



Ministère de l'Enseignement
Supérieur et de la Recherche
Scientifique

ANNÉE UNIVERSITAIRE

2018/2019

EPI -Polytec
ÉCOLE POLYTECHNIQUE

ÉCOLE PRIVÉE INTERNATIONALE
D'INGÉNIEURS

RAPPORT DE PROJET DE FIN D'ANNÉE

Spécialité

GENIE INFORMATIQUE

Intitulé

**Conception et Réalisation d'une Solution
Modulaire sous Odoo pour un centre de beauté**

Lieu du stage

Smart2do

Réalisé par

Med Oussama BEN JAZIA

Asma AZAIEZ

Encadré par

M Houssein CHTIOUI

M Nawfel BEN YAHIA

Dédicace

A nos parents

Grâce à leurs tendres encouragements et leurs grands sacrifices, ils ont pu créer le climat affectueux et propice à la poursuite de nos études. Aucune dédicace ne pourrait exprimer notre gratitude, notre considération et nos profonds sentiments envers eux. Nous prions le bon Dieu de les bénir, de veiller sur eux, en espérant qu'ils seront toujours fiers de nous.

A tous nos professeurs

Leur générosité et leur soutien nous obligent de leurs témoigner

Notre profond respect et notre loyale considération.

A tous nos amis et nos collègues

Ils vont trouver ici le témoignage d'une fidélité et d'une amitié infinie

Med Oussama Ben Jazia & Asma Azaiez

Remerciements

Avant de commencer la présentation de ce travail, on profite de l'occasion pour remercier tous les membres du jury et toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce projet de fin d'année.

On exprime nos profondes gratitudee et respectueuse

Reconnaissance à notre encadrant :

M. Houssemeddine Chtioui

Pour sa bonne volonté d'accepter de nous encadrer, pour tout le temps qu'il nous a octroyé, pour tous les conseils qu'il nous a prodigué et pour la confiance qu'il nous a accordé tout au long de notre stage.

On tient à remercier d'avantage notre encadreur professionnel

M. Naoufel BEN YAHYA

Et toute l'équipe de Smart2 Do, on cite :

Rihab CHATBOURI & Amina TOUAY

Pour le temps qu'ils nous ont consacré tout au long de cette période, sachant répondre à toutes nos Interrogations.

Et enfin on tient aussi à remercier toute autre personne ayant contribué d'une façon ou d'une autre, au bon déroulement de notre projet de fin d'année.

Sommaire

INTRODUCTION GENERALE.....	1
CHAPITRE 1 : ANALYSE DES BESOINS ET ETUDE PREALABLE.....	2
1. INTRODUCTION :	2
2. PRESENTATION DE L'ORGANISME D'ACCUEIL (SIEGE DE TRAVAIL) :	2
2.1 L'offre / Les produits :	3
2.2 Domaine d'interventions :	3
3. CONTEXTE DU PROJET :	4
3.1 Problématique :	4
3.2 Étude des solutions proposées sur le marché :	4
3.3 Critique de l'existant :	5
3.4 Solutions Proposées	6
4. SPECIFICATION DES BESOINS :	6
4.1 Besoins Fonctionnels :	6
4.2 Besoins non fonctionnels :	7
5. RESULTATS ATTENDUS :	8
6. LES DIAGRAMMES DES CAS D'UTILISATION :	8
6.1 Identification des acteurs :	9
6.2 Identification des cas d'utilisation :	9
6.3 Diagramme de cas d'utilisation général :	9
6.4 Description de cas d'utilisation général :	10
6.5 Diagramme de cas d'utilisation « Gestion Clients » :	11
6.6 Diagramme de cas d'utilisation « Gestions des utilisateurs » :	11
6.7 Diagramme de cas d'utilisation « Gestions des Commandes » :	12
6.8 Diagramme de cas d'utilisation « Gestions des Services » :	12
6.9 Diagramme de cas d'utilisation « Gestions des Chaises » :	13
6.10 Diagramme de cas d'utilisation « Gestion des RDV » :	13
6.11 Diagramme de cas d'utilisation « Gestions de Compatibilité » :	14
6.12 Diagramme de cas d'utilisation « Gestions de Configuration » :	14
7. CONCLUSION :	14
CHAPITRE II : ETUDE CONCEPTUELLE DU SYSTEME	15
1. INTRODUCTION :	15
2. PROCESSUS UNIFIE (UP) :	15
3. LES PRINCIPES FONDAMENTAUX DU PROCESSUS UNIFIE(UP) :	15
4. CONCEPTION DETAILLEE :	16
4.1 Vue statique :	17

4.1.1	Diagramme des Cas d'Utilisation :	17
4.1.2	Diagramme de Classe :	17
4.2	<i>Vue dynamique :</i>	18
4.2.1	Diagramme de séquence :	18
5.	CONCLUSION :	21
CHAPITRE III: REALISATION.....		22
1.	INTRODUCTION	22
2.	ENVIRONNEMENT MATERIEL :	22
3.	ENVIRONNEMENT DE TRAVAIL	22
3.1	<i>Environnement de conception UML :</i>	22
3.2	<i>Environnement de développement :</i>	23
3.2.1	Outil utilisé :	23
3.2.2	Langage utilisé:	24
4.	ODOO : [4].....	25
4.1	<i>Description ODOO :</i>	25
4.2	<i>Avantages d'ODOO :</i>	25
4.3	<i>Architecture Odoo :</i>	26
4.4	<i>Les principaux modules Odoo :</i>	26
5.	SCENARIO COMPLET D'UTILISATION DE LA SOLUTION	27
6.	CONCLUSION.....	29
7.	CONCLUSION GENERALE ET PERSPECTIVES	30
8.	BIBLIOGRAPHIE	31

Liste des figures

FIGURE 1: LOGO DE SOCIETE SMART2DO	2
FIGURE 2: DOMAINE D'INTERVENTIONS DE SMART2DO TECHNOLOGIES.....	3
FIGURE 3: LOGO BEAUTY SPA MANAGEMENT	5
FIGURE 4: LOGO SPA SALON MANAGEMENT.....	5
FIGURE 5: DIAGRAMME CAS D'UTILISATION GENERAL.....	10
FIGURE 6: DIAGRAMME DE CAS D'UTILISATION « S'AUTHTENTIFIER »	10
FIGURE 7: DIAGRAMME DE CAS D'UTILISATION DE « GESTION DES CLIENTS »	11
FIGURE 8:DIAGRAMME DE CAS D'UTILISATION « GESTION DES UTILISATEURS»	12
FIGURE 9: DIAGRAMME DE CAS D'UTILISATION DE GESTION DE COMMANDE.....	12
FIGURE 10: DIAGRAMME DE CAS D'UTILISATION DE « GESTION DES SERVICES »	12
FIGURE 11: DIAGRAMME DE CAS D'UTILISATION DE « GESTION DES CHAISES »	13
FIGURE 12: DIAGRAMME DE CAS D'UTILISATION DE « GESTION DES RDV»	13
FIGURE 13: DIAGRAMME DE CAS D'UTILISATION DE GESTION DE COMPTABILITE »	14
FIGURE 14: DIAGRAMME DE CAS D'UTILISATION « GESTION DE CONFIGURATION ».....	14
FIGURE 15:DIAGRAMME DE CLASSE.....	17
FIGURE 16: DIAGRAMME DE SEQUENCE SYSTEME « AUTHENTIFICATION »	18
FIGURE 17: DIAGRAMME DE SEQUENCE SYSTEME « GESTION DES CLIENTS »	19
FIGURE 18: DIAGRAMME DE SEQUENCE SYSTEME DE GESTION RDV	20
FIGURE 19:LOGO STARUML.....	23
FIGURE 20: LOGO DE VISUAL PARADIGM.....	23
FIGURE 21: LOGO ATOM.....	23
FIGURE 22: LOGO PYTHON	24
FIGURE 23: LOGO XML	24
FIGURE 24: LOGO UML.....	25
FIGURE 25: LOGO ODOO	25
FIGURE 26: ARCHITECTURE ODOO	26
FIGURE 27: MODULES D'ODOO.....	27
FIGURE 28: LA LISTE DES RDVS CREES	28
FIGURE 29: LA CREATION D'UN NOUVEAU RDV	28
FIGURE 30: LA CREATION D'UNE CHAISE	28
FIGURE 31: LA LISTE DES COMMANDES.....	28
FIGURE 32: L'INTERFACE DE GESTION DE FACTURE	29

Liste des tables

TABLE 1: PLANNING DE STAGE	3
TABLE 2: COMPARATIF DES SOLUTIONS	5
TABLE 3 DESCRIPTION DU : DIAGRAMME DE CAS D'UTILISATION « S'AUTHTENTIFIER »	11

Introduction générale

Durant ces trente dernières années, l'informatique a subi des bouleversements considérables. Les avancées technologiques du traitement de l'information ont eu des conséquences capitales sur le rôle de l'outil informatique.

Si les premières applications ont permis d'automatiser les activités opérationnelles des organisations (gestion de production, gestion commerciale et financière, ressources humaines), aujourd'hui les systèmes d'information prennent en charges des niveaux de gestion de plus en plus stratégiques.

