

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

الجمهورية الجزائرية الديمقراطية الشعبية

MINISTRY OF HIGHER EDUCATION

AND SCIENTIFIC RESEARCH

وزارة التعليم العالي والبحث العلمي

HIGHER SCHOOL IN APPLIED SCIENCES



المدرسة العليا في العلوم التطبيقية
École Supérieure en
Sciences Appliquées

المدرسة العليا في العلوم التطبيقية



Mémoire de fin d'étude

Pour l'obtention du diplôme d'Ingénieur

Filière : Génie Industriel

Spécialité : Management Industriel et Logistique

Présenté par :

BOUACHRIA Sabrina

LARBI MOKHTAR Hayet

Thème

**Conception d'un système d'information
pour l'entreprise Aquatec
Biotechnologie (Groupe kherbouche)**

Soutenu publiquement, le 10/2020, devant le jury composé de :

M Mehdi SOUIER	MCA	ESM. Tlemcen	Président
M Fouad MALIKI	MCB	ESSA. Tlemcen	Directeur de mémoire
M OUSSAMA SEKKAK	Ingénieur	Groupe Kherbouche	Co- Directeur de mémoire
M Mohammed BENNEKROUF	MCB	ESSA. Tlemcen	Examineur 1
M IMANE KOULOUGHLI	MCB	ESSA. Tlemcen	Examineur 2

Année universitaire : 2019 /2020

Remerciements

Tout d'abord, nous remercions Dieu de nous avoir donné la force, la patience et la volonté d'accomplir ce projet en dépit de la situation actuelle.

Nous adressons toute notre reconnaissance et nos chaleureux remerciements à :

Notre godfather, la source de motivation et d'amour paternel chef de filière de GI notre professeur et encadreur MS FOUAD MALIKI pour son temps et ses efforts dévoués tout au long de ces 3 dernières années.

Toute l'équipe professoral et administrative de la filière génie industriel spécialement et toute l'école en général pour les connaissances et le savoir-faire qu'ils nous ont données pour pouvoir atteindre ce point de notre vie académique.

Notre soutien technique, le Co-encadreur Ms OUSSAMA SEKKAK pour tous ses conseils et son aide dans la réalisation de notre modeste travail.

La promotion la plus spéciale et unique de tous les temps GI 2017 2020 ; notre deuxième famille pour tous les moments de support et de joie.



Je dédie ce modeste travail à :

A mon cher père Sahraoui, qui m'accompagne toujours par ses prières, son amour et son soutien.

Ma réussite est la tienne ! à ma mère ma vie Dhabî Bent Larbi. Maman je t'aime trop, je te remercie pour le soutien moral et matériel et pour l'amour et la paix que tu me portes toute ma vie

Puisse dieu, le très Haut vous accorde santé, bonheur et longue vie en sorte que jamais je ne déçoive.

A ma très chère et adorable sœur Fatiha, aucune dédicace ne serait exprimée tout l'amour que j'ai pour toi, ma deuxième mère.

A mes frères Ahmed, Ghoulam Allah et Mohamed.

A ma binôme et amie Hayet.

A toutes mes amis qui sont toujours là pour moi.





Je dédie ce mémoire à tous les membres de ma famille : ma mère aimante **DJAZIA**, mon père le plus attentionné **MOHAMMED**, ma sœur très encourageante **FATIMA** et mon frère compatissant **ANISS**.

Ma grand-mère bien-aimée **CHERIFA** « paix a son âme », elle était la plus reconnaissante mais aussi la plus fière de moi et je ne pourrais atteindre nulle part où je suis sans son soutien à tous les niveaux.

Ma grand-mère la plus gentille **KHADIJA**, qui m'a donné tous les remèdes naturels et qui s'est toujours inquiétée de ma santé et de ma prospérité.

Je dédie également ce travail à tous mes amis et à ma binôme **SABRINA**

Tous mes voisins et tous les professeurs qui m'ont appris un jour.



Table de matière:

Remerciements	I
Dédicace	I
Dédicace	II
Liste des figures	XI
Liste des tableaux	XII
Liste des acronymes	XIII
Introduction générale.....	XIII

Chapitre 01 : Généralité sur les outils et langages utilisés.

1. Introduction.....	1
2. Généralité sur les systèmes d'informations (SI)	1
2.1. Définition des systèmes d'information.....	1
2.2. Historique et évolution des systèmes d'information	1
2.3. Représentation des systèmes d'information	3
3. Généralité sur le langage de modélisation UML	4
3.1. Historique.....	4
3.2. Définition	5
3.3. Notation.....	5
3.4. Les diagrammes d'UML.....	8
3.5. Caractéristiques d'UML	10
4. Python :.....	11
5. XML.....	11
6. Les ERPs.....	11
6.1. Historique.....	11
6.2. Définition	12
6.3. Architecture technique	12
6.4. Architecture modulaire des ERP's	13
6.4.1. Les principaux éditeurs d'ERP.....	13
6.4.2. Architecture modulaire.....	13
6.5. Pourquoi un ERP open source	14

6.6.	Caractéristiques des ERP	14
7.	Odoo.....	15
7.1.	Historique des versions.....	15
7.2.	Odoo et ses principales fonctionnalités.....	16
7.3.	Les caractéristiques d'Odoo.....	17
7.4.	Architecture technique d'Odoo	17
7.5.	Avantages d'Odoo	18
8.	Conclusion	18

Chapitre 02 : Etude préalable et discussion des besoins.

1.	Introduction.....	19
2.	Analyse de l'existant	19
3.	Présentation du projet	19
3.1.	Motivation de projet	19
3.2.	Problématique	19
3.3.	Description du contexte	19
3.4.	Objectifs de projet	20
4.	Démarche de développement	20
4.1.	Etude comparative	20
4.1.1.	Model en cascade.....	20
4.1.2.	Model en V.....	21
4.1.3.	Le modèle incrémental.....	22
4.1.4.	Model spirale.....	23
5.	Présentation de l'entreprise.....	24
5.1.	Présentation du Groupe KHERBOUCHE.....	24
5.2.	Présentation de AQUATEC Biotechnologie.....	25
5.2.1.	Activité principale.....	25
5.2.2.	Activités secondaires	25
5.2.3.	Fiche technique de l'entreprise.....	26
5.2.4.	Organisation de l'entreprise	26
6.	Analyse des flux.....	26
6.1.	Définition.....	26
6.2.	Types du flux	27
6.3.	Mode du flux.....	27

6.4.	Analyse des flux au sein de l'entreprise	27
7.	Analyse et discussion des besoins	29
7.1.	La notion de besoin	29
7.2.	Expression des besoins	30
7.2.1.	Les besoins fonctionnels	30
7.2.2.	Les besoins techniques.....	30
8.	Conclusion	32

Chapitre 03 : réalisation et implémentation du module gestion de production.

1.	Introduction.....	30
2.	Modélisation structurale du système.....	30
2.1.	Diagramme de classe	30
2.1.1.	Les règles de Gestion	30
2.1.2.	Dictionnaire de données :.....	31
2.1.3.	Diagramme de classe de système.....	32
3.	Modélisation comportementale du système	32
3.1.	Les diagrammes de cas d'utilisations des processus métiers de système	33
3.1.1.	Administration générale	33
3.1.2.	Gestion des prévisions	34
3.1.3.	Gestion de la production	35
3.1.4.	Gestion des achats et approvisionnement.....	37
3.1.5.	Gestion commerciale	38
3.2.	Les diagrammes de séquences du système.....	39
3.2.1.	Diagramme de séquence d'authentification	40
3.2.2.	Diagramme de séquence de la gestion de vente	41
3.2.3.	Diagramme de séquence de la gestion des prévisions.....	42
3.2.4.	Diagramme de séquence de la gestion de production	43
4.	Conclusion	44

Chapitre 03 : Conception du module gestion de production.

1. Introduction.....	44
2. Environnement de travail	44
2.1. Environnement matériel.....	44
2.2. Environnement logiciel.....	44
3. Le codage	44
3.1. Structure de module « manufacturing Aquatec »	44
4. L'implémentation	48
5. La manipulation	48
5.1. Insertion les coordonnées de l'entreprise.....	49
5.2. Ajouter un utilisateur	49
5.3. Ajouter un article.....	49
5.4. Insertion de nomenclature.....	51
5.5. Lancement un ordre de fabrication.....	51
5.6. Déconstruction d'un ordre de fabrication	52
6. Conclusion	52
Conclusion générale générale	53
Résumé.....	54
Abstract	54
ملخص	54

Liste des figures

Figure 1.1 : évolution des Systèmes d'Information	3
Figure 1.2 : Représentation d'un Système d'Information.....	4
Figure 1.3 : représentation de classe en UML.	5
Figure 1.4 : représentation d'un objet en UML.....	6
Figure 1.5 : représentation d'une interface en UML.	6
Figure 1.6 : représentation d'une collaboration en UML.	6
Figure 1.7 : représentation de cas d'utilisation en UML.....	7
Figure 1.8 : représentation d'acteur en UML.....	7
Figure 1.9 : représentation de composant en UML.	7
Figure 1.10 : représentation de l'état initial en UML.....	7
Figure 1.11 : représentation de l'état final en UML.....	7
Figure 1.12 : Architecture technique d'ERP.....	13
Figure 1.13 : Architecture modulaire d'ERP	14
Figure 1.14 : logo d'Open ERP Odoo.	17
Figure 1.15 : Architecture trois tiers d'Odoo.....	18
Figure 2.1 : processus en cascade.....	21
Figure 2.2 : processus en V.....	21
Figure 2.3 : processus incrémental.	23
Figure 2.4 : processus spirale.....	24
Figure 2.5 : Station de potabilisation.....	25
Figure 2.6 : poste locale de télégestion SOFREL	25
Figure 2.7 : schéma récapitulatif des flux.....	28
Figure 2.8 : schéma récapitulatif des besoins.	30
Figure 3.1 : diagramme de classe.	32
Figure 3.2 : diagramme de cas d'utilisation « gestion d'administration ».....	33
Figure 3.3 : Exemple de PIC.....	34
Figure 3.4 : diagramme de cas d'utilisation « gestion de prévision ».....	34
Figure 3.5 : Exemple de PDP.....	35
Figure 3.6 : diagramme cas d'utilisation « gestion de la production ».....	36
Figure 3.7 : diagramme cas d'utilisation « gestion des achats et approvisionnement ».....	37
Figure 3.8 : diagramme cas d'utilisation « gestion commerciale ».....	38
Figure 3.9 : Diagramme de séquence d'authentification.	40
Figure 3.10 : Diagramme de séquence de la gestion de vente.	41
Figure 3.11 : diagramme de séquence de la gestion des prévisions.	42
Figure 3.12 : Diagramme de séquence de la gestion de production.....	43
Figure 4.1: structure de module mrp_aquatec.....	45
Figure 4.2: fichier init de module.....	45
Figure 4.3: fichier manifest de module.....	46
Figure 4.4:exemple d'un module	46
Figure 4.5:exemple de vu de mrp production.	47
Figure 4.6: exemple d'une action.....	47
Figure 4.7 : Interface des applications de odoo.....	48

Figure 4.8 : module de production installé	48
Figure 4.9 : Les coordonnées d'entreprise.....	49
Figure 4.10 : interface d'ajout un utilisateur.....	49
Figure 4.11 : Interface d'ajout d'un article	50
Figure 4.12 : interface de réapprovisionnement.....	50
Figure 4.13 : interface de prévision d'un article	50
Figure 4.14 : Liste des articles ajoutés	51
Figure 4.15 : nomenclateur de kite solaire 150w	51
Figure 4.16 : ordre de fabrication.....	52
Figure 4.17 : interface de déconstruction d'un ordre de fabrication	52

Liste des tableaux

Tableau 1.1 : Les types de relation en UML avec notation.	8
Tableau 1.2 : Historique des versions d'Odoo.	16
Tableau 2.1 : Carte d'identité de l'entreprise AQUATEC Biotechnologie.....	26
Tableau 2.2 : les flux au sein de l'entreprise.....	29
Tableau 3.1 : dictionnaire de donnée.....	32
Tableau 4.1 : composantes d'un fichier manifest.....	46

Liste des acronymes

Acronyme	Signification
ATO	Assemble to order
CRM	Customer Relationship Management
DAC	Diagramme d'activité
DCL	Diagramme de classe
DCO	Diagramme de communication
DCU	Diagramme de cas d'utilisation
DCP	Diagramme des composants
DET	Diagramme d'état-transition
DGI	Diagramme global d'interaction
DOB	Diagramme d'objet
DPL	Diagramme de déploiement
DSC	Diagramme de structure composite
DES	Diagramme de séquence
DSS	decision support system
DTP	Diagramme de temps
ERP	Entreprise ressource planning
ES	Expert system
IA	Intelligence artificielle
GNU AGPL	Affero General public license
GNU GPL	General public license
KMS	Les systèmes de gestion des connaissances
MOA	Maitre d'ouvrage
MOE	Maitre d'œuvre
MRP	Manufacturing requirements planning
MTO	Make to order
MTS	Make to stock
OMG	Object Management Group
OMT	Object Modeling Technique
OOSE	Object-oriented software engineering
PDP	Plan directeur de production
PIC	Plan industriel et commercial
PME	Petites et moyennes entreprises
SI	Système d'information
SQuaRE	software quality requirements and evaluation
TPE	Les très petites entreprises
XML	Extensible Markup Language

Introduction générale

De nos jours, les industries se développent de plus en plus parallèlement au marché ; les clients sont de plus en plus exigeants et tout retard ou défaut peut entraîner des pertes importantes pour l'entreprise.

