



Ecole Supérieure des Communications de Tunis



République Tunisienne

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Carthage

École Supérieure des communications de Tunis

## Rapport de stage d'été

---

# Conception et Développement d'un module ERP Orienté Startup

---

Réalisé au sein de SmartWaves

Département Informatique, Réseaux

Réalisé par

Barkaoui Chaker

Encadreur Académique :

Asma BEN LETAIFA ,SUP'COM

Encadreur Professionnel:

Kais Ben Rhouma, SmartWaves

Année universitaire 2020/2021

# Contents

<b>List of Figures</b>	<b>3</b>
<b>List of Tables</b>	<b>3</b>
<b>Remerciement</b>	<b>4</b>
<b>Introduction</b>	<b>5</b>
<b>1 Etude préliminaire du projet</b>	<b>6</b>
1.1 Avant Propos . . . . .	6
1.2 Cadre du projet . . . . .	7
1.3 Etude de l'existant . . . . .	10
<b>2 Modélisation et Conception du module BMC</b>	<b>16</b>
2.1 Spécification des besoins généraux . . . . .	16
2.2 Diagramme de cas d'utilisation générale . . . . .	19
2.3 Module BMC:Spécification et analyse des besoins . . . . .	20
2.4 Module BMC:Conception . . . . .	22
2.5 Pilotage du projet . . . . .	25
<b>3 Environnement et phase de réalisation</b>	<b>27</b>
3.1 Environnement de travail . . . . .	27
3.2 Sprint 1 : Implémentation du frontend InnovationLab . . . . .	29
3.3 Sprint 2 : consommation de l'API . . . . .	31
<b>4 Analyse ,test et déploiement</b>	<b>33</b>
4.1 Aperçu sur le travail réalisé . . . . .	33
4.2 Déploiement de l'application . . . . .	34
<b>Conclusion</b>	<b>39</b>
<b>Bibliographie</b>	<b>40</b>

# List of Figures

1.1	Les partenaires de SmartWaves . . . . .	7
1.2	Vu globale d'un ERP . . . . .	8
1.3	Comparatif entres les ERP . . . . .	10
1.4	Les modules ERP les plus utilisés . . . . .	11
1.5	Statistique utilisateurs potentiels . . . . .	12
1.6	Interface de Swiver . . . . .	13
1.7	Business Model Canvas . . . . .	15
2.1	Parties prenante du projet . . . . .	17
2.2	Diagramme de cas d'utilisation général . . . . .	19
2.3	Diagramme de cas d'utilisation d'InnovationLab . . . . .	21
2.4	Architecture 3-tier . . . . .	22
2.5	Architecture Flux . . . . .	23
2.6	Diagramme de classe d'InnovationLab . . . . .	24
2.7	Diagramme de séquence d'InnovationLab . . . . .	25
2.8	Modèle de cycle de vie Scrum de notre projet . . . . .	26
2.9	Equipes et rôles Scrum . . . . .	26
3.1	Bundles utilisées . . . . .	28
3.2	Backlog du sprint 1 : Implémentation du frontend InnovationLab . . . . .	29
3.3	Redux pattern . . . . .	30
3.4	Niveaux de Richardson . . . . .	31
4.1	Interface d'InnovationLab : phase d'ajout d'un BMC . . . . .	33
4.2	Interface complète d'InnovationLab . . . . .	34
4.3	Noms de domaine implémentés . . . . .	34
4.4	Fichier de config SSH . . . . .	35
4.5	Processus de validation de l'agent [7] . . . . .	36
4.6	Fichier de config OpenDKIM . . . . .	37
4.7	Fichier de config principale de Fail2Ban . . . . .	37
4.8	Règle de filtrage Regex pour l'authentification . . . . .	38

# Remerciements

Je tiens à remercier dans un premier temps, toute l'équipe de SMART-WAVES pour m'avoir aidé à réaliser ce présent rapport.

Je remercie Madame Asma Ben LETAIFA, enseignante à l'École Supérieure des Communications de Tunis (SUP'COM) et ma maîtresse de stage, pour sa confiance et les connaissances qu'elle a su partager avec moi. Je la remercie aussi pour sa disponibilité et la qualité de son encadrement à SUP'COM, qui m'a fourni les outils nécessaires au bon déroulement de mon stage. Je tiens à la remercier spécialement car elle fut la première à me soutenir dans ma démarche de stage.

Je saisis cette occasion pour adresser mes profonds remerciements aux responsables et au personnel de l'École Supérieure des Communications de Tunis.

Un grand merci à ma mère et mon père, pour leurs conseils ainsi que leur soutien inconditionnel, à la fois moral et économique.

Enfin, je remercie les membres du jury qui ont accepté d'évaluer ce travail.

# Introduction

Depuis une décennie, les progiciels de gestions intégrés se sont grandement répandus. Dans les pays développés chaque entreprise petite, moyenne ou grande dispose d'un ERP et l'utilise sans cesse dans sa gestion des processus pour améliorer la gestion des ressources humaines, la gestion comptable et financière, l'aide à la décision, mais aussi la vente, la distribution, l'approvisionnement et le commerce électronique. C'est pour cela que les entreprises sont amenées à avoir un ERP qui leur permet de mieux se présenter d'une manière particulière dans le domaine commercial. En Tunisie on trouve quelques ERP traditionnels qui présentent ces modules (gestion des ressources humaine, achat, vente, finance). Mais les coûts de ces ERP sont généralement exorbitants pour les PME et manquent en matière d'optimisation et d'automatisation de Workflow. Dans ce cadre, Nous proposons de développer les modules inventaire, vente et distribution ainsi qu'administration tout en offrant une intégration entre ces modules. Notre objectif dans ce stage est de développer un nouveau module nommé Innovation qui nous offre la possibilité de présenter les BMCs des PME.

Le présent rapport synthétise le travail que nous avons effectué dans cette perspective. Il est organisé comme suit :

- Le premier chapitre intitulé Etude préliminaire du projet est consacré à la présentation de l'organisme d'accueil, ainsi que la mise en contexte du projet à mettre en place et le problème qu'il résout.
- La spécification et l'analyse des besoins seront présentées dans le deuxième chapitre intitulé Modélisation et conception de la solution dans lequel nous étudions en détail les besoins fonctionnels et non fonctionnels ainsi que la modélisation de ces besoins par le recours au diagramme de cas d'utilisation. La présentation des différents diagrammes reliés à la méthodologie adoptée qui aideront à une meilleure compréhension du fonctionnement de l'application.
- Le troisième chapitre intitulé Environnement et phase de réalisation sera dédié à présenter les outils techniques utilisés, les différents phases, la création de l'entité et le déploiement.
- Le quatrième chapitre intitulé Analyse, test et évaluation de performance présentera la partie réalisée et le scénario de test de performances et le déploiement de l'application.

Enfin nous clôturons ce rapport par une conclusion générale dans laquelle nous ferons le bilan du travail réalisé tout en évaluant les possibilités d'amélioration.

# Chapter 1

## Etude préliminaire du projet

### Introduction

Ce chapitre décrit le contexte généralisé du projet, d'une part il intègre la présentation de son environnement de stage par rapport à l'organisme d'accueil, d'autre part il détaille les objectifs à atteindre tout en exposant la problématique, on terminera par une analyse et critique de l'existant.

### 1.1 Avant Propos

#### 1.1 Présentation de SMART WAVES

SMART WAVES est une jeune entreprise tunisienne "Startup" de développement, créée en Décembre 2018 par Mr Kais Ben Rhouma et Mr Ahmed Mefteh.

- Activité : Startup structure juridique SUARL orientée développement, implémentation, intégration et consulting (ERP/CRM).
- Équipe : Elle est composée de 9 personnes (consultant fonctionnel, développeurs, Intégrateurs).
- Site web : [Smartwavesis.com](http://Smartwavesis.com)

## 1.2 Stratégies de SMART WAVES

SMART-WAVES s'est donnée pour objectif de numériser l'ensemble des acteurs du marché tunisien en fournissant un service ERP accessible à tout le monde grâce à sa simplicité et son prix raisonnable.

La société compte aussi à moyen terme, faire évoluer leurs produits dans une démarche environnementale (Un module de gestion de l'environnement et normalisé suivant l'ISO 14000 et l'RSE). Aussi leurs vision comprend l'intégration de l'aspect juridique avec la loi Startup'act et la prise en considération des différents scénarii possibles du business process et en offrant le meilleur Workflow.

## 1.3 Partenaires de SmartWaves

Parmi les partenaires de SMART WAVES :



Figure 1.1: Les partenaires de SmartWaves

## 1.2 Cadre du projet

### 2.1 Mise en contexte

#### 2.1.1 Définition du terme ERP

Un ERP (Entreprise Ressources Planning), aussi appelé Progiciel de gestion intégré (PGI) est une solution logicielle qui permet la gestion de l'ensemble des processus d'une entreprise et ce par le biais d'une décomposition en plusieurs modules.

L'ensemble des aspects de l'entreprise sont ainsi intégrés dans une base de données unique et qui se caractérise par un moteur à « Workflow ». En effet, l'ERP permet de générer un processus automatique qui comprend tant les différentes opérations de gestion (processus business) ainsi que leurs points d'intégration. Les différents modules financiers, comptables, ressources humaines, logistique, production sont en adéquation entre eux, ils s'intègrent de façon automatique afin de donner la meilleure réponse possible suivant une situation donnée. Le moteur à « Workflow » est déterminé par les meilleurs pratiques dans les meilleurs bords .

Aujourd'hui on parle aussi d'ERP II qui offre en plus par rapport à un ERP classique l'intégration des fournisseurs et des clients de façon automatisée dans la base de données de l'entreprise pour une excellence opérationnelle.