L'ERP est basé sur la construction des applications informatiques de l'entreprise (comptabilité, gestion de stocks, etc.) sous forme de modules indépendants. Ces modules partagent une base de données commune, permettant la communication de données entre les applications.

Dans le cadre de notre travail, nous intéressons à concevoir une application ERP qui permet la gestion d'un centre de beauté au sein de l'entreprise Smart2do.

Au premier chapitre, nous présenterons le cahier des charges avec les analyses nécessaires argumenté par des solutions orientées conceptuelles, organisationnelles et techniques.

Dans le deuxième chapitre, nous parlerons de notre réflexion conceptuelle autour de ce projet, dans une partie dédiée à la modélisation, avec les diagrammes nécessaires de chaque processus.

Dans le troisième chapitre, « la réalisation », nous décrivons l'environnement du développement matériel et logiciel et nous présentons les différentes fonctionnalités de notre application à travers des captures d'écran.

Enfin, une conclusion générale comportant l'ensemble d'apports et limites de l'ensemble du travail proposé, ainsi que les perspectives éventuelles dans l'optique d'améliorer ce qui est réalisé.

Chapitre 1 : Analyse des besoins et Etude Préalable

1. Introduction :

La réalisation de cette application nécessite une étude approfondie sur certaines notions qui touchent non seulement, au cadre général du projet, mais aussi à sa réalisation. Cette étude vise en générale à faciliter le processus de conception et de développement des applications informatiques. Elle englobe un ensemble des moyens, des méthodes et des formalismes, mis à la disposition du concepteur pour mieux analyser, comprendre, critiquer et améliorer le système actuel du travail, dans le but de l'informatiser totalement ou partiellement.

Ce chapitre sera consacré, en premier lieu, à la présentation d l'organisme d'accueil.

2. Présentation de l'organisme d'accueil (Siège de travail) :

La société Smart2Do Technologies est considérée comme nouvellement créée, elle est fondée en 19 avril 2016 à SAHLOUL Sousse avec un capital social de 31600 DT et elle est constituée sous la forme juridique SUARL.

La société Smart2Do Technologies propose une activité de services. En effet, elle est spécialisée dans l'édition des logiciels, la conception des services, les solutions innovantes et les solutions ERP destinées aux PME.

Cette jeune société a pour vocation d'accompagner les sociétés dans le développement de leurs activités à travers :

- ✓ L'accès aux technologies les plus récentes d'outils de gestion et l'aide à la décision.
- ✓ Le développement des solutions logicielles innovantes, application Web et Mobile.
- ✓ La figure suivante présente le logo de la société:



Figure 1: Logo de société Smart2Do

2.1 L'offre / Les produits :

Odoo/Open ERP, une solution web complète pour la gestion des différents départements de l'entreprise (achat, vente, comptabilité, gestion de la fabrication, gestion de stock...).

Grâce à une expertise métier et technique nous offrons à nos clients une solution adaptée à leurs besoins dans différents secteurs d'activités (Commercial, Industriel, Service...), nous garantissons également une bonne communication et un service de support de qualité pour toute réclamation ou demandes.

2.2 Domaine d'interventions :

La figure suivante représente le domaine d'interventions de l'organisme d'accueil:

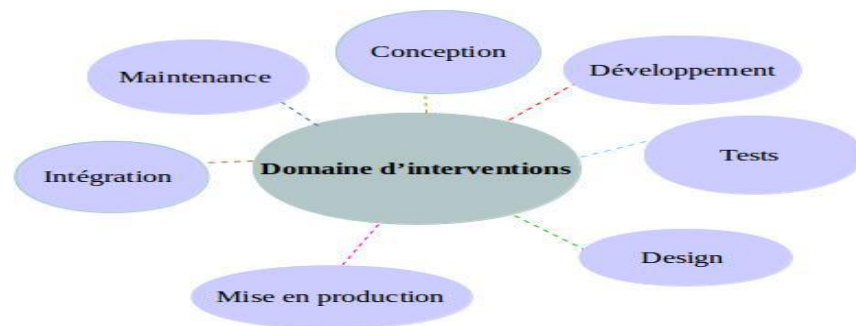


Figure 2: Domaine d'interventions de Smart2do Technologies

2.1 Le Stage :

Phase/Semaine	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	
Etude de projet												
Formation												
Conception												
Rapport V1												
Integration module Odoo Classique												
Development des modules Odoo												
Rapport V2												
Déploiement sur un serveur												
Test et validation												
Rapport Version Finale												

Table 1: Planning de stage

3. Contexte du projet :

3.1 Problématique :

Un centre de beauté représente une des organisations ayant des ressources et des activités dont la gestion nécessite une application informatique fiable, facile à maintenir et sécurisé pour bien gérer ses activités.

Parmi les problèmes à l'intérieur de ce centre, on cite :

- Problème d'archivage: mauvaise gestion des données sous forme papier accentue la perte de temps lors de la recherche et cause parfois de la perte de plusieurs fiches des clients.
- Problème de communication: le partage d'un dossier du client entre différents acteurs et l'échange des informations est parfois très difficile.
- Problème de planification et suivie: les propriétaires ou l'agent de réception du centre ne disposent pas de compte pour le suivi de leurs plannings et la supervision de leurs patients.

3.2 Étude des solutions proposées sur le marché :

Avant de spécifier les besoins il est nécessaire de réaliser une étude de l'existant, c'est une étape essentielle de tout projet. Il s'agit d'une étude permettant de comprendre la problématique du projet et de détecter les avantages et inconvénients des solutions proposées actuellement sur le marché afin d'en profiter pour la réalisation de notre projet.

Nous avons mené une étude sur deux applications qui s'inscrivent dans le même cadre que notre projet :

✓ Beauty SPA Management [1]

Salon SPA point of Sale est un module Odoo développé par Serpent Consulting Services Pvt. Ltd ce module pour objectif de donner aux clients une nouvelle expérience en matière d'utilisation des services du centre de spa. Ce module aide le client à traiter sa réservation en ligne pour des services de spa. Le module est intégré à Odoo Accounting et Website, offrant une pléthore de fonctions.

L'un des principaux avantages du système de gestion du spa est qu'il offre à vos clients les informations de réservation par courrier électronique. Le client est équipé du statut de sa réservation, améliorant ainsi l'expérience et la construction de marque habituelles de votre entreprise.



Figure 3: Logo Beauty SPA management

✓ SPA Salon Management :

Le module SPA Salon développée de odoo développée par Ltd permet la Réservation et ordonnancement facile Configuration et gestion du personnel, esthéticienne et attribution des chaires aux clients ainsi l'analyse des revenus.



Figure 4: logo SPA salon Management

3.3 Critique de l'existant :

✓ Comparatif des solutions

Nous avons présenté une comparaison entre les différentes solutions dans le tableau ci-dessous :

	Beauty SPA Management	SPA Salon Management
Payante	Oui	Oui
Version	V11, V12	V10, V11, V12
Avantages	<ul style="list-style-type: none"> Solutions bien déterminées pour la gestion des centres de SPA Bien organisées et faciles à utilisées 	
Inconvénients	Très chers	

Table 2: Comparatif des solutions

Il existe un nombre d'applications pour aider la gestion des centres de SPA. Mais ces applications soit coûtent très cher, soit ne réussissent pas à proposer toutes les fonctionnalités demandées.

3.4 Solutions Proposées

Notre projet a pour objectif de réaliser une application informatique :

- Facile à intégrer dans un environnement de travail d'un centre de beauté
- Automatiser les tâches qui se traitent manuellement.
- Diminuer la complexité des traitements et la perte de temps, garantir l'intégrité de l'information lors des recherches sur les archives
- La disponibilité des données des clients à n'importe quel moment d'une manière plus facile et sécurisé
- Préparer des propres comptes utilisateurs pour suivre leurs notifications

4. Spécification des besoins :

L'expression des besoins sert à traduire les besoins du client en termes d'exigences et de contraintes techniques. Ainsi, dans cette partie nous présenterons donc l'ensemble des besoins fonctionnels et non fonctionnels du projet à développer.

4.1 Besoins Fonctionnels :

Les besoins fonctionnels ou besoins métiers représentent les actions que le système doit exécuter, il ne devient opérationnel que s'il les satisfait. Notre application doit couvrir principalement les besoins fonctionnels suivants :

✓ Gérer les utilisateurs :

Les utilisateurs de notre application qui seront repartis dur des sessions sont :

- Le secrétaire : gère les opérations de Comptabilité : la facturation et le payement et la gestion des commandes.
- La réceptionniste: faires les plannings, les appels, création des nouveaux clients, faire les réservations des chaises, gérer les prestations
- Le Propriétaire : c'est l'Administrateur, faire un contrôle sur le système et géré la configuration globale du système

✓ **Gérer la réservation des biens :**

- Réservations des chaises

✓ **Gestion de comptabilité :**

Parmi les fonctionnalités de l'application :

Il est possible de gérer les encaissements les opérations de paiement et les factures par la secrétaire

Après avoir reçu le soin demandé, le client passe à la caisse pour payer, il peut choisir le mode du paiement (soit espèce ou par chèque, ou carte bancaire) l'application est capable de délivrer une facture correspondant aux services effectués dans le centre

✓ **Gestion RDV :**

Notre application permet à l'utilisateur de gérer les rendez-vous selon la disponibilité du calendrier par l'agent de la réception.