L'utilisation optimale des ressources est donc cruciale pour chaque entreprise, et les ERP répondent tout aussi bien à ce besoin

L'ERP est un investissement informatique majeur pour les PME, qui peut être considéré comme une décision stratégique.

Aujourd'hui, plus de 2000 entreprises investissent dans des logiciels intégrés pour gérer leurs activités commerciales en conservant la fluidité et la disponibilité de l'information

Contexte :

Le présent document est le résultat du travail qui s'inscrit dans le cadre du projet de fin d'étude effectuée au sein de la société Aquatec biotechnologie en vue d'obtenir le diplôme d'ingénieur en génie industriel option management industriel et logistique à l'école supérieure de science appliquée Tlemcen.

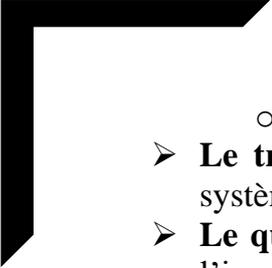
Notre projet intitulé « Conception d'un système d'information pour l'entreprise Aquatec Biotechnologie (Groupe kherbouche) » consiste à concevoir un système d'information pour l'entreprise en question afin d'améliorer ses performances en termes de traçabilité et de sauvegarde des flux d'informations. Nous utilisons le langage UML afin de modéliser les différents modules à développer ; l'implémentation des modules se fera avec l'open ERP ODDO.

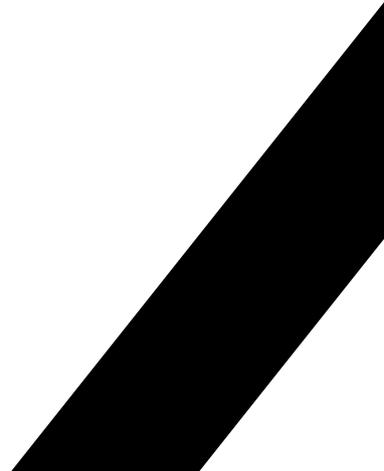
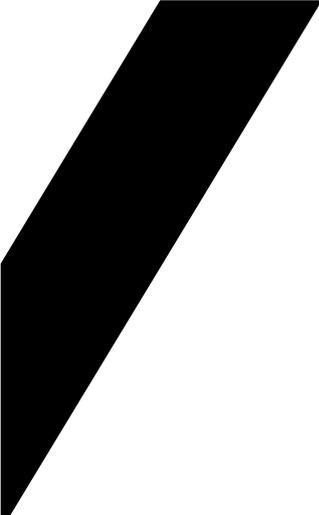
L'objectif principale du projet c'est de mettre en place un système de gestion de production.

Pour bien mener ce projet, nous avons choisi de suivre un mode incrémental pour le développement, qui est plus adapté pour la conception des logiciels, elle permet aussi de réagir avec une grande vitesse aux modifications (mise à jour), et elle est compatible avec toute taille de projet.

Les chapitres de ce mémoire sont présentés comme suit :

- **Le première chapitre** présente des Généralités sur les outils et langages utilisés pour réaliser ce projet.
- **Le deuxième chapitre** intitulé étude préalable et discussion des besoins, dans ce chapitre en vas présenter :
 - Les détails de notre projet.
 - L'organisme d'accueil.

- 
- Le cahier de charge de l'entreprise.
 - **Le troisième chapitre** contient tous les diagrammes liés à la conception de système, ces derniers sont conçus en UML à l'aide de LucidChart.
 - **Le quatrième chapitre** qui présente le fruit de notre projet (le module réalisé, l'implémentation et la manipulation).



Chapitre 01 : Généralité sur les outils et langages utilisés

1. Introduction

Dans ce chapitre, nous mettons le point sur les outils et langages utilisés pour la réalisation de notre projet. Nous commençons par une présentation des systèmes d'information (S.I), suivi d'une définition du langage de modélisation unifié UML (Unified Modeling Language). Ce chapitre est clôturé par une description des progiciels de gestion intégrée (PGI/ERP), et du progiciel ODOO utilisé dans le cadre de ce travail.

2. Généralité sur les systèmes d'informations (SI)

L'ère dans laquelle nous entrons actuellement est celle des données. Le monde est envahi par l'information en particulier les entreprises. Ces informations doivent être collectées, stockées, classées et catégorisées tout en garantissant leur intégrité et l'accessibilité aux différents utilisateurs. Ceci est assuré par les S.I.

2.1. Définition des systèmes d'information

Dans toute entreprise, les systèmes d'informations représentent l'entité la plus fondamentale qui permet la gestion et la communication entre toutes les ressources de l'organisation (personnel, matériel, logiciel ou procédures).

Le S.I est composé de deux sous-systèmes un social et l'autre fonctionnel; dont le système social est composé de la structure organisationnelle et des personnes liées au SI, elle se soucie de l'intégration des salariés dans l'entreprise favorisant la vie sociale et la culture d'entreprise par la diffusion de l'information.

Le système fonctionnel (technique) regroupe toutes les technologies (hardware, software et équipements de télécommunication).

2.2. Historique et évolution des systèmes d'information ¹

Les systèmes d'information ont existé depuis longtemps. L'homme a toujours été l'épine dorsale des SI.

La croissance du domaine des SI a fait des progrès importants au cours des 50 dernières années. A mesure que le domaine s'est développé, de nouvelles spécialités et communautés de recherche sont apparues, et le niveau de recherche a augmenté de façon spectaculaire. Et nous pouvons diviser cette période sur 5 phases de 1950 jusqu'à aujourd'hui :

- La première période (1950 à 1960) - Système de traitement des transactions :

Depuis l'année 1950 et après le développement du premier ordinateur d'entreprise en 1951 par J. Lyon et Cie jusqu'à 1960, le rôle de la plupart des SI était assez simple puisqu'ils étaient principalement utilisés pour le traitement électronique des données (TED). Ils effectuent des activités comme le traitement des transactions, la tenue des dossiers et la comptabilité. L'informatique est souvent définie comme l'utilisation d'ordinateurs pour enregistrer, classer, manipuler et résumer des données. Elle est également appelée « Computer Business Research ».

¹S. ALTER. Management Perspective. Benjamin Cummings Publishing Company, 1996.

- La deuxième période (1960 à 1970) - système de bureautique (office automation system) :

Au cours de cette période, un autre rôle a été ajouté à l'utilisation des ordinateurs, à savoir le traitement des données en rapports informatifs utiles.

Ce nouveau rôle vise à développer une application métier qui fournit aux utilisateurs finaux des rapports de gestion prédéfinis qui donneraient aux gestionnaires les informations dont ils ont besoin pour prendre des décisions.

Cette époque marque également la période de développement où l'attention des organisations est passée lentement de la simple automatisation des processus opérationnels de base dans les années 1950 à la consolidation du contrôle au sein de la fonction de traitement des données.

- La troisième période (1970 à 1980) - système de gestion des informations :

Systèmes d'information de gestion (SGI) - fournissent des informations sous forme de rapports et d'affichages prédéfinis pour soutenir la prise de décision opérationnelle

Généralement, les SGI génèrent trois types d'informations de base : détaillées, résumées et d'exception. Les rapports d'information détaillée confirment généralement les activités de traitement des transactions. Un rapport de commande détaillé est un exemple de rapport détaillé.

L'information sommaire consolide les données dans un format qu'une personne peut examiner rapidement et facilement, tandis que l'information sur les exceptions filtre les données pour signaler l'information qui est hors d'une condition normale.

- La quatrième période (1980 à 1990) système d'aide à la décision (DSS decision support system) :

Dans cette deuxième ère, les progrès technologiques ont continué à s'envoler. Le principal progrès a été l'introduction des ordinateurs personnels (PC). Avec l'introduction des PC, les organisations ont commencé à répartissent leur puissance de calcul et de traitement dans toute l'organisation. Au fur et à mesure que l'éventail des utilisateurs s'est élargi, les organisations ont adopté une orientation de gestion plus forte par rapport à leur approche traditionnellement technique de l'exploitation des SI. Le mouvement a commencé à se concentrer sur le " système informatique interactif " pour aider les décideurs à résoudre les problèmes.

- La cinquième période (1990 - aujourd'hui) - système informatique exécutif :

C'est également au cours de cette période que le matériel informatique, les logiciels et les télécommunications ont évolué rapidement que le réseau TCP/IP ou l'Internet ont été largement adoptés. Cela devient un nouveau phénomène dans l'industrie des TI.

Cette période marque un changement important de la technologie des SI et de l'environnement des entreprises. La commercialisation de l'Internet a permis l'apparition de nouveaux modes de communication et de nouvelles façons de faire des affaires qui n'étaient pas possibles aux époques précédentes. L'Internet permet la diffusion des connaissances dans différentes parties du monde, indépendamment du temps et de l'espace.

En outre, des percées ont été réalisées dans le développement et l'application des techniques d'intelligence artificielle (IA) aux systèmes d'information des entreprises. Avec un besoin moindre d'intervention humaine, les travailleurs du savoir pourraient être libérés pour s'occuper de tâches plus complexes : Les systèmes experts (ES) et les systèmes de gestion des connaissances (KMS) sont interdépendants.

Au cours de cette période, du milieu à la fin des années 1990, on a assisté à l'émergence révolutionnaire des systèmes de planification des ressources de l'entreprise (ERP). Cette forme de système d'information stratégique propre à l'organisation intègre toutes les facettes d'une entreprise, y compris sa planification, sa fabrication, ses ventes, sa gestion des ressources, ses ressources humaines et son marketing - pratiquement toutes les fonctions de l'entreprise.

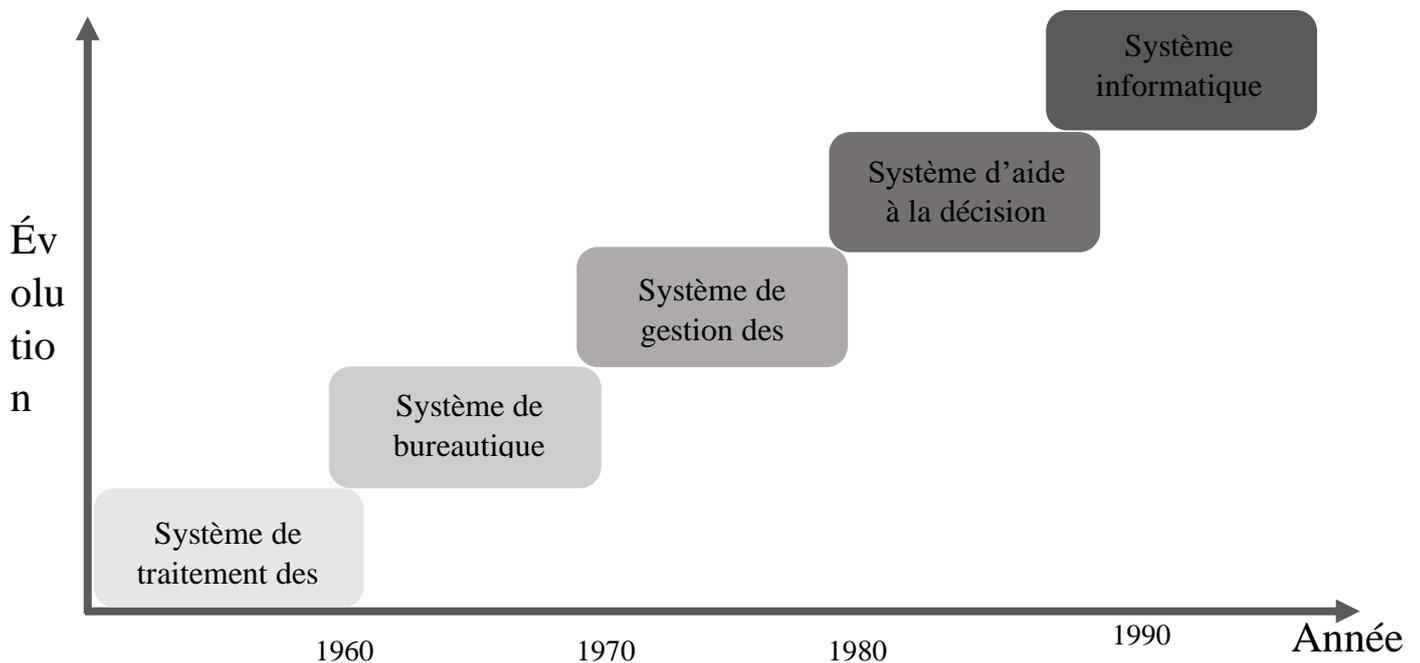


Figure 1.1 : évolution des Systèmes d'Information

2.3. Représentation des systèmes d'information

L'objectif de cette représentation est d'avoir une visibilité détaillée de tous les facteurs, incluant les liens ainsi que les impacts entre eux.

