d'une solution adaptatif et évolutif de Dynamics CRM déjà existante sera un vrai gain temps me direz-vous, néanmoins la mise en place d'une solution évolutive "non officiel et non développer par SQLI" peut engendrer des problèmes à cause de la mauvaise conformité des données avec les solutions existantes dans l'organisation d'où l'importance de créer des solutions propres à notre CRM et qui correspondent parfaitement aux attentes de l'organisation concernée.

La première solution consiste à implémenter au sein d'une organisation CRM un système de tags, et la deuxième consiste à la définition de critères afin de permettre à l'Agent Customer Service d'être notifiés. Les deux solutions comptent être installables sur Dynamics 365.

### 1.2.4. Objectifs et fonctionnalités attendus

#### **Objectif : « Solution Tag »**

Le but ultime de la première solution est d'apporter un filtrage facile et efficace aux données, chaque donnée aura désormais un ensemble de tags qui caractérise les thèmes qui lui sont relatifs, ainsi la recherche et le reporting ne seront plus des tâches complexe, des tags prédéfinis seront donc mis à disposition ainsi l'auto-complétions sans oublier de laisser bien évidemment libre cours à l'utilisateur de créer ses propres tags.

**Objectif : « Solution Notification »**

Concernant la deuxième solution est la mise en place de critères pour notifier les utilisateurs de CRM des différentes informations et changements comme “Qualified Leads” ou “Create Account” et donner à l'utilisateur la main pour définir des critères afin que les Agents Customer Service soient notifiés, en ajoutant “entity notification” qui permet la création de sa propre notification en liaison avec d'autre “entity” de CRM. Ces notifications seront journalisées en temps réel afin de garder une traçabilité, qui évitera tout éventuels problèmes ou confusions. Ainsi il permet aux utilisateurs d'être en permanence au courant des différents changement, des dernières informations et des différentes tâches qu'ils leurs ont été attribuer.

**Synthèse :**

Au cours de ce chapitre, qui représente un point de départ de mon projet, j'ai présenté l'organisme d'accueil, ses secteurs d'activités et sa méthodologie. Une entreprise qui accompagne ses clients dans la mise en œuvre et le pilotage de leur transformation digitale. Par la suite, j'ai présenté le projet, le contexte du stage, la problématique ainsi que les objectifs visés.

Le prochain chapitre portera sur le déroulement du stage, la conduite du projet, où je vais faire le tour de mon projet et finalement j'ai éclairé le planning suivi lors de la réalisation du projet et le choix du cycle du développement.

# Chapitre II :

## Le Déroulement Du Stage

*Ce chapitre présente les conditions de déroulement du stage ainsi que la démarche adoptée pour le mener à bien, ensuite il donne une vue générale qui inclut la conduite du projet et le cycle de développement.*

## 1.1. Déroulement du stage :

Mon stage chez SQLI s'est déroulé en trois grandes phases, une première phase consacrée aux formations et à l'apprentissage des bonnes pratiques adoptées chez SQLI. Une seconde, dédiée au développement de l'application dans le cadre de projet de fin d'études.

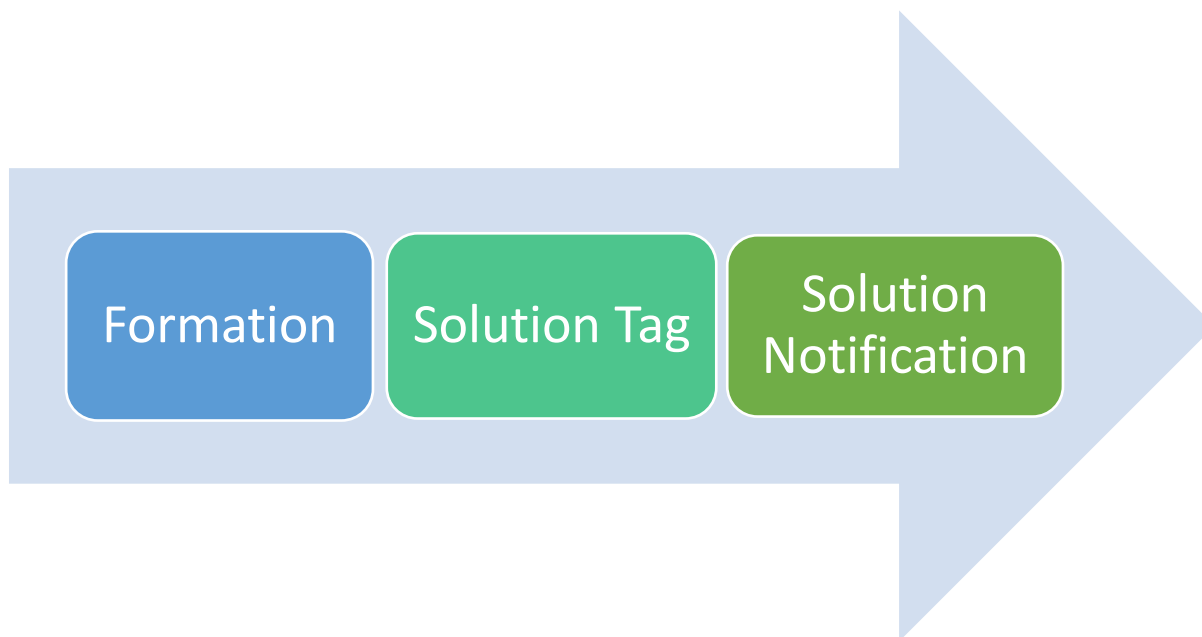


Figure 5 : conduite du stage

Dans le premier mois et demi du stage, j'ai eu une série de formations en matière de développement des applications web avec HTML, CSS, JS, Angular6, ASP MVC5, WebSocket – SignalR et Microsoft Dynamics CRM.

Les écoles d'ingénieurs au Maroc n'assurent pas une formation sur les CRM, Pour cela, il nous fallait dans un premier temps d'apprendre l'architecture générale CRM et ces composants.

- **Dynamics CRM :**

L'objectif de Microsoft Dynamics 365 est d'unifier les données qui résident dans divers systèmes pour donner aux commerciaux une vision plus complète des clients et un point d'entrée plus pénétrant pour interagir avec eux. Il s'agit pour cela de combiner les données et d'éviter d'avoir à les saisir à nouveau. Ce nouveau service, dont la sortie est prévue pour le quatrième trimestre 2016, vise également les silos de données, en permettant aux commerciaux d'entrer les informations une seule fois et de les partager entre plusieurs systèmes au lieu de répéter l'opération.

Microsoft Dynamics CRM est un progiciel de gestion de la relation client développé par Microsoft. Le produit se concentre principalement sur les secteurs Ventes, Marketing et Service, mais Microsoft a commercialisé Dynamics CRM en tant que plate-forme et encourage les partenaires à utiliser son Framework propriétaire (.Net) pour le personnaliser. Il fait partie de la famille d'application Microsoft Dynamics.

- **WebSocket :**

WebSocket est un standard du Web désignant un protocole réseau<sup>1</sup> de la couche application et une interface de programmation du World Wide Web visant à créer des canaux de communication full-duplex par-dessus une connexion TCP pour les navigateurs web. Le protocole a été normalisé par l'IETF dans la RFC 6455<sup>2</sup> en 2011 et l'interface de programmation<sup>3</sup> par le W3C.

- **SignalR :**

SignalR peut être utilisé pour ajouter n'importe quelle sorte de fonctionnalité Web « en temps réel » à votre application ASP.NET. Alors que le chat est souvent utilisé comme un exemple, vous pouvez faire beaucoup plus. Chaque fois qu'un utilisateur actualise une page Web pour voir de nouvelles données, ou que la page implémente un long sondage pour récupérer de nouvelles données, c'est un candidat à l'utilisation de SignalR. Les exemples incluent les tableaux de bord et les applications de surveillance, les applications collaboratives (telles que l'édition simultanée de documents), les mises à jour des progrès de l'emploi et les formulaires en temps réel.

Aussi il permet également de nouveaux types d'applications Web qui nécessitent des mises à jour à haute fréquence à partir du serveur, par exemple, les jeux en temps réel.

D'autre part, il fournit une API simple pour créer des appels de procédure à distance serveur à client (RPC) qui appellent les fonctions JavaScript dans les navigateurs clients (et d'autres plates-formes client) à partir du code .NET côté serveur. SignalR comprend également l'API pour la gestion des connexions (par exemple, les événements de connexion et de déconnexion) et le regroupement des connexions.

- **PowerApps :**

Power Apps est une suite d'applications, de services, de connecteurs et une plateforme de données qui fournissent un environnement de développement applicatif rapide dans le but de concevoir des applications personnalisées et adaptées à vos besoins



métier. Grâce à Power Apps, vous permet de créer rapidement des applis métier personnalisées qui se connectent à vos données métier stockées *soit* sur la plateforme de données sous-jacente (Common Data Service), *soit* dans diverses sources de données locales et en ligne (SharePoint, Excel, Office 365, Dynamics 365, SQL Server, etc.).

Les applications créées à l'aide de Power Apps fournissent une logique métier et des capacités de flux de travail enrichies pour transformer vos processus métier manuels en processus numériques et automatisés. De plus, les applications conçues à l'aide de Power Apps ont une conception réactive et elles peuvent fonctionner de manière transparente dans un navigateur ou sur des appareils mobiles (téléphone ou tablette). Power Apps « démocratise » l'expérience de création d'applications métier personnalisées en permettant aux utilisateurs de créer des applications métier personnalisées enrichies de fonctionnalités sans écrire de code.

Power Apps fournit également une plate-forme extensible qui permet aux développeurs professionnels d'interagir par programme avec les données et les métadonnées, d'appliquer une logique métier, de créer des connecteurs personnalisés et de s'intégrer aux données externes.

- **Asp.net MVC :**

Le Framework ASP.NET MVC nous a été un moyen puissant pour créer le site Web dynamique qui nous a permis une séparation claire des préoccupations et nous a offert un contrôle total du balisage pour un développement agréable et agile. ASP.NET MVC inclut de nombreuses fonctionnalités qui permettent un développement rapide et convivial pour la création d'une application sophistiquée utilisant les dernières normes web.

## 1.2. La conduite du projet :

Dans un premier temps, nous allons parler du cycle de développement que nous avons adopté au cours de notre projet, pour passer ensuite à la planification du projet.

### 1.2.1. La méthodologie choisie

L'approche Agile propose de réduire considérablement l'effet tunnel des approches traditionnelles de type cycle en V ou cycle en cascade en donnant davantage de visibilité, en impliquant le client du début à la fin du projet et en adoptant un processus itératif et incrémental. Elle considère que le besoin ne peut être figé et propose au contraire de s'adapter aux changements de ce dernier. Mais pas sans un minimum de règles.

La méthode SCRUM implique que le projet progresse à travers la mise en place de séries de « sprints ». Et ceux à travers le découpage des tâches en un ensemble des sous tâches pour les traiter sous forme de sprints.

### • Pourquoi la méthode de SCRUM ?

Dans le cadre du projet plusieurs facteurs ont eu une influence pour décider de travailler en mode Agile avec des itérations courtes :

- ✓ La nécessité d'avoir une visibilité permanente sur l'avancement des travaux de conception et de développement.
- ✓ Réaliser des tests en temps réel et éviter l'effet tunnel
- ✓ Identifier et adresser les difficultés dès le démarrage du projet
- ✓ Adapter certaines exigences en cours de projet

Ainsi, Nous avons choisi la méthode SCRUM puisqu'elle accomplit ses besoins.

### • Les acteurs de SCRUM du projet :

Il existe trois rôles principaux à « pourvoir » : le Product Owner, le SCRUM Master, et le membre de l'équipe.

**Le Product Owner** : Il représente les entreprises, les clients ou les utilisateurs, et guide l'équipe vers la construction du bon produit ; il porte la vision du produit à réaliser. Il est responsable de maximiser la valeur du produit et du travail de l'équipe de développement. Mr. BOUFADDI Mahmoud représente le product owner du projet.

**Le Scrum Master** : Il est responsable de la compréhension, de l'adhérence et de la mise en œuvre de la méthode. C'est le leader ; il est au service de l'équipe ; il assiste chaque rôle de l'équipe dans son activité et promet le changement des interactions entre les rôles dans le but de maximiser la valeur du produit de l'équipe. Son autorité s'exerce sur le processus de développement (définition de la durée des itérations, des modalités de tenues et de l'ordre du jour des réunions Scrum, etc.) ; mais il ne dispose d'aucune autorité sur les membres de l'équipe.

**Membre de l'équipe** : dans la méthode SCRUM, l'équipe est responsable de la réalisation opérationnelle des travaux. L'équipe est d'ailleurs composée de 2 personnes

Multitâches. C'est toute l'équipe qui est responsable du résultat final de chaque sprint. La manière dont sont exécutées les tâches est très libre mais cette liberté est néanmoins cadrée par l'obligation de répondre aux objectifs du sprint.

- **Application de la méthodologie Scrum :**

Durant le projet auquel nous avons participé, nous avons appliqué la méthode agile Scrum ainsi :

Nous faisons part d'une équipe de développement formée de dix membres en fixant la durée de sprint à trois semaines ou deux semaines. Chaque sprint vise à mettre en œuvre un nombre fixe d'éléments du Product backlog dans ce qu'on appelle un sprint backlog, lors de la réunion de planification du sprint, toute l'équipe participe au découpage et chiffrage des tâches.

Tout au long du sprint, nous faisons une réunion quotidienne en visioconférence appelé mêlée, chaque membre de l'équipe décrit le travail à faire le jour même, les progrès de la veille, et tous les blocs effacés. Lorsque suffisamment d'items du backlog sont réalisés, l'équipe effectue les tests d'intégration et génère la documentation nécessaire pour le déploiement du produit.