✓ **Gestion des Reporting :**

L'application donne l'opportunité aux utilisateurs de faire le « Reporting » c'est-à-dire faire un rapport mensuel ou annuel sous forme de courbes de tous les dépenses et les recettes.

4.2 Besoins non fonctionnels :

Les besoins non fonctionnels présentent des exigences qui ne concernent pas spécifiquement le comportement du système mais plutôt identifient des contraintes internes et externes du système. Les principaux besoins non fonctionnels devant être pris en charge par notre application se résument par les points suivants :

✓ **Les Sessions :**

L'application n'offre la possibilité d'accéder à l'ensemble de ses fonctionnalités que via un système de session, à travers lequel chaque utilisateur doit obligatoirement entrer à l'application par son propre session qui contient ses fonctionnalités.

✓ **Fiabilité :**

Le système doit réduire les risques d'erreurs et garantir la pertinence des informations.

✓ **Extensibilité :**

Le code doit être clair pour permettre de futures évolutions ou améliorations.

✓ **Intégrité :**

Garantir la cohérence des données à chaque mise à jour aussi bien d'insertion, de modification et ou de suppression.

✓ **Ergonomie :**

L'application offre une interface conviviale et facile à utiliser.

5. Résultats attendus :

Que l'application se déroule convenablement, de conserver ces fonctionnalités dans l'application, et d'améliorer s'il est possible les performances du système ainsi que les bases des données existantes.

6. Les diagrammes des cas d'utilisation :

Un cas d'utilisation définit une manière d'utiliser le système et permet d'en décrire les exigences fonctionnelles.

D'après Bittner et Spence, «Un cas d'utilisation défini simplement, permet de décrire une séquence d'événements qui, pris tous ensemble, définissent un système faisant quelque chose d'utile» [2].

- Acteur : rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système étudié. Un acteur participe à au moins un cas d'utilisation.
- Cas d'utilisation (use case) : ensemble de séquences d'actions réalisées par le système produisant un résultat observable intéressant pour un acteur particulier. Collection de scénarios reliés par un objectif utilisateur commun.
- Association : utilisée dans ce type de diagramme pour relier les acteurs et les cas d'utilisation par une relation qui signifie simplement « participe à ».
- Inclusion : le cas d'utilisation de base en incorpore explicitement un autre, de façon obligatoire, à un endroit spécifié dans ses enchaînements.
- Extension : le cas d'utilisation de base en incorpore implicitement un autre, de façon optionnelle, à un endroit spécifié indirectement dans celui qui procède à l'extension
- Généralisation : les cas d'utilisation descendants héritent de la description de leur parent commun. Chacun d'entre eux peut néanmoins comprendre des relations spécifiques supplémentaires avec d'autres acteurs ou cas d'utilisation

6.1 Identification des acteurs :

Chaque acteur a des tâches spécifiques à faire selon les besoins du système, alors pour faire simplifier le diagramme de cas d'utilisation, on a divisé le diagramme selon les acteurs du système :

- Le propriétaire : Autrement dit l'administrateur, c'est la personne qui a tous les droits d'accès du système, il peut faire toutes les tâches des autres acteurs : la réceptionniste et la secrétaire, il gère davantage le contrôle et la configuration globale du système.
- La réceptionniste: C'est la personne qui a le droit de gérer les RDV, les réservations des chaises, les services, les chaises, les clients et aussi les journaux.
- La secrétaire : C'est la personne qui gère les commandes, la facturation et le Reporting.

6.2 Identification des cas d'utilisation :

Ces cas d'utilisation principaux ont été mis en évidence par l'expression de besoins préliminaire à savoir :

- Authentification
- Gestion des RDV
- Gestion de Comptabilité
- Gestion des Clients
- Gestion des Services
- Gestion des chaises
- Gestion des Utilisateurs
- Gestion de Reporting
- Gestion de Journalisation et d'archivage
- Gestion de Configuration et Contrôle

6.3 Diagramme de cas d'utilisation général :

On présente le diagramme de cas d'utilisation globale qui illustre toutes les fonctionnalités de du système :



Figure 5: Diagramme cas d'utilisation général

6.4 Description de cas d'utilisation général :

Afin de décrire les interactions entre les cas d'utilisation, nous présentons ces derniers de façon textuelle. Il s'agit donc d'associer à chaque cas d'utilisation un nom, un objectif, les acteurs qui y participent, les prés et post-conditions et des scénarios. Dans notre description textuelle, nous présentons les scénarios nominaux, exceptions avec les près et les post-conditions.

✓ Cas d'utilisation « Authentification » :

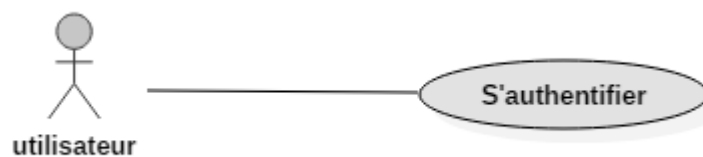


Figure 6: Diagramme de cas d'utilisation « s'authentifier »

Sommaire Identification	
Acteur	Propriétaire, Secrétaire, Réceptionniste
Résumé	Les utilisateurs se connectant au système pour accéder à leurs propres comptes
Description du scénario : Authentification	

Précondition	L'utilisateur doit avoir un compte Odoo
Scénario nominal	Saisir login et password
Scénario d'exception	<ul style="list-style-type: none"> login ou password incorrect le système renvoie un message d'erreur et l'invite à recommencer

Table 3 Description du : Diagramme de cas d'utilisation « s'authentifier »

6.5 Diagramme de cas d'utilisation « Gestion Clients » :

Après avoir s'authentifier, la réceptionniste a le droit de traiter les opérations de gestion des clients: ajouter, supprimer, modifier et consulter

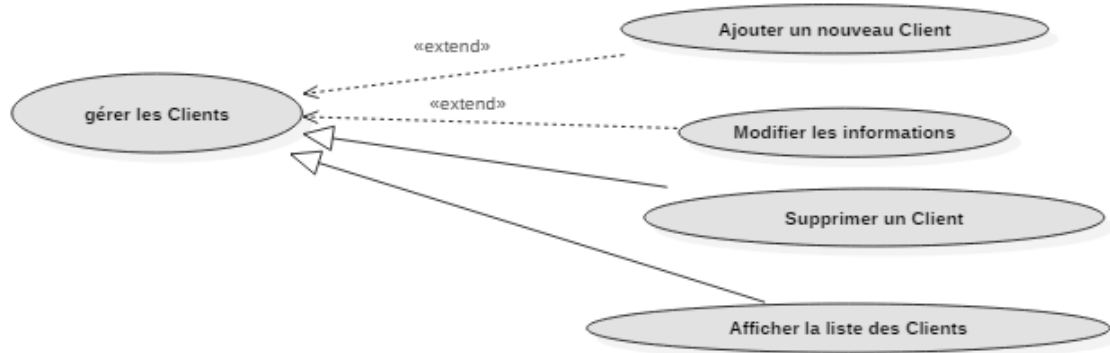


Figure 7: Diagramme de cas d'utilisation de « gestion des clients »

6.6 Diagramme de cas d'utilisation « Gestions des utilisateurs » :

L'administrateur peut effectuer la création d'un utilisateur en lui attribuant les privilèges souhaités. L'administrateur consulte la liste des utilisateurs, sélectionne un utilisateur pour le modifier ou le supprimer.

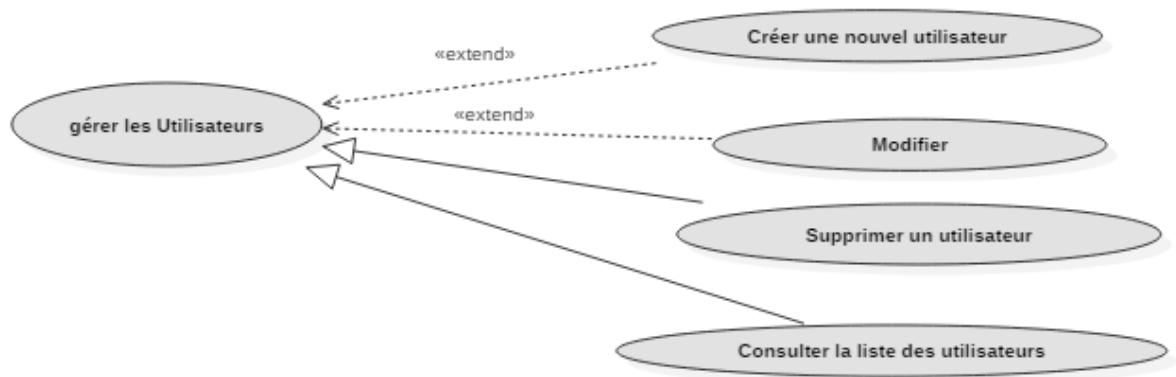


Figure 8:Diagramme de cas d'utilisation « Gestion des utilisateurs»