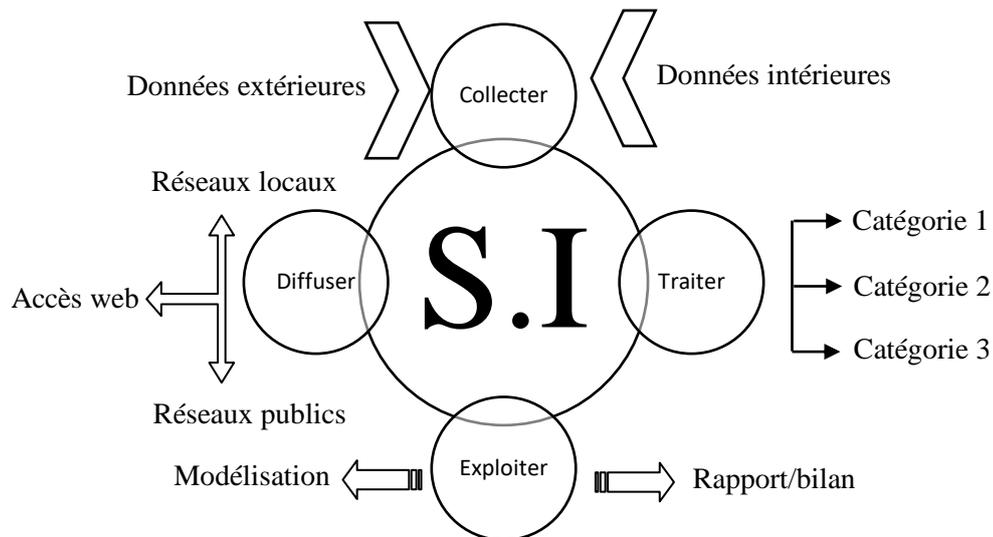


Figure 1.2 : Représentation d'un Système d'Information.

3. Généralité sur le langage de modélisation UML

3.1. Historique

Les langages de modélisation orientés objet sont apparus à la fin des années 1970 et dans les années 1980. De la fin des années 1980 au début de la décennie suivante, le nombre de langages de modélisation a explosé. La consolidation de ces différents systèmes était inévitable, et au milieu des années 1990, UML est apparu comme une combinaison de trois méthodes :

OMT (Object Modeling Technique) de James Rumbaugh, qui s'est avérée très utile lors de la manipulation de systèmes sensibles.

BOOCH de Grady Booch qui s'est illustrée comme une méthode très flexible pendant la conception et la construction des objets.

OOSE (Object-oriented software engineering) de Ivar Jacobson qui a fourni une excellente façon de contourner les cas d'utilisation. Elle a également une approche puissante pour la conception de haut niveau.

Cela a créé une stabilité et une prévisibilité, permettant aux développeurs de s'appuyer sur un système solide.

UML a été reconnu comme une norme par l'Object Management Group (OMG) en 1997. L'Object Management Group est responsable de la gestion d'UML depuis son adoption comme standard.

En 2005, l'Organisation internationale de normalisation a approuvé UML en tant que norme ISO. Il est utilisé dans diverses industries pour créer des modèles orientés objet.

La dernière version d'UML est la 2.5.1 qui a été publiée en décembre 2017. (slide share, 2014)

Les grandes étapes de la diffusion d'UML peuvent se résumer comme suit :

1994-1996 : rapprochement des méthodes OMT, BOOCH et OOSE et naissance de la première version d'UML.

23 novembre 1997 : version 1.1 d'UML adoptée par l'OMG.

1998-1999 : sortie des versions 1.2 à 1.3 d'UML.

2000-2001 : sortie des dernières versions suivantes 1.x.

2002-2003 : préparation de la V2.

10 octobre 2004 : sortie de la V2.1.²

3.2. Définition

UML signifie Unified Modeling Language (langage de modélisation unifié). C'est un standard qui est principalement utilisé pour créer des modèles de documentation orientés objet et significatifs pour tout logiciel présent dans le monde réel. Il nous fournit un moyen de développement des modèles riches qui décrivent le fonctionnement de tout système logiciel/matériel.

La conception en UML est utilisée pour organiser le code de programme de manière plus efficace et visuelle, les programmeurs créent souvent des " objets " qui sont des ensembles de données structurées dans les programmes.

3.3. Notation

UML est populaire pour ses notations graphiques c'est pourquoi ces notations UML sont les éléments les plus importants de la modélisation. L'utilisation efficace et appropriée de ces derniers est très importante pour réaliser un modèle complet et significatif.

Les notations sont divisées en deux types :

➤ Notation structurelle :

Elles sont considérées comme les noms des modèles UML Voici la liste des notations structurelles :

- Classe :

Une classe est utilisée pour représenter divers objets. Elle est utilisée pour définir les propriétés et les opérations d'un objet. En UML, on peut aussi représenter une classe abstraite (Une classe dont les fonctionnalités ne sont pas définies).

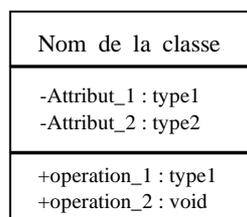


Figure 1.3 : représentation de classe en UML.

²JOSEPH GABAY, DAVID GABAY. UML 2 analyse et conception. DUNOD, 2007.

- Objets :

Un objet est une entité qui est utilisée pour décrire le comportement et les fonctions d'un système. La classe et l'objet ont les mêmes notations. La seule différence est que le nom d'un objet est toujours souligné en UML.

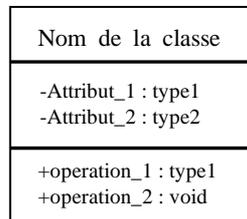
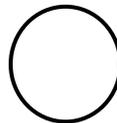


Figure 1.4 : représentation d'un objet en UML.

- Interface :

Une interface est similaire à un modèle sans détails d'implémentation, Lorsqu'une classe implémente l'interface, elle implémente également la fonctionnalité.



Nom-de-interface

Figure 1.5 : représentation d'une interface enUML.

- Collaboration :

La collaboration représente des responsabilités. En général, les responsabilités sont dans un groupe.

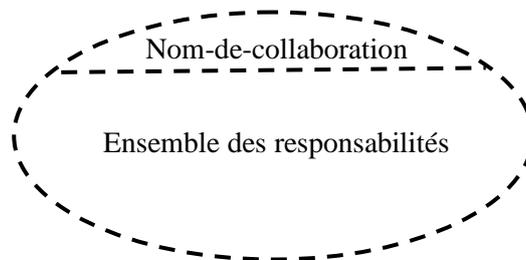


Figure 1.6 : représentation d'une collaboration en UML.

- Cas d'utilisation :

Le cas d'utilisation est représenté comme une ellipse avec un nom à l'intérieur. Il peut contenir des responsabilités supplémentaires.

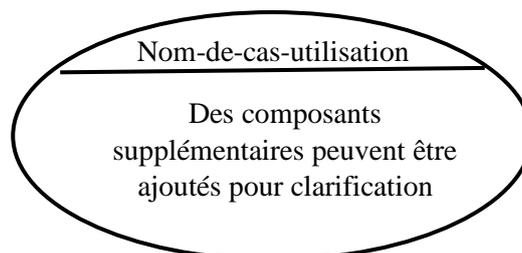
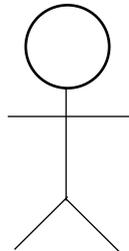


Figure 1.7 : représentation de cas d'utilisation en UML.

- Acteur :

Un acteur est utilisé dans un diagramme de cas d'utilisation pour décrire les entités internes ou externes.



Nom-d-acteur

Figure 1.8 : représentation d'acteur en UML.

- Composants :

Le composant est utilisé pour représenter toute partie d'un système.

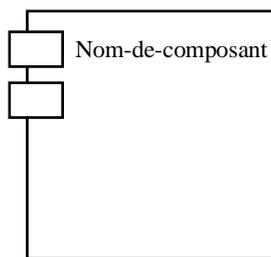


Figure 1.9 : représentation de composant en UML.

➤ Notations comportementales :

- L'état initial :

L'état initial est défini pour montrer le début d'un processus. Cette notation est utilisée dans presque tous les diagrammes.



Figure 1.10 : représentation de l'état initial en UML

- L'état final :

L'état final est utilisé pour montrer la fin d'un processus. Cette notation est également utilisée dans presque tous les diagrammes pour décrire la fin.



Figure 1.11 : représentation de l'état final en UML

➤ Notations relationnelles :

Un modèle n'est complet que si les relations entre les éléments sont décrites correctement. La relation donne une signification appropriée à un modèle UML. Voici les différents types de relations disponibles en UML :

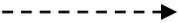
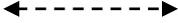
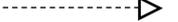
Les relations	Descriptions	Notations
Dépendance	L'élément source est dépendant de l'élément cible et peut être affecté par les changements qui lui sont apportés.	
Généralisation	Aussi appelée la relation parent-enfant. Ce type de relation est utilisé pour représenter le concept d'héritage.	
Association	Elle représente la relation entre deux éléments d'un système.	
Réalisation/ Implémentation	Dans ce cas, un élément décrit une responsabilité qui n'est pas implémentée et les autres éléments implémentent la fonctionnalité mentionnée par le premier élément. La relation de réalisation est largement utilisée tout en désignant les interfaces.	
Agrégation	Une association binaire entre une propriété et un ou plusieurs objets.	
Composition	Est une forme forte d'agrégation. Elle s'agit d'une relation entière/partie.	

Tableau 1.1 : Les types de relation en UML avec notation.

3.4. Les diagrammes d'UML

La définition de diagramme selon « Cambridge dictionary » est “un plan simple qui représente une machine, un système, une idée, etc., souvent dessinée pour expliquer son fonctionnement”.

La clé pour réaliser un diagramme UML est de connecter des formes qui représentent un objet ou une classe avec d'autres formes pour illustrer les relations et le flux d'informations et de données. Les normes UML actuelles prévoient 13 types de diagrammes différents répartis en deux grandes catégories : les diagrammes structurels et les diagrammes de comportement (d'interaction)

➤ Les diagrammes structurels :

Il montre la hiérarchie ou la structure des différents composants ou modules du système et montre comment ils se connectent et interagissent les uns avec les autres.

- Diagramme de classe (DCL):

Les diagrammes de classes sont le principal élément de construction de toute solution orientée objet. Ils décrivent la structure statique d'un système (classes, attributs, opérations et relations entre les classes).

- Diagramme de paquetage (DPA):

Comme son nom l'indique, diagramme de paquet organise les éléments d'un système en groupes apparentés (paquets) afin de minimiser les dépendances entre ces paquets.

- Diagramme d'objet (DOB) :

Les diagrammes d'objets décrivent la structure statique d'un système à un moment donné. Ils peuvent être utilisés pour tester la précision des diagrammes de classes.

- Diagramme des composants (DCP) :

Un diagramme de composants affiche la relation structurelle des composants d'un système logiciel. Ces diagrammes sont surtout utilisés pour travailler avec des systèmes complexes comportant de nombreux composants.

- Diagramme de structure composite (DSC) :

Les diagrammes de structure composite sont utilisés pour montrer la structure interne d'une classe.

- Diagramme de déploiement (DPL) :

Les diagrammes de déploiement décrivent l'architecture technique de logiciel avec la représentation des ressources physiques de système, y compris les nœuds, les composants et les connexions. Ces diagrammes sont utiles lorsque votre solution logicielle est déployée sur plusieurs machines, chacune ayant une configuration unique.

➤ Les diagrammes comportementaux :

Les diagrammes comportementaux montrent ce qui devrait se passer dans un système. Ils décrivent comment les objets interagissent les uns avec les autres pour créer un système fonctionnel pour une finalité souhaitée.