### 2.1.2 Types d'ERP

On peut caractériser les projets ERP en 3 catégories distinctes suivant la taille du projet ou la taille de l'entreprise qui compte intégrer cet ERP. On distingue donc 3 niveaux :

- Niveau 1 : Les plus puissants mais les plus coûteux, ils sont généralement implémentés dans le secteur industriel.
- Niveau 2 : S'adressant aux entreprises de taille moyenne, ils offrent une panoplie de produits qui n'a rien à envier au précédent, mais est limité par sa capacité de traitement. De nombreuses entreprises les adoptent ce qui permet à ces dernières de concurrencer les plus grandes entreprises et soutenir leurs croissances.
- Niveau 3 : Destiné aux petites entreprises leurs prix et fonctionnalités varie fortement d'un éditeur à un autre.



Figure 1.2: Vue globale d'un ERP

### 2.2 Le Business Model

Le business model a pour fonction de décrire la manière dont une entreprise crée de la valeur et assure ainsi sa propre pérennité. C'est la traduction concrète de la "bonne idée" de départ de l'entrepreneur, et de la manière dont il va l'exploiter sur le marché. Très concrètement, cela traduit la manière dont l'entreprise fait des affaires. [1]

### 2.3 Problématiques Résolus par SMART WAVES

Notre projet consiste à la Mise en place d'un ERP tout en prenant en compte les déficiences que présentent les entreprises au niveau de leurs performances et leur processus métier afin d'automatiser et d'optimiser la gestion des processus aux seins de l'entreprise, ainsi qu'offrir une coordination des services pour atteindre une meilleur traçabilité et une minimisation des gâchis de ressources.



### 2.4 Solutions proposées par SMART WAVES

Mettre une solution ERP pour les petites et moyennes entreprises, ainsi que les startups suite à la déficience en terme organisationnel. La solution doit permettre l'automatisation du flux des entreprises (RH, Achat, etc...) mais aussi garantir une prise de décision plus rapide grâce à des fonctionnalités adaptées aux différents domaines existants.

### 2.5 Modules développés par SMART WAVES

Intégration d'une solution de gestion optimisée, développement d'un ERP orienté petites et moyennes entreprises, startups et artisans, automatiser les flux de gestion (Audit, business process, Workflow, Workload, Paramétrable).

La solution qui sera développée doit comprendre au minimum les modules suivants pour pouvoir être commercialisée :

Un module Achat

Un module vente et distribution

Un module de gestion RH

Un module de gestion d'inventaire

Un module d'Innovation (dans le cadre de ce stage)

Les modules en question doivent interagir entre eux grâce à des points d'intégration qui sont paramétrables. Ils doivent aussi prendre en considération les spécifications de chacune des entreprises et leurs domaines d'activité.

### 2.6 Les utilisateurs clés et les utilisateurs finaux

Utilisateurs clés : Responsables IT, chef d'entreprise, Consultant fonctionnel, Intégrateur...

Utilisateurs finaux : Association, Artisans, Agence immobilière, Startup, Industrie..

### 2.7 Charte de projet ERP

Dans un souci d'excellence opérationnelle, SmartWaves s'est donnée comme ligne directrice de suivre la norme internationale de management de projet ERP PMBOK (Project Management Body of Knowledge). Parmi les composantes les plus importantes de cette norme, on dénote la charte ainsi que l'OTP (organigramme technique de projet), nous allons sur cette partie expliciter notre charte pour l'ERP startup suivant une cartographie mentale.

## 1.3 Etude de l'existant

### 3.1 Analyse de l'existant

Bien qu'il existe un grand nombre d'éditeurs d'ERP de nos jours sur le terrain la plus part ne répondent pas aux exigences fonctionnelles requis par le marché Tunisien pour les micro-entreprises. En effet, les solutions annexes sont généralement délaissées lors de l'intégration fonctionnelle.

Certes, il existe des solutions qui ont fait leurs preuves à l'échelle internationale mais leurs coûts d'implémentation et de suivi demeurent relativement élevés.

Comme illustrées dans le tableau ci-dessous, nous allons donner un comparatif entre différentes solutions Open Source et Éditeur.





	ERP Editeur	ERP Open Source
Avantages	Fiabilité, pérennité, adaptabilité, assistance, support	Moins cher, modifiable, extensible, plus facile à intégrer
Inconvénients	Dépendance éditeur, flexibilité, adaptabilité, <u>cout</u>	Maintenance, gestion opérationnelle, rigidité
Quelques ERP correspondant à ces critères	 	 

Figure 1.3: Comparatif entres les ERP

Les géants sur le marché tel que SAP, Oracle ou Dynamics présentent une excellence opérationnelle mais outre leurs coûts exorbitants pour les PME/TPE, ils ne prennent pas en considération la situation actuelle tunisienne (Tel que Startup act) dans les process flow juridique locaux.

Les modules ERP les plus utilisés dans les entreprises :

Généralement les entreprises sont dotées d'un système comptabilité étant un domaine générique à obligation légale. Le schéma suivant décrit bien les modules ERP les plus utilisés au sein des entreprises et nous donne un préavis sur les modules qu'on devra adopter dans notre ERP.

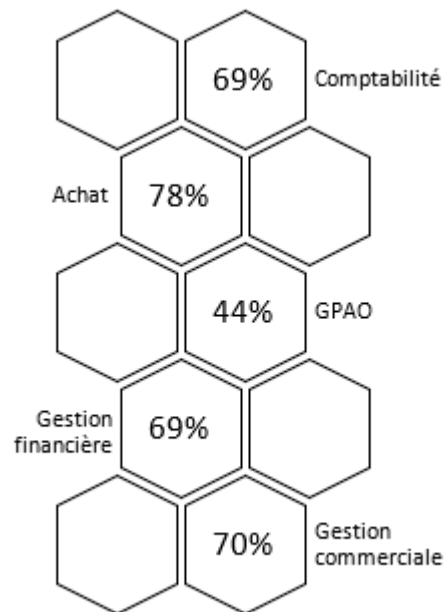


Figure 1.4: Les modules ERP les plus utilisés

Comme on le constate sur la figure ci-dessus, bien que le coût exorbitant de l'ERP fait généralement douter certaines entreprises quant à sa nécessité, il intéresse néanmoins des acteurs dont les chiffres d'affaires sont dans la moyenne et ceci démontre qu'un système qui centralise toutes les solutions informatiques et optimise la prise de décision attire bien plus que d'autres les PME et les Startup.

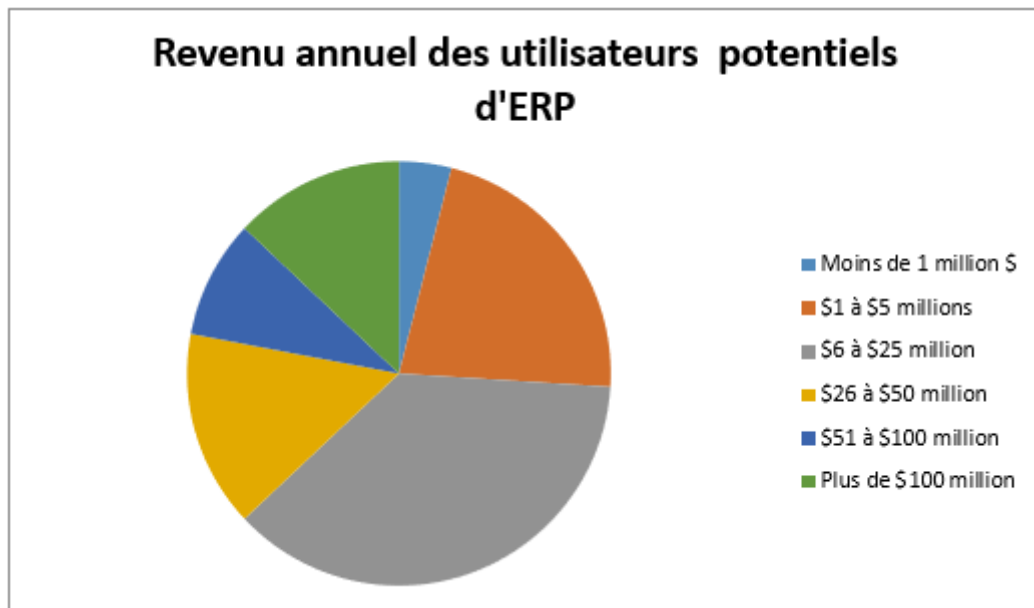


Figure 1.5: Statistique utilisateurs potentiels

### 3.2 Critique de l'existant

Vu la demande croissante des entreprises en matière d'ERP, le marché n'a guère d'offre qui satisfait leurs besoins et prend en considération leurs limites budgétaires, à part quelques outils financiers qui existent en Tunisie comme Swiver. Ces derniers ne sont pas des ERP proprement parlés et manquent en effet en matière d'automatisation et d'optimisation de Workflow de l'entreprise, les modules de gestion de production, de projet et d'entrepôt restent aussi inexistantes.

### 3.3 Swiver

Nous avons fait une analyse de la solution existante et nous avons trouvé ses limites :

- Swiver n'est pas destiné aux PME comme par exemple : les artisans.
- Swiver ne contient pas un module juridique qui prend en charge les documents juridiques.
- Le blog de Swiver n'est pas très détaillé.
- Swiver ne présente pas un modèle économique qui définit ce que l'entreprise va vendre, auprès de quels clients, dans quel but, de quelle manière, et pour quel bénéfice.



Figure 1.6: Interface de Swiver

### 3.4 Solution proposée

Après une étude comparative des solutions existantes, nous allons prendre en charge les inconvénients et mettre une solution ERP qui présente un outil pour dresser un état des lieux du modèle économique d'une entreprise. Nous allons donc proposer d'ajouter un InnovationLab permettant aux clients d'ajouter le Business Model Canvas adapté à leur besoins.