6.7 Diagramme de cas d'utilisation « Gestions des Commandes » :

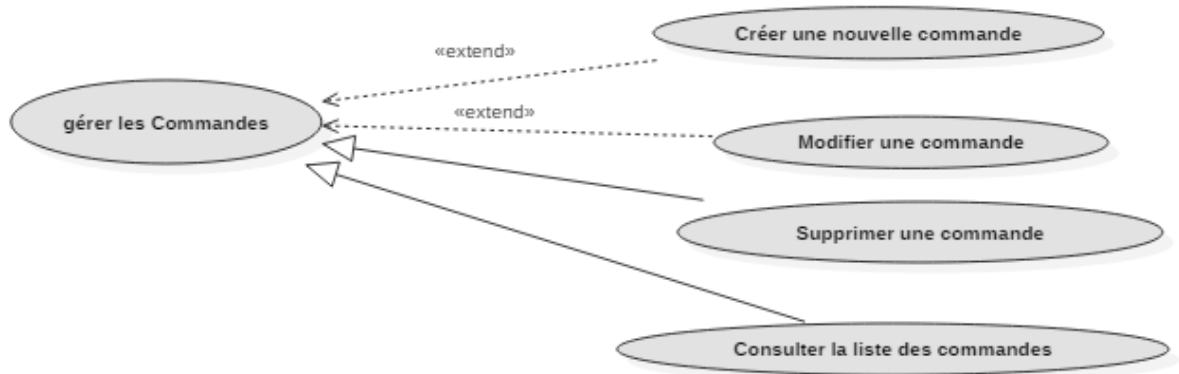


Figure 9: Diagramme de cas d'utilisation de gestion de commande

6.8 Diagramme de cas d'utilisation « Gestions des Services » :

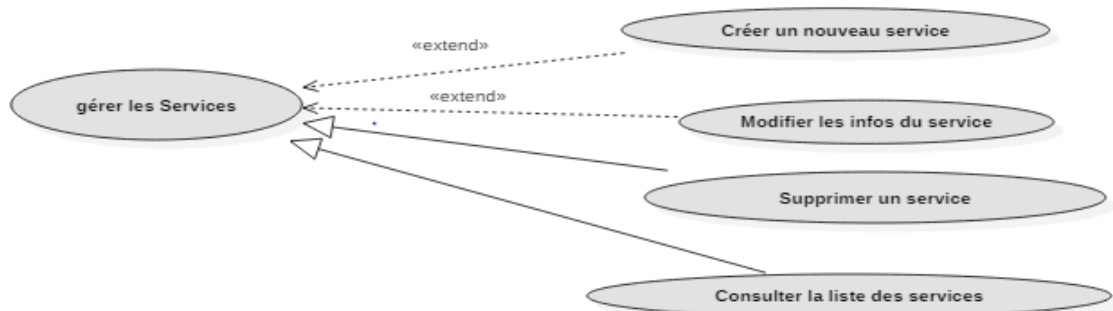


Figure 10: Diagramme de cas d'utilisation de « gestion des services »

6.9 Diagramme de cas d'utilisation « Gestions des Chaises » :

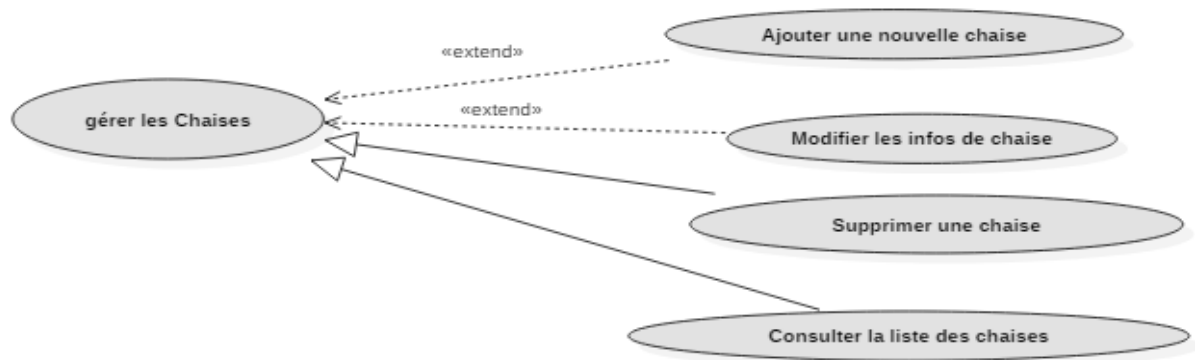


Figure 11: Diagramme de cas d'utilisation de « gestion des chaises »

6.10 Diagramme de cas d'utilisation « Gestion des RDV » :

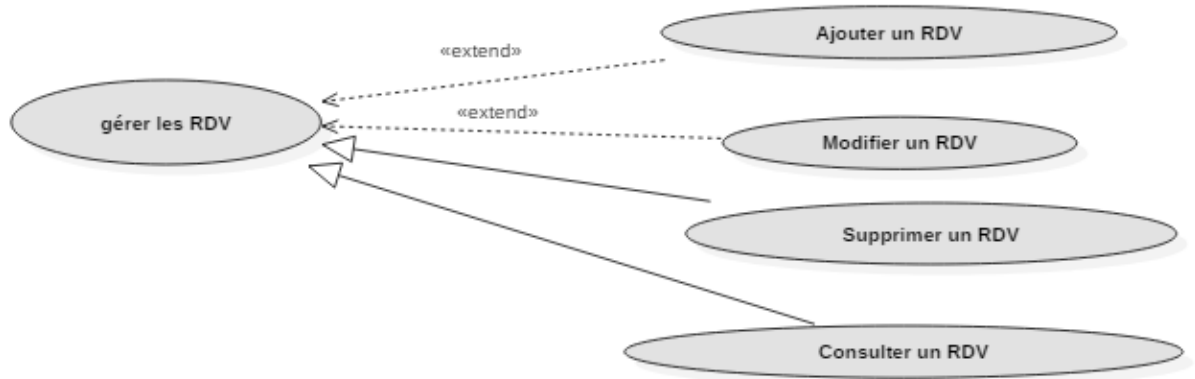


Figure 12: Diagramme de cas d'utilisation de « Gestion des RDV »

6.11 Diagramme de cas d'utilisation « Gestions de Compatibilité » :

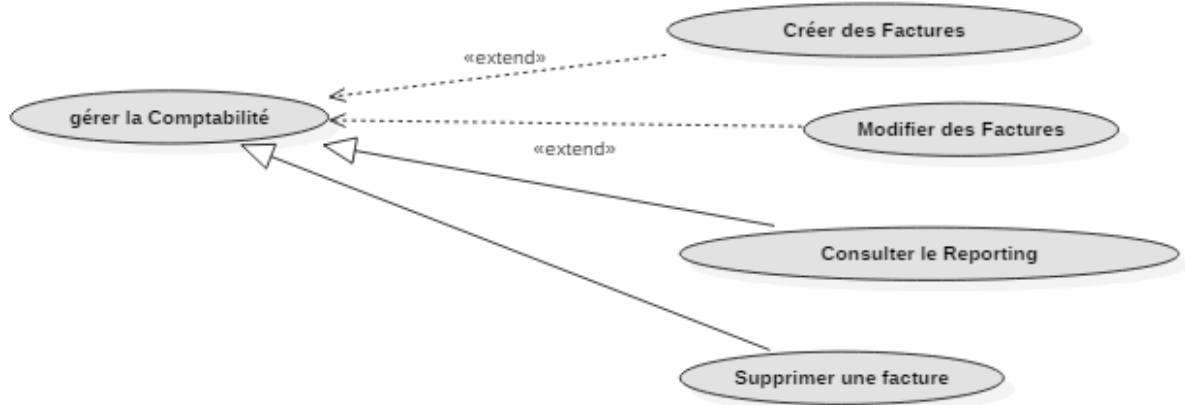


Figure 13: Diagramme de cas d'utilisation de Gestion de comptabilité »

6.12 Diagramme de cas d'utilisation « Gestions de Configuration » :

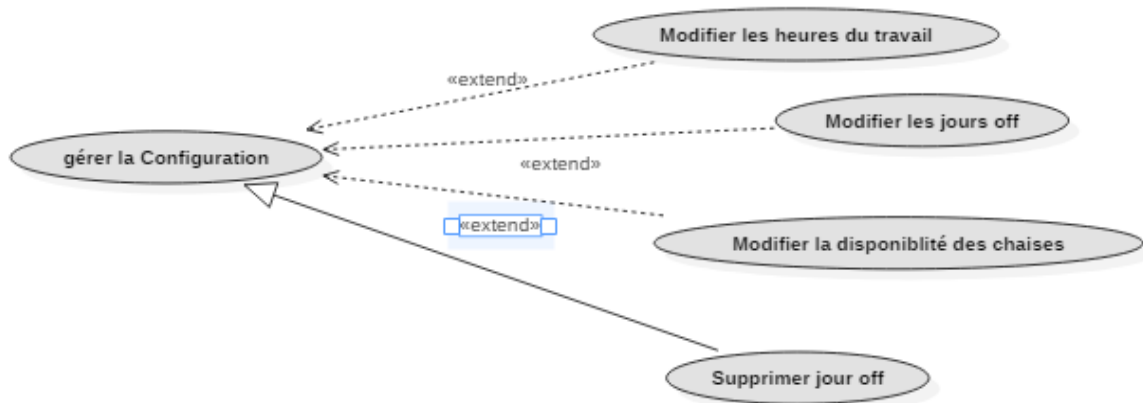


Figure 14: Diagramme de cas d'utilisation « Gestion de configuration »

7. Conclusion :

Ce chapitre a été consacré pour la présentation de cadre du ce projet, le stage qu'on a effectué, aussi la spécification des besoins fonctionnels et non fonctionnels du système résultant, ce qui correspondait aux différentes activités de la première phase du cycle de développement du notre système. Dans le chapitre suivant, nous étudierons la phase de conception.