- Diagramme de cas d'utilisation (DCU) :

Les diagrammes de cas d'utilisation donnent une vue d'ensemble graphique des acteurs impliqués dans un système, des différentes fonctions nécessaires à ces acteurs et de la façon dont ces différentes fonctions interagissent.

- Diagramme d'activité (DAC) :

Illustrer la nature dynamique d'un système en modélisant le flux de contrôle d'une activité à l'autre. Une activité représente une opération sur une certaine classe du système qui se traduit par un changement d'état du système. C'est très important au début de tout programme car cela aidera le programmeur à identifier les principaux acteurs et processus impliqués dans le système.

- Diagramme de séquence (DSE) :

Les diagrammes de séquence en UML montrent comment les objets interagissent les uns avec les autres et l'ordre dans lequel ces interactions se produisent. Il est important de noter qu'ils montrent les interactions pour un scénario particulier.

- Diagramme d'état-transition (DET) :

Il décrit le comportement dynamique d'un système en réponse à des stimuli externes. Les diagrammes d'état sont particulièrement utiles pour modéliser les objets réactifs dont les états sont déclenchés par des événements spécifiques.

- Diagramme de communication (DCO):

Les diagrammes de communication sont un autre moyen de visualiser l'information plus couramment représentée par les diagrammes de séquence. Ils sont plus simples que les diagrammes de séquence et ne montrent que les messages qui passent entre les objets ou les rôles dans un programme logiciel, une infrastructure ou tout type de processus

- Diagramme global d'interaction (DGI) :

Les diagrammes de vue d'ensemble des interactions combinent les éléments des diagrammes d'activités avec les diagrammes de séquences pour montrer le déroulement de l'exécution du programme. Vous pouvez également utiliser un diagramme de vue d'ensemble des interactions pour déconstruire un scénario complexe qui, autrement, nécessiterait l'illustration de plusieurs chemins if-then-else sous la forme d'un seul diagramme de séquence.

- Diagramme de temps (DTP) :

Les diagrammes de synchronisation sont très similaires aux diagrammes de séquence. Ils représentent le comportement des objets dans un intervalle de temps donné.

3.5. Caractéristiques d'UML

Le langage de modélisation unifié « UML » est largement utilisé et accepté comme le langage de description des programmes. Il est utilisé dans une variété de buts et ses caractéristiques en font toujours un choix idéal pour les programmeurs, dont nous pouvons citer les plus importantes :

- ✓ Une représentation visuelle : qui aide le programmeur à mieux comprendre les interactions entre les entités et aussi les finalités attendues du programme.
- ✓ Réutilisabilité et lisibilité (adaptation) : Le diagramme est destiné à être compris par tout type de programmeur et permet d'expliquer les relations dans un programme de manière simple. De plus, un diagramme de modèle UML est facile à modifier, alors que la reprogrammation d'une section de code peut être fastidieuse et prendre beaucoup de temps.
- ✓ Un cout de développement réduit.
- ✓ Standards : actuellement UML est le langage de programmation standard en ce qui concerne la programmation orientée objet (accepté par l'Object Management Group (OMG) comme la norme pour le développement de logiciels de modélisation).

4. Python :

Python est un langage de programmation open source le plus utilisé par les informaticiens et les développeurs, il est facile à apprendre et à utiliser. Il permet au programmeur de focaliser sur ce qu'ils font plutôt que la façon dont ils font.

Python est orienté objet, extensible, dynamique, portable et gratuit.

5. XML

L'acronyme XML signifie Extensible Markup Language traduit en français langage de balisage extensible, est un langage informatique ou un métalangage pour être plus précis, il s'agit d'une version améliorée de l'HTML.

XML a pour mission de formaliser des données textuelles, il se classe dans la catégorie des langages des descriptions c'est-à-dire ce n'est ni langage de programmation ni langage de requête. Il est utilisé pour décrire les données avec des balises et des règles personnalisables.

6. Les ERPs

Vu l'évolution progressive et la compétitivité dans le marché, les entreprises doivent mettre en place un nouvel outil de gestion afin d'avoir un gain du temps et une efficacité.

Nous présentons en ce qui suit les progiciels de gestion intégré (PGI) ainsi que leurs architectures. Nous avons opté pour l'utilisation du PGI open source Odoo qu'on va détailler par la suite.

6.1. Historique

L'ERP (Entreprise resource planning) est relativement récent. Cet outil tire son origine du MRP.

Dans le début des années 60, les systèmes MRP (Manufacturing requirements planning) ont été introduits par l'ingénieur américain Joseph Orlicky. La mise en œuvre du MRP répond à un besoin déjà existant qui est de mettre en place des solutions de planification. La première entreprise à avoir mis en place un tel système est **Toyota** alors qu'elle avait besoin d'évaluer la quantité de composants nécessaires à la fabrication de ses véhicules.

Afin de répondre à ce besoin, l'ingénieur américain développe donc le concept du MRP. Il s'agit à l'époque d'un système destiné uniquement aux unités de production.³

Après l'apparition du MRP et dans les mêmes années Oliver Wight et George Plossl mettent au point le MRPAu (MRP2). D'où une évolution en trois phases :

³<https://www.ecommerce-nation.fr/>

- MRP0 (littéralement « planification des besoins en matières 0 ») : méthode de calcul des besoins matière mise au point en 1965 .
- MRP1: première application industrielle de la gestion intégrée des flux de production mise au point en 1971.
- MRP2 (littéralement « planification des ressources pour la fabrication 2 ») : en plus du calcul des besoins nets en matières premières et composants, effectue une planification desancements en tenant compte des capacités des ressources par période mise au point en 1979.
- À partir de 1990 environ, la logique introduite par le MRP s'étend progressivement à l'ensemble des fonctions de l'entreprise pour donner l'ERP.

6.2. Définition

Le terme PGI vient de "Progiciel de Gestion Intégrée" traduit en anglais par Enterprise Resource Planning (ERP). ERP est le terme le plus couramment utilisé.

Un ERP c'est un groupe de modules reliés à une base de données unique, permet de gérer l'ensemble des processus dans une entreprise en intégrant plusieurs fonctions de gestion : la gestion des ressources humaines, gestion de production, gestion de stock, gestion des commandes, autrement l'ERP représente la colonne vertébrale de l'entreprise.

6.3. Architecture technique

L'architecture d'un ERP se compose principalement d'un serveur ERP sur lequel est présente une base de données unique et disponible pour tous les employés. Elle implique l'utilisation de différents réseaux :

- Multi sites.
- Intranet, extranet.
- Serveur ERP à distance.

C'est la mise en réseau des données qui permettra leur gestion décentralisée.

Le déploiement d'un ERP est la plupart du temps client/serveur comme le décrit le schéma ci-dessous :

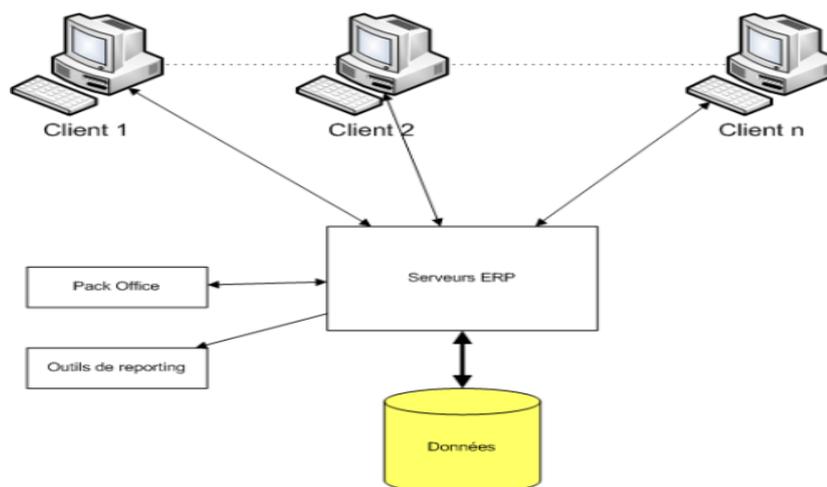


Figure 1.12 : Architecture technique d'ERP

6.4. Architecture modulaire des ERP's

Avant de découvrir l'architecture modulaire et ses modules, nous allons voir les principaux acteurs du marché des ERP (au niveau mondial).

6.4.1. Les principaux éditeurs d'ERP

On distingue deux types d'ERP : les ERP propriétaires, édités par des sociétés, ce qui implique l'achat d'une licence, et les ERP open source qui sont « gratuits ».

Parmi les principaux ERP propriétaires on peut citer : SAP, ORACLE-Peoplesoft, SSA GLOBAL, Geac, SAGE.

Parmi les ERP gratuits on peut citer : Odoo 1^{er} des ERP gratuits, Dolibarr, openConcerto, Neogia, OFBIz.

6.4.2. Architecture modulaire

Un ERP est constitué d'un ensemble de modules qui fonctionnent les uns avec les autres. Cela est possible grâce à :

- L'utilisation d'une base d'informations unique.
- La mise en réseau des différents postes de travail.
- La compatibilité garantie entre les différents modules.

Un ERP est modulaire dans le sens où il est possible de n'avoir qu'une ou plusieurs applications, en même temps, ou peu à peu. Les applications modulaires telles que les ERP

permettent d'être sûr de la compatibilité des modules entre eux, ils s'imbriquent comme des blocs de Lego et fonctionnent ensemble (pas de vérification de compatibilité à effectuer)

Voici un exemple d'architecture modulaire qui tend à représenter tous les ERP :

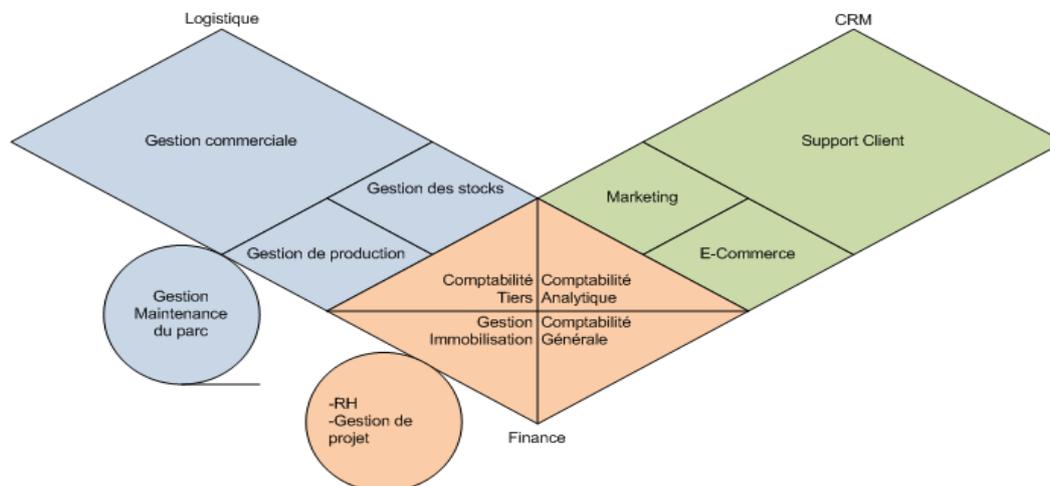


Figure 1.13 : Architecture modulaire d'ERP

Pour notre travail nous allons choisir l'ERP open source Odoo, on va le détailler par la suite. ⁴

6.5. Pourquoi un ERP open source

- Les entreprises utilisent les ERPs open source pour les économies de licence d'installation.
- L'ERP open source garantit l'accès aux informations en temps réel.
- L'ERP open source séduit les petites et moyennes entreprises (PME) et les très petites entreprises (TPE) pour disposer d'un outil de gestion complet au meilleur coût en leur apportant rapidement un vrai bénéfice en termes de compétitivité.

6.6. Caractéristiques des ERP

- Permet d'assurer la cohérence et l'homogénéité des informations
- Permet d'éviter la redondance d'information entre différents SI de l'entreprise.
- La mise à jour des données est effectuée en temps réel et transférer aux autres modules liés.

⁴<https://erp.ooreka.fr/>

- Un ERP est un outil multilingue et multidevise, il est donc adapté au marché mondial, en particulier aux multinationales.
- Permet d'accélérer l'optimisation des processus de gestion.
- Une meilleure maîtrise de production, vente, stock.
- Favorise la communication interne au sein de l'entreprise et garantie l'intégrité et l'unicité du système d'information.

7. Odoo

Odoo est anciennement connue sous le nom Open ERP, c'est un programme de gestion d'entreprise proposant une vaste suite de modules qui va permettre de centraliser les données. C'est un outil collaboratif permettant de gérer principalement le partage d'informations au sein de l'entreprise qui une fois implanté va activement solliciter tous les services de l'entreprise et une collaboration à travers cet outil va être mise en place. Odoo est l'ERP le plus évolutif et le plus utilisé au monde.