### 3.5 Business Model Canvas

Toute entreprise ou projet débute par une idée ; Le business model est alors la traduction de cette idée de départ pour la concrétiser en une structure qui génère un profit ou crée de la valeur. Le Business Model Canvas est tout simplement un outil qui permet de structurer et de visualiser votre modèle économique. Ce type de visualisation vous aide à structurer vos idées, à pousser à la réflexion et à la créativité et à s'assurer que vous avez cerné la question et vous l'avez abordé sous tous ses angles importants.

Les composantes du Business Model Canvas Le modèle proposé par Alexander Osterwalder du business model canva comprend 9 composantes :

1. Segmentation Clients : Dans votre modèle économique il faut commencer par connaître vos clients. Il est très important de bien connaître ses clients et les différents segments auxquels ils appartiennent, afin de leur offrir un produit ou un service qui leur correspondent.
2. Proposition de valeur : Ceci est l'offre que votre entreprise, projet ou service fait à ses clients. Pour bien formuler votre proposition de valeur, il faut savoir les besoins que l'entreprise va satisfaire à ses clients, sa valeur ajoutée, ses points forts par rapport à la concurrence.

3. Canaux de distribution : Dans cette composante l'entrepreneur va indiquer où son produit/service sera vendu ainsi que les relations qui existent entre ces canaux. Les canaux de distribution peuvent être directs, tels que la vente sur internet ou indirects tels la vente à des grossistes.

4. Relation Client : La relation client détermine comment l'entrepreneur va interagir avec ses différents segments de clients. Ceci comprend la stratégie de communication et de fidélisation des clients.

5. Sources de revenus : Les revenus sont tous les flux d'argent qui sont générés par l'entreprise. les revenus peuvent être issus de ventes directes en « one shot », d'abonnements récurrents, par la location, par l'octroi d'une licence, frais de courtage, etc . . .

6. Ressources clés : Avant de se lancer, il est essentiel de clarifier dans son Business Model les différents ressources dont vous aurez besoin. Ceci nous amène à fixer les ressources humaines, financières et matérielles nécessaires au lancement de son projet.

7. Partenaires clés : Il n'est pas toujours nécessaire d'apporter toutes les ressources clés soit même, il est possible, et même recommandable d'avoir recours à des partenaires dans certains cas.

8. Activités Principales : Ceci nous amène à évoquer la ou les activités clés. Par exemple, la chaîne logistique, le développement logiciel . . .

9. Structure de coûts : Le point crucial pour chaque entreprise est le coût. Il est donc primordial de bien maîtriser ses coûts. Il faut tout d'abord connaître les coûts nécessaires à la réalisation de proposition de valeur, à l'entretien des relations clients et à la génération de revenus, les différents types de coûts pour votre projet, Coûts fixes, coûts variables, économie d'échelle.



Figure 1.7: Business Model Canvas

## Conclusion

Tout au long de ce chapitre, nous avons consacré une section pour lister les différents concepts clés puis nous avons étudié et critiqué les solutions existantes et nous avons pris comme exemple concret le Swiver pouvoir positionner notre solution. Nous avons fini par détailler notre solution proposée.

## Chapter 2

# Modélisation et Conception du module BMC

### Introduction

Dans ce chapitre, nous présentons l'analyse et la spécification de notre projet. Nous commençons par présenter les patrons de conception utilisés par la suite les besoins fonctionnels et non fonctionnels de l'application, spécifier les diagrammes de cas d'utilisation qui aideront à comprendre le comportement fonctionnel de la solution. Le présent chapitre est consacré à l'analyse des besoins fonctionnels et non fonctionnels ainsi que la spécification des exigences de notre projet. Nous entamons le chapitre par identifier les acteurs et faire une étude des besoins. Ensuite, nous détaillons les différents cas d'utilisation par des diagrammes des cas d'utilisation et des diagrammes de séquence.

## 2.1 Spécification des besoins généraux

### 1.1 Parties prenantes du projet

Une partie prenante représente chaque acteur du projet qui pourrait apporter une valeur ajoutée au produit final ou influencer ses exigences, les relever nous permettra d'avoir une idée générale sur notre application et les fonctionnalités qu'elle doit adopter.

Le tableau ci-dessous décrit les parties prenantes de notre projet :



Stakeholder	Description
Administrateur de l'entreprise	La personne responsable de l'administration des utilisateurs, sa tâche principale est l'affectation des modules spécifiques à chacun et la configuration du profil de l'entreprise.
Responsable VD (Vente et distribution)	La personne affectée par l'administrateur de l'entreprise au module VD, son rôle est la gestion des commandes, de la facturation et des produits.
Responsable d'inventaire	La personne affectée par l'administrateur de l'entreprise au module inventaire, son rôle est la gestion du flux et des emplacements.
Product owner	Il a le rôle de nous guider dans le développement afin de suivre les bonnes pratiques en ERP et de spécifier les fonctionnalités que la solution doit implémenter.
Ministère de l'économie numérique	Le rôle de ce ministère est la régularisation des outils financiers utilisés par les entreprises ainsi que l'adoption de lois relative aux startups.

Figure 2.1: Parties prenante du projet

## 1.2 Les besoin fonctionnels

Les besoins fonctionnels définissent les attentes et les fonctionnalités de l'application à développer par rapport aux acteurs. Au cours de cette étape, nous allons présenter les différentes fonctionnalités offertes par notre projet.

Authentification : L'authentification devra être « Stateless » à travers JWT.

Gérer compte : Un administrateur peut ajouter un nouvel utilisateur en lui envoyant un email de vérification.

Configuration du profil de l'entreprise : Un administrateur peut modifier ou supprimer les données relatives à son entreprise.

Gestion des utilisateurs : Un administrateur peut consulter ou désactiver le compte d'un utilisateur, il peut aussi lui affecter de nouveaux modules.

Gestion des commandes : Un responsable VD peut créer ou consulter une commande, il peut aussi changer son statut et générer la facturation.

Gestion des produits : Un responsable VD peut créer ou consulter un produit ou le désactiver.

Gestion des clients : Un responsable VD peut consulter l'historique des clients.

Consultation du reporting : Un responsable VD peut faire un rapport de l'activité de son entreprise.

Gestion des emplacements : Un responsable inventaire peut ajouter un emplacement spécifique à un entrepôt.

Gestion du flux : Un responsable inventaire peut gérer le flux entrée/sortie d'un entrepôt.

### **1.3 Les besoins non fonctionnels**

Les besoins non fonctionnels représentent les propriétés, et contraintes d'implémentation de notre application, certains critères doivent être pris en considération tels que :

- La sécurité : L'application devra être hautement sécurisée, les informations ne doivent pas être accessibles à tous. Cette contrainte est primordiale vu l'architecture SaaS adoptée.
- La modularité : Chaque module de l'application devra être indépendant des autres, ceci pour souligner les différentes fonctionnalités et rôles remplis pour chaque module.
- L'extensibilité : L'application devra être hautement extensible afin de faciliter l'ajout d'autres modules ultérieurement.
- L'ergonomie : L'application devra être simple et facile à manipuler même par des non experts.

## 2.2 Diagramme de cas d'utilisation générale

La figure 5 Ci-dessous représente le diagramme de cas d'utilisation général de notre projet nécessaire pour décrire le comportement de notre application.

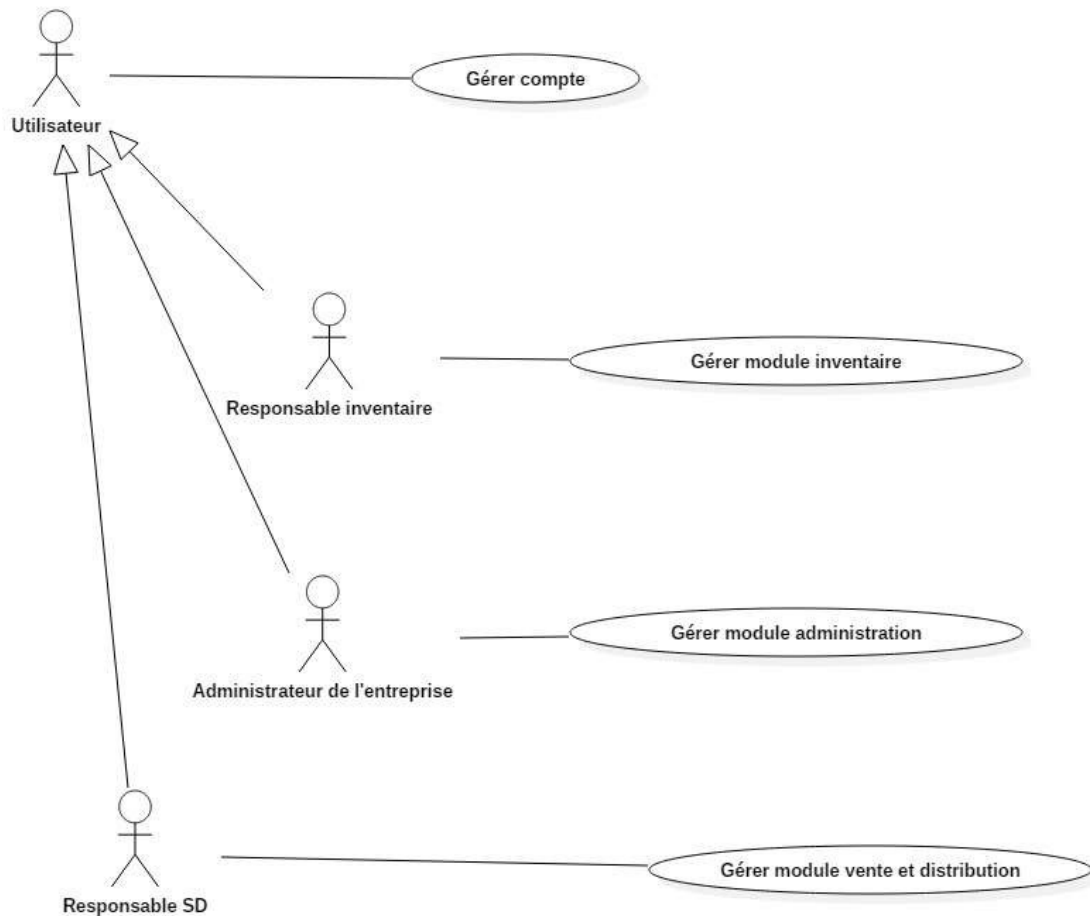


Figure 2.2: Diagramme de cas d'utilisation général

## 2.3 Module BMC:Spécification et analyse des besoins

Dans cette section nous évoquons une étape importante dans le déroulement du projet qui sert à déterminer les différentes fonctionnalités attendues du système pour ses acteurs.