Chapitre II : Etude conceptuelle du système

1. Introduction :

La partie conceptuelle est l'un des éléments fondamentaux de chaque cycle de réalisation et de mise en œuvre d'un produit, qui a pour objectif d'analyser et modéliser les fonctionnalités de ce dernier et définir les droits d'accès pour les acteurs. Nous élaborons dans ce chapitre l'étude conceptuelle de notre application via la modélisation UML de point de vue statique ainsi que de point de vue dynamique. Nous allons mettre l'accent sur les diagrammes du cas d'utilisation pour décrire les différentes activités apportées par l'utilisateur d'application.

Puis on procède à illustrer l'articulation des données et des traitements dans le diagramme de classe, Ensuite nous allons donner une vue détaillée à propos du déroulement de chaque activité grâce aux diagrammes des séquences.

A la fin de cette partie, on modélise la solution physique de l'application à travers l'architecture déployée dans le diagramme de déploiement.

2. Processus Unifié (UP) :

Dans le cadre de modélisation et conception de notre application on a choisi d'utiliser le langage UML.

Le développement d'UML a commencé en 1994, c'est un langage graphique de modélisation des données et des traitements, c'est une formalisation très aboutie et non–propriétaire de la modélisation objet utilisé en génie logiciel.

Dans ce projet nous allons opter d'utiliser le diagramme des cas d'utilisation, le diagramme de classes, le diagramme des séquences et le diagramme de déploiement. ^[3]

3. Les principes fondamentaux du Processus Unifié(UP) :

Le Processus Unifié est un processus de développement logiciel « itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques » :

- Itératif et incrémental : le projet est découpé en itérations de courte durée (environ 1 mois) qui aident à mieux suivre l'avancement global. À la fin de chaque

itération, une partie exécutable du système final est produite, de façon incrémentale.

- Centré sur l'architecture : tout système complexe doit être décomposé en parties modulaires afin de garantir une maintenance et une évolution facilitées. Cette architecture (fonctionnelle, logique, matérielle, etc.) doit être modélisée en UML et pas seulement documentée en texte.
- Piloté par les risques: les risques majeurs du projet doivent être identifiés au plus tôt, mais surtout levés le plus rapidement possible.
- Les mesures à prendre dans ce cadre déterminent l'ordre des itérations.
- Conduit par les cas d'utilisation: le projet est mené en tenant compte des besoins et des exigences des utilisateurs. Les cas d'utilisation du futur système sont identifiés, décrits avec précision et priorisés

4. Conception détaillée :

UML est une approche orientée objet de modélisation qui permet de modéliser un problème d'une manière standard. UML évite de se définir comme une méthodologie, comme son nom l'indique, c'est un langage « visuel » qui permet d'exprimer la compréhension d'un système :

Il comporte 9 principaux diagrammes regroupés dans deux vues différentes:

- Vue Statique (cinq diagrammes structurels) :
 - Diagramme de Cas d'utilisation.
 - Diagramme de Classes.
 - Diagramme d'Objets.
 - Diagramme de Composants.
 - Diagramme de Déploiement.
- Vue Dynamique (quatre diagrammes comportementaux)
 - Diagramme de Séquence
 - Diagramme d'états transitions.
 - Diagramme de Collaboration.
 - Diagramme d'activité

A cet effet on présente quelques diagrammes de modélisation, qu'on a jugé les plus importants pour la compréhension du fonctionnement du système.

4.1 Vue statique :

4.1.1 Diagramme des Cas d'Utilisation :

Les diagrammes de cas d'utilisation représentent toutes les interactions des utilisateurs avec le système, comme nous avons déjà les décrit dans le chapitre précédent "Analyse des besoins et étude préalable"

4.1.2 Diagramme de Classe :

Le diagramme de classes représente les classes constituant le système et les associations entre elles. Les diagrammes de classes expriment de manière générale la structure statique d'un système, en termes de classe et de relations entre ces classes. De même qu'une classe décrit un ensemble d'objets, une association décrit un ensemble de liens ; les objets sont des instances de classes et les liens sont des instances de relations.

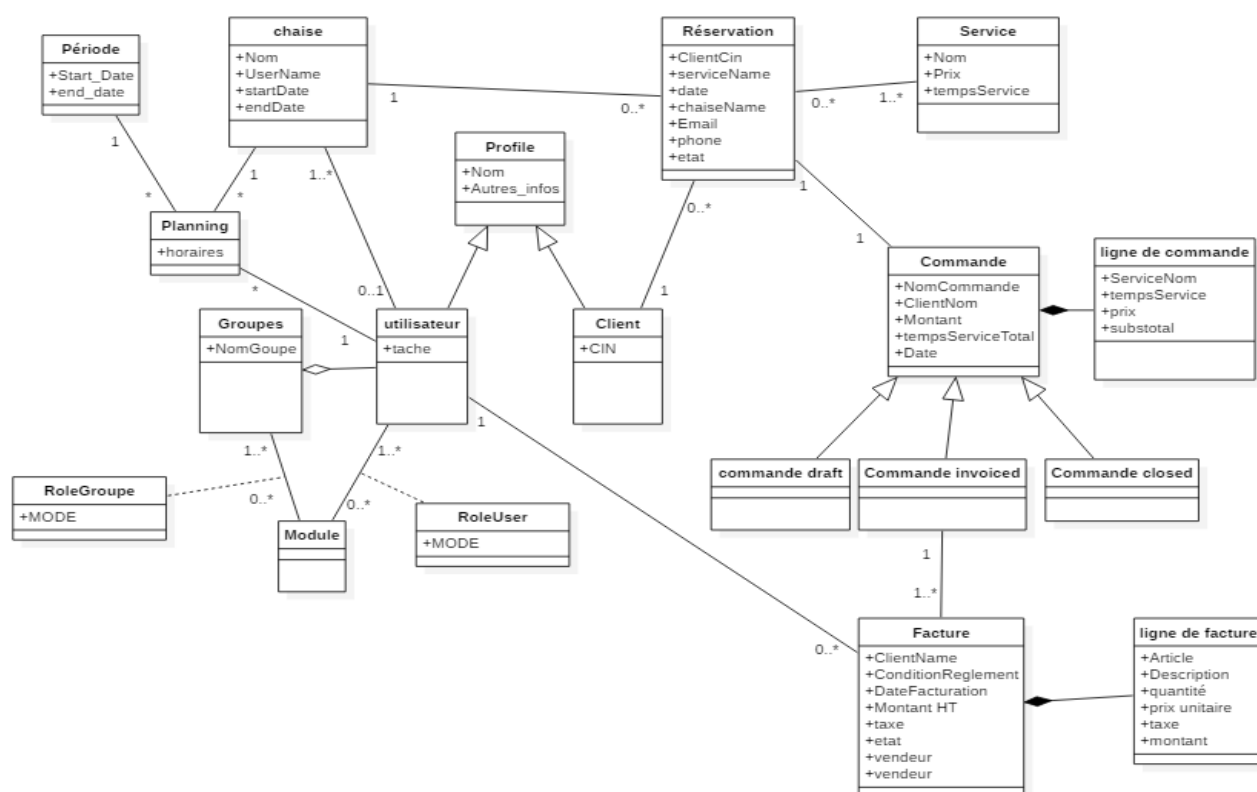


Figure 15:Diagramme de classe

D'après l'étude du système existant et des différents diagrammes de cas d'utilisation, nous avons pu dégager les principales classes illustrées dans la figure ci-dessus pour avoir une vue plus claire du système étudié. A partir de ce diagramme, on dégage les classes correspondant

dans l'application à développer. Ce diagramme de classe ne contient pas des méthodes. Ces procédures seront évoquées après le diagramme de séquence.

4.2 Vue dynamique :

4.2.1 Diagramme de séquence :

Un diagramme de séquence représente les interactions entre les objets de point de vue temporel. Il permet de détailler un diagramme de cas d'utilisation.