7.1. Historique des versions

Nom du logiciel	Version	Date de lancement	Changement significatif	Licence du logiciel
Tiny ERP	1.0	Février-05	Première publication	GNU GPL
	2.0	Mai-05		GNU GPL
	3.0	Septembre-05		GNU GPL
	4.0	Décembre-06		GNU GPL
OpenERP	5.0	Avril-09		GNU GPL
	6.0	Janvier-11	Première publication sous AGPL, premier client Web	GNU AGPL
	6.1	Février-12	Client web en Ajax, Fin du support pour le client riche (GTK+).	GNU AGPL
	7.0	22 décembre-12	Web client amélioré et facilité d'utilisation	GNU AGPL

Odoo	8.0	18 septembre-14	Support pour le CMS : construction de site internet, e-commerce, point de vente, vente et business intelligence	GNU AGPL
	9.0	01 octobre-15	Première publication des éditions Community sous licence LGPLV3 et Enterprise sous licence propriétaire.	GNU LGPL v3
	10.0	05 octobre-16	Fonctionnalités de fabrications renouvelées.	GNU LGPL v3
	11.0	Vers le 4 octobre-17	Studio améliorés, service de support améliorés, rapports améliorés.	GNU LGPL v3
	12.0	Octobre 2018		
	13.0	Octobre 2019		
	14.0	Octobre 2020		

Légende :

-  Anciennes versions ou fin de maintenance.
-  Anciennes versions avec maintenance étendue.
-  Version actuelle
-  Version en cours de développement.

Tableau 1.2 : Historique des versions d'Odoo.⁵

7.2. Odoo et ses principales fonctionnalités

Odoo est une suite d'applications de gestion dont les modules principaux sont communs à la majorité des entreprises commerciales et organisations à but non lucratif, ses principales fonctionnalités :

- Gestion de la relation client (CRM).
- Gestion des ventes.

⁵<https://fr.wikipedia.org/wiki/Odoo/>

- Gestion des stocks.
- Gestion de fabrication.
- Gestion des achats.
- Gestion des Ressources Humaines.
- Gestion de projet.
- Comptabilité / Finance.
- Contrôle de gestion / Comptabilité analytique.

7.3. Les caractéristiques d'Odoo

- Cohérent au niveau des données gérées (partage d'une base de données unique et commune).
- Plus de 4 500 Modules et fonctionnalités.
- Environ 2 000 000 d'utilisateurs à travers le monde.
- Structure modulaire permet d'ajouter de nouveaux modules pour étendre les fonctionnalités.



Figure 1.14 : logo d'OpenERP Odoo.

7.4. Architecture technique d'Odoo

D'un point de vue de l'architecture technique, Odoo est construit autour de trois composants :

- Serveur de base de données PostgreSQL : qui contient toutes les bases de données, chacune contenant toutes les données et la plupart des éléments de la configuration du système Odoo.
- Serveur d'application Odoo : qui contient toute la logique de l'entreprise et garantit l'exécution optimale d'Odoo. Une couche de serveur est dédiée à

la communication et à l'interface avec la base de données postgresQL, le moteur ORM.

- Le client WEB : Qui fournit l'interface utilisateurs, il exécute dans un navigateur web en tant qu'application javascript. Le client envoie une requête au serveur, récupère les données et affiche le résultat (par exemple une liste de clients) de différentes manières (sous forme de formulaires, listes, calendriers, ...). Lors d'action de l'utilisateur, il envoie des requêtes pour modifier les données sur le serveur.

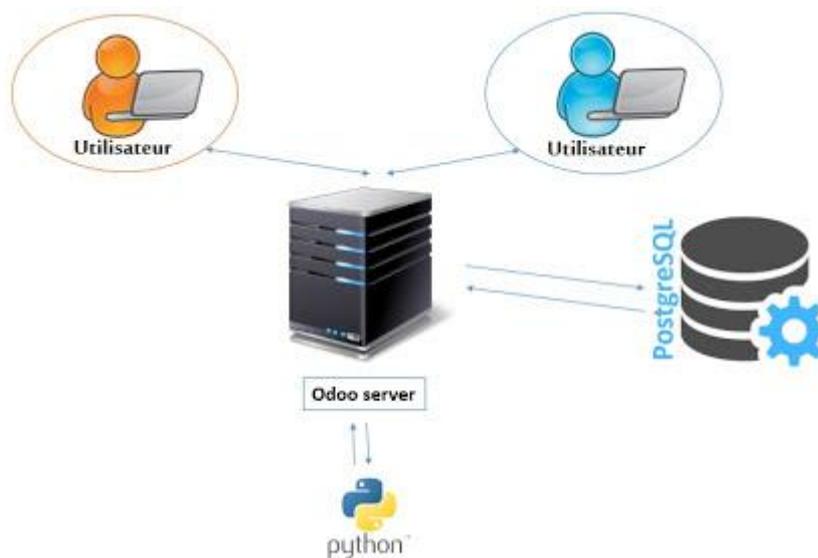


Figure 1.15 : Architecture trois tiers d'Odoo.

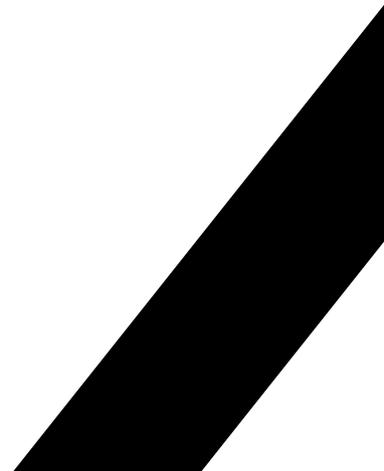
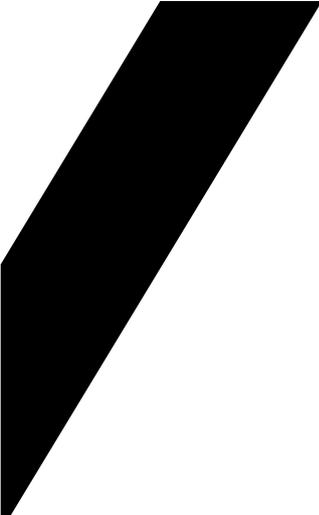
7.5. Avantages d'Odoo

- ✓ Version open source gratuite.
- ✓ Logiciel personnalisable selon vos besoins.
- ✓ Interface web très compétitive.
- ✓ Amélioration de l'activité et de la productivité.
- ✓ Gain de temps quotidien.
- ✓ De nombreuses langues disponibles.
- ✓ Service de haute qualité et reconnu.
- ✓ Les coûts d'intégration les plus faibles.

8. Conclusion

L'utilisation des systèmes d'information est devenue ces dernières années un atout incontournable pour les entreprises, ceci nécessite l'utilisation d'outils et de langages

permettant la création de ces systèmes de puis la conception jusqu'à l'implémentation à l'image du langage UML et de l'ERP Odoo détaillés dans ce chapitre.



Chapitre 02 : étude préalable et discussion des besoins

1. Introduction

Dans ce chapitre nous allons vous présenter l'entreprise ou nous allons faire notre stage projet de fin d'étude, par la suite l'étude de l'existant ainsi que la présentation de notre projet prenant en considération les besoins techniques, fonctionnelles et les résultats attendus pour finir par un cahier de charge.

2. Analyse de l'existant

L'entreprise utilise actuellement les logiciels standards(PCCOMPTA, DLGI plus le office world et Excel) principalement pour la gestion des flux et la communication entre les départements (production, RH, stock, finances...), ce qui laisse une grande lacune dans le système qui est la gestion de la production incluant les prévisions, la gestion des ventes et des achats, et aussi le « reporting » ; qui joue un rôle important dans la documentation ; la traçabilité des flux d'information et l'archivage.

Dans le cadre de la mise à niveau du système d'information, et des processus métier AQUATEC a décidé de remplacer le logiciel existant par un ERP à fin d'améliorer ses performances en termes de traçabilité et de sauvegarde des flux d'informations et pour le calcul des prévisions des demandes clients et la planification de la production pour aider les responsables à prendre les décisions rapidement et efficacement.

3. Présentation du projet

3.1. Motivation de projet

Avec la croissance illimitée du secteur de l'industrie, maintenir à la fois la productivité et l'efficacité est un besoin crucial pour chaque entreprise, là où la perte de temps, de ressources ou d'efforts est très coûteuse et diminue les bénéfices.

Pour éviter une telle perte, chaque entreprise devrait apprendre à gérer ses activités et ses informations de manière efficace et fonctionnelle, c'est pourquoi de nombreuses entreprises utilisent les ERP.

L'ERP est un logiciel de gestion des processus d'entreprise qui permet aux organisations d'utiliser un système d'applications intégrées pour gérer un processus d'entreprise et automatiser diverses fonctions.

3.2. Problématique

La problématique dans notre projet consiste à:

- Optimiser des flux.
- Optimiser des ressources lors de la production.
- Planifier de la production (ventes, achats, prévisions, stocks).

3.3. Description du contexte

Ce projet consiste à concevoir un système d'information pour l'entreprise afin d'améliorer ses performances en termes de traçabilité et de sauvegarde des flux d'informations. Nous utilisons le langage UML afin de modéliser les différents modules à développer. L'implémentation des modules se fera avec l'open ERP ODDO notamment le module de gestion de la production.

3.4. Objectifs de projet

- Faciliter l'acquisition et l'archivage des données de l'entreprise (Délocaliser l'information).
- Améliorer les performances du système de production par le développement d'un système d'aide à la décision de gestion de production.

4. Démarche de développement

Pour le développement de notre ERP on a choisi la démarche incrémentale qui consiste à décomposer les besoins du système en sous-systèmes (modules); telle que chaque composant accomplir une fonctionnalité bien définie qui réponde à l'exigence.

4.1. Etude comparative

4.1.1. Model en cascade

Dans l'approche en cascade "The Waterfall», tout le processus de développement de logiciels est divisé en phases distinctes. Dans ce modèle, le résultat d'une phase sert généralement d'entrée pour la phase suivante de manière séquentielle.

➤ Les avantages :

- ✓ Simple et facile à comprendre et à utiliser
- ✓ Facile à gérer en raison de la rigidité du modèle.
- ✓ Les phases sont traitées et achevées une à la fois.
- ✓ Étapes bien comprises.

➤ Les inconvénients :

- ✓ Aucun logiciel fonctionnel n'est produit avant la fin du cycle de vie.
- ✓ Les risques et les incertitudes sont élevés.
- ✓ Pas un bon modèle pour les projets complexes et orientés objet.
- ✓ Il est difficile de mesurer les progrès par étapes
- ✓ Ne peut pas s'adapter à l'évolution des besoins.

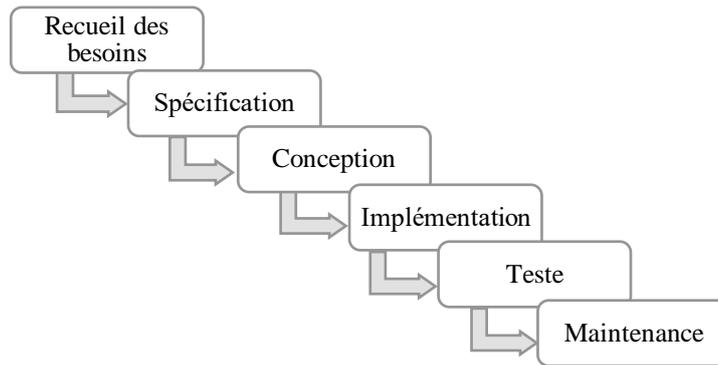


Figure 2.1 : processus en cascade.

4.1.2. Model en V

Ce modèle se concentre sur l'exécution des processus de manière séquentielle, comme le modèle de la cascade, mais en accordant plus d'importance aux tests.

➤ Les avantages :

- ✓ Un modèle très discipliné et les phases sont réalisés une à une.
- ✓ Il fonctionne bien pour les petits projets où les besoins sont très bien compris.
- ✓ Il est simple et facile à comprendre et à utiliser.
- ✓ Facile à gérer en raison de la rigidité du modèle.

➤ Les inconvénients :

- ☒ Risque et incertitude élevés.
- ☒ Pas un bon modèle pour les projets complexes et orientés objet.
- ☒ Mauvais modèle pour les projets longs et continus.
- ☒ Aucun logiciel fonctionnel n'est produit avant la fin du cycle de vie.