A la fin de chaque sprint, nous réunissons en visioconférence pour effectuer la revue de sprint, pour une durée de quatre heures, afin de valider le logiciel qui a été produit. La revue commence par une présentation des différentes tâches réalisées durant le sprint, suivie d'une démonstration du logiciel produit.

À la suite de la revue, nous organisons la rétrospective du sprint, pour une durée de quatre heures, pendant laquelle tous les membres de l'équipe se réunissent en visioconférence pour discuter les points forts du sprint, les points à améliorer ainsi que les actions à mettre en place pendant les prochains sprints. Ces points sont saisis dans le tableau de bord ci-dessous individuellement par chaque membre de l'équipe avant la réunion.

### **1.2.2. Déroulement du projet**

- **L'outil de gestion de projet :**

Dans cette partie, nous allons expliquer l'implémentation de la méthode Scrum sous JIRA tout en démontrant le cycle de développement adopté. Avant d'entamer cela, une brève présentation de l'outil Atlassian Jira s'impose.

Ainsi nous avons utilisée est Team Foundation Server, communément appelé TFS. C'est un utilitaire de gestion de code source intégré à Visual Studio par Microsoft, permet en plus de faire de la collection de données, des rapports, ainsi que faire le suivi d'un projet et s'adapte très bien à la méthode de travail SCRUM.

### ➤ Présentation de Atlassian

Atlassian est un éditeur de logiciel australien fondé en 2002. Il développe des produits orientés vers les gestionnaires de projet (manager) et aux développeurs de logiciels. Actuellement, son produit le plus connu est JIRA.

### ➤ Présentation de Jira

Jira est un système de suivi de bugs, de gestion des incidents et de gestion de projets développé par Atlassian. Il combine des méthodes agiles telles que XP (eXtreme-Programming), Scrum, Lean développement, aidant les équipes à fournir rapidement un niveau supérieur de qualité de code en meilleure harmonie avec les besoins des clients. Parmi ses fonctionnalités les plus importantes nous citons :

- ✓ Une interface très configurable.
- ✓ Un suivi de tout type d'activité (tâches, jalons, livrables, anomalies...).
- ✓ Une liaison du projet à un outil de documentation collaboratif « wiki ».

Le workflow d'une demande est représenté par le graphe dans la figure suivante :

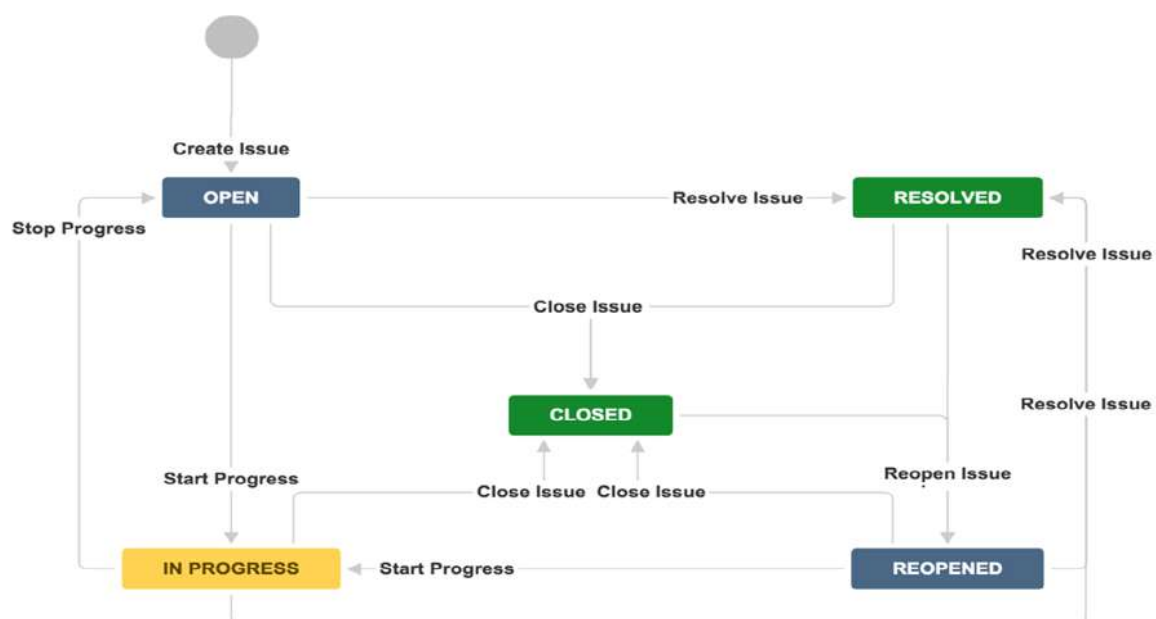


Figure 6 : Graphe de traitement d'une demande client sous Jira.

## ➤ Scrum sous Jira

Via une interface graphique (UX) Jira Software nous retrouvons dans la barre de tâches latérale, nos tableaux de bord, projets et demandes.

Pour démarrer un projet, il suffit de cliquer sur le bouton "créer un projet". Avec, à la clé, un choix à faire : Scrum ou Kanban, en choisissant l'une ou l'autre, l'UX implémentera ensuite dynamiquement les bonnes pratiques et le vocabulaire de chacune.

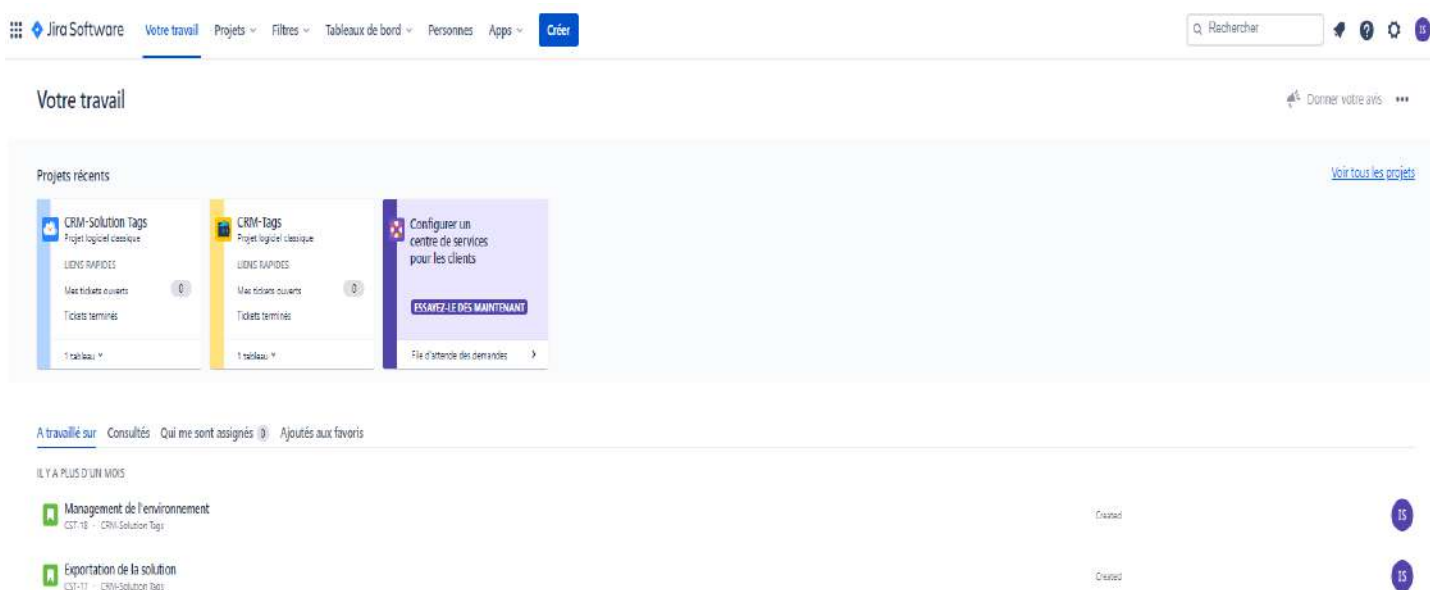


Figure 7 : Interface Jira.

- **La spécification du backlog :**

Au démarrage de projet, l'équipe avec le manager qui représente en même temps notre client ont mis en place le backlog du projet qui est une liste des fonctionnalités attendues du produit final (identification des key-users, création des user stories). Ces fonctionnalités sont classées par priorité et réparties par les sprints de projet. Ce qui permet de définir l'ordre de réalisation.

- **Avant le sprint planning :**

Au début de chaque sprint on se réunit pour discuter les fonctions à produire et les user stories les plus importantes qui doivent obligatoirement être réalisées au cours du sprint, par la suite le chef de projet valide le développement nécessaire à faire, le SCRUM master, essaye à son tour de traduire ces fonctionnalités en des interfaces faciles à utiliser. Au cas où il y a une complexité sur un point mal traité, l'équipe de projet consulte à tout moment le manager (le client) pour que l'interface soit à l'image exacte attendue.

- **Durant le Sprint planning :**

Au début de chaque sprint nous nous réunissons avec la présence du SCRUM master et toute l'équipe de développement pour établir le sprint planning. C'est une réunion où nous estimons, en points de complexité ou d'effort, les tâches à réaliser pendant le sprint.

Ensuite nous parcourons chaque histoire utilisateur (user story : élément du backlog) et nous ajoutons des sous tâches pour chaque histoire, chaque sous tâche correspondant à une tâche technique de cette story.

- **SCRUM daily :**

Pour la mêlée quotidienne nous nous mettons d'accord pour le faire chaque matinée à 9h d'une durée de 5min à 10min. Chacun de nous répond à la première question du SCRUM daily : Qu'as-tu fait hier ? Il s'agit, pour chaque membre de l'équipe, de parler des tâches sur lesquelles il a travaillé. Pour chacune de ces tâches, il va indiquer si la tâche est en cours ou si elle est finie.

Ensuite chaque participant répond à la deuxième question du SCRUM daily : Que prévois-tu de faire jusqu'à aujourd'hui ? Il s'agit de parler des tâches sur lesquelles il prévoit de travailler. Pour chacune de ces tâches, il va indiquer s'il estime que la tâche sera finie ou pas lors de la prochaine mêlée. En dernier lieu chacun de nous répond à la troisième question du SCRUM daily : Qu'est-ce que te gêne pour réaliser ton travail aussi efficacement que possible ? L'objectif de cette question est l'identification des problèmes.

### 1.2.3. Planification Prévisionnelle

La planification est parmi les phases d'avant-projet consistant à prévoir le déroulement du projet en mettant en place un planning prévisionnel décrivant les étapes à suivre à fin d'aboutir au résultat final souhaité.

Pour mettre en œuvre ma planification, je suis basé sur le diagramme de Gantt, couramment utilisé en gestion de projet, est l'un des outils les plus efficaces pour représenter visuellement l'état d'avancement des différentes activités (tâches) qui constituent un projet. La colonne de gauche du diagramme énumère toutes les tâches à effectuer, tandis que la ligne d'en-tête représente les unités de temps les plus adaptées au projet (jours, semaines, mois etc.). Chaque tâche est matérialisée par une barre horizontale, dont la position et la longueur représentent la date de début, la durée et la date de fin.

Ci-dessous la planification par diagramme de Gantt de mon projet :



Figure 8 : Diagramme de Gantt



### 1.3. Choix du cycle du développement du projet : cycle semi-itératif et incrémental

Toutes les méthodes Agiles débutent par des phases séquentielles, courtes mais bien réelles, d'exploration, d'architecture et de planning. Un usage totalement itératif de ces méthodes n'est cependant pas exclu mais ne peut s'appliquer qu'à de très petits projets. Il existe une multitude de Modèles de cycle de vie (en cascade, en V, par incrément, en B, en spirale, etc.). Face à cette multitude de modèles, et avant de pouvoir décider notre cycle de développement à adopter dans le cas de notre projet nous avons choisi de faire la comparaison sur trois cycles de vie à savoir :

- ❖ **Cycle de développement en V :** Ce type de cycle de développement est classique (non adapté aux projets agiles). Il propose des livraisons à longues durées. Il est rejeté pour éviter l'effet tunnel que peut produire une simple mauvaise compréhension d'une exigence : Les anomalies ne sont détectées qu'à une étape tardive.
- ❖ **Cycle de développement itératif :** Ce type de cycle de développement est appliqué dans le cas des projets à grande durée (de délai > 6 mois, et charge > 400 JH). Il propose d'itérer sur tout le processus de la réalisation du projet, commençant par la définition jusqu'à la construction. Il est conseillé de l'adopter dans les projets où les besoins fonctionnels évoluent rapidement et à tout moment, ce qui justifie la nécessité de refaire l'analyse et la conception à chaque itération.
- ❖ **Cycle de développement semi-itératif :** Ce cycle de développement est appliqué dans le cas des projets à moyenne durée. Il propose d'itérer sur la phase de construction seulement, tout en gardant les autres phases en cascade. Il est recommandé dans le cas des projets agiles.

Dans le cas de notre projet, Le cycle semi-itératif paraît le meilleur choix : Nous garderons les premières étapes du cycle de vie du référentiel SKILLS « Projet » en séquentiel, et nous itérons en sprint sur la phase de construction :



Figure 9 : cycle de vie de chaque solution



## Synthèse :

Dans ce chapitre, on a décrit le déroulement de stage en donnant dans un premier lieu un résumé sur les formations offertes et une idée globale sur la méthodologie de conduite du projet à projet à travers une description de l'application de la méthodologie Scrum au sein des équipes et de son implémentation sous l'outil Jira est nécessaire. Enfin, on a présenté le planning et le cycle de développement adopté pour notre projet.