- Authentification :

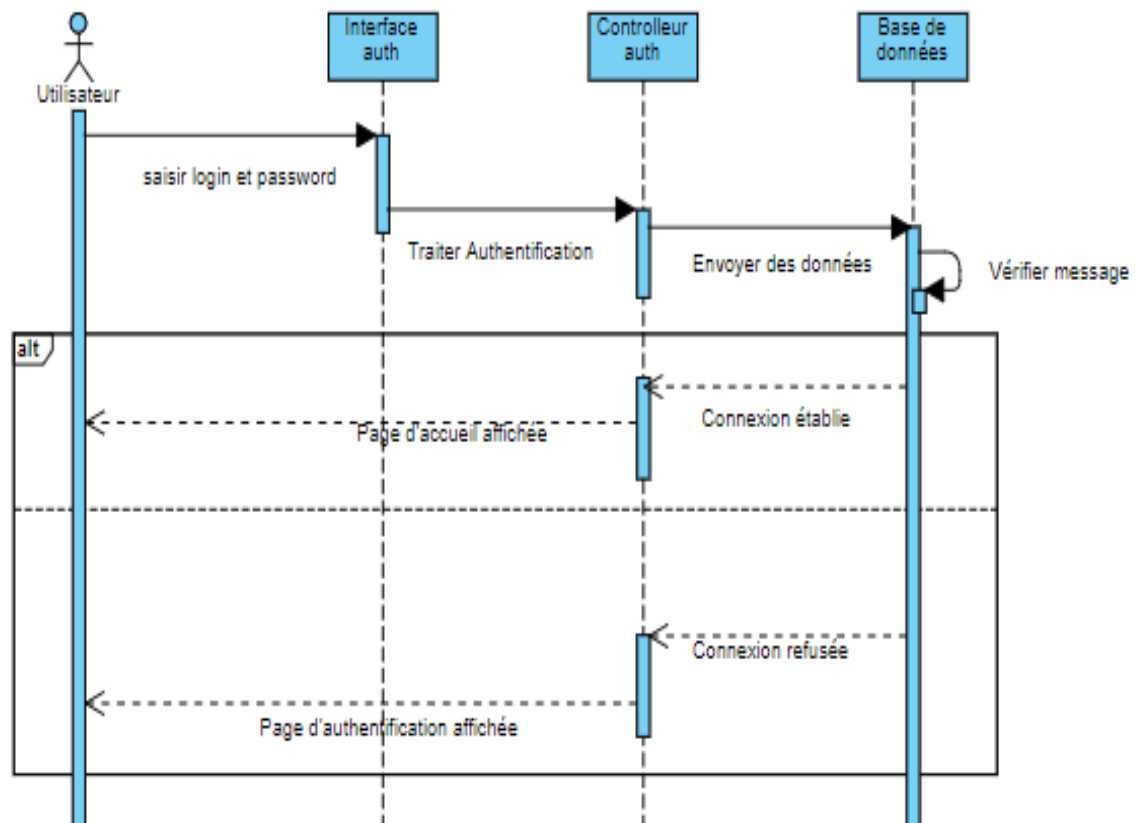


Figure 16: Diagramme de séquence Système « Authentification »

Pour accéder à l'application l'utilisateur doit saisir les paramètres d'authentification Login et password à chaque fois où il veut manipuler l'application.

Le système vérifie la validité des paramètres, s'ils ont accepté l'utilisateur est passé vers la page d'accueil de l'application, si non le système renvoie un message d'erreur.

- Gestion des clients :

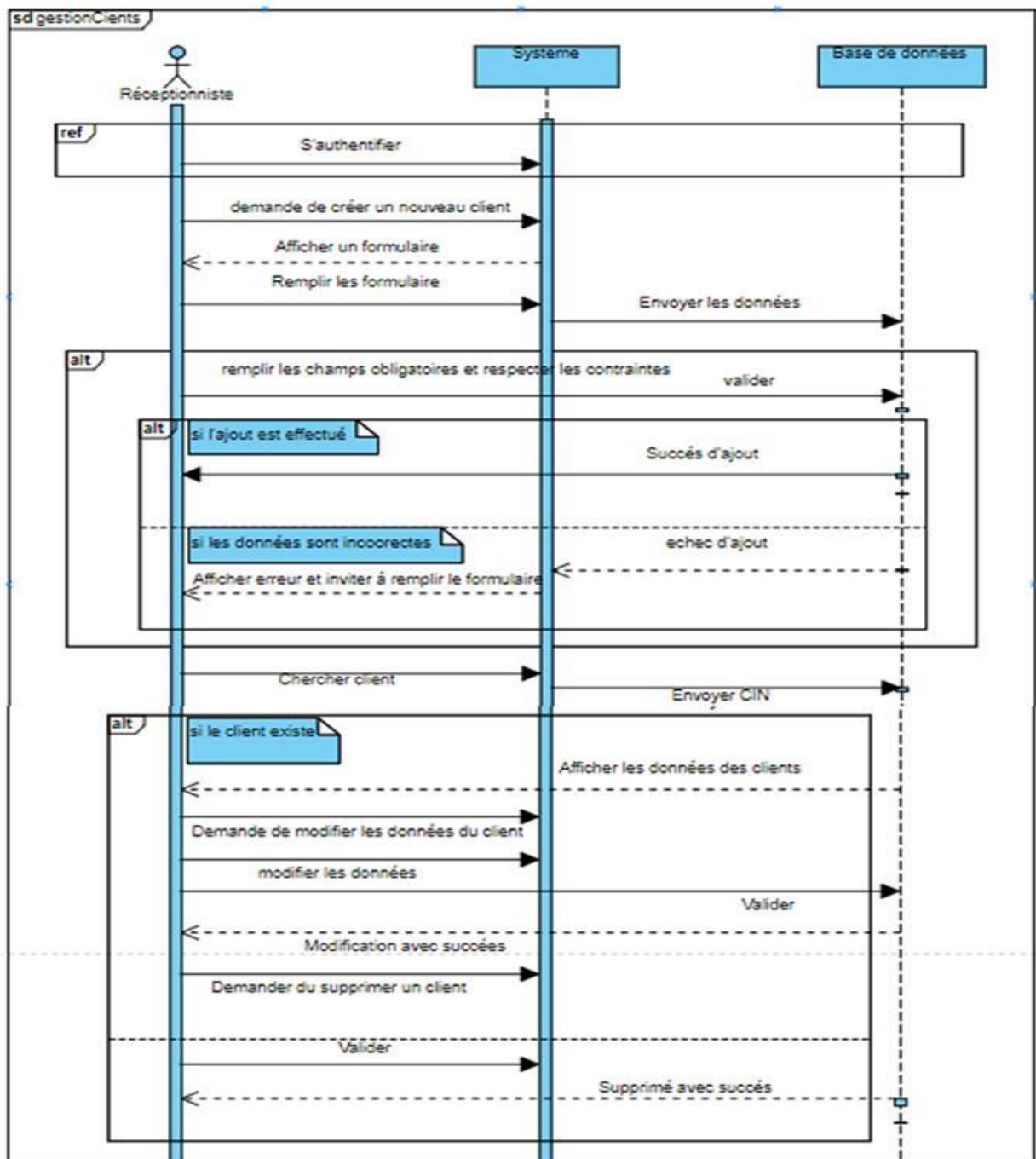


Figure 17: Diagramme de séquence système « gestion des clients »

La réceptionniste doit ajouter un nouveau client. Elle demande le formulaire « client » puis elle remplit les données relative au client et valide. Les champs obligatoires dans cette partie sont : Nom Ainsi le numéro de la C.I.N doit être égale à huit chiffres sinon il y aura un message d'erreur. Par la suite le système informe l'utilisateur que l'ajout a été effectué avec succès.

Pour supprimer un client, la réceptionniste cherche le patient à l'aide des filtres de recherche par son numéro de carte CIN, Si le client existe les donnée s'affichent dans la vue, la réceptionniste sélectionne le client et le supprime.

- Gestion des RDV :