### 3.1 Analyse des besoins fonctionnels

Les besoins fonctionnels représentent les divers services attendus par l'utilisateur. nous allons présenter les besoins fonctionnels pour chaque acteur. Le système doit permettre à un administrateur :

\*\* d'accorder à un utilisateur le droit de gérer InnovationLab Le système doit permettre à un utilisateur de :

1. S'authentifier
2. Ajouter un BMC
3. Consulter un BMC
4. Ajouter, modifier ou supprimer des partenaires
5. Ajouter, modifier ou supprimer des activités clés
6. Ajouter, modifier ou supprimer des ressources clés
7. Ajouter, modifier ou supprimer des propositions de valeurs
8. Ajouter, modifier ou supprimer des relations clients
9. Ajouter, modifier ou supprimer des canaux de distribution
10. Ajouter, modifier ou supprimer des segmentations client
11. Ajouter, modifier ou supprimer des coûts
12. Ajouter, modifier ou supprimer des revenus
13. Convertir le BMC en format pdf
14. Imprimer le BMC

### 3.2 Aperçu fonctionnel global

Le diagramme des cas d'utilisation général suivant présente les fonctionnalités offertes par InnovationLab ainsi que les relations entre les différents cas d'utilisation.

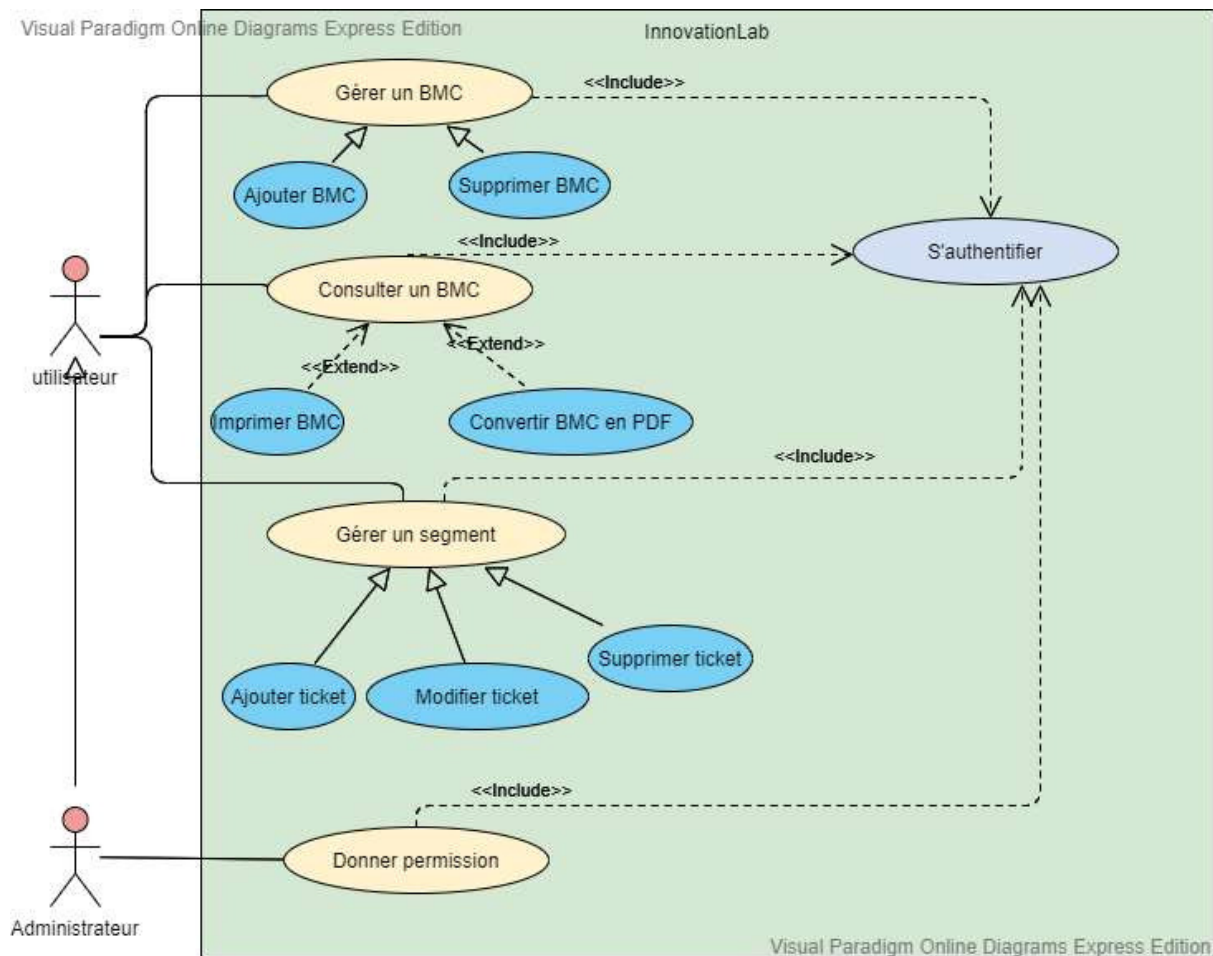


Figure 2.3: Diagramme de cas d'utilisation d'InnovationLab

### 3.3 Analyse des besoins non fonctionnels

Les besoins non fonctionnels représentent les propriétés et les contraintes d'implémentation de notre application. les critères qui doivent être pris en considération pour aboutir à une solution adéquate :

- La modularité : Chaque module de l'application devra être indépendant des autres, ceci pour souligner les différentes fonctionnalités et rôles remplis pour chaque module.
- l'extensibilité : L'application devra être hautement extensible afin de faciliter l'ajout d'autres modules ultérieurement.
- L'ergonomie : L'application devra être simple et facile à manipuler même par des non experts.

## 2.4 Module BMC:Conception

Dans cette section, nous allons présenter la conception de notre projet. Nous commençons par définir l'architecture physique et logique de notre système ensuite nous allons évoquer la conception détaillée.

### 4.1 Architecture physique

L'architecture physique représente les composants matériels du système . Après spécification des besoins, nous avons opté pour l'architecture trois tiers qui présente trois couches

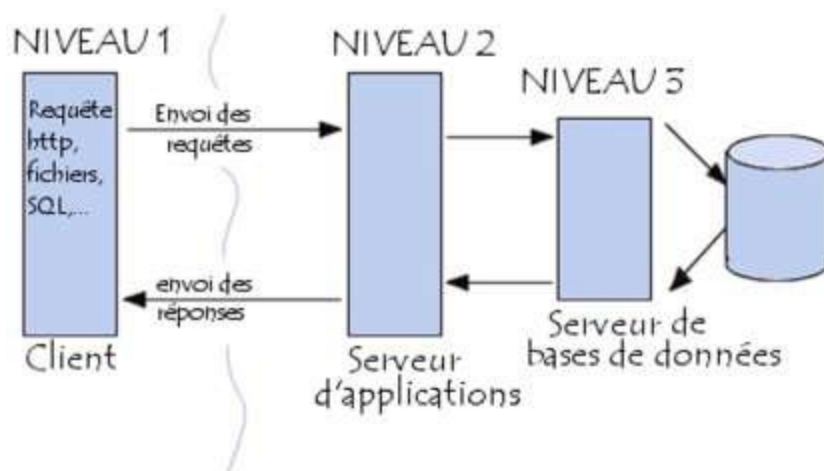


Figure 2.4: Architecture 3-tier

## 4.2 Architecture logique

L'architecture logique représente la manière avec laquelle nous allons concevoir notre système pour répondre aux besoins spécifiés. Dans notre projet, nous avons utilisé l'architecture Flux. FLUX est une architecture pour application web proposée par Facebook dans le but de construire des applications avec React en évitant les problèmes confrontés lors du développement d'applications web massives dont la quantité s données est grande. [5] La figure suivante illustre le cycle de l'architecture Flux.



Figure 2.5: Architecture Flux

Chaque action de l'utilisateur est un tour complet de cette boucle. Cette boucle se présente de la manière suivante :

- L'Interface est le point d'entrée de l'architecture. Elle est votre librairie de gestion d'interface préférée (React, VDOM, etc. . . ). Elle capte les actions de l'utilisateur .
- A partir de cette action utilisateur un objet action sera créé. Il contient les informations nécessaires à identifier l'action de l'utilisateur .
- Les actions sont envoyées à un Dispatcher. C'est un objet Observer/Observable.
- Le store va créer un nouvel état de l'application en se passant sur chaque action reçue depuis le dispatcher et de l'état courant de la donnée stocké en son sein.