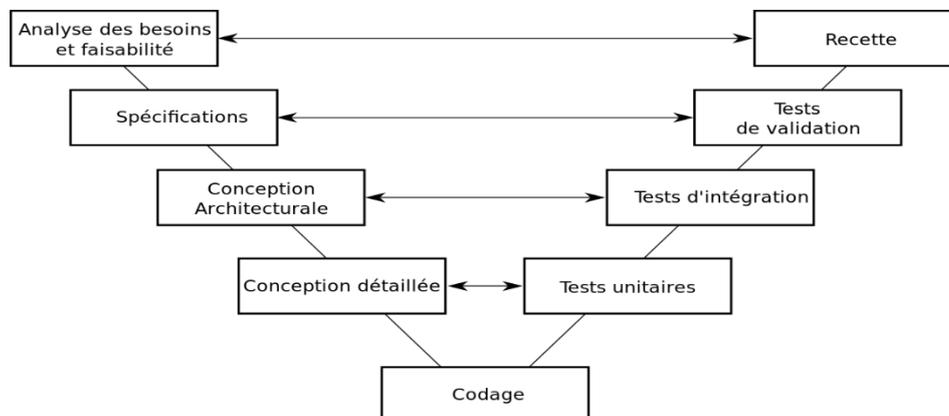


Figure 2.2 : processus en V.

4.1.3. Le modèle incrémental

Le modèle incrémental est un processus de développement de logiciels dans lequel les exigences sont décomposées en plusieurs modules autonomes du cycle de développement de logiciels. Le développement incrémental se fait par étapes, depuis la conception de l'analyse, la mise en œuvre, les tests/vérifications, la maintenance.

Chaque itération passe par les phases d'exigences, de conception, de codage et de test. Et chaque version suivante du système ajoute des fonctions à la version précédente jusqu'à ce que toutes les fonctionnalités conçues aient été mises en œuvre.

➤ Les avantages :

- ✓ Le logiciel sera généré rapidement pendant le cycle de vie du logiciel
- ✓ Il est souple et moins coûteux de modifier les exigences et le champ d'application
- ✓ Tout au long des étapes de développement, des changements peuvent être apportés
- ✓ Les erreurs sont faciles à identifier

➤ Les Inconvénients :

- ☒ Il faut une bonne planification pour concevoir
- ☒ Des problèmes peuvent survenir en raison de l'architecture du système en tant que telle, les exigences n'étant pas toutes collectées à l'avance pour l'ensemble du cycle de vie du logiciel
- ☒ Chaque phase d'itération est rigide et ne se chevauche pas
- ☒ La correction d'un problème dans une unité nécessite la correction dans toutes les unités et prend beaucoup de temps

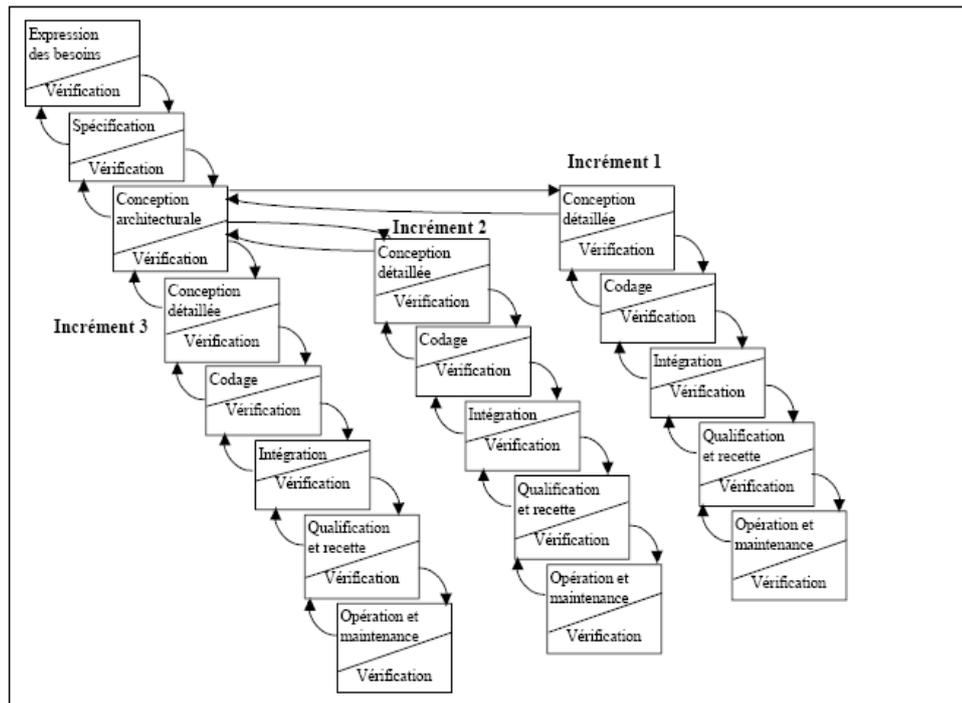


Figure 2.3 : processus incrémental.

4.1.4. Model spirale

Ce modèle est une combinaison du modèle de processus de développement itératif (modèle incrémental) et du modèle de développement linéaire séquentiel (cascade), qui met fortement l'accent sur l'analyse des risques. Il permet des rejets incrémentiels du produit ou un raffinement incrémentiel à chaque itération autour de la spirale.

➤ Les avantages :

- ✓ Il est possible de s'adapter à l'évolution des besoins.
- ✓ Permet une utilisation étendue des prototypes.
- ✓ Les utilisateurs voient le système à un stade précoce.
- ✓ Le développement peut être divisé en petites parties et les parties à risque peuvent être développées plus tôt, ce qui contribue à une meilleure gestion des risques.

➤ Les Inconvénients :

- ☒ Le processus est complexe
- ☒ La fin du projet peut ne pas être connue à l'avance.

☒ Le grand nombre d'étapes intermédiaires exige une documentation excessive.

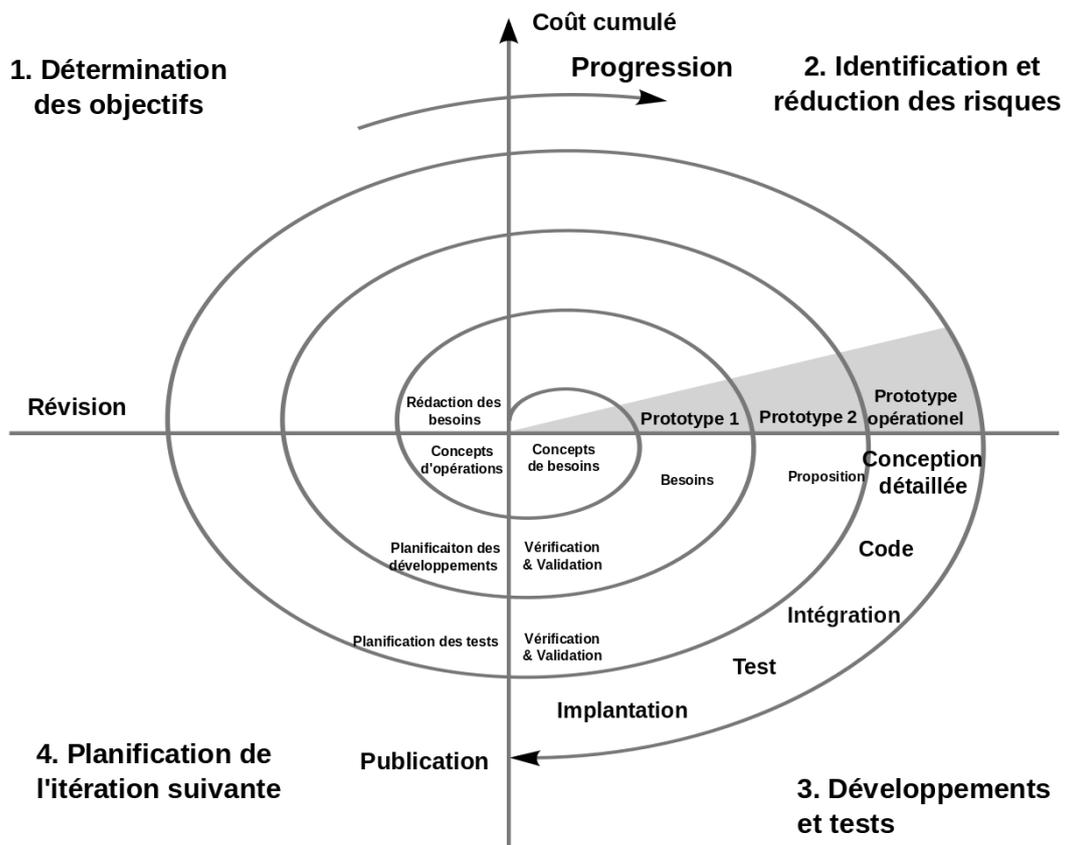


Figure 2.4 : processus spirale

5. Présentation de l'entreprise

5.1. Présentation du Groupe KHERBOUCHE

C'est un groupe présent dans l'agro-industrie, l'ingénierie de l'eau et la réalisation hydraulique englobe neuf filiales, qui sont actives sur l'ensemble du territoire national. D'une complémentarité et interopérabilité avérées, les filiales du groupe chacune dans sa spécialité. Le groupe emploie un effectif de plus de 2000 employés, toute spécialité confondue.

➤ Les filiales du groupe :

1. INTER ENTREPRISE.
2. AGRO INDUSTRIE.
3. AGRODEEL.
4. AQUATEC.
5. ELALF.
6. CANAL PLAST.
7. HUILE D'OLIVE Tlemcen.
8. ARBOR ACRES Algérie.
9. INTERPREFA.

5.2. Présentation de AQUATEC Biotechnologie

AQUATEC biotechnologie l'une des neuf filiales du groupe KHERBOUCHE, créée en 2000, spécialisée dans l'installation et l'exploitation des stations de traitement des eaux principalement et d'autres activités secondaires.

5.2.1. Activité principale

La production industrielle (assemblage, réalisation) des stations de traitement des eaux :

- Station de déminéralisation.
- Station de potabilisation.
- Station d'épuration
- Equipement de piscine.

5.2.2. Activités secondaires

- La télégestion et supervision.
- Les énergies renouvelables.
- La détection des fuites d'eau et de gaz.
- Les pompes de chantier, assainissement et aérateurs.



Figure 2.5: Station de potabilisation



Figure 2.6 : poste locale de télégestion SOFREL

5.2.3. Fiche technique de l'entreprise

Nom de l'entreprise	AQUATEC biotechnologie
Statut juridique	SARL
Date de création	Créée en l'an 2000
Capital social	60.000.000
Activité	-traitement des eaux. - télégestion et supervision. - énergies renouvelables. - détection des fuites d'eau et de gaz. - pompes de chantier, assainissement et aérateurs.
Effectif	35 personnes
Contact	Tél : 043 27 23 62. Fax : 043 27 43 44. Email : aquatec@groupekherpouche.com
Siège social	Z.I. Chetouane, Desserte N°3 13000 Tlemcen.

Logo d'entreprise



Tableau 2.1 : Carte d'identité de l'entreprise AQUATEC Biotechnologie.

5.2.4. Organisation de l'entreprise

La SARL AQUATEC Biotechnologie est organisée comme suit :

- LA direction.
- Service de l'administration générale.
- Service commercial et Marketing.
- Service technique.

6. Analyse des flux

6.1. Définition

L'analyse des flux dans une entreprise est une technique d'analyse qui cherche à comprendre comment le flux circule pour objectif de mettre en place une démarche de progrès et construire une carte des flux lisible et documentée.

6.2. Types du flux

- Flux physique : se compose de matière première, produits semi finis et produits finis.
- Flux d'information : l'information peut être sur papier ou informatique.
- Flux financier : coût de : stock, de production et de contrôle.

6.3. Mode du flux

- Flux poussé : faire une prévision avant de produire, ce mode est proposé par le MRP pour avoir un stock MTS.
- Flux tiré : attendre la commande pour produire pour avoir un MTO.
- Mixe des deux : pour avoir un ATO.

6.4. Analyse des flux au sein de l'entreprise

Dans cette partie on va voir les flux échanger dans chaque activité de l'entreprise.

- Installation des stations de traitement des eaux :

Le client envoie une commande (analyse d'eau+le débit) au commercial de l'entreprise et ce dernier envoie cette commande au bureau d'étude des commandes.

Le bureau d'étude contacte le fournisseur et lui envoie l'analyse d'eau et le débit pour qu'il puisse déterminer les équipements nécessaires pour la réalisation de la station demandée par le client.

L'entreprise établit le bon de commande et fait une négociation avec le fournisseur concernant les équipements de la station par la suite le fournisseur envoie le bon de commande finale.

L'entreprise établit une fiche technique et financière du produit et les envoie au client.

Si le client valide sa commande l'entreprise fait la production industrielle (assemblage) de la station au bout de 3 mois.