# Chapitre III :

## La spécification des besoins et Conception du système

*Ce chapitre présente L'analyse et la spécification des besoins représentent la première phase du cycle de développement d'un logiciel. Nous commençons en premier lieu par une spécification des besoins auxquels doit répondre l'application, passant ensuite à l'analyse de ces besoins à travers l'introduction des acteurs et les diagrammes de cas d'utilisation relatifs à ces acteurs.*

On va également décrire l'aspect fonctionnel à travers les différents diagrammes UML élaborés dans la phase de conception. Ces diagrammes ont servi à bien structurer le projet et à assurer un bon développement.

## 1. Méthodologie utilisée :

### 1.1. Introduction

La complexité des systèmes informatiques conduit de plus en plus à l'utilisation d'outils standards d'aide à la spécification, à la conception et au développement. Ces outils sont basés, si possible, sur des méthodes et des langages standard. Une méthode d'analyse et de conception a pour objectif permettre de formaliser les étapes préliminaires du développement d'un système afin de rendre ce développement plus fidèle aux besoins du client.

### 1.2. Choix de la méthode

Notre choix a été le langage UML comme langage de modélisation, car la notation UML est la plus appropriée pour des projets orientés objet. Ce choix peut être justifié également par plusieurs raisons :

- ❖ Le processus de développement adopté se base sur les diagrammes UML.
- ❖ La notation UML facilite la compréhension et la communication d'une modélisation objet.
- ❖ UML est aujourd'hui un standard, adopté par les grands constructeurs de logiciels du marché

## 2. Besoins fonctionnels :

### 2.1. Solution Tag

L'administrateur en plus de personnaliser le système de Tag et consulter les entités a aussi le droit d'émettre les actions qui caractérisent l'Utilisateurs CRM, soit la création de l'enregistrement, qui implique si désirer, la sélection d'un tag parmi les tags existants, la consultation de la liste des tags, ou l'ajout à la liste des tags suivie par la création d'un nouveau tag puis la saisie du tag dans les deux cas de figure. L'utilisateur CRM peut aussi visualiser es enregistrements et visualiser les enregistrements créés.

## 2.2. Solution Notification

L'administrateur CRM effectue des développements spécifiques selon les capacités de l'outil (CRM), tout en les maintenant et en assurant un support fonctionnel aux utilisateurs. L'administrateur CRM définit et assume les besoins de développement permettant de faire évoluer les fonctionnalités de l'outil. Il joue un rôle de conseil auprès des utilisateurs. Parmi ses rôles, la gestion des utilisateurs, il veille à l'installation et interconnexions réseau, la gestion de la messagerie CRM, la gestion des diverses administrations opérationnelle et la maintenance. Il est aussi garant du bon fonctionnement technique du CRM et effectue les mises à jour du logiciel.

L'administrateur global est chargé de créer une entité de configuration des relations qui permettra la création d'une relation N-N entre le système de notification et les entités de l'organisation, la suppression de la relation N-N entre système de notification et les entités de l'organisation ou encore la visualisation des relations déjà existantes.

L'administrateur du centre de notification, compte à lui est responsable de la création d'une nouvelle notification, qui peut être une notification simple ou une notification complexe se basant sur les Critère « fetchXML ». L'administrateur du centre de notification a aussi la main pour supprimer une notification, la replanifier ou encore modifier ses paramètres. Toutes ces actions apparaissent dans le système de notification et l'utilisateur CRM sera notifié en se basant sur l'entité de configuration des relations déjà existante.

## 3. Besoins non fonctionnels

Les besoins non fonctionnels sont importants car ils agissent de façon indirecte sur le résultat et sur le rendement de l'utilisateur, ce qui fait qu'ils ne doivent pas être négligés, pour cela il faut répondre aux exigences suivantes :

- **Fiabilité**

Les solutions doivent fonctionner de façon cohérente sans erreurs et doit être satisfaisante aux exigences proposer par l'utilisateur.

- **Les erreurs**

Les ambiguïtés doivent être signalées par des messages d'erreurs bien organisés pour bien guider l'utilisateur et le familiariser avec mes solutions.

- **Ergonomie et bonne Interface**

Les solutions doivent être adaptées à l'utilisateur sans qu'il ne fournisse aucun effort (utilisation claire et facile) de point de vue navigation entre les différentes pages, ainsi ils doivent respecter les normes et le style impose par Microsoft Dynamics.

- **Sécurité**

Nos solutions doivent respecter la confidentialité des données personnelles des clients qui reste l'une des contraintes les plus importantes et restrictive les droits d'utilisation pour chaque rôle d'utilisation.

- **Aptitude à la maintenance et la réutilisation**

Nos solutions doivent être conforme à une architecture standard et claire permettant sa maintenance et sa réutilisation dans d'autres organisations CRM.

## 4.Étude conceptuelle

### 4.1 Solution Tag

- **Diagrammes des cas d'utilisation :**

La figure ci-dessus représente le diagramme de cas d'utilisation de l'administrateur et utilisateur normal de CRM. Nous y retrouvons comme convenu l'acteur principal et ses rôles. Nous remarquons également les relations d'inclusions qui lient les différents cas d'utilisation. Par la suite chacun de ces cas sera détaillé dans un diagramme.

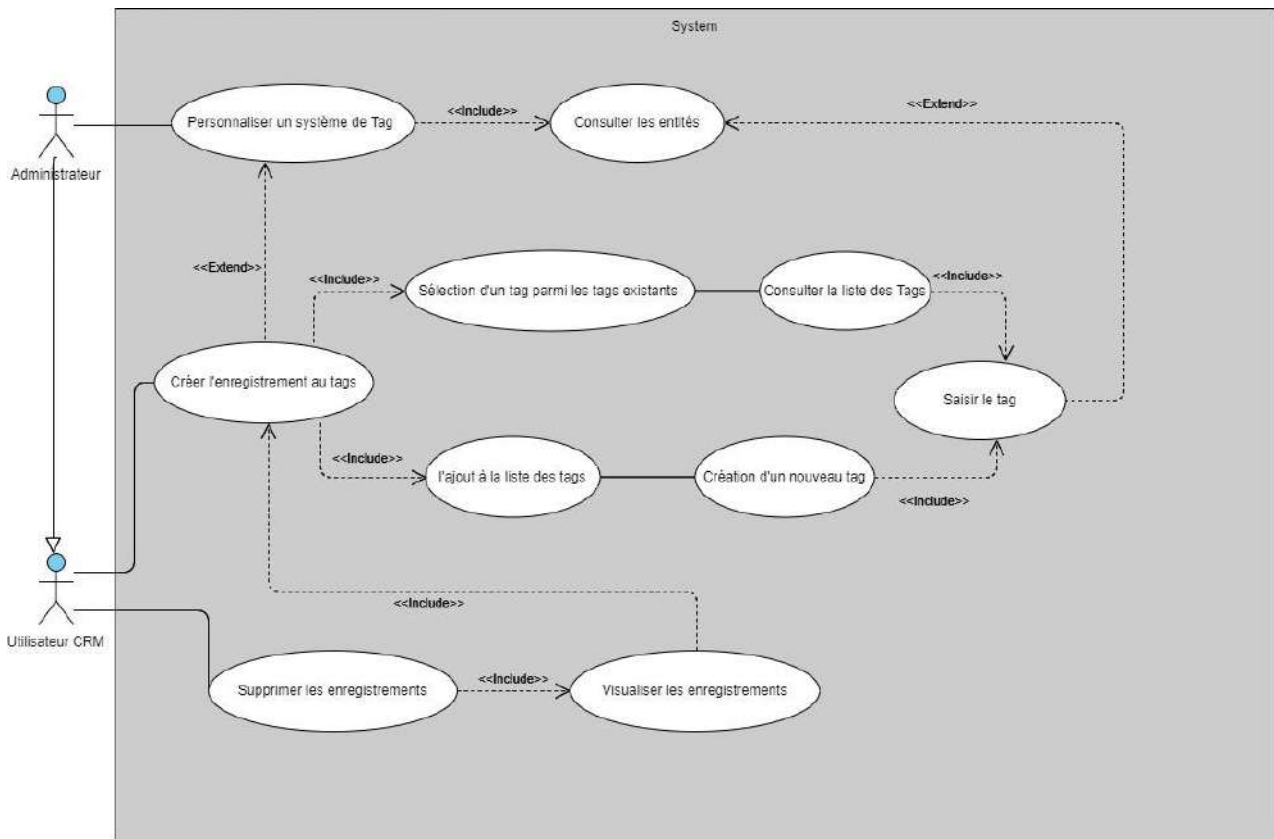


Figure 10 : Diagramme de cas d'utilisation pour la solution de tag

## • Diagrammes de séquence

Dans ce qui suit, on va décrire l'aspect fonctionnel à travers les différents diagrammes de séquence élaborés dans la phase de conception. Ces diagrammes ont servi à bien structurer le projet et à assurer un bon développement.

### Rôle d'administrateur

L'administrateur peut dans ce cas de figure avoir recours à quatre cas d'utilisation, à savoir, installer la solution, créer entité de tag et l'ajout du système de tag dans l'entité spécifique et visualiser les entités qui contient ce système.

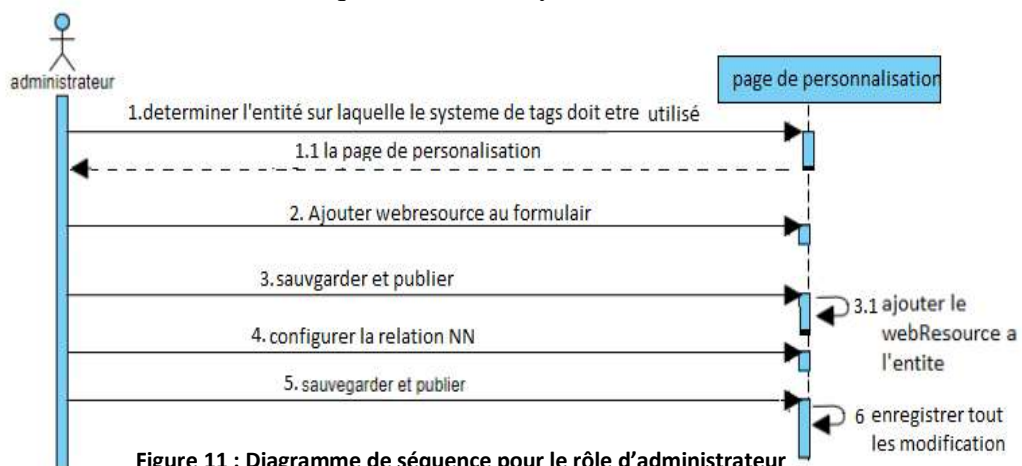


Figure 11 : Diagramme de séquence pour le rôle d'administrateur

### Rôle d'utilisateur

L'utilisateur peut dans ce cas de figure avoir recours à plusieurs cas d'utilisation, à savoir, accéder à l'entité qui contient le système de tag, aussi il peut choisir un tag parmi les tags existants ou créer un nouveau tag ainsi il a la possibilité de voir tous les tags liés à un entité bien définie.

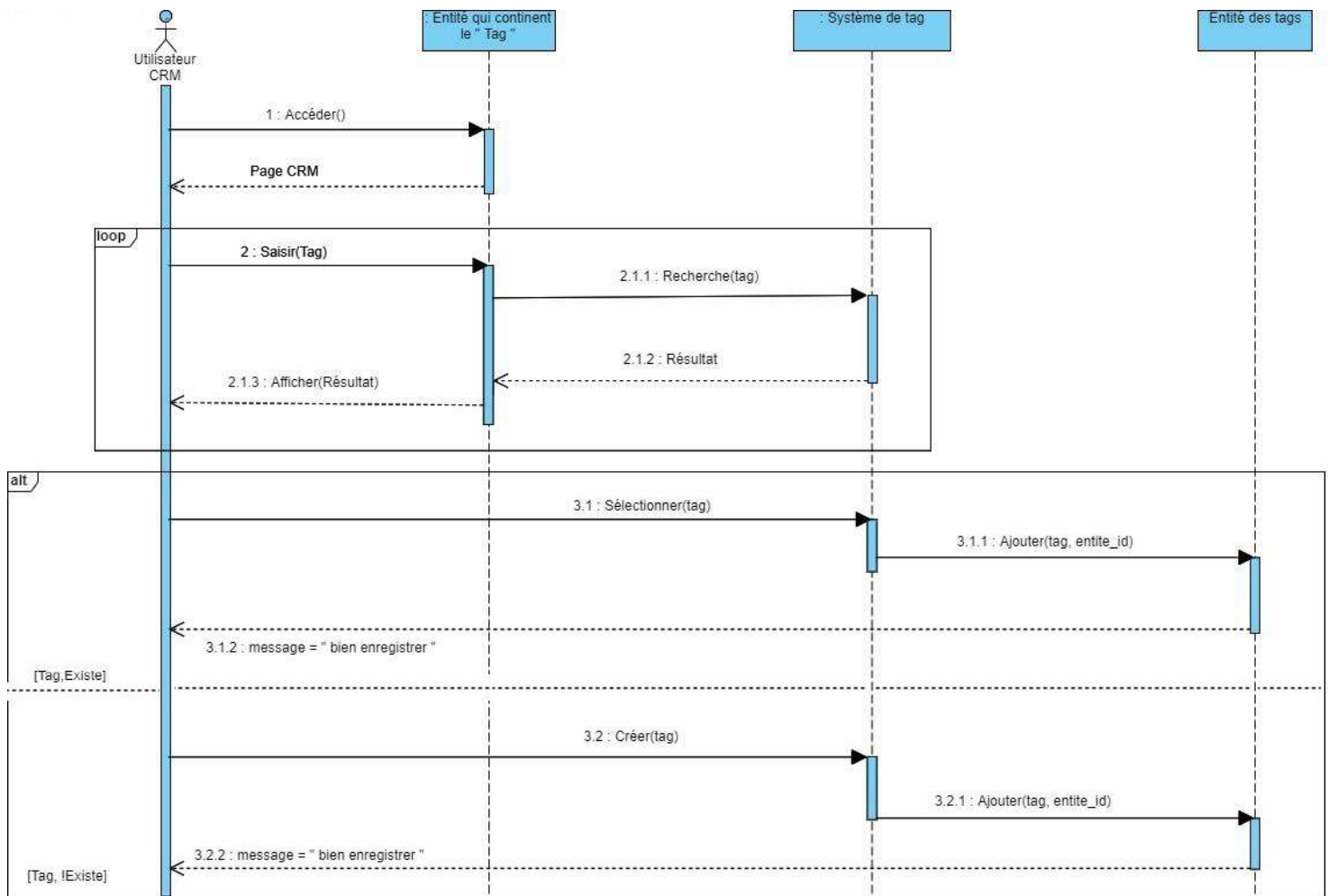


Figure 12 : Diagramme de séquence pour le rôle d'utilisateur

### • Diagrammes d'activité

Un diagramme d'activité est utilisé pour représenter le comportement d'une méthode ou le déroulement d'un cas d'utilisation, dans lequel les nœuds correspondent à l'exécution d'actions ou d'activités, cheminement de flots de contrôle (toutes les instructions, branches, chemins...) et de flots de données (toutes les définitions de variable, les utilisations...).