## 4.3 Conception détaillée

Afin de bien modéliser l'architecture de notre système, nous allons présenté un diagramme de classe. Il sert d'avantage à présenter l'aspect statique et l'aspect dynamique détaillant les classes constituant notre système et les relations entre elles ainsi que le séquençement de notre solution.

## 4.4 Aspect statique

L'aspect statique identifie les propriétés des objets et leurs liaisons avec les autres objets. Pour bien mettre en évidence cet aspect, nous avons opté pour le diagramme de classe.

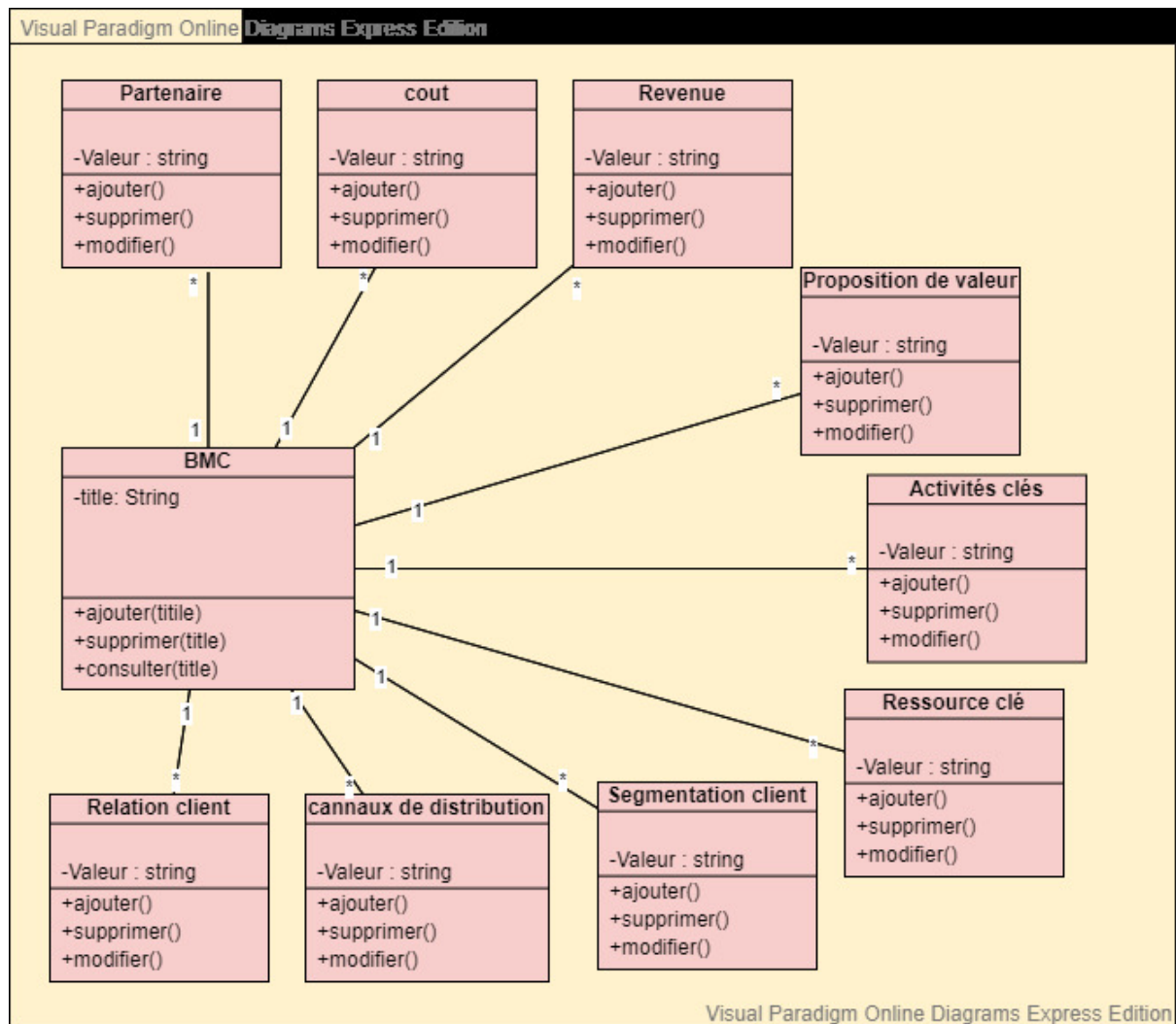


Figure 2.6: Diagramme de classe d'InnovationLab



### 4.5 Aspect dynamique

L'aspect dynamique présente le comportement du système et l'évolution des objets dans le temps. Pour représenter l'enchaînement dans notre système, nous avons opté pour le diagramme de séquence.

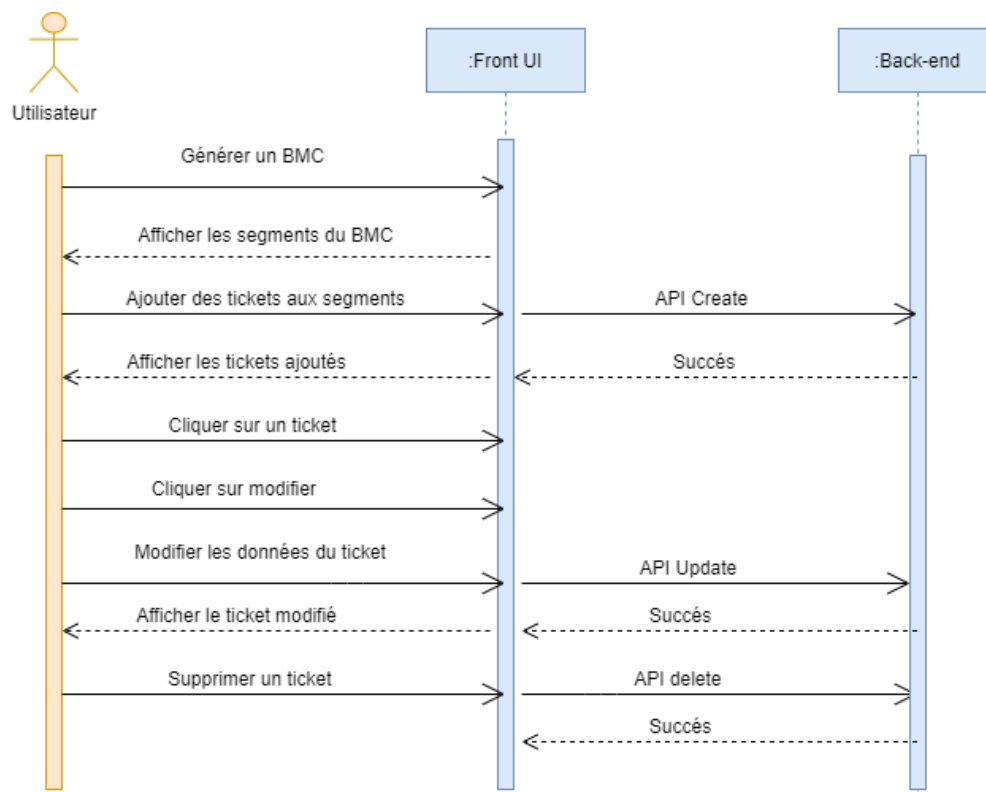


Figure 2.7: Diagramme de séquence d'InnovationLab

## 2.5 Pilotage du projet

### 5.1 La méthodologie Scrum

Pour réaliser ce projet, nous avons adapté la méthodologie Scrum.

La méthodologie Scrum est un processus qui commence par découper notre planning en plusieurs itérations ou sprint, chaque itération va compter entre deux à quatre semaines en fonction de la taille de l'équipe, de la nature du produit et du contexte de l'application. A la fin de chaque sprint, une livraison d'incrément de produit qui va certes garantir un résultat partiel mais qui reste fonctionnel et opérationnel. Dans chaque période de Sprint, on doit proposer une stratégie de développement pour réaliser les incréments de fonctionnalités demandé par l'utilisateur. Elle consiste à décomposer ces fonctionnalités en tâches et leurs attribuer une durée (en heure).

Pour réaliser cela, nous illustrons dans la figure suivante l'approche pour le développement de notre solution.

La figure 3 représente le modèle de cycle de vie Scrum adopté dans notre projet.

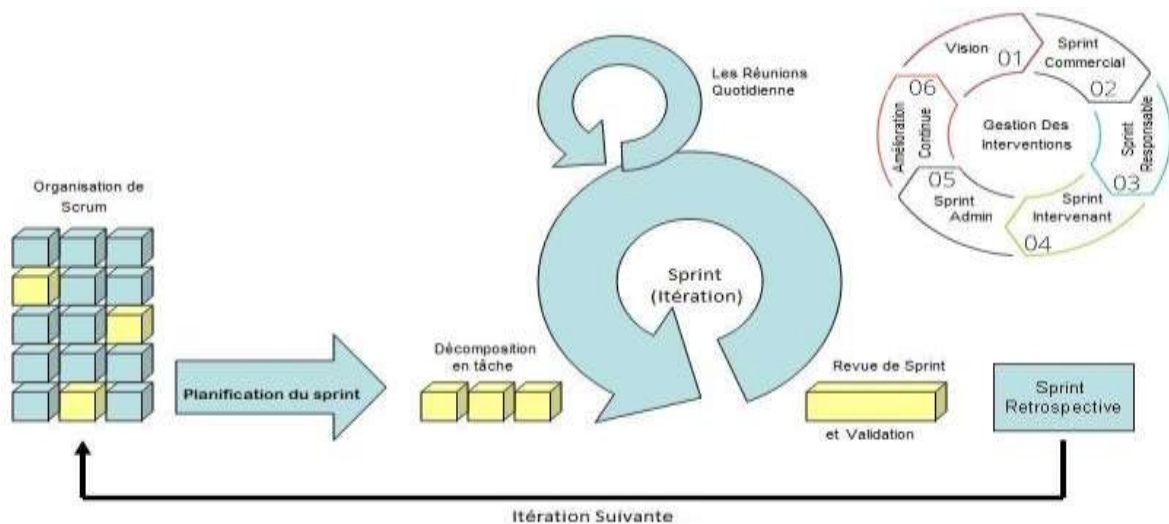


Figure 2.8: Modèle de cycle de vie Scrum de notre projet

## 5.2 Equipes et rôles

Nous commençons tout d'abord par définir les rôles de chacun dans le projet.