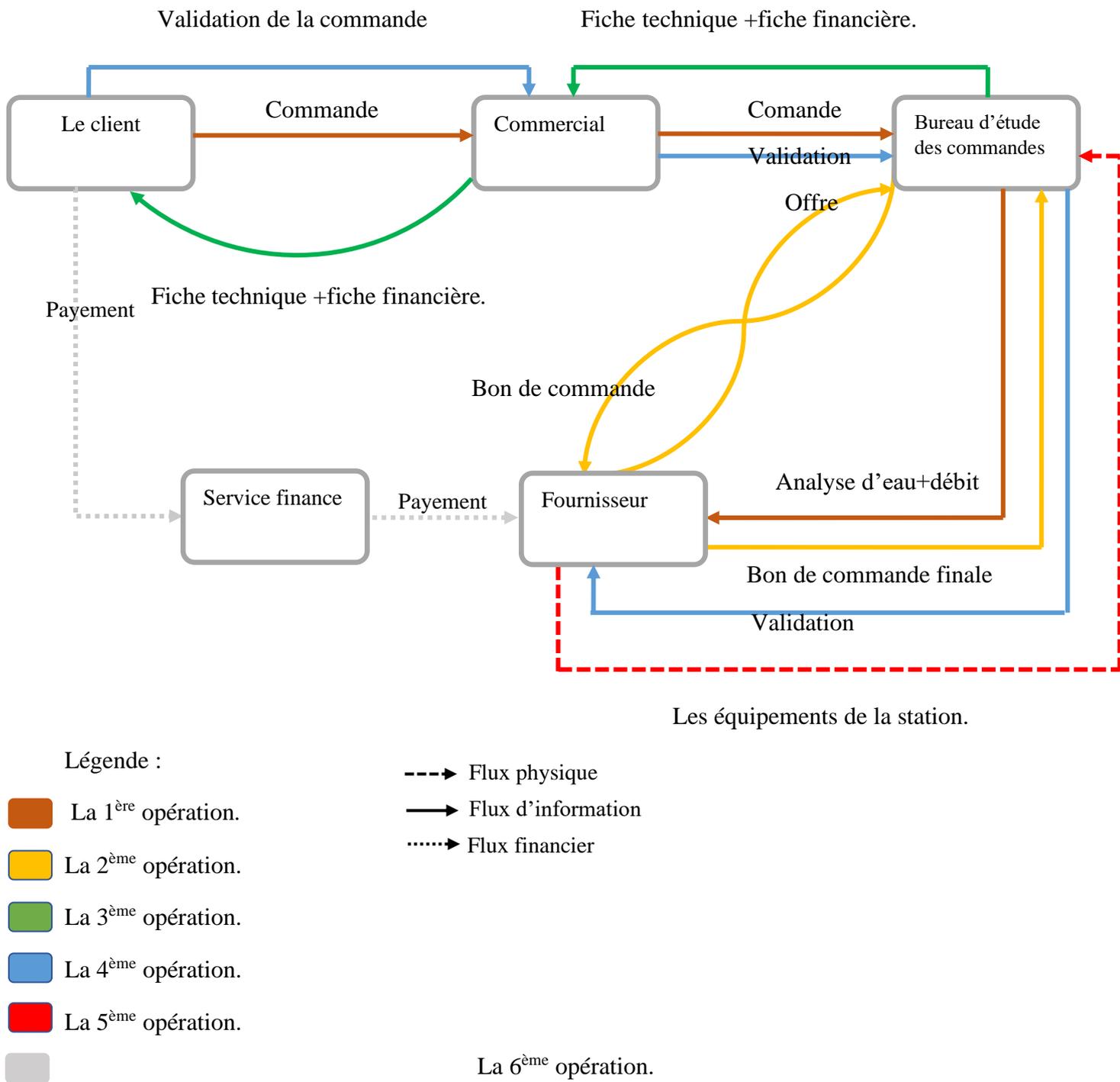


Figure 2.7 : schéma récapitulatif des flux.

- Installation des stations de traitement des eaux :
- Résumé :

Flux d'information	Flux physique	Flux financier
<ul style="list-style-type: none">- Commande du client.- Fiche produit.- Offre du fournisseur.- Bon de commande.- Fiche technique du produit.- Fiche financière du produit.- Validation.	<ul style="list-style-type: none">- Les équipements de la station.- Les panneaux solaires	<ul style="list-style-type: none">- Les factures

Tableau 2.2 : les flux au sein de l'entreprise.

7. Analyse et discussion des besoins

7.1. La notion de besoin

Le besoin en littérature est défini comme la nécessité d'avoir quelque chose là où son existence est nécessaire. Mais ces besoins peuvent être différents d'une situation à l'autre, par exemple les besoins de l'être humain sont clairement expliqués dans la pyramide de Maslow, alors que les besoins des systèmes d'information se diffèrent.

Les besoins en systèmes d'information peuvent être décomposés en deux grandes parties :

- Les besoins fonctionnels :

Définit une fonction d'un système ou de son composant. C'est une description du service que le logiciel doit offrir.

Une fonction n'est rien d'autre que des entrées dans le système logiciel, son comportement et ses sorties. Il peut s'agir d'un calcul, d'une manipulation de données, d'un processus métier, d'une interaction avec l'utilisateur...

Les exigences fonctionnelles du logiciel nous aident à saisir le comportement prévu du système.

- Les besoins non-fonctionnels (techniques) :

Spécifie l'attribut de qualité d'un système logiciel. Ils jugent le système logiciel sur la base de la réactivité, de la convivialité, de la sécurité, de la portabilité et d'autres normes non fonctionnelles qui sont essentielles à la réussite du système logiciel.

Les besoins non-fonctionnels permettent d'imposer des contraintes ou des restrictions sur la conception du système.

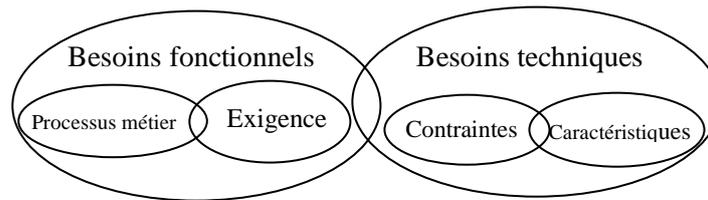


Figure 2.8 : schéma récapitulatif des besoins.

C'est pourquoi on a réalisé une étude pour recueillir ces besoins, les analyser et en sortir avec une vision claire de ce que notre client veut vraiment (un cahier de charges).

7.2. Expression des besoins

Après les nombreuses réunions et la longue discussion avec toutes les parties impliquées dans le système (MOA), nous (MOE) avons finalement pu identifier les besoins de l'entreprise en fonctionnement et contraints.

7.2.1. Les besoins fonctionnels

- Gestion d'achats et ventes.
- Gestion de production industrielle.
- Etablir les documents et les factures (bon de commandes, bon d'achats, fiche technique, ...).

7.2.2. Les besoins techniques

7.2.2.1. Les caractéristiques

Pour définir les caractéristiques techniques de notre progiciel on a utilisé la norme ISO/IEC 25010 (SQuaRE) définissant un langage commun pour modéliser les qualités d'un logiciel.

- Adéquation fonctionnelle :

Les modules développés doivent respecter les spécifications fournis par la MOA. Le système doit être facilement paramétrable et disposer d'interfaces plus paramétré selon le besoin.

- Efficacité des performances

Cette caractéristique représente la performance par rapport à :

- La quantité de ressources utilisées.
- Le temps de réponse.

Dans des conditions données.

- Compatibilité :

Le système doit être capable d'échanger des informations avec d'autres systèmes, et/ou remplir les fonctions requises tout en partageant le même environnement matériel ou logiciel et en gardant toujours la coexistence et l'interopérabilité de tous les systèmes....

- Utilisabilité :

Le système doit être utilisé par des utilisateurs déterminés (accessibilité) pour atteindre des objectifs précis avec efficacité, efficience et satisfaction dans un contexte d'utilisation déterminé.

- Fiabilité :

Le système doit remplir des fonctions déterminées dans des conditions déterminées et pendant une période déterminée. D'où notre système doit être :

- Mature : répond aux besoins attendus dans des conditions normales de fonctionnement.
- Disponible : opérationnel et accessible lorsqu'il est nécessaire à l'utilisation.
- Récupérable : la possibilité de récupérer les données directement affectées et rétablir l'état souhaité du système en cas d'interruption ou de panne.

- Sécurité :

Degré de protection des informations et des données de système, telle que les personnes ou d'autres produits ou systèmes aient le degré d'accès aux données approprié à leurs types et niveaux d'autorisation.

- Maintenabilité :

Le système peut être modifié pour l'améliorer, le corriger ou l'adapter aux changements de l'environnement et des exigences avec l'implémentation d'un code simple et compréhensible.

- La portabilité :

Le système peut être transféré d'un matériel, d'un logiciel ou d'un autre environnement opérationnel ou d'utilisation à un autre.

Donc en veillant toujours à ce que notre système soit adaptable, installable et remplaçable.

7.2.2.2. Contraint liée au projet

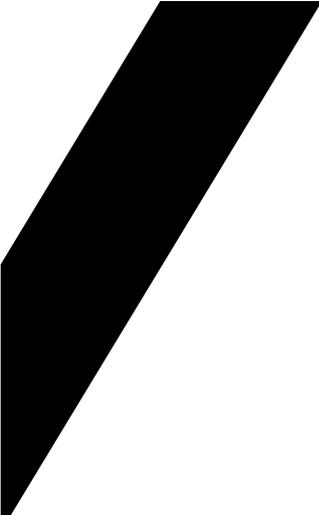
Lors de la planification de notre projet, nous avons envisagé certains obstacles qui pourraient affecter la réalisation de notre projet ou la qualité de notre travail.

Précisant certaines de ces contraintes :

- Le temps : notre projet est dû à 20/20, ce qui nous laisse environ trois mois de travail et toute anomalie ou imprécision peut retarder le projet.
- Le budget : concernant la recherche, le développement et la mise en œuvre.
- Compétences techniques : un manque d'expertise dans le domaine informatique.
- Insertion des données : dans la phase de mise en œuvre, une base de données devrait être remplie d'informations et d'archives passées qui peuvent être considérées comme un travail très fatigant.

8. Conclusion

Après avoir établi une étude bien détaillée de notre projet par la détermination exhaustive de cahier de charge ; nous pouvons à présent passer à la phase « conception ».



Chapitre 03 : Conception du module gestion de la production

1. Introduction

Après avoir vu la situation actuelle de l'entreprise ainsi que les besoins et contraintes fonctionnels, il est temps de commencer la conception, qui est considérée comme une phase d'importance cruciale pour la réalisation de notre système.

La conception vise à traduire les besoins fonctionnels et les contraintes issues du cahier des charges et de la spécification des exigences ; cette étude nous permet aussi d'avoir une bonne réflexion sur le fonctionnement du système à travers un langage visuel simple afin qu'il soit compris par toutes les parties intervenantes.

2. Modélisation structurale du système

2.1. Diagramme de classe

Le diagramme de classe est utilisé non seulement pour visualiser, décrire et documenter les différents aspects statiques d'un système ; mais aussi pour construire le code exécutable du logiciel

2.1.1. Les règles de Gestion

Le diagramme de classe du système étudié est basé sur les règles de gestion suivantes :

- Un employé est titulaire d'aucun ou un et un seul compte.
- Un compte appartient à un et un seul employé.
- Un employé peut exécuter une ou plusieurs tâches.
- Une tâche peut être affectée à un seul employé.
- Un employé est affecté à un seul service.
- Un service contient plusieurs employés.
- Un service peut accomplir une ou plusieurs tâches.
- Une tâche peut être accomplie dans un seul service.
- Un service peut contacter plusieurs fournisseurs.
- Un fournisseur peut être contacté par un seul service.
- Un fournisseur peut livrer plusieurs composants.
- Un composant est livré par un seul fournisseur.
- Un article peut être composé de plusieurs composants.
- Un composant appartient à un seul article.
- Un client peut établir une ou plusieurs commandes.
- Une commande appartient à un seul client.

- Un client possède un seul registre client.
- Une commande contient un ou plusieurs articles.
- Un article peut faire partie d'une ou de plusieurs commandes.
- Une commande a une seule fiche technique.
- Une fiche technique est affectée à une seule commande.
- Une commande a une seule fiche financière.
- Une fiche financière appartient à une seule commande.