### ▪ Configuration de la solution

Afin de personnaliser le système CRM, nous choisissons dans un premier lieu l'entité qui est contenue dans le système de tag, on ajoute le web ressource dans un deuxième lieu, puis on spécifie les paramètres d'entrée. Une fois ces étapes achevées il est nécessaire de sauvegarder pour pouvoir oublier par la suite.

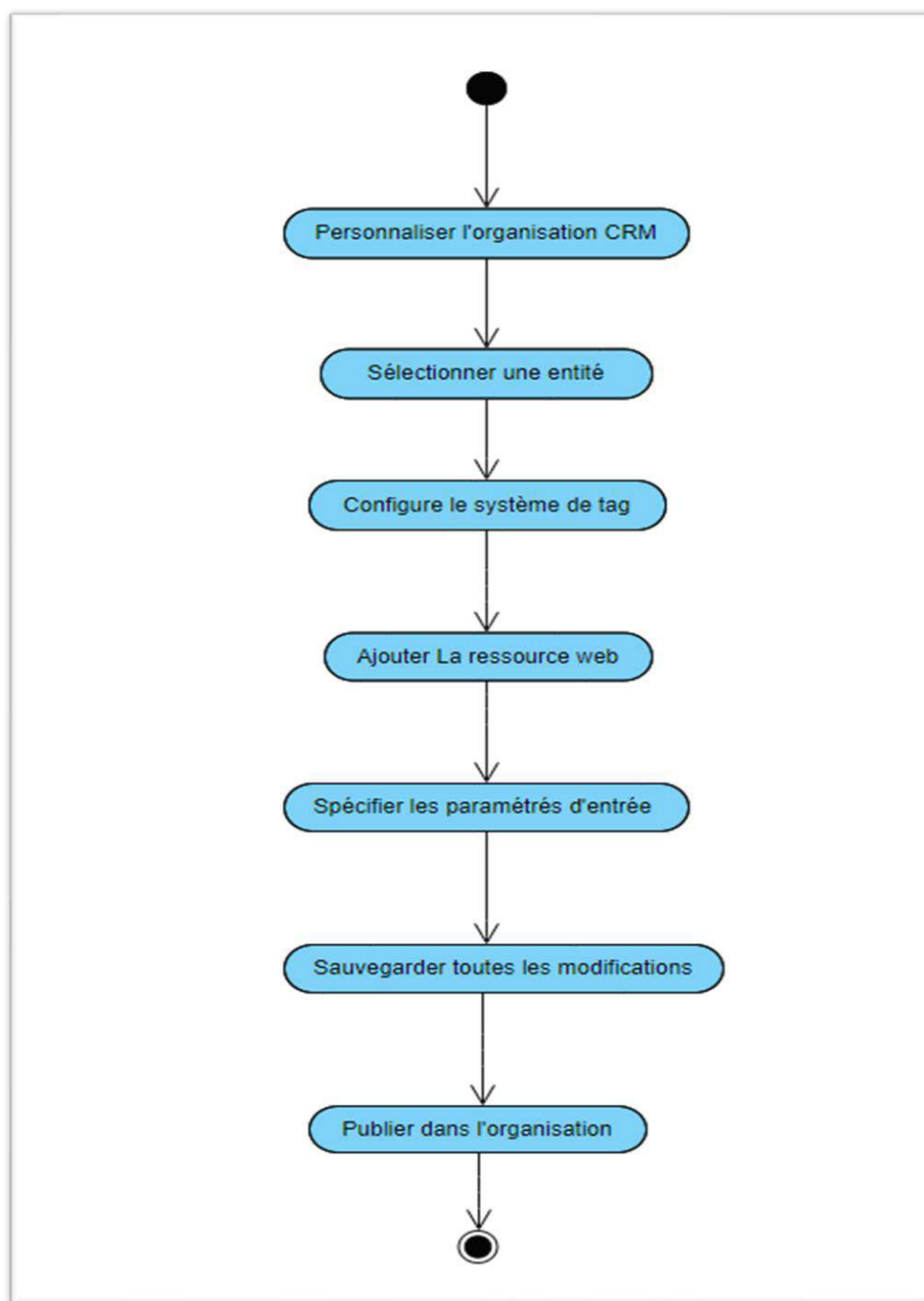


Figure 13 : Diagramme d'activité de la configuration



### ▪ Ajout d'un tag

Une fois qu'on accède à une entité contenue dans le système de tag, on peut saisir un tag qui impliquera le filtrage des tags déjà existant et qui permet de savoir si le tag voulu existe déjà, si le tag n'apparaît pas on peut l'ajouter en le créant ou encore s'il figure parmi les tags déjà existants.

Par la suite, on peut tout simplement le choisir. Une fois que c'est fait, on ajoute le tag sélectionné à l'enregistrement, puis on peut refaire ces étapes en boucles pour ajouter autant de tags que l'on souhaite.

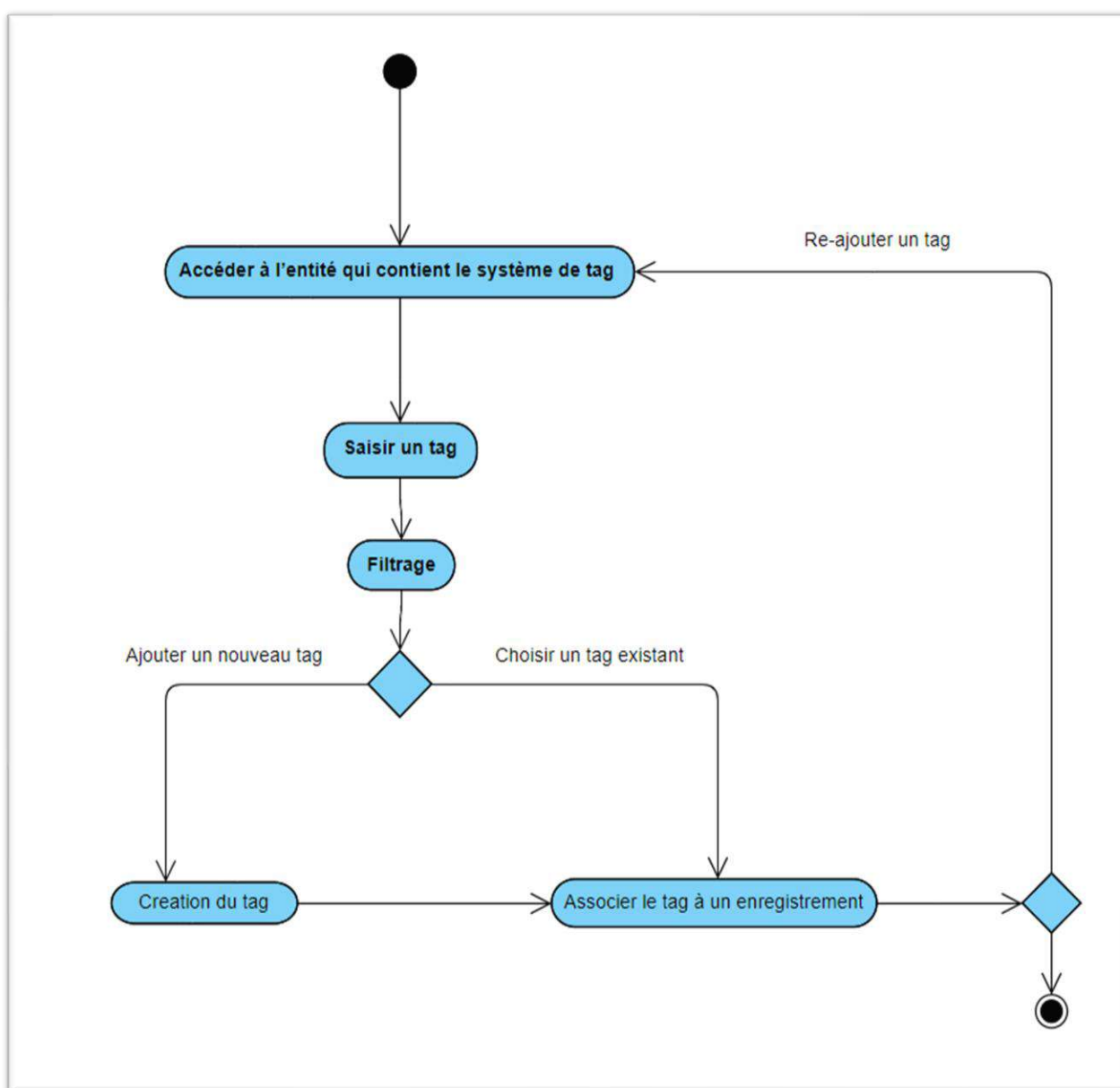


Figure 14 : Diagramme d'activité d'ajout

### ▪ Suppression d'un tag

Une fois qu'on accède à une entité contenue dans le système de tag, on peut voir tous les tags associés à cette entité ainsi on a modélisé le tag pour il soit supprimable.

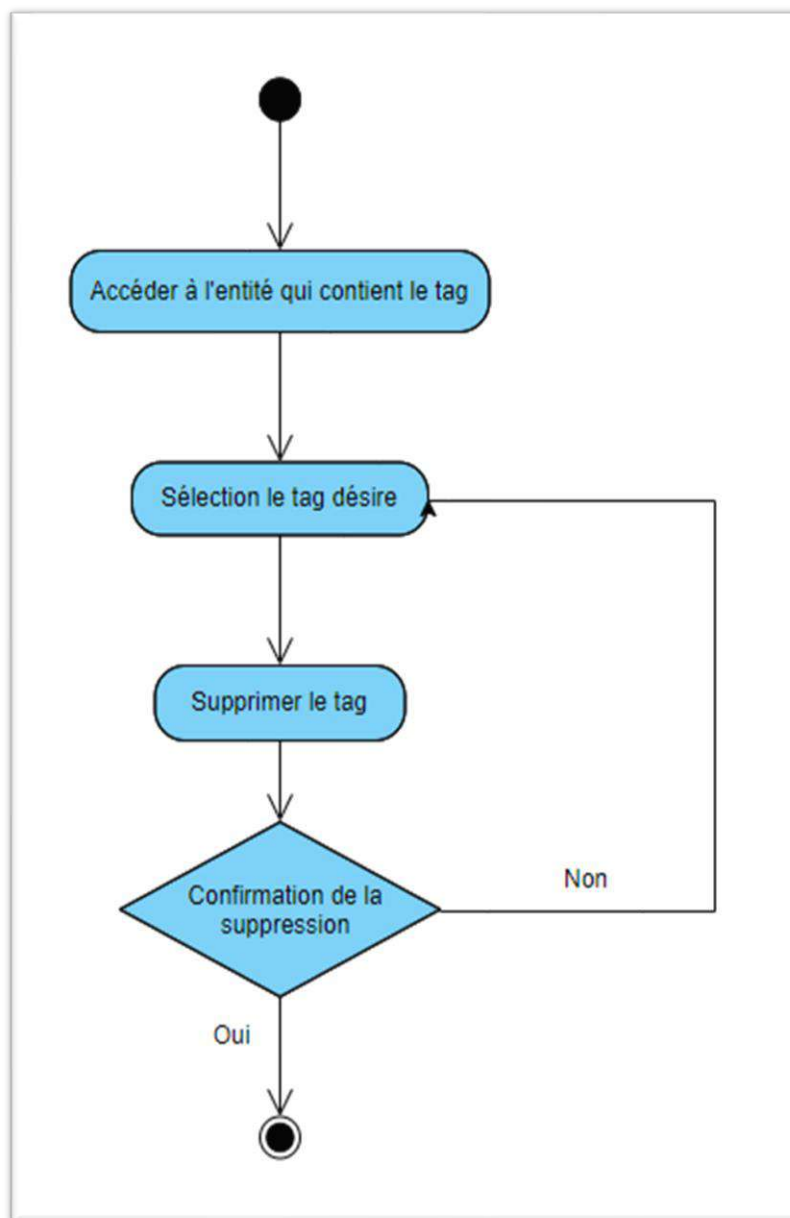


Figure 15 : Diagramme d'activité de la suppression

## 4.2 Solution Notification

### • Diagrammes des cas d'utilisation :

J'ai tout d'abord repéré les différentes actions qu'un des deux administrateurs voudront faire. Cela m'a permis de construire le diagramme de cas d'utilisation suivant :