Rôle Scrum	Personnes affectés
Product Owner	Le chef de projet dans SMART-WAVES: <u>Mr Ben Rhouma Kais</u>
Scrum Master	Le dirigeant de l'équipe au sein de SMARTWAVES: <u>Mr Meftah Ahmed</u> .
Team	L'équipe de conception et développement

Figure 2.9: Equipes et rôles Scrum

## Conclusion

Tout au long de ce chapitre, nous avons consacré une section pour analyser et spécifier les besoins fonctionnels et non fonctionnels de notre projet et nous avons présenté la conception globale et détaillée que nous avons opté pour réaliser notre solution. Le suivant chapitre va évoquer les différentes phases de réalisation.

# Chapter 3

## Environnement et phase de réalisation

### Introduction

Dans ce chapitre, nous présentons l'environnement matériel et logiciel ainsi que les technologies utilisées pour le développement de notre solution. Dans un deuxième lieu, nous exposons les sprints de notre travail en présentant des captures élaborant les différentes fonctionnalités.

### 3.1 Environnement de travail

Cette section présente l'environnement de travail que nous avons utilisé pour réaliser notre application.

- Environnement Matériel : Nous avons travaillé avec une machine dotée d'un microprocesseur Intel Core i7 avec une fréquence de 2.70GHz et 12GB de RAM. Cette machine fonctionne sous Microsoft Windows 10 64 bits.

- Environnement logiciel : Nous allons commencer par l'exposition de l'environnement logiciel et les langages de programmation utilisés.

\*Visual Studio Code : c'est un éditeur extensible de code source développé par Microsoft. Il est utilisé avec plusieurs langages de programmation, tel que Java, JavaScript, Node.js et C ++. Il intègre plusieurs fonctionnalités et outils de développement, notamment le débogage, la mise en évidence de la syntaxe, un Git intégré... Classé comme l'éditeur le plus populaire, il est le plus adéquat pour garantir une productivité maximale des développeurs .

\*GitLab : c'est un logiciel libre et collaboratif de forge basé sur git et développé par GitLab Inc. GitLab donne la possibilité de gérer ses dépôts Git . Il assure l'hébergement des projets web et la gestion de versions des codes sources ainsi que la gestion de tout le processus de développement. Ce logiciel performant est utilisé par des grandes entreprises tel que IBM, Sony.. .

\*React : est un Framework Javascript pour le développement d'interface utilisateur. Il est principalement utilisé comme une base pour la réalisation d'application web « Single page », il se démarque par rapport à ses concurrents par sa flexibilité et ses performances

à travers l'utilisation du DOM virtuel, ne mettant à jour dans l'interface seulement les composantes qui nécessite d'être changé, cela se fait par un stockage local des structures des données et une évaluation des changements par rapport à la structure existante.

\*Symfony : est un ensemble de composants PHP ainsi qu'un Framework MVC libre écrit en PHP. Il fournit des fonctionnalités modulable et adaptable qui permettent d'accélérer le développement web. L'avantage principal de Symfony par rapport aux autres Framework est l'utilisation de Bundles qui se traduisent par un ensemble de fonctionnalités qui sont totalement indépendant du reste du projet offrant ainsi la modularité souhaitée, chaque Bundle représentera donc un module distinct qu'on va développer.

L'implémentation de Bundle tiers nous aidera aussi dans la réalisation de l'API, ci-dessous un tableau représentatif de l'ensemble de Bundles utilisées lors du processus de développement.

Nom du Bundle	Description
SecurityBundle	Garantit la gestion d'accès RBAC ainsi que l'authentification et la sécurisation des ressources.
MonologBundle	Offre un logging du réseau, de la base donnée ainsi que les web services et l'accès aux fichiers serveur.
SwiftMailerBundle	Offre un service d'envoi d'email qui s'intègre avec le serveur d'email.
DoctrineBundle	L'ORM le plus utilisé par les applications PHP, offre une interface entre la base de données et l'application.
FOSRestBundle	Offre des outils pour développement du serveur REST comme la sérialisation et facilite le traitement des requêtes http.
LexikJWTAuthentication	Facilite la configuration du serveur d'authentification JWT.
NelmioApiDocBundle	Offre des annotations qui facilite la génération de documentation pour l'api REST.
NelmioCORSBundle	Offre une interface de configuration des règles CORS de l'API.
KnpsnappyBundle	Une librairie qui convertit un fichier HTML en pdf, nécessaire pour la génération des facturations.

Figure 3.1: Bundles utilisées

-Technologies et langages:

PHP[8] : Un langage de programmation libre utilisé pour le développement web, c'est un langage non typé avec une syntaxe assez facile, il intègre aussi tous les concepts de l'orienté objet.

JSX : Une extension du langage JavaScript, il va de pair avec le Framework ReactJS vu qu'il permet un couplage entre le code logique de JavaScript et les balises de html permettant un ainsi débogage plus facile et s'aligne avec la philosophie Soc de React à travers les fonctionnalités des « Components ».

## 3.2 Sprint 1 : Implémentation du frontend InnovationLab

Après la planification des sprints et la spécification des besoins , nous allons commencer par le sprint 1 qui présente l'implémentation de l'interface InnovationLab. Ce sprint a duré 4 semaines.

Feature	User story	Priority
Implémentation du role "utilisateur"	En tant qu'un utilisateur, je peux m'authentifier avec un login et mot de passe	Élevée
	En tant qu'un utilisateur, je peux ajouter un BMC	Élevée
	En tant qu'un utilisateur, je peux consulter un BMC existant	Élevée
	En tant qu'un utilisateur, je peux ajouter , supprimer ou modifier un partenaire	Élevée
	En tant qu'un utilisateur je peux ajouter, modifier et supprimer des activités clés	Élevée
	En tant qu'un utilisateur je peux ajouter, modifier et supprimer ressources clés	Élevée
	En tant qu'un utilisateur je peux ajouter, modifier et supprimer des propositions de valeur	Élevée
	En tant qu'un utilisateur je peux ajouter, modifier et supprimer des relations clients	Élevée
	En tant qu'un utilisateur je peux ajouter, modifier et supprimer des segmentations clients	Élevée
	En tant qu'un utilisateur je peux ajouter, modifier et supprimer des canaux de distributions	Élevée
	En tant qu'un utilisateur je peux ajouter, modifier et supprimer des couts	Élevée
	En tant qu'un utilisateur je peux ajouter, modifier et supprimer des revenus	Élevée
	En tant qu'un utilisateur je peux imprimer et convertir en format pdf la table BMC	Normale

Figure 3.2: Backlog du sprint 1 : Implémentation du frontend InnovationLab

## 2.1 Frameworks et technologies utilisées

\*React : c'est une bibliothèque JavaScript open source développée par Facebook .React rend la création des interfaces utilisateurs interactives. Elle mettra à jour, de façon optimale, juste les composants qui en auront besoin

Redux : c'est une bibliothèque open-source JavaScript de gestion d'état pour applications web. Elle est plus couramment utilisée avec des bibliothèques comme React ou Angular pour la construction d'interfaces utilisateur. Semblable à (et inspirée de) l'architecture Flux, elle a été créée par Dan Abramov et Andrew Clark.Redux est une petite bibliothèque conçue pour être un conteneur d'état d'application. Elle est influencée par la programmation fonctionnelle.

## 2.2 Justification du choix de Redux

Le choix de Redux était bien étudié et ceci pour ces raisons :

- Redux permet une gestion de “states globaux”. nous pouvons pallier aux faiblesses des props qui ne sont accessibles qu'en lecture seule ou des states qui sont liés à un unique composant, et donc seulement localement.
- Redux permet donc de faciliter le développement de toute application dès que celle-ci implique l'existence de composants dépendants les uns des autres.[8]

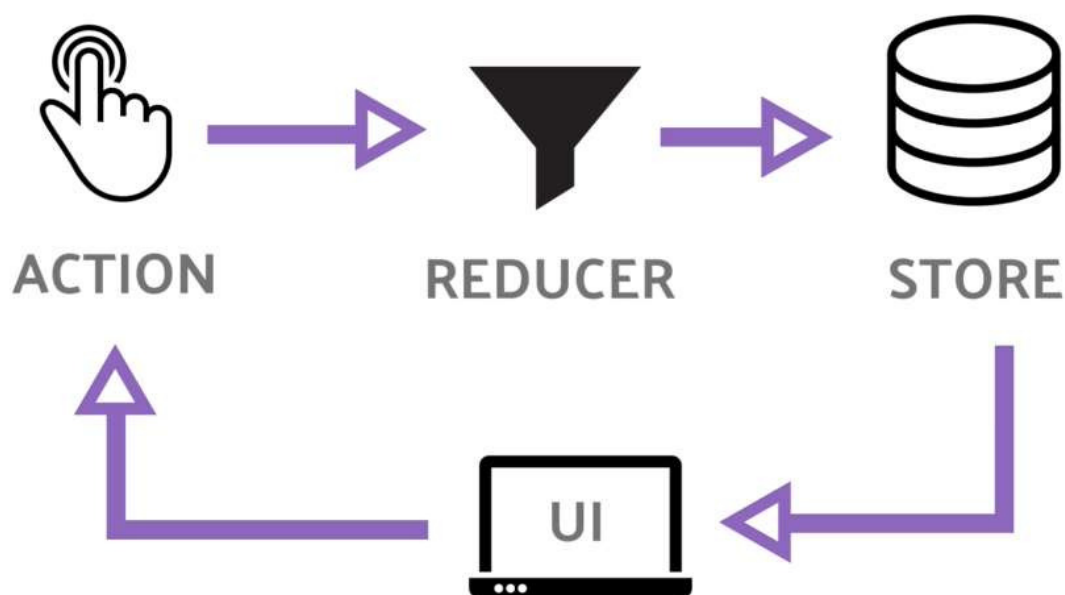


Figure 3.3: Redux pattern

### 3.3 Sprint 2 : consommation de l'API

Après la phase de l'implémentation , nous avons commencer par le sprint 2 qui présente la consommation des API . Ce sprint a duré 2 semaines. Dans ce qui suit, nous allons présenté les outils que nous avons utilisé pour mener à bien cette tâche.