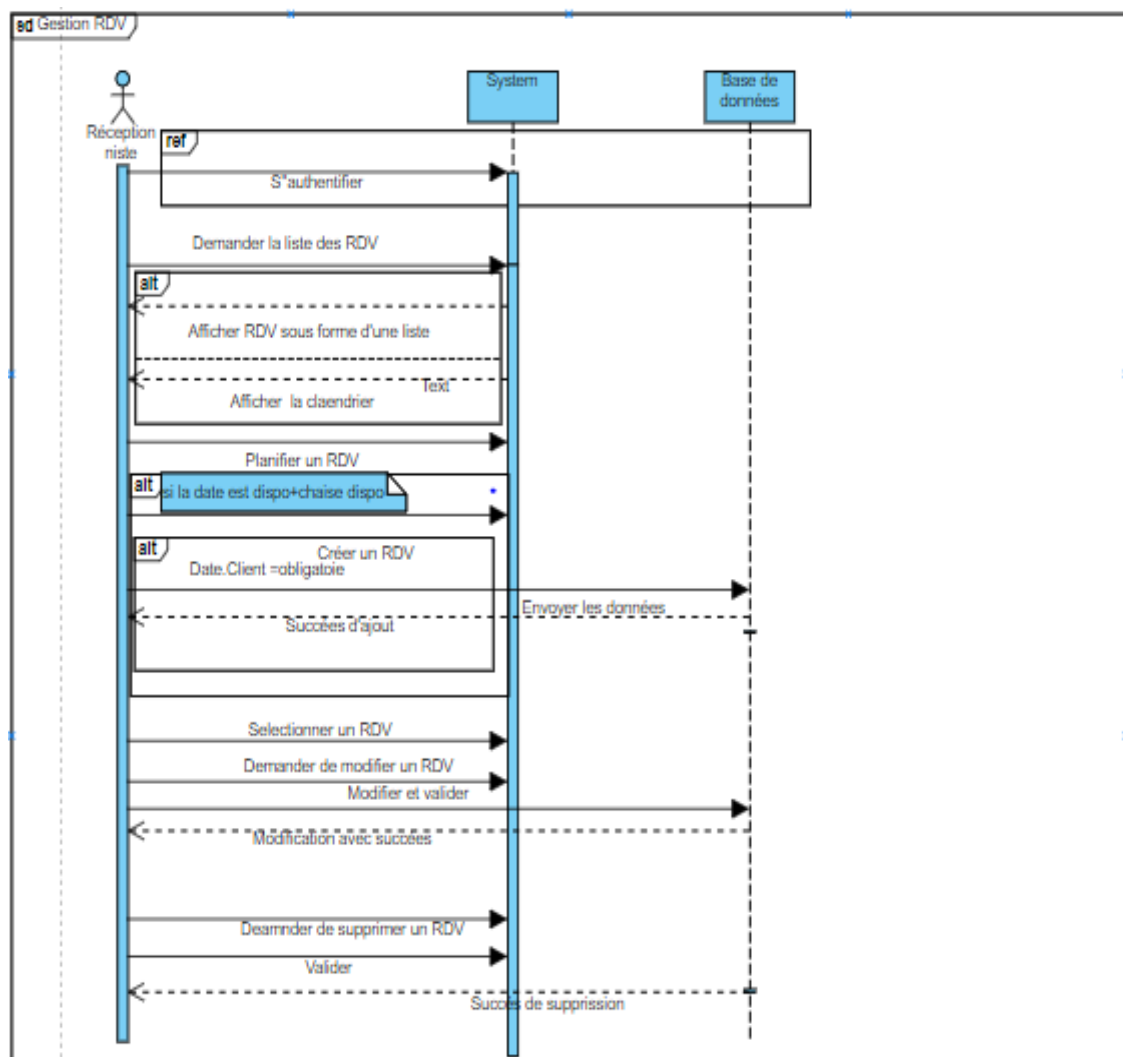


Figure 18: Diagramme de séquence Système de Gestion RDV

La réceptionniste doit planifier un nouveau rendez-vous. Elle doit accéder au calendrier ou la liste des RDV pour consulter l'ensemble des RDV créés, une fois la liste ou le calendrier s'affichent, la réceptionniste doit sélectionner une date pour ajouter le rendez-vous, puis elle doit remplir les détails de ce dernier et valider. Les champs obligatoires dans cette partie sont : Date RDV, le numéro du CIN du client. La date du rendez-vous doit être supérieure ou égale à la date

d'aujourd'hui et la date choisie doit comporter une chaise disponible sinon il y aura un message d'erreur. Puis l'ajout est effectué avec succès et le RDV sera affiché dans la liste des RDV et le calendrier. On peut par la suite modifier ou supprimer un RDV.

5. Conclusion :

Dans ce chapitre, nous avons conçu et documenté le code que nous devons produire. Dans cette phase, toutes les questions concernant la manière de réaliser le système à développer ont été élucidées. Le produit obtenu est un modèle graphique (ensemble de diagrammes) prêt à être codé. Dans le chapitre suivant nous allons étudier en détails les outils et les langages utilisés durant la phase de construction.

Chapitre III: Réalisation

1. Introduction

Après avoir clôturé la phase de conception, la solution étant déjà choisie et bien étudiée. Nous présentons dans ce chapitre quelques détails de réalisation. Tout d'abord, nous présentons un aperçu sur les technologies et les outils utilisés pour réaliser notre application. Finalement, nous allons élaborer une présentation des différentes interfaces créées.

2. Environnement matériel :

Lors de la réalisation de ce projet, nous avons utilisé notre ordinateur dont la configuration est présentée ci-dessous :

PC-Portable / RAM	1. ASUS X541UJ/ 8Go
Disque dur	1To
Processeur	Intel Core i7-7500U 7è Gén
Système d'exploitation	LINUX

Tableau 3 : Environnement matériel

3. Environnement de travail

Dans cette partie, nous spécifions les logiciels utilisés, l'environnement de conception, l'environnement de la base de données utilisées lors de la réalisation de notre projet. Enfin, nous indiquons les outils et le langage choisi.

3.1 Environnement de conception UML :

✓ StarUml :

C'est un outil de conception UML innovant consacré à rendre la modélisation d'un système facile et rentable en transformant les conditions et les détails d'un système en des solutions

graphiques spécifiques et détaillées. Cet outil offre une gamme complète d'outils logiciels de modélisation de tous types de diagrammes, de planification, de modélisation de classe etc.

On a utilisé cet outil dans l'élaboration des diagrammes des cas d'utilisations et le diagramme des classes.



Figure 19:logo StarUML

✓ **Visual Paradigm :**

Visual Paradigm est un outil UML CASE prenant en charge UML 2, SysML et la notation de modélisation de processus métiers du groupe de gestion des objets. Outre la prise en charge de la modélisation, il offre des fonctionnalités de génération de rapports et d'ingénierie de code, y compris la génération de code.

On a opté d'utilisé cet outils dans la création des diagrammes de séquence pour son flexibilité et la variété de ses fonctionnalités.



Figure 20: Logo de Visual Paradigm

3.2 Environnement de développement :

3.2.1 Outil utilisé :

✓ **Atome**

Un éditeur de texte sophistiqué pour le code, le balisage et la prose. Il génère automatiquement un index à l'échelle du projet de chaque classe, méthode et fonction.



Figure 21: Logo Atom

3.2.2 Langage utilisé:

- **Python**

Dans le développement de notre application nous avons utilisé le langage de programmation interprété Python. Il est un langage portable, dynamique, extensible, qui permet une approche modulaire et orientée objet de la programmation. [1]



Figure 22: Logo Python

- **XML (extensible Markup Language)**

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



Figure 23: Logo XML

- **La base des données PostgreSQL**

Est un système de gestion de base de données relationnelle et objet (SGBDRO). Il constitue un système libre distribué sous la licence BSD. Ce système a beaucoup de concurrents tels que. MYSQL, ORACLE, SQL server...

- **Le langage de modélisation unifié UML**

C'est un langage de modélisation objet officiellement approuvé, en 1997, par L'OMG. UML unifie des méthodes de conception logicielle orientées objet telles que Booch, OMT et OOSE qui coexistaient jusque-là sans beaucoup de comptabilité entre elles. UML les rassemble

non pas en proposant une synthèse, mais en créant un langage de modélisation, c'est-à-dire une notation unique pour faciliter la conception de programmes.



Figure 24: Logo UML

4. Odoo : [4]

4.1 Description ODOO :

Anciennement OpenERP ou Tiny ERP, c'est un Progiciel de Gestion Intégré(PGI) Open Source. C'est un Framework par-dessus lequel sont construites les briques du PGI (CRM, gestion des ventes, des achats, de la production, de la RH, etc.). Il propose des fonctionnalités qu'on attend d'un environnement de développement moderne, sans : un ORM (Object-Relational Mapping), une gestion des utilisateurs, des groupes et des droits d'accès, différents types d'écrans (listes, arbres, formulaires, diagrammes de Gantt, Kanban, calendriers), une gestion des workflows, une édition de rapports, gestion des traductions, etc.



Figure 25: Logo ODOO

4.2 Avantages d'ODOO :

Afin d'atteindre l'objectif de notre projet, nous avons opté pour Odoo :

- Pour son intégration: permet de consolider les applications, de centraliser les données et rendre le travail quotidien plus simple, au sein d'une interface agréable et unifiée

- Pour son modèle économique: solution ERP open-source
- Pour son support de document : bien documenter
- Pour l'intégration des technologies ouvertes et modernes : python pour la programmation, PostgreSQL pour la base de données, fortes intégration des services web et développement d'une interface web mobile particulièrement compétitive.