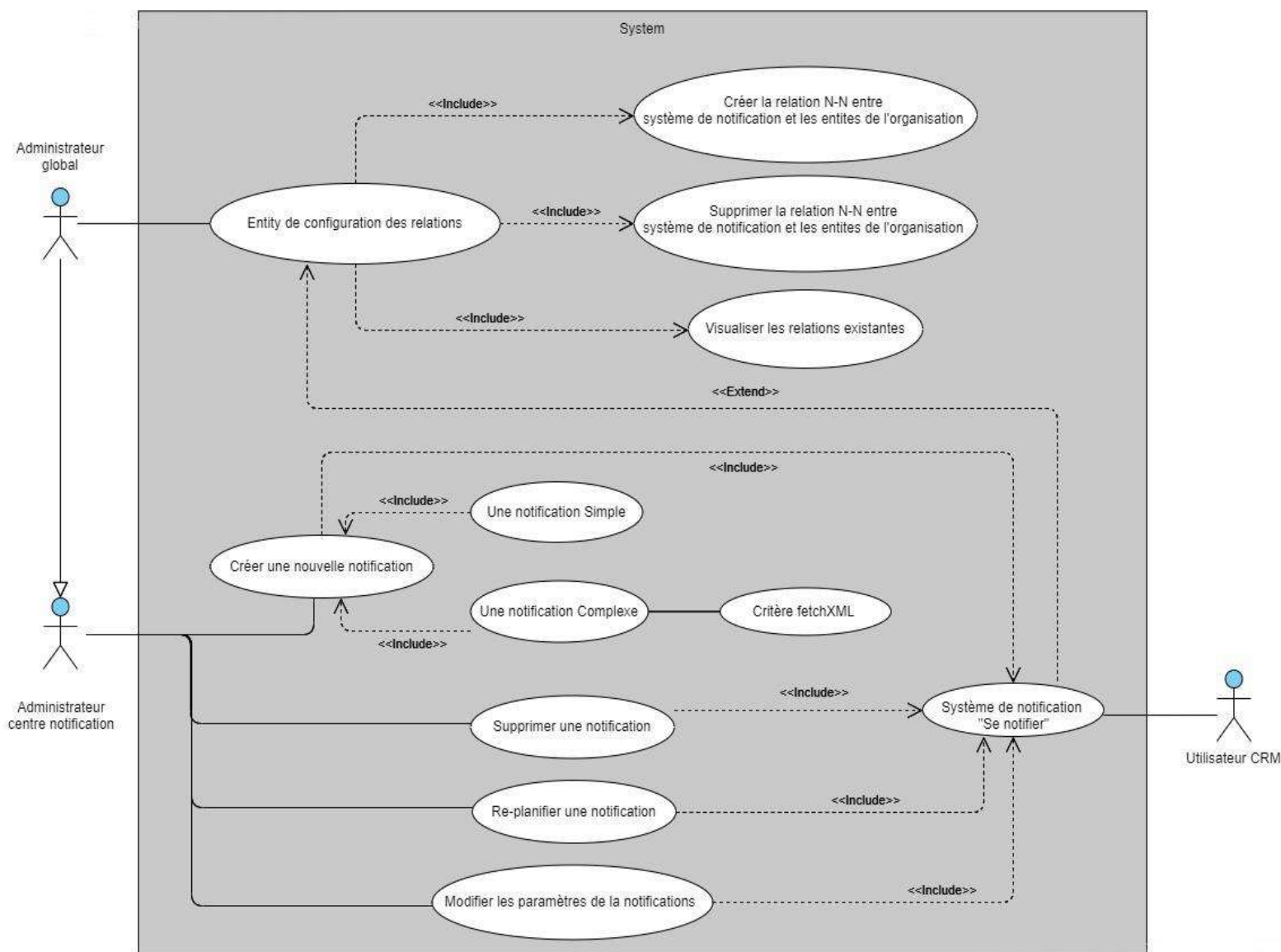


Figure 16 : Diagramme de cas d'utilisation pour la solution de tag

Ce diagramme comporte deux administrateurs, l'un qui peut configurer le système de notification et les différents aspects au tour comme les relations entre les entités. Et l'autre, Il peut ensuite créer, modifier, supprimer et visualiser les notifications programmées, l'utilisateur dans ce cas il est juste notifier.

## • Diagrammes de séquence

### Rôle administrateur global

L'administrateur global peut dans ce cas de figure avoir recours à un cas d'utilisation, primordial à savoir, c'est la configuration de la solution et l'adaptation des paramètres selon les caractéristiques de l'organisation cible.

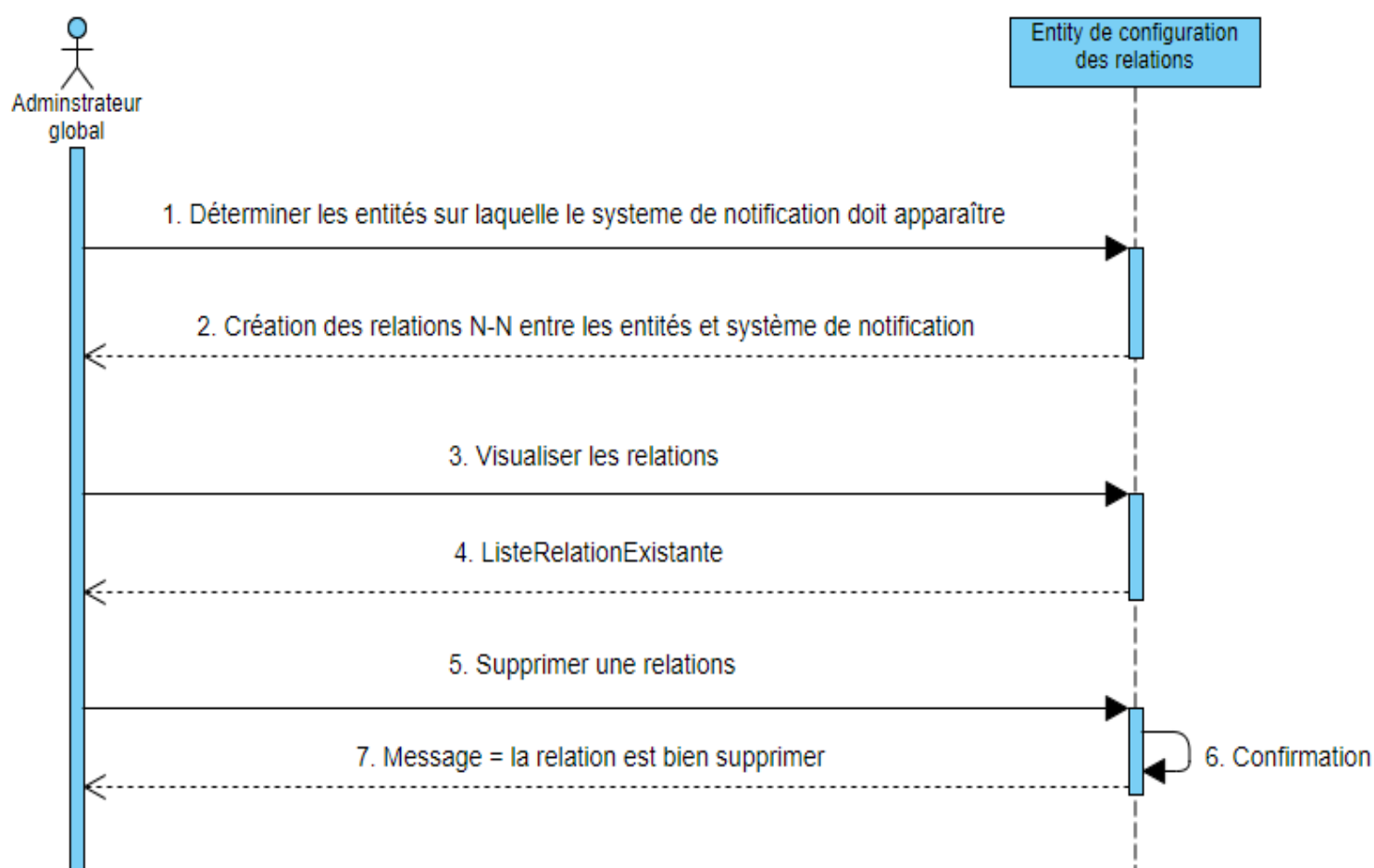


Figure 17 : Diagramme de séquence pour l'administrateur global

### Rôle d'administrateur centre notification

L'administrateur centre notification peut dans ce cas de figure avoir recours à plusieurs cas d'utilisation, à savoir, accéder à l'entité de système de notification ensuite il peut créer, modifier, supprimer et visualiser les notifications programmées,

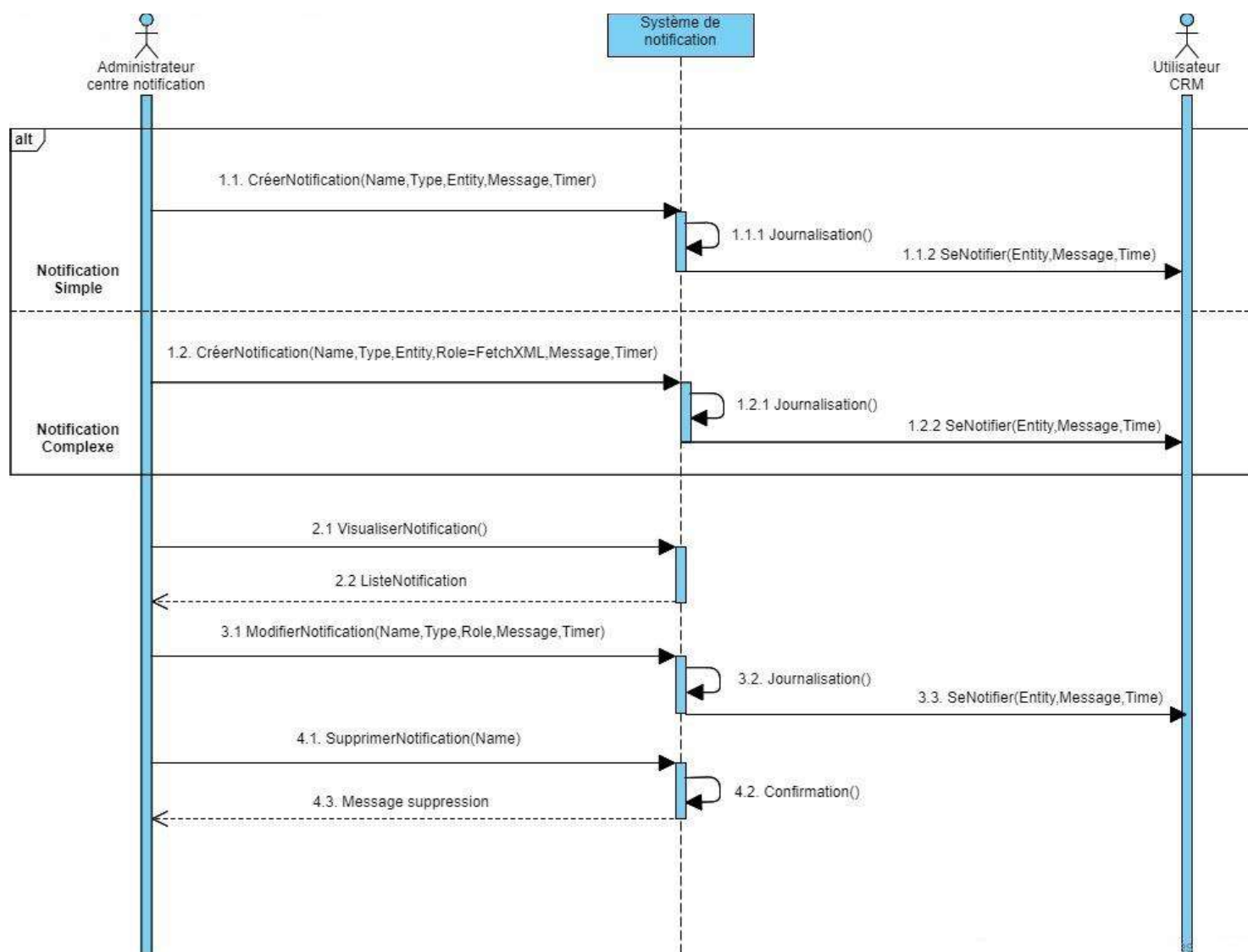


Figure 18 : Diagramme de séquence pour l'administrateur de centre notification

## • Diagrammes d'activité

L'administrateur de centre de notification a le contrôle total pour gérer les notifications, choisir les utilisateurs cible à notifier dans l'organisation CRM, personnaliser les paramètres de la notification et aussi afficher les notifications courantes.