Nous allons ainsi expliquer l'importance de cette phase pour aboutir un système répondant aux besoins des clients. Une API RESTful est un style architectural pour une interface de programme d'application (API) qui utilise des requêtes HTTP pour accéder et utiliser des données. Ces données peuvent être utilisées pour les types de données GET, PUT, POST et DELETE, qui font référence à la lecture, la mise à jour, la création et la suppression d'opérations concernant les ressources.

Une API pour un site Web est un code qui permet à deux logiciels de communiquer entre eux. L'API décrit la manière appropriée pour un développeur d'écrire un programme demandant des services à un système d'exploitation ou à une autre application.

#### Les niveaux de Richardson [4]

Pour implémenter notre API nous utiliserons La classification de Richardson ; créée par Leonard Richardson après l'analyse de centaines d'API, il les a classés par la suite en quatre niveaux de maturité. Le digramme ci-dessous représente les niveaux en question.

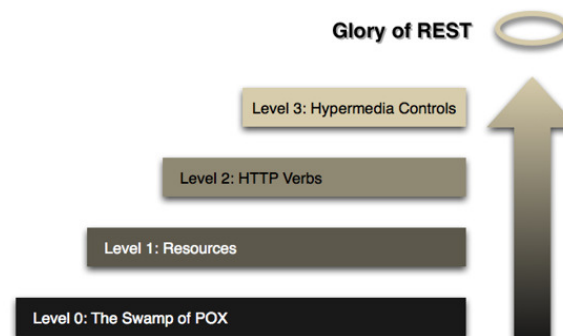


Figure 3.4: Niveaux de Richardson

Le niveau 0 : Pas d'utilisation de http méthodes spécifique ni d'URI, on utilise simplement le protocole http pour échanger des données avec une seule ressource et la même méthode (généralement POST).

Le niveau 1 : Utilisation d'URI multiple pour chaque opération mais la même requête http.

Le niveau 2 : Utilisation d'URI multiple mais aussi des méthodes spécifique à chaque type d'opération (POST, PUT, GET, DELETE) et des code de réponse http qui prennent en considération le traitement qui a eu lieu dans le serveur.

Le niveau 3 : Api auto découvrable à travers l'implémentation HATEOS, joignant à la réponse d'autres URI.

### Consommation de l'API avec Axios

Axios est un client HTTP based-promise facile à utiliser pour le navigateur et node.js. Étant donné qu'Axios est based-promise, nous pouvons tirer parti de l'async et await un code plus lisible et asynchrone. Avec Axios, nous avons la possibilité d'intercepter et d'annuler la demande, il dispose également d'une fonctionnalité intégrée qui offre une protection côté client contre la falsification de demandes intersites.[2] Les fonctionnalités de l'Axios

- Interception des demandes et des réponses.
- Gestion des erreurs simplifiées.
- Protection contre XSRF.
- Prise en charge de la progression du téléchargement
- Délai de réponse
- La possibilité d'annuler les demandes
- Prise en charge des anciens navigateurs
- Transformation automatique des données JSON

### Conclusion

Ce chapitre illustre la phase de réalisation de notre projet durant la période de notre stage d'été. Nous avons commencé par présenter l'environnement de travail en expliquant les différents Framework et outils utilisés. Ensuite, nous avons détaillé les sprints en mettant l'accent sur les aspects théorique et les outils adaptés. Dans le chapitre suivant, nous allons présenter le déploiement de notre application.



# Chapter 4

## Analyse ,test et déploiement

### Introduction

Dans ce chapitre, nous allons présenter la partie réalisée et le test de fonctionnalité ,présenter les outils utilisés pour le déploiement l'application ainsi que la configuration qui a été mise en place.

### 4.1 Aperçu sur le travail réalisé

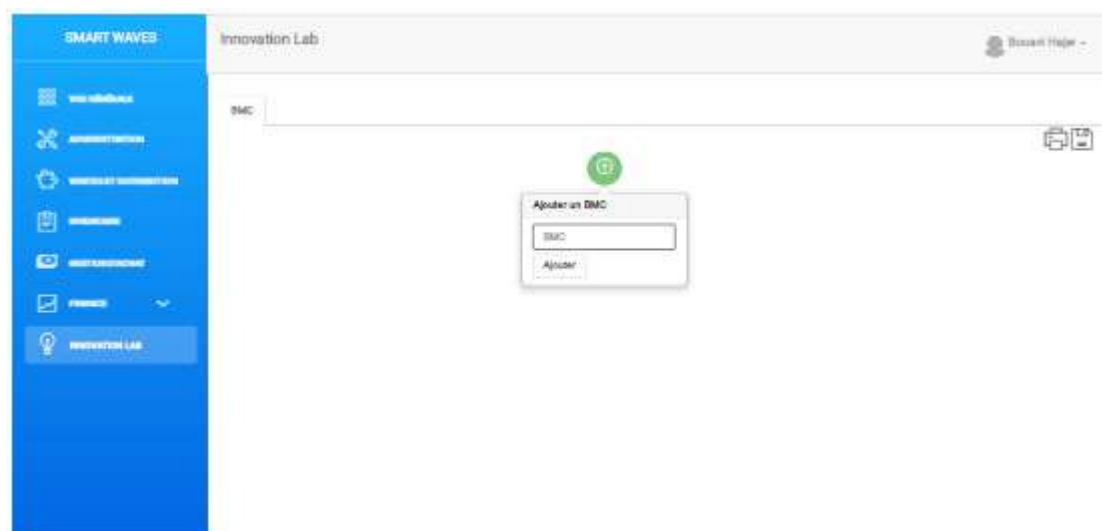


Figure 4.1: Interface d'InnovationLab : phase d'ajout d'un BMC

Cette interface présente les BMCs réalisés et le bouton qui nous permet d'ajouter une nouvelle interface pour créer un nouveau modèle BMC. Aussi les options d'impression le BMC ou Convertir le BMC en format pdf.

Dans cette deuxième interface on peut ajouter , modifier ou supprimer les partenaires, activités et ressources clés, des propositions des valeurs, des relations clients, des canaux de distribution, des segmentations clients, des coûts et des revenus de chaque BMC.



Figure 4.2: Interface complète d’InnovationLab

## 4.2 Déploiement de l’application

### 2.1 Configuration des noms de domaines

Pour l’hébergement des services que nous avons implémentés nous utiliserons la plateforme « OVH.com ». Ci-dessous la liste des noms de domaines hébergés.

Nom de domaine	Description
Api.Smartwavesis.com	Sous domaine spécifique à l’Api REST.
Demo.smartwavesis.com	Sous domaine spécifique à l’application web.
Mail.smartwavesis.com	Sous domaine spécifique au serveur d’email implémenté.

Figure 4.3: Noms de domaine implémentés

Pour l’implémentation future de l’application, chaque client aura son propre sous domaine pour accéder à l’application web.

## 2.2 Configuration SSH

SSH ou SecureShell est un protocole de communication sécurisée qui permet d'établir une communication chiffrée entre deux hôtes distants. Pour des raisons de sécurité l'authentification s'établira en utilisant la cryptographie asymétrique sans avoir à utiliser une clé secrète, ceci ce fait par le partage de la clé publique sur le système ou on souhaite se connecter et crypter par la suite la communication avec la clé privée.

Le protocole SFTP (SSH File Transfer Protocol) sera utilisé par la suite en dessus de SSH pour échanger des fichiers de manière sécurisé. Ceci a été principalement utilisé pour déployer le code sur le serveur.

```
1 ##### /etc/ssh/sshd_config
2
3 Port xxxx
4 AllowUsers user
5 PermitRootLogin no
6 PubkeyAuthentication yes
7 AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2
8 PasswordAuthentication no
9 ChallengeResponseAuthentication no
10 UsePAM yes
11 X11Forwarding yes
12 PrintMotd no
13 AcceptEnv LANG LC_*
14 Subsystem sftp /usr/lib/openssh/sftp-server
```

Figure 4.4: Fichier de config SSH

## 2.3 Configuration HTTPS

Le protocole HTTPS est la combinaison entre HTTP et un chiffrement TLS, ceci étant primordiale afin de garantir la confidentialité et l'intégrité des paquets échangés mais aussi il permet de valider l'authenticité du serveur avec un certificat d'authenticité. L'implémentation de ce protocole est essentiel vu l'aspect SaaS de notre application [6] .

La certification TLS pour notre application sera obtenue à travers l'autorité de certification « Lets Encrypt », lancée en 2015 elle permet de fournir des certificats gratuits selon la norme X.509.