4.3 Architecture Odoo :

Odoo est basé sur une architecture trois-tiers : base de données (PostgreSQL), traitement (le serveur Odoo), présentation (client Web) et utilise, assez naturellement, le paradigme MVC (Modèle-Vue-Contrôleur).^[5]

Nous avons présenté l'architecture d'odoo dans la figure ci-dessous:

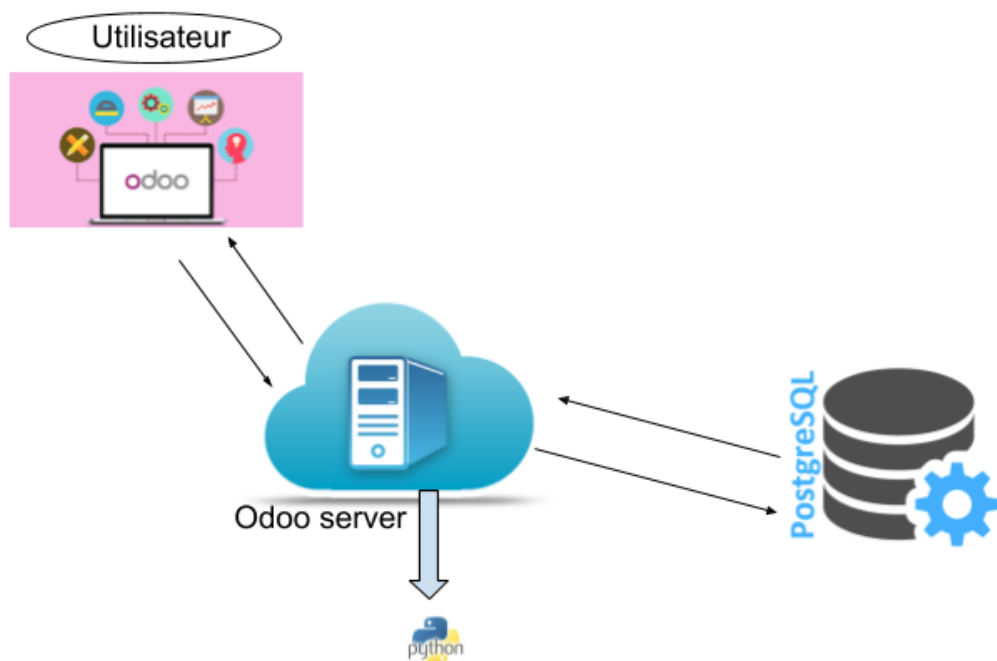


Figure 26: Architecture Odoo

4.4 Les principaux modules Odoo :

- CRM (Customer Relationship Management): ce module permet de créer et entretenir une relation mutuellement bénéfique avec les clients
- Ressources humaines

- Logistique: gérer toute la chaîne logistique, de l'achat des marchandises en passant par l'approvisionnement et le stockage jusqu'à la préparation des commandes et leurs livraisons
- Comptabilité: ce module destiné aux salariés professionnels de la finance et de la comptabilité [4]

Dans la figure 27, nous avons présenté les modules odoo:



Figure 27: Modules d'Odoo

5. Scénario complet d'utilisation de la solution

Nous allons présenter dans cette partie certaines interfaces principales développées pour notre application qui enrichissent les différents cas d'utilisation.

- Les interface de gestion des RDV :



Figure 29: la création d'un nouveau RDV

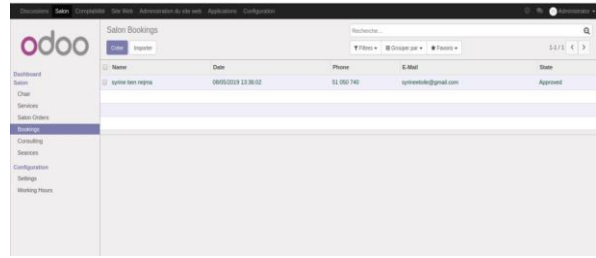


Figure 28: la liste des RDVs créés

- L'interface de gestion des chaises :

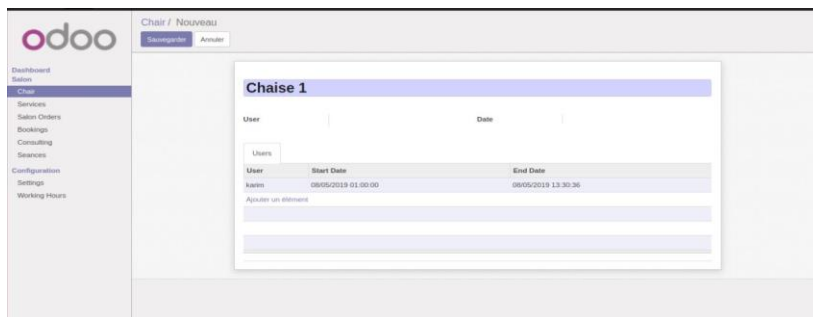


Figure 30: la création d'une chaise

- L'interface de gestion des commandes :

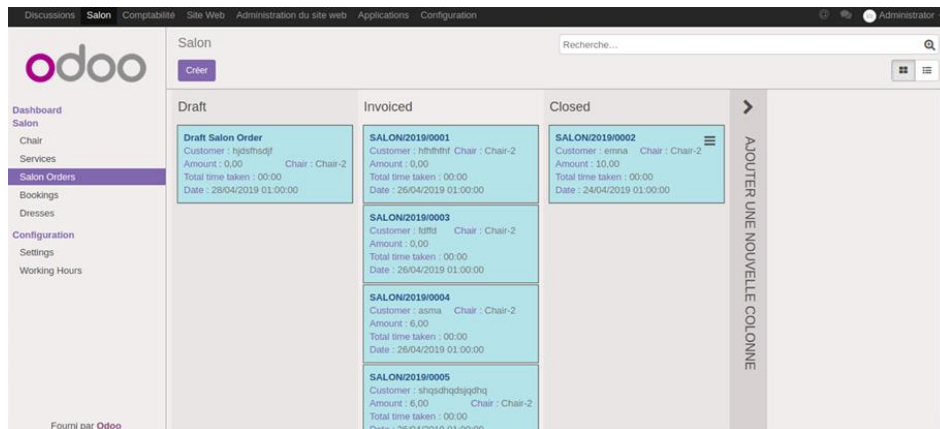


Figure 31: la liste des commandes

- **L'interface de gestion de facture :**

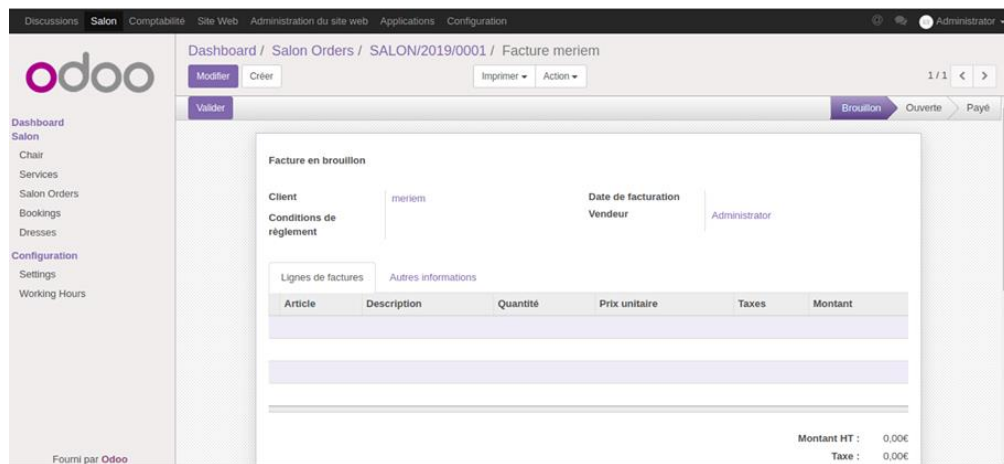


Figure 32: l'interface de gestion de facture

6. Conclusion

Nous avons consacré ce chapitre pour présenter les environnements que nous avons utilisés pour le développement de notre jeu et ceci, tout en expliquant les différents outils mis en jeu.

Puis nous avons présenté l'architecture fonctionnelle et des captures d'écran décrivent tous les scénarios possibles dans le jeu.

7. Conclusion générale et perspectives

Le stage que nous avons effectué dans la société Smart2Do Technologies, était une occasion pour découvrir le monde professionnel, celui de l'entreprise et particulièrement celui des ERP: intérêt majeur de Smart2Do Technologies, ce stage nous a permis de travailler dans le monde de l'open source, et plus précisément des progiciels de gestion intégrée tel que OpenERP. Ce qui nous a conduits à approfondir nos connaissances dans ces domaines, notamment le Framework odoo.

Le présent travail consiste à la mise en œuvre d'un module « Gestion d'un un centre de beauté» dans Odoo permet d'avoir une vision claire sur les fonctionnements de centre, de gérer les ressources du centre, ainsi de faciliter la communication entre les utilisateurs. Cet outil s'est avéré très efficace, sécurisé et facile à utiliser.

Finalement, notre travail est encore d'actualité et ne s'arrête pas à ce niveau. En effet plusieurs perspectives s'offrent à ce projet : Pour rendre l'application répond plus aux tendances du marché, nous proposons d'intégrer un module WEB qui permet aux clients de réserver leurs places en ligne, le client peut aussi voir les chaises disponibles et les détails de sa commande Cela peut être un plus si le fondateur de la société SMART2DO pense à fournir une application mobile pour ces clients. Outre, on a planifié d'intégrer un module de gestion des stocks pour les produits utilisés dans les différents services du salon Finalement, vu l'accomplissement de projet, nous souhaitons très fortement qu'il soit le fruit du progrès, de l'évolution et qu'il reste à la hauteur des exigences de la société, nous souhaitons par ailleurs la satisfaction de SMART2DO et de tous ses responsables, ainsi que les membres de jury.

8. Bibliographie

- [1] http://sebastien.nameche.fr/supports/Le_framework_Odoo_v0.1_20151125.pdf
- [2] <https://dumas.ccsd.cnrs.fr/dumas-01535649/document>
- [3] <https://www.qualitystreet.fr/2007/02/03/le-processus-unifie-en-bref>
- [4] https://www.odoo.com/fr_FR/
- [5] <http://apcpedagogie.com/connaitre-larchitecture-dodoo/>
- [6] <https://www.eyrolles.com/Informatique/Livre/apprendre-a-programmer-avec-python-3-9782212134346/> (ouvrage électronique)