Ci-dessous les différents scénarios possibles faite par l'administrateur de centre notification :

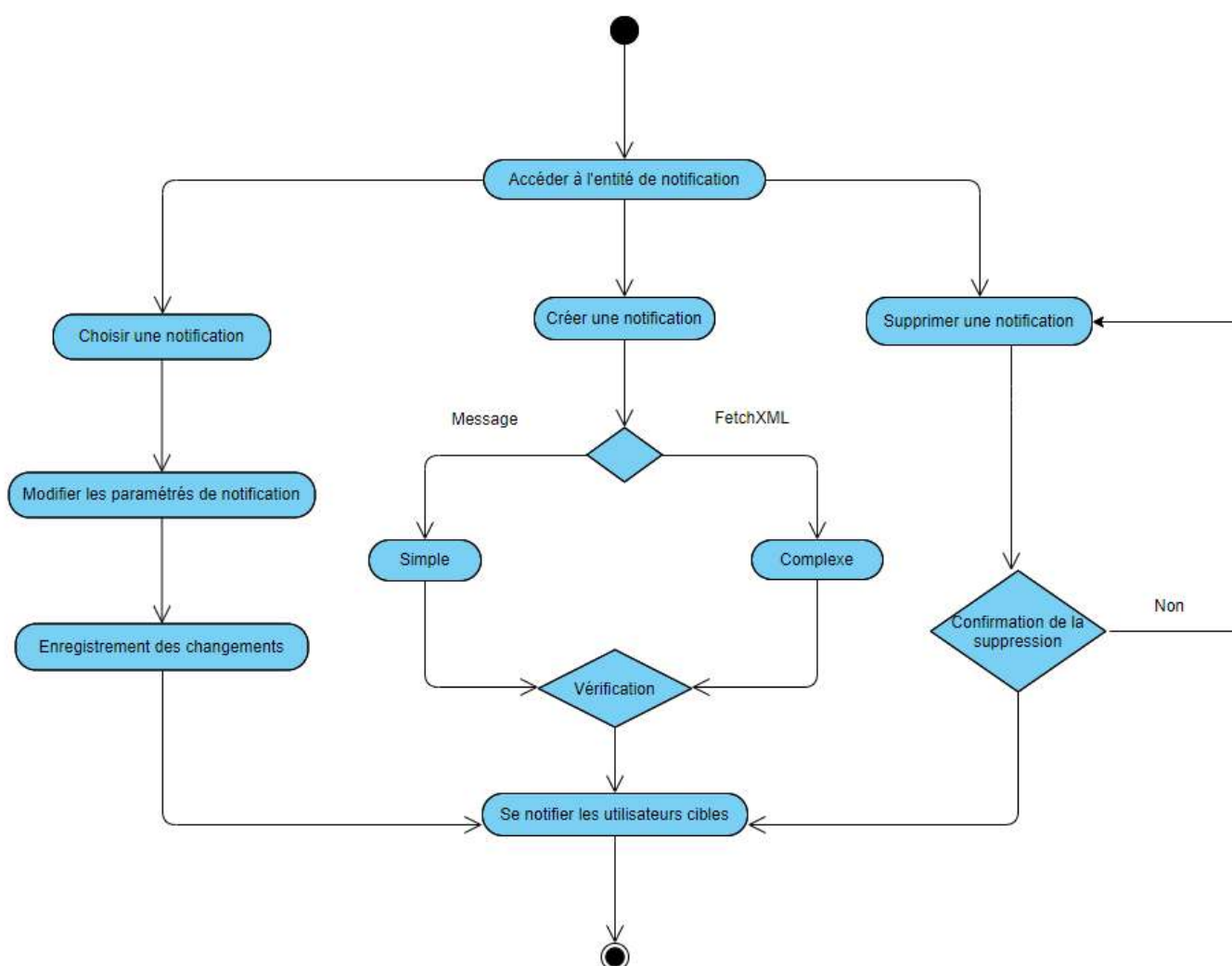


Figure 19 : Diagramme d'activité de la gestion des notifications

## Synthèse :

Ce chapitre a permis de couvrir les différents besoins fonctionnels et non fonctionnels des acteurs du système. En fournissant une analyse détaillée de ces besoins grâce aux diagrammes des cas d'utilisation relatifs à l'acteur unique réagissant avec le système qui est l'utilisateur de l'application CRM. Dans ce chapitre j'ai essayé également de présenter une étude conceptuelle de la partie réalisée du système grâce aux diagrammes de séquences et d'activités.

# Chapitre IV :

## Etude Technique et Réalisation

*Cette partie présente L'analyse Technique et la phase de Réalisation. Nous commençons en premier lieu par une spécification des besoins technique auxquels doit répondre aux exigences de notre projet, passant ensuite à la réalisation du projet à travers l'introduction des photos descriptives.*

## 1. Etude technique

L'une des contraintes majeures que nous nous posons lors de la conception d'une solution CRM est sa maintenabilité. La solution produite doit être facile à maintenir, facile à enrichir et étendre, avec des composants facilement réutilisables. Il est vrai que le coût en temps et effort d'une architecture respectant ces critères paraît élevé. Mais il ne faut pas négliger toute la facilité qu'elle nous procurera en étendant et maintenant notre application... Des avantages qui valent cet investissement.



Figure 20 : Logo de Microsoft Dynamics

Microsoft Dynamics CRM vise à proposer des solutions technologiques permettant de renforcer la communication entre l'entreprise et ses clients afin d'améliorer la relation avec la clientèle en automatisant les différentes composantes de la relation client,

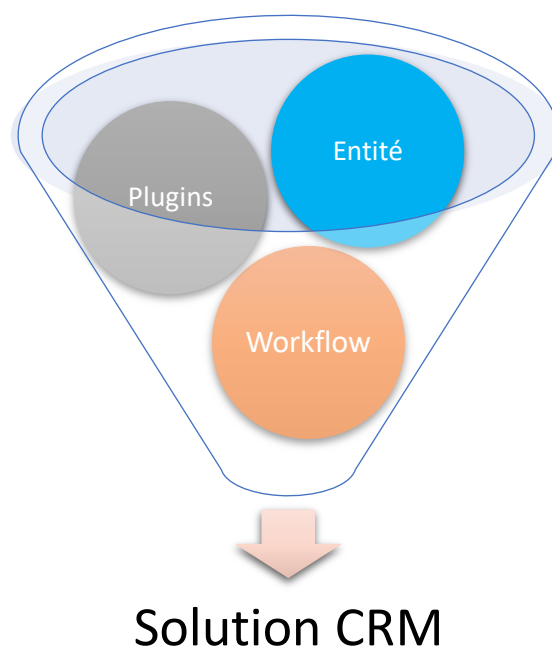


Figure 21 : Architecture de la solution CRM



## Solution Tag

- **Entités :**
  - **Entité de tag :** contient la liste des tags existants et ajouter par l'utilisateur.
  - **Entité de liaison entre le système et l'entité tag :** intermédiaire
- **Plugins :**
  - **Ajout :** sert à prendre les informations entrantes et l'envoie les paramètres sortants vers le workflow.
- **Workflow :**
  - **Création :** d'un enregistrement contient les informations reçoit par le plugin.
- **Web Ressource**
  - **Suppression d'un tag**
  - **Changement de la langue**
  - **Ouvrir le formulaire de tag**
  - **Désigne de la zone de tag**

## Solution notification

- **Entités :**
  - **Entité de notification :** Où on initialise et planifie une notification
  - **Entité de configuration du système de notification :** Gérer le système
- **Plugins :**
  - **Ajout :** sert à prendre les informations entrantes et l'envoie les paramètres sortants vers le workflow.
  - **Modification :** Changer les paramètres de notification
  - **Suppression :** Annuler la planification de notification
- **Workflow :**
  - **Création :** d'un enregistrement contient les informations reçoit par le plugin.
  - **Désactivation :** d'un enregistrement
- **Web Ressource**
  - **Changement de la langue**
  - **Ouvrir le formulaire de notification**
  - **Désigne de la zone d'affichage de la notification**

## 2. Outils et technologies utilisés

### 2.1. Framework .Net

Le Framework .Net est une plate-forme de développement pour créer des applications pour Windows, Windows Phone, Windows Server et Windows Azure. Le .Net Framework fournit un environnement d'exécution managé, un développement et un déploiement simplifiés et l'intégration à une grande variété de langages de programmation, notamment Visual Basic et Visual C#.

### 2.2. CSharp

C# est le langage par excellence de .Net. Ce langage est comme dit précédemment un langage de programmation orienté objet, étant un carrefour entre différents langages comme le Java, le C++, ou encore le Visual Basic, tout en restant un langage à part. C'est un langage à typage fort, c'est à dire que le type des variables est fixé à la compilation et que celle-ci peut vérifier les erreurs de typage (à la différence du langage PHP par exemple). Le C# est très polyvalent, il permet de coder de simples applications consoles jusqu'à de gros programmes avec une multitude de fenêtres, en passant par les jeux.

### 2.3. MS Dynamics CRM web services

Les services Web Microsoft Dynamics CRM permettent aux ressources externes d'interagir avec les données CRM et les métadonnées qui définissent l'instance CRM spécifique à laquelle vous vous connectez. Il existe deux services Web proposés par CRM

✚ **IOrganizationService** : Il s'agit du principal service Web utilisé pour accéder aux clients, aux ventes, au marketing et aux données de service (opérations CRUD) ; En outre, il expose les métadonnées qui définissent la "structure" spécifique de l'organisation CRM à laquelle vous accédez.

✚ **IDiscovery** : Ce service Web permet à un développeur d'examiner les éventuelles organisations multiples qui sont hébergées dans une seule instance de CRM (multitenancy) pour déterminer quelle est l'organisation et les endpoints appropriés à utiliser pour l'accès aux données. Les services Web Microsoft Dynamics CRM sont principalement utilisés dans notre projet car ils représentent les portails (permettant aux utilisateurs non-CRM d'interagir avec les données CRM) ou pour permettre aux applications externes d'interagir avec les données des serveurs.

## 2.4. WCF comme services

Pour offrir la possibilité de réutilisation de nos composants métier (services métier) dans d'autres applications, ainsi d'assurer la communication entre le CRM et l'application web, nous avons choisi de les publier en web services dans la couche Services (voir architecture applicative). A ce niveau, deux choix nous étaient possibles : Web API ou WCF.

WCF (Windows Communication Foundation) est le modèle de programmation unifié de Microsoft permettant de générer des applications orientées service. Il permet aux développeurs de générer des solutions transactionnelles sécurisées et fiables qui s'intègrent à plusieurs plateformes et interagissent avec les investissements existants. Il représente le standard SOAP pour Microsoft (supporte plusieurs protocoles http, TCP, UDP...).




Web API est une infrastructure qui facilite la création de services HTTP qui atteignent une large gamme de clients, y compris les navigateurs et les appareils mobiles. C'est une plate-forme idéale pour générer des applications RestFull sur le .NET Framework. Il représente la procédure REST pour Microsoft (supporte le protocole http seulement).

### Contrainte technique :

Nous utilisons le web service pour communiquer des données sensibles entre l'application web et le CRM, et WCF propose un niveau de sécurité plus haut, donc nous opterons pour WCF comme type de nos services.

## 2.5. Log4Net pour la journalisation

Log4Net est un outil open source de journalisation pour le Framework .NET, simple et efficace, il permet de logger des informations vers diverses cibles pour détecter tout problème fonctionnel. Nous l'avons configuré pour :

-  Gérer les logs de manière dynamique, sans avoir à modifier le code existant.
-  La journalisation sur plusieurs niveaux de filtrage de messages : Information, warning, error, debug, fatal et trace...
-  La journalisation se fait sur un fichier cible selon la date, l'heure, le niveau et la source du problème dans le projet.

## 3. Environnement de travail

### 3.1. Visual Studio Entreprise 2019 comme IDE

Visual Studio est l'IDE officiel de Microsoft. Il est sous plusieurs éditions : Entreprise, Community et Professional. Nous avons choisi de travailler avec l'édition « Entreprise » version 2019. Cette édition supporte un ensemble de plugins orientés production des applications entreprises.

### 3.2. Forge logicielle TFS 2017

Team Foundation Server est un serveur professionnel permettant aux équipes de partager du code, d'assurer le suivi des travaux dans un package unique. Il s'agit d'un ensemble d'outils de collaboration qui s'intègrent parfaitement à Visual Studio, pour un travail efficace sur des projets logiciels de toutes les formes et toutes les tailles. Il offre les fonctionnalités suivantes :

#### a) Gestion du code source

La gestion du code source permet le stockage et la collaboration sur le code depuis n'importe quel emplacement grâce aux projets d'équipe privés assortis d'un contrôle de version.

#### b) Intégration et déploiement continus

La fonctionnalité d'intégration continue jumelée à celle du déploiement continu nous a permis de garder toujours le dernier build de notre application consultable sur le cloud. En effet, notre application passe d'abord les tests d'intégration avec succès, en suite elle est packagée est déployée sur notre serveur cloud Azure. Les rapports d'intégration et du déploiement sont générés sur Visual Studio Online.

### 3.3. Internet Information Services


Internet Information Services (IIS) est un serveur Web des différents systèmes d'exploitation Windows. IIS est un serveur flexible, sécurisé et gérable pour héberger n'importe quoi sur le Web. Du streaming multimédia aux applications Web, l'architecture évolutive et ouverte d'IIS est prête à gérer les tâches les plus exigeantes. Nous avons travaillé avec IIS pour exécuter l'application web en mode développement, ainsi que pour

héberger le projet des services web pour qu'il soit consommable à la fois par l'application web que pour l'organisation CRM.

### 3.4. Dynamics CRM on Prémisse 2019

Dynamics CRM on premise représente la base de données, notre source d'une part et notre destination d'autre part. Dynamics CRM contient trois modules présents de base dans l'application :

 Le module Ventes

 Le module Marketing

 Le module Service

Chaque module gère des entités spécifiques. Une entité représente une table dans une base de données. Elle comporte des champs contenant ses différentes informations. Par exemple, l'entité Contact va contenir le nom et le prénom du contact, son numéro de téléphone, son adresse email, etc.