Le processus d'obtention et de renouvellement de certificat implémenté par « Lets Encrypt » est totalement automatisé facilitant ainsi grandement la configuration et permettant de se passer du processus usuel qui est la création manuelle, la validation et l'installation des certificats.

Le processus de validation de notre domaine est comme suit, l'agent (le serveur web) aura à résoudre un challenge émis par l'autorité de certification pour vérifier que l'agent possède bien le nom de domaine en question, le challenge peut soit être un DNS Record ou mettre une ressource spécifique dans un URI. Suite à ceci, l'agent crypte le challenge avec sa clé privée, l'autorité aura par la suite le rôle de décrypter et de valider le challenge.

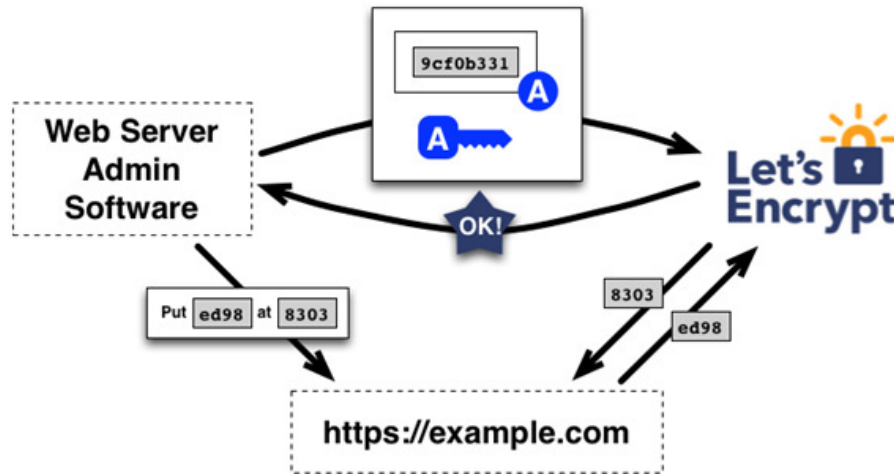


Figure 4.5: Processus de validation de l'agent [7]

La figure ci-dessus explicite les étapes nécessaires, dans cet exemple le challenge en question est de mettre « ed98 » dans l'URN 8303. Par la suite l'agent aura à crypter « 9cf0b331 » avec sa clé privée et l'envoyer à l'autorité.

## 2.4 Configuration du serveur de messagerie

Pour l'implémentation sur serveur de la messagerie nous utiliserons l'outil PostFix qui est Open Source et facilement configurable. Nous utiliserons aussi l'outil OpenDKIM pour valider l'authenticité des Email envoyés à travers un cryptage à clé publique, les Email lors de leurs envoi seront cryptés puis décryptés par le destinataire en récupérant la clé publique de la source à travers son adresse DNS [8].

La figure ci-dessus représente la configuration implémentée par OpenDKIM.

```

1 ##### /etc/opendkim.conf
2
3 OversignHeaders      From
4 TrustAnchorFile      /usr/share/dns/root.key
5 PidFile              /var/run/opendkim/opendkim.pid
6
7 Syslog               yes
8 SyslogSuccess        Yes
9 LogWhy               yes
10 UMask                002
11 Canonicalization     relaxed/simple
12 Mode                 sv
13 SubDomains           no
14 #ADSPAction           continue
15 AutoRestart          Yes
16 AutoRestartRate      10/1h
17 Background           yes
18 DNSTimeout           5
19 SignatureAlgorithm    rsa-sha256
20 UserID                opendkim:opendkim
21 Socket                inet:12345@localhost
22 KeyTable              refile:/etc/opendkim/KeyTable
23 SigningTable          refile:/etc/opendkim/SigningTable
24 ExternalIgnoreList    refile:/etc/opendkim/TrustedHosts
25 InternalHosts         refile:/etc/opendkim/TrustedHosts
26

```

Figure 4.6: Fichier de config OpenDKIM

## 2.5 Configuration du pare-feu applicative Fail2Ban

Pour assurer la sécurité de l'application contre les intrusions nous utiliserons l'outil Fail2Ban qui est un IPS qui permet de détecter les attaques communes tel que les attaques par force brute, un TCP SYN flood, un DoS, ou quelconque Scan vertical du réseau.

Fail2ban opère en surveillant les fichiers log tel que « auth.log » et « access.log », il peut bannir un IP spécifique pour une durée déterminé si un comportement suspect est détecté en appliquant des modifications sur la configuration de NetFilter ou TCP Wrapper.

```

1 # /etc/fail2ban/jail.local
2
3 [DEFAULT]
4 # liste des IP qui ne seront jamais bannies (ah, un garde-fou contre les faux-positifs !)
5 ignoreip = 127.0.0.1, 192.168.1.50
6 # durée d'un ban, en secondes (soit 10mn ici)
7 bantime = 600
8 # Une IP est bannie si "maxretry" actions suspectes sont trouvées durant les "findtime" dernières
  secondes
9 findtime = 600
10 maxretry = 3
11
12 # L'IP est bannie dès le premier échec d'authentification SSH
13 [ssh]
14 enabled = true
15 maxretry = 1
16
17 # L'IP est banie après 3 échec d'authentification Apache
18 [apache]
19 enabled = true
20 maxretry = 3

```

Figure 4.7: Fichier de config principale de Fail2Ban

Les figures ci-dessus représentent les règles de filtrage appliqué à l'URI d'authentification de notre application ainsi que la configuration par défaut de Fail2Ban.

```
1 # /etc/fail2ban/filter.d/symfony-auth.conf
2 [Definition]
3 failregex = ^<HOST> .* "POST /login_check HTTP/1.1" 302
4 ignoreregex =
```

Figure 4.8: Règle de filtrage Regex pour l'authentification

### Conclusion

Dans ce chapitre nous avons présenté les technologies utilisées pour déployer notre application ainsi que la configuration. Nous clôturerons ce travail par une conclusion générale qui récapitulera tout le travail réalisé.

## Conclusion et perspectives

Les progiciels de gestion intégrés ont pu balayer ce monde. Ils sont utilisés pour désigner une catégorie des systèmes d'informations d'entreprise, qui comportent un ensemble de modules fonctionnels, couvrant les achats, les stocks, la production, les ventes, la distribution, ainsi que les salaires...

Aujourd'hui, les ERP doivent apporter aux entreprises beaucoup de souplesse, avec des logiciels à l'architecture plus ouverte, facilitant l'intégration de processus externes et de nouvelles applications, répondant aux besoins des utilisateurs. Les utilisateurs s'attendent à un ERP offrant un outil de gestion économique plus spécifiquement un business model contenant les détails du vente, de l'achat et de la stratégie.

Pour mener à bien notre projet, nous avons commencé dans un premier temps par une étude des solutions existantes du marché voire ses limites pour analyser les besoins fonctionnels et non fonctionnels. Afin d'aboutir relever les exigences définies dans la première phase, nous avons divisé notre travail sur deux sprints. Un premier sprint qui présente l'implémentation de l'interface du module InnovationLab. Durant ce sprint nous avons fait face à des difficultés au niveau du développement avec des outils avancés à savoir le Redux qui assure la gestion de la partie données et controller. Un deuxième sprint qui englobe la phase de la consommation de l'API, étape primordiale pour suralimenter notre solution React avec des données.

Nous estimons avoir réalisé notre objectif à savoir l'amélioration des fonctionnalités de SmartWaves et l'ajout du module InnovationLab qui répond aux besoins spécifiques des clients.

Bien que notre application répond efficacement aux exigences et aux besoins définis précédemment, SmartWaves peut être amélioré grâce à d'autres perspectives et fonctionnalités telles que :

- La digitalisation des activités et événements culturels en mettant en place d'un système qui prend en considération le besoin métiers du domaine artistique et culturel (Cinéma, festivals de musique, artisanats etc.) et lui offre des services dont il a besoin.
- Création d'un système connecté aux déclarations fiscales et à la documentation financière relative à l'environnement culturel.

# Bibliographie

[1] Business model de l'entreprise : de quoi s'agit-il et comment le construire ? Consulté le 06 08, 2020, sur <https://ccibusiness-builder..fr/guide-creation/le-business-model/>

[2] Shedrack Akintayo. Consuming REST APIs In React With Fetch And Axios. medium.Consulté le 11 08, 2020 ,sur <https://www.smashingmagazine.com/2020/06/rest-api-react-fetch-axios/>

[3] Alex Bachuk. Redux An Introduction.Consulté le 11 08, 2020 Smashing Magazine. ,sur <https://www.smashingmagazine.com/2016/06/an-introduction-to-redux/>

[4] Fowler, M. (s.d.). Richardson Maturity Model. Consulté le 15 08, 2020, sur <https://martinfowler.com/articles/richardsonMaturityModel.html>

[5] Linkvalue. React et Architecture Flux. medium. Consulté le 21 08, 2020 ,sur <https://blog.link-value.fr/react-et-architecture-flux-d763c5e0ecf8>

[6] Hyper Text Transfer Protocol Secure.(s.d.). Consulté le 25 08, 2020, sur <https://fr.wikipedia.org/wiki>

[7] Lets Encrypt. (s.d.). How It Works. Consulté le 29 08, 2020, sur <https://letsencrypt.org/how-it-works/>

[8] Archlinux. (s.d.). OpenDKM. Consulté le 06 09, 2020, sur <https://wiki.archlinux.org/index.php/OpenDKIM>

[9] Aymen ZAOUALI. Pourquoi utiliser Redux? medium.Consulté le 05 09, 2020 ,sur <https://medium.com/@aymenzaouali/pourquoi-utiliser-redux-f38b55430d13>