2.1.2. Dictionnaire de données :

Nom de la classe	Attributs	Désignation	Type
Employé	Id_emp	Un code d'identification d'un employé.	
	Nom	Nom de l'employé.	
	Prénom	Prénom de l'employé.	
	Adresse	Adresse de l'employé	
	Id_cmp	Identification de compte d'utilisateur	
Compte utilisateur	Mot_passe	Mot de passe de compte utilisateur	
	Adresse	Adresse de compte utilisateur.	
Tâche	Nom_tâche	Le nom de la tâche faire.	
	Des_tâche	Désignation de la tâche.	
Article	Id_art	Identification d'article.	
	Ref_art	Reference d'article	
	type_art	Type article.	
Type_article	Prix_art	Prix d'article.	
	Id_co	Identification de composant.	
	Nom_co	Nom de composant.	
Composant	Prix_co	Prix de composant.	
	Code_ser	Code du service.	
	Nom_ser	Nom de service.	
Service	Id_fr	Identification du fournisseur.	
	Nom_fr	Nome du fournisseur.	
	Contact_fr	Le contact du fournisseur.	
Fournisseur	Id_cl	Identification du client.	
	Nom_cl	Nom du client.	

Client	Prénom_cl	Prénom du client.
	Contact_cl	Cordonnées du client.
Registre client	Ref_reg	Référence du registre.
	Details_cl	Details du client.
	Id_cmd	Identification de la commande.
	Nom_cmd	Nome de la commande.
Commande	Type_cmd	Type de la commande.
	Date_début	Date de réception de la commande.
	Date_fin	Date de livraison de la commande.
	Id_tech	Identification de la fiche technique.
Fich technique	Id_cmd	Identification de la commande traitée dans la fiche technique.
	List_cmp	List des composants pour la commande.
	Id_ff	Identification de la fiche financière.
Fiche financière	Montant_cmd	Montant totale de la commande traitée.

Tableau 3.1 : dictionnaire de donnée.

2.1.3. Diagramme de classe de système

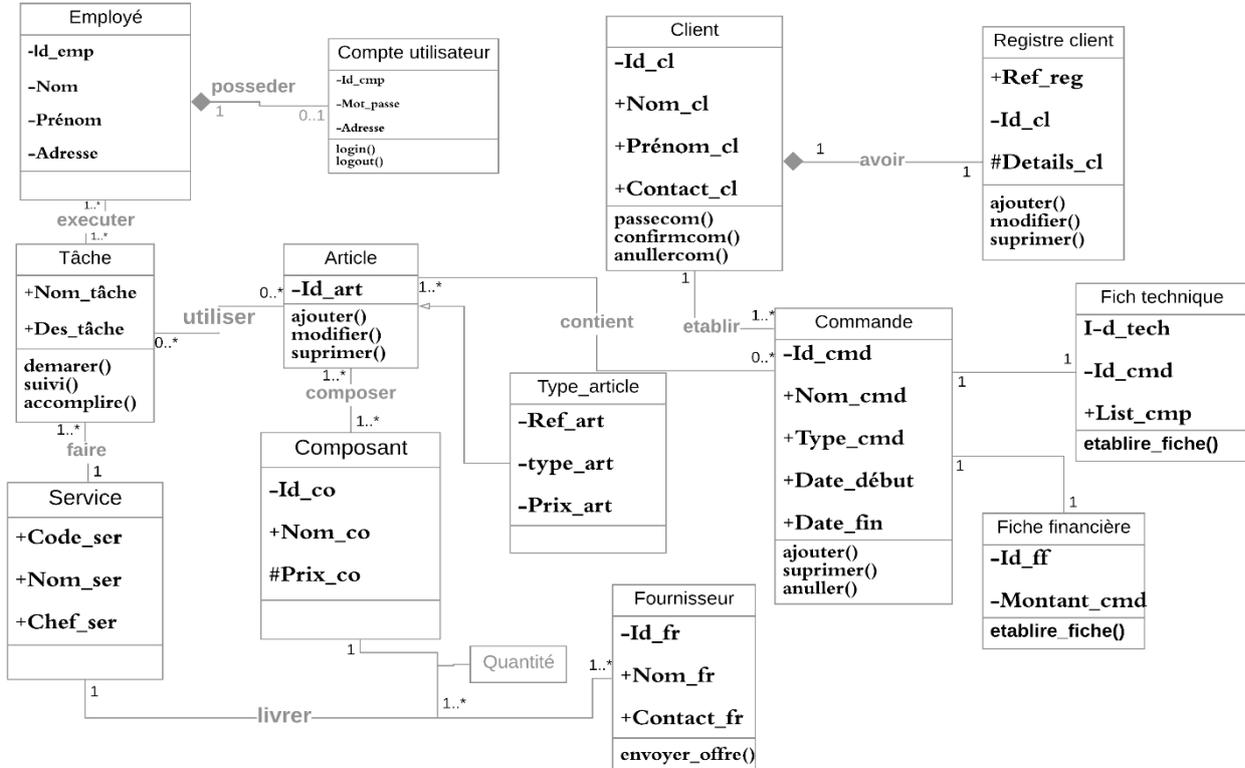


Figure 3.1: diagramme de classe.

3. Modélisation comportementale du système

Dans cette partie de la modélisation on va voir les diagrammes de cas d'utilisation aussi que celle de séquences.

3.1. Les diagrammes de cas d'utilisations des processus métiers de système

3.1.1. Administration générale

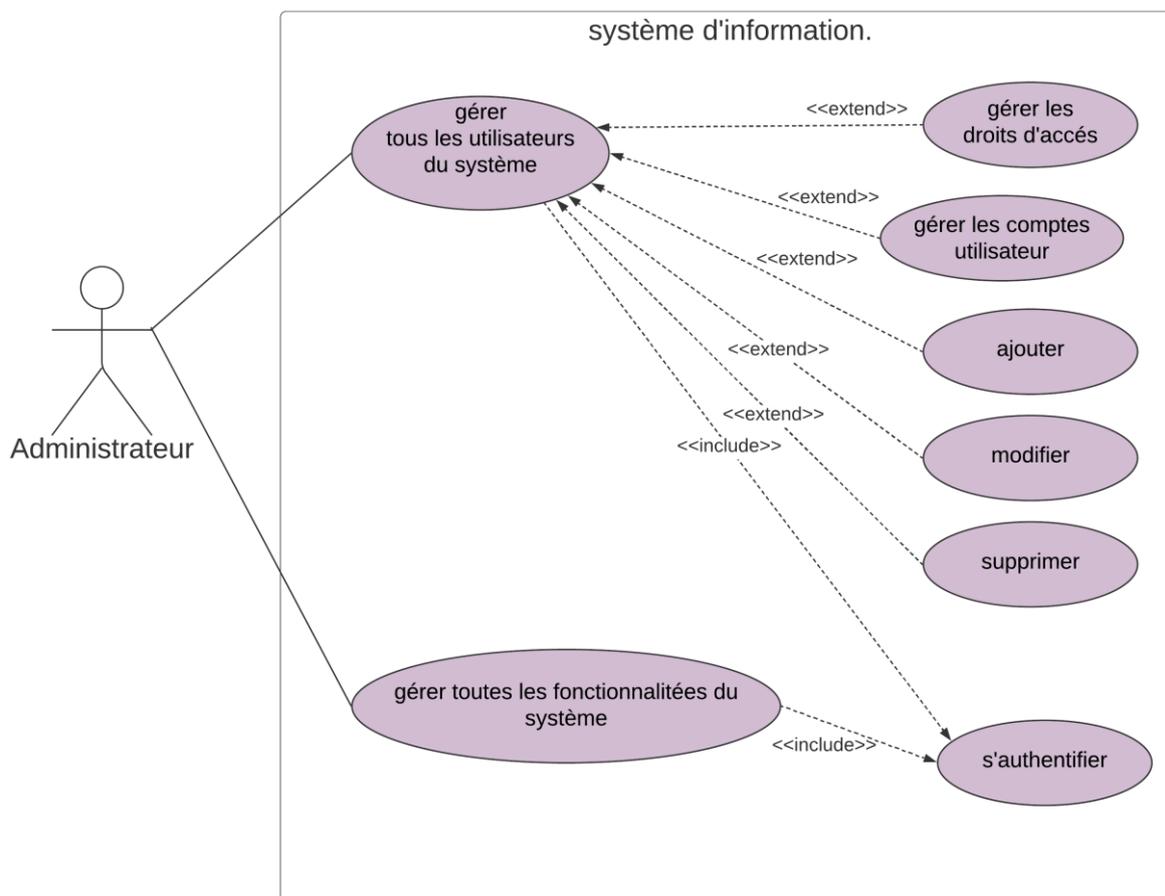


Figure 3.2 : diagramme de cas d'utilisation « gestion d'administration ».

➤ Les acteurs du système :

- Acteur administrateur : il a pour rôle d'assurer le fonctionnement du système d'information en assurant l'accessibilité des données et la sécurité des informations. L'administrateur a aussi la possibilité de gérer les droits d'accès des utilisateurs du système ainsi toutes ses fonctionnalités.

➤ Les scénarios fournis par ce système :

- Gérer les utilisateurs du système : L'administrateur gère les utilisateurs qui ont le droit d'accès au système ainsi que leur compte, il est aussi capable d'ajouter, modifier ou supprimer des utilisateurs.

- Gérer toutes les fonctionnalités du système : L'administrateur contrôle et gère toutes les fonctions existantes sur Odoo (modules, paramètres...).
- S'authentifier : L'authentification est obligatoire pour chaque utilisateur afin de pouvoir accéder au système.

3.1.2. Gestion des prévisions

Ce processus consiste à faire l'étude prévisionnelle à travers la réalisation d'un PIC (plan industriel et commercial), ce plan stratégique est celui que l'entreprise va utiliser pour planifier la production et les ressources.

➤ Exemple explicatif de PIC :

ventes	jan.	fév.	mars	avr.	mai	juin	juil.	août
prévisionnel	1000	1000	1000	1000	1000	1020	1020	1040
réel	1020	1020	1020					
écart	20	20	20					
écart en %	2	2	2					

production	jan.	fév.	mars	avr.	mai	juin	juil.	août
prévisionnel	980	1000	1020	1040	1040	1040	1040	1040
réel	960	980	980					
écart	-20	-20	-40					
écart en %	-2	-2	-4					

stock	jan.	fév.	mars	avr.	mai	juin	juil.	août
prévisionnel	500	460	420	420	460	480	500	500
réel	460	420	380					
écart	-40	-40	-40					
écart en %	-8	-8,7	-9,5					

6

Figure 3.3 : Exemple de PIC.

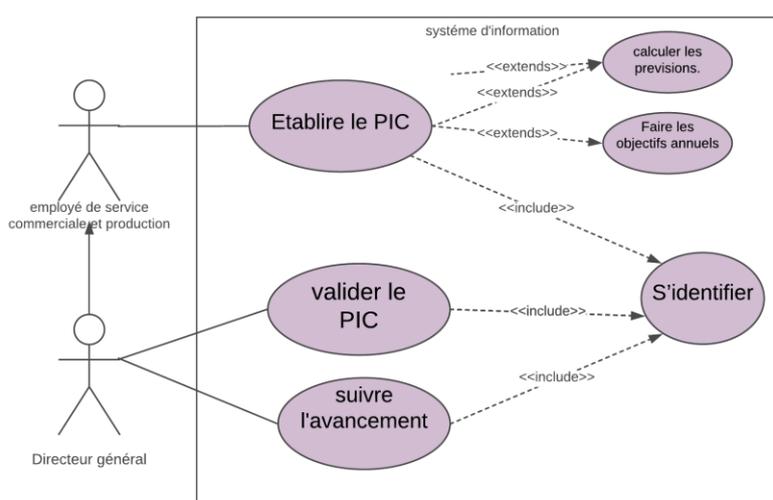


Figure 3.4 : diagramme de cas d'utilisation « gestion de prévision ».

⁶<http://ressources.auneg.fr/nuxeo/site/esupversions>

➤ Les scénarios fournis par ce système :

- Etablir le PIC : faire les prévisions des ventes, des approvisionnements en matières premières et des quantités à produire par famille de produit en se basant sur les commandes et les ventes des mois passés.
- Contacter le service de la production pour vérifier les ressources et la capacité de production.
- Valider le PIC pour passer au PDP.

3.1.3. Gestion de la production

La gestion de la production consiste à gérer l'ensemble des ressources (humaines, finances, matérielles...) afin de garantir une rentabilité et une qualité supérieure des livrables.

Pour cette raison l'entreprise établit un plan directeur de production (MPS master production Schedule) pour planifier la production par famille de produit, basant sur les résultats obtenus par le PIC.

➤ Exemple explicatif de PDP :

	1	2	3	4	5	6	7
prévisions de vente	10	40	60	80	90	100	100
com m andes fermes	70	40	30	10	4		
prévisionnel disponible	160	80	190	100	6	106	6
PDP (fin)			200			200	
PDP (début)		200			200		


zone ferme

Figure 3.5 : Exemple de PDP.