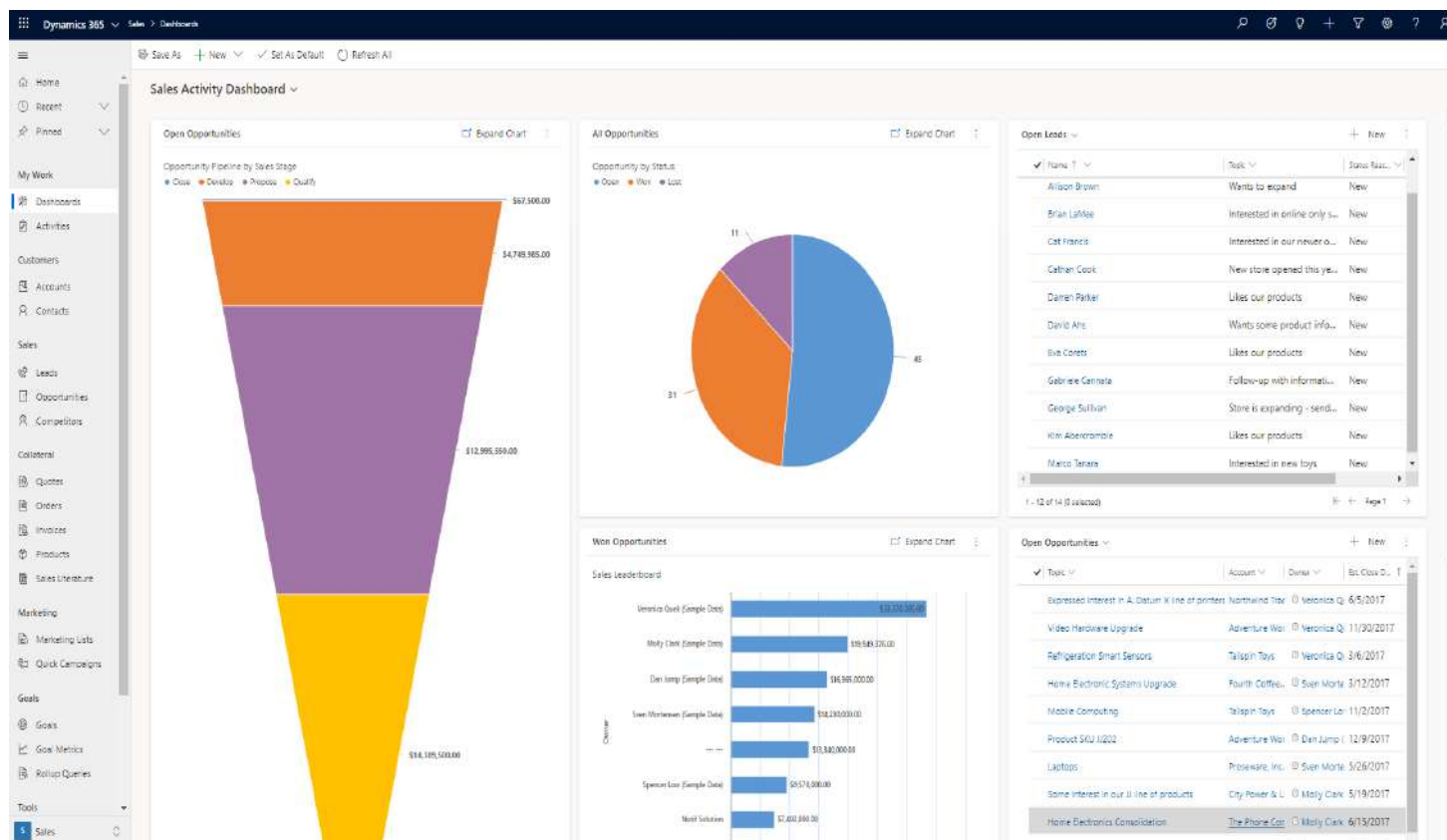


Figure 22 : Microsoft Dynamics CRM Interface.

## 4. Réalisation

On va détailler dans ce chapitre ce qu'on a réalisé de notre backlog, ainsi que quelques captures de deux solutions.

- **Solution tag**

Tout commence par la création d'un nouveau tag

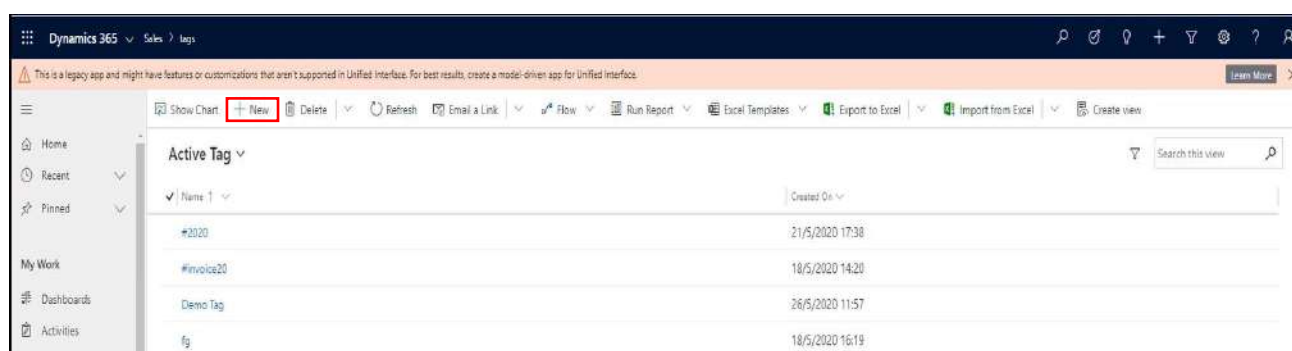


Figure 23 : Entité de tag

On saisie le nom du tag désiré puis on l'enregistre par la suite il apparaît parmi mes tag déjà existants.



Figure 24 : Création d'un nouveau tag

Maintenant on doit se diriger vers l'entité où se trouve notre web ressource. Pour notre cas présent, notre entité est le devis. Lors de la création d'une nouvelle entité, le système de tag n'apparaît pas, ce dernier n'apparaît qu'un fois notre nouvelle entité est bien créée et enregistrée.

**Avant l'enregistrement**

Figure 25 : Entité de devis contient le web ressource avant l'enregistrement

**Après l'enregistrement**

Figure 26 : Entité de devis contient le web ressource après l'enregistrement

Place à l'étape de l'ajout d'un tag à l'enregistrement. On s'aperçoit que dans la liste déroulante figurent deux types de tags, en haut ceux créer par moi-même, et puis en bas les tags créés par d'autres personnes.



## SALES INFORMATION

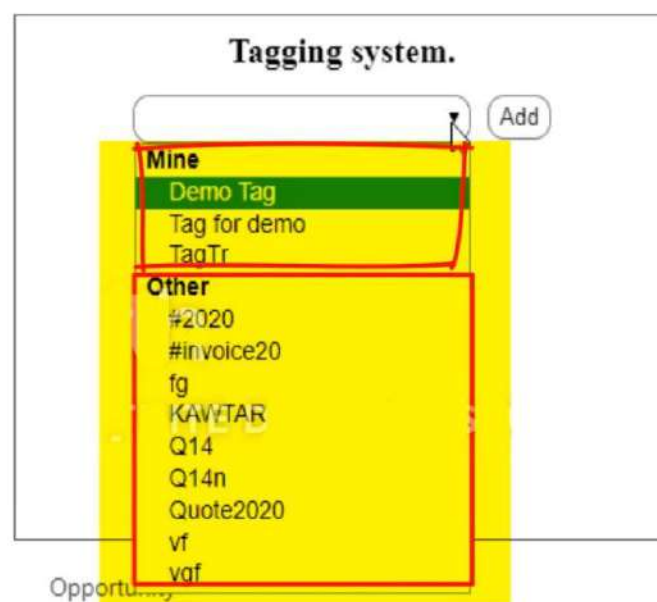


Figure 27 : Système de Tag

Lors de l'ajout d'un tag à l'aide au bouton add le tag apparaît comme suit :

## SALES INFORMATION

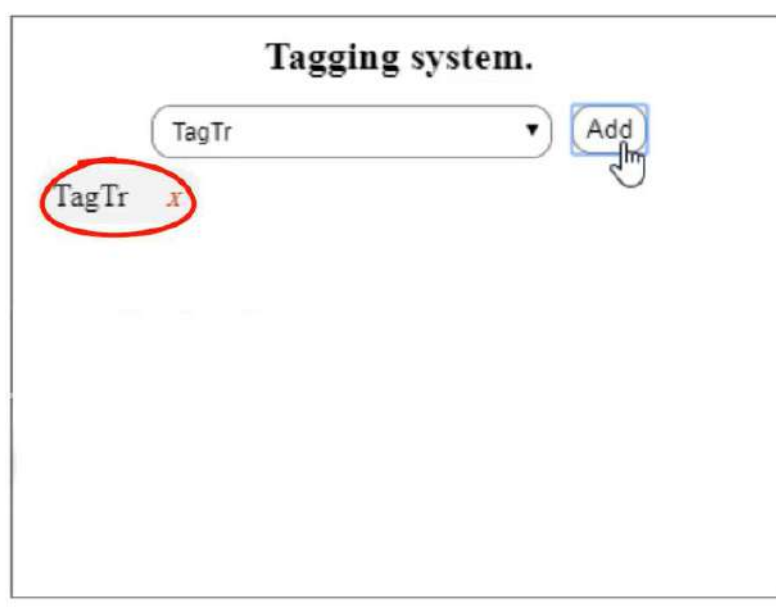


Figure 28 : Associe d'un tag a un enregistrement

On sélectionne le TagTr, on a donc accès à l'information du tag, on a aussi le possible de rajouter plusieurs tags.



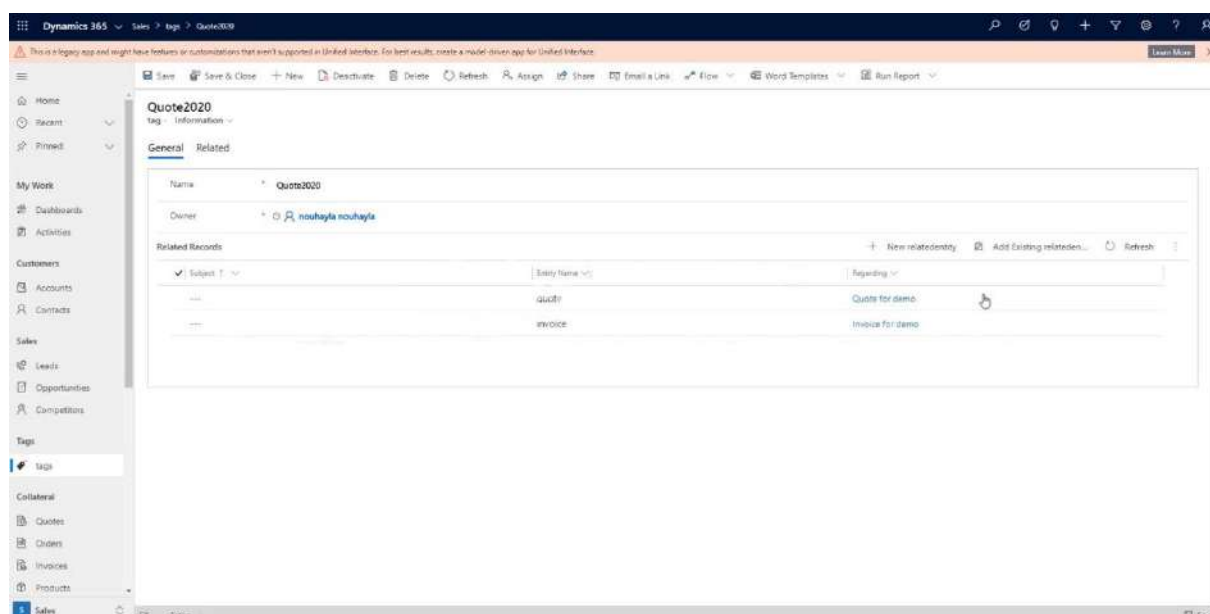


Figure 29 : Information sur le tag

On sort de l'enregistrement puis on y revient et on se rend compte que tous les sont intacts. On passe à présent à une nouvelle entité et à un nouvel enregistrement. On peut donc voir qu'on a la possibilité d'utiliser n'importe quel tag dans n'importe quel enregistrement. On peut voir ici les emplacements où ce tag a été appelé et le type de chacun aussi.

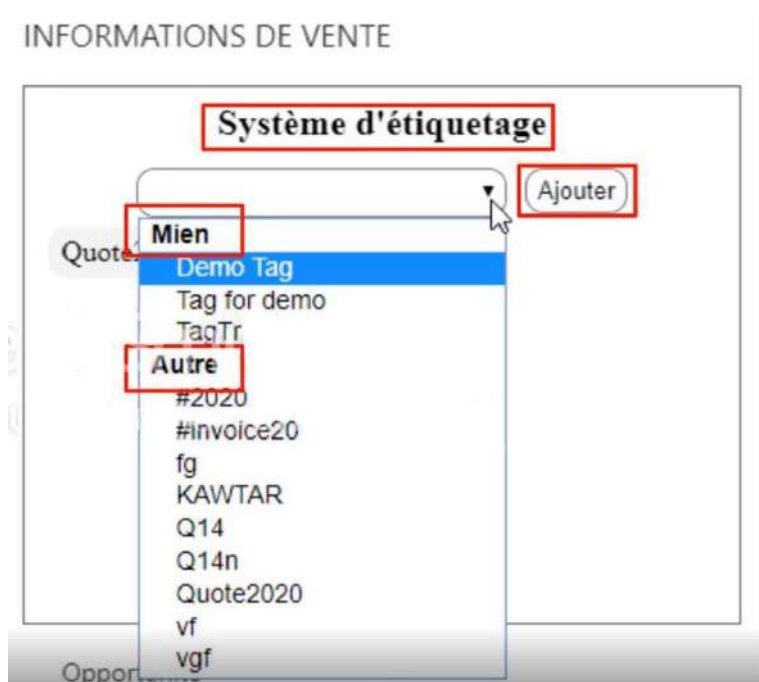


Figure 30 : Système de traduction

Le changement du langage de l'organisation implique comme on peut le voir un changement aussi au niveau du système de tag.

## • Solution notification

L'interface de notification, Il montrera toutes les notifications ainsi il peut ajouter une notification.

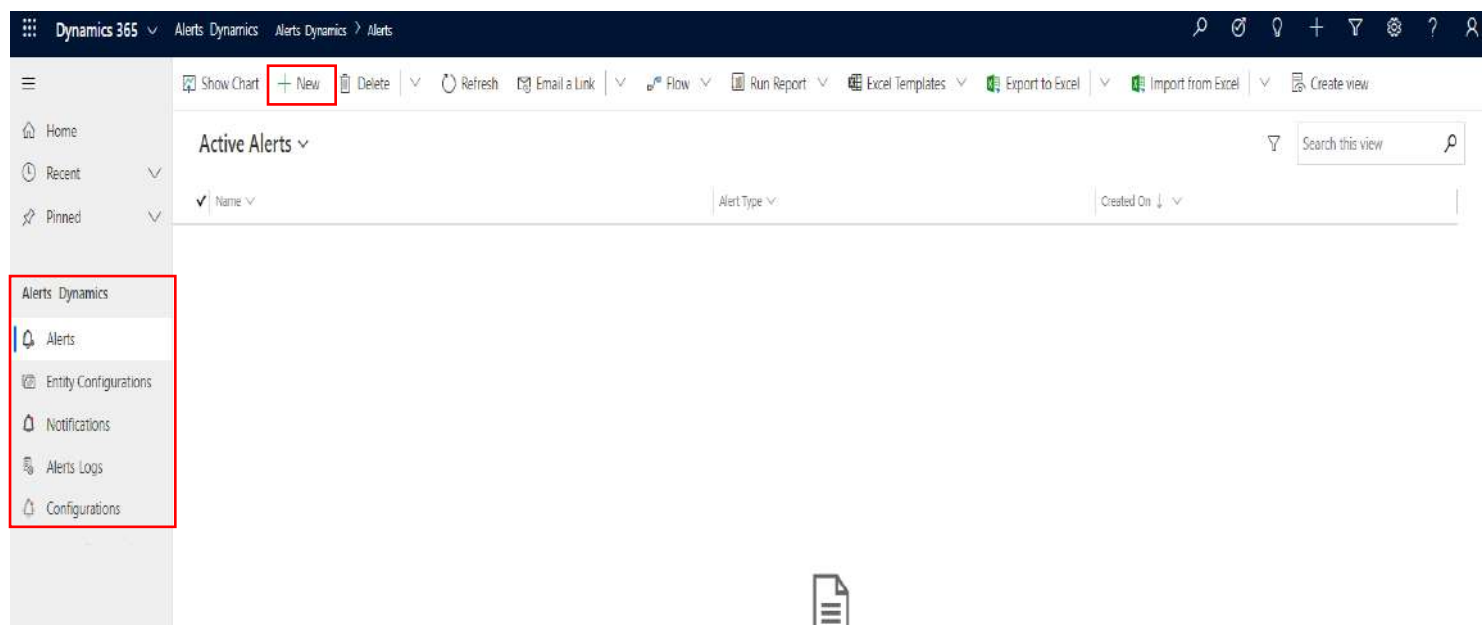


Figure 31 : Interface des notifications

On crée une nouvelle notification après on saisit les paramètres puis on l'enregistre par la suite il apparaît parmi les notifications déjà existantes.

On a deux types d'alerte :

- **Annonce Simple**

### New Alert

Notif Solution  
Owner

#### General

Name	* Notification CRM
Alert Type	* Announcement

Figure 32 : Alerte de type annonce simple

- **Annonce basée sur des règles**

## New Alert

Notif Solution  
Owner

## General

Name	*	Notification CRM
Alert Type	*	Rule Based
Entity Configuration	*	---
Rule Mode	*	---

Interval	
Poll Interval	*
Interval	---

Figure 33 : Alerte de type annonce basée sur des règles

On a aussi de type de condition pour planifier une nouvelle notification.

- **Simple**

## New Alert

Notif Solution  
Owner

## General

Name	*	Notification CRM
Alert Type	*	Rule Based
Entity Configuration	*	---
Rule Mode	*	Simple
View	*	<input type="text"/>

Interval	
Poll Interval	*
Interval	---

Figure 34 : Alerte de type annonce basée sur des règles simple

- **FetchXML**

## New Alert

Notif Solution  
Owner

## General

Name	*	Notification CRM
Alert Type	*	Rule Based
Entity Configuration	*	---
Rule Mode	*	Advanced
Fetch XML	*	---

Interval	
Poll Interval	*
Interval	---

Figure 35 : : Alerte de type annonce basée sur des règles complexes

## Synthèse :

Nous avons vu les différents outils et technologies utilisés pour mettre en œuvre ce projet. Par la suite, nous avons décrit dans ce dernier chapitre le fonctionnement des tâches réalisées durant la période de stage en donnant un aperçu sur les fonctionnalités de base de deux solutions à travers un ensemble de captures d'écran.

## **Conclusion générale**

Dans mon projet de fin d'études, le travail a été réalisé au profit de l'agence SQLI Oujda. Dans le cadre de ce projet qui consiste à développer deux solutions installables pour Dynamics CRM, l'un va permettre aux utilisateurs de procéder par utilisation de tags dans une organisation CRM et l'autre va le notifier se basant sur son statut d'Agent Customer Service ainsi qu'un ensemble d'autre critères.

Ainsi mes objectifs ont été atteints, à savoir faciliter la vie aux utilisateurs de l'organisations CRM, en rendant cette expérience plus facile et moins complexe ainsi qu'en permettant un gain de temps énorme grâce à un triage de données plus efficaces et efficient. J'ai aussi réussi à contribuer en l'amélioration de la communication entre collaborateurs et permettre une diffusion plus fluide de l'information.

Pour ce faire, nous avons adopté une méthode agile à savoir SCRUM, qui est une solution efficace pour modifier les pratiques de travail et apporter de la souplesse et de la réactivité dans notre système. Dans chaque sprint, on sélectionne les éléments prioritaires de la liste des exigences fonctionnelles pour les réaliser au cours des deux semaines passant par la conception afin d'aborder le processus de développement pour la mise en œuvre de notre système.

En termes de perspectives, le projet dans sa version actuelle n'est pas encore achevé, le rapport ne couvre que les itérations 0, 1, 2, 3, 4 et 5, on est actuellement dans l'itération 6 qui a pour objectif d'intégrer des nouvelles fonctionnalités de suivi des utilisateurs, et aussi intégrer un module de notifications plus puissant pour augmenter la performance du canal de transmission des notifications.

Dans une vision professionnelle et pour conclure, ce projet fut une première grande expérience dans ma carrière. Il m'a permis de mieux appréhender l'activité globale de l'organisation qui nous a accueilli, et il m'a également offert l'occasion de vivre sur le terrain avec des opérationnels afin de mieux comprendre leur vie dans l'organisme, leur méthodologie de travail, les problèmes qu'ils rencontrent et comment ils les résolvent. Tout ça m'a offert une bonne préparation à mon insertion professionnelle.



# Bibliographie

- [1] "Importance of Software Documentation" by Noela Jemutai Kipyegen and William P. K. Korir, Department of Computer Science, Egerton University Njoro, Kenya. Trouvé sur : <http://ijcsi.org/papers/IJCSI-10-5-1-223-228.pdf>
- [2] "Plugin Development for Dynamics 365" by Alex Shlega. Trouvé sur : <https://itaintboring.com/downloads/training/Plugin%20Development.pdf>
- [3] "Microsoft Dynamics CRM 2016 - Guide des nouvelles fonctionnalités". Trouvé sur : [https://www.bcsys.fr/docs/Microsoft\\_Dynamics\\_CRM\\_2016.pdf](https://www.bcsys.fr/docs/Microsoft_Dynamics_CRM_2016.pdf)

# Webographie

- [W1] Intranet du référentiel de la méthodologie de réalisation des projets chez SQLI url : [www.SKILLS.sqli.comllo](http://www.SKILLS.sqli.comllo), Date de consultation : Avril 2020
- [W2] Le Groupe SQLI. Récupéré depuis url : [www.sqli.com/fre/le-groupe](http://www.sqli.com/fre/le-groupe), Date de consultation : Avril 2020
- [W3] La communauté Microsoft .NET francophone url : [www.dotnetfrance.com](http://www.dotnetfrance.com), Date de consultation : Mai 2020
- [W4] Le réseau des développeurs Microsoft url : [www.msdn.microsoft.com](http://www.msdn.microsoft.com), Date de consultation : Mai 2020
- [W5] Le site officiel de Dynamics 365 url : [www.microsoft.com/frfr/dynamics365/home](http://www.microsoft.com/frfr/dynamics365/home), Date de consultation : Mai 2020
- [W6] La plateforme des formations en ligne du .NET url : [www.dotnettricks.com/learn](http://www.dotnettricks.com/learn), Date de consultation : Juin 2020
- [W7] La documentation de Visual Studio Online url : [www.visualstudio.com](http://www.visualstudio.com), Date de consultation : Juin 2020
- [W8] Le journal du net url : [www.journaldunet.com/solutions/0104/010424\\_eai.shtml](http://www.journaldunet.com/solutions/0104/010424_eai.shtml), Date de consultation : Juillet 2020

# Annexes

-  **Annexe A : Le back-log de la solution de tag**
-  **Annexe B : Qu'est-ce que le CRM ?**

# ANNEXE A : Back-log de la solution de tag

Backlog 18 tickets		Créer un sprint	
➤ Réalisation d'un thème ( Front-End )	CST-1	↑	-
➤ Création d'une liste des tags	CST-2	↑	-
➤ L'ajout des entités nécessaires et ses champs liés	CST-3	↑	-
➤ Implémentation de la ressource web	CST-4	↑	-
➤ L'ajout d'un tag à la liste des tags	CST-5	↑	-
➤ Création des liaisons et des relations essentielles entre le système de Tag et les entités	CST-6	↑	-
➤ Affichage d'une liste de suggestion pour les utilisateurs	CST-7	↑	-
➤ Filtrage de la liste des tags en se basant sur un ensemble de conditions prédéfinis	CST-8	↑	-
➤ L'auto-complétion lors de la saisie	CST-9	↑	-
➤ Validation des tags ( Regex )	CST-10	↑	-
➤ L'ajout d'un record lié au tag	CST-11	↑	-
➤ Suppression des records existants	CST-12	↑	-
➤ Affichage d'un message d'erreur ( Non conformité détectée ou existence du tag )	CST-13	↑	-
➤ Présentation des différents détails de records et des tags sur un tableau de bord	CST-14	↑	-
➤ Configuration des rôles de sécurité	CST-15	↑	-
➤ Mise en place d'un système de traduction	CST-16	↑	-
➤ Exportation de la solution	CST-17	↑	-
➤ Management de l'environnement	CST-18	↑	-

## ANNEXE B : Qu'est-ce que le CRM ?

Le CRM (Customer Relationship Management, ou en français GRC, gestion de la relation client) vise à proposer des solutions technologiques permettant de renforcer la communication entre l'entreprise et ses clients afin d'améliorer la relation avec la clientèle en automatisant les différentes composantes de la relation client :

- **L'avant-vente** : il s'agit du marketing, consistant à étudier le marché, c'est-dire les besoins des clients et à démarcher les prospects. L'analyse des informations collectées sur le client permet à l'entreprise de revoir sa gamme de produits afin de répondre plus précisément à ses attentes. L'Enterprise Marketing Automation (EMA) consiste ainsi à automatiser les campagnes marketing.

- **Les ventes** : L'Automatisation des forces de ventes, consiste à fournir des outils de pilotage aux commerciaux afin de les assister dans leurs démarches de prospection (gestion des prises de contact, des rendez-vous, des relances, mais aussi aide à l'élaboration de propositions commerciales, ...).

- **La gestion du service clientèle** : le client aime se sentir connu et reconnu de l'entreprise et ne supporte pas devoir récapituler, à chaque prise de contact, l'historique de sa relation à l'entreprise.

- **L'après-vente** : consistant à fournir une assistance au client notamment via la mise en place de centres d'appel (appelés généralement Call centers, Help Desk ou HotLine) et via la mise en ligne d'informations de support technique.

L'objet du CRM est d'être plus à l'écoute du client afin de répondre à ses besoins et de le fidéliser. Un projet de CRM consiste donc à permettre à chaque secteur de l'entreprise d'accéder au système d'information pour être en mesure d'améliorer la connaissance du client et lui fournir des produits ou services répondant au mieux à ses attentes.<sup>1</sup>

---

*Cette définition est tirée du site <http://www.jsx-conseils.fr/nous/crm/> consulté le 09/07/2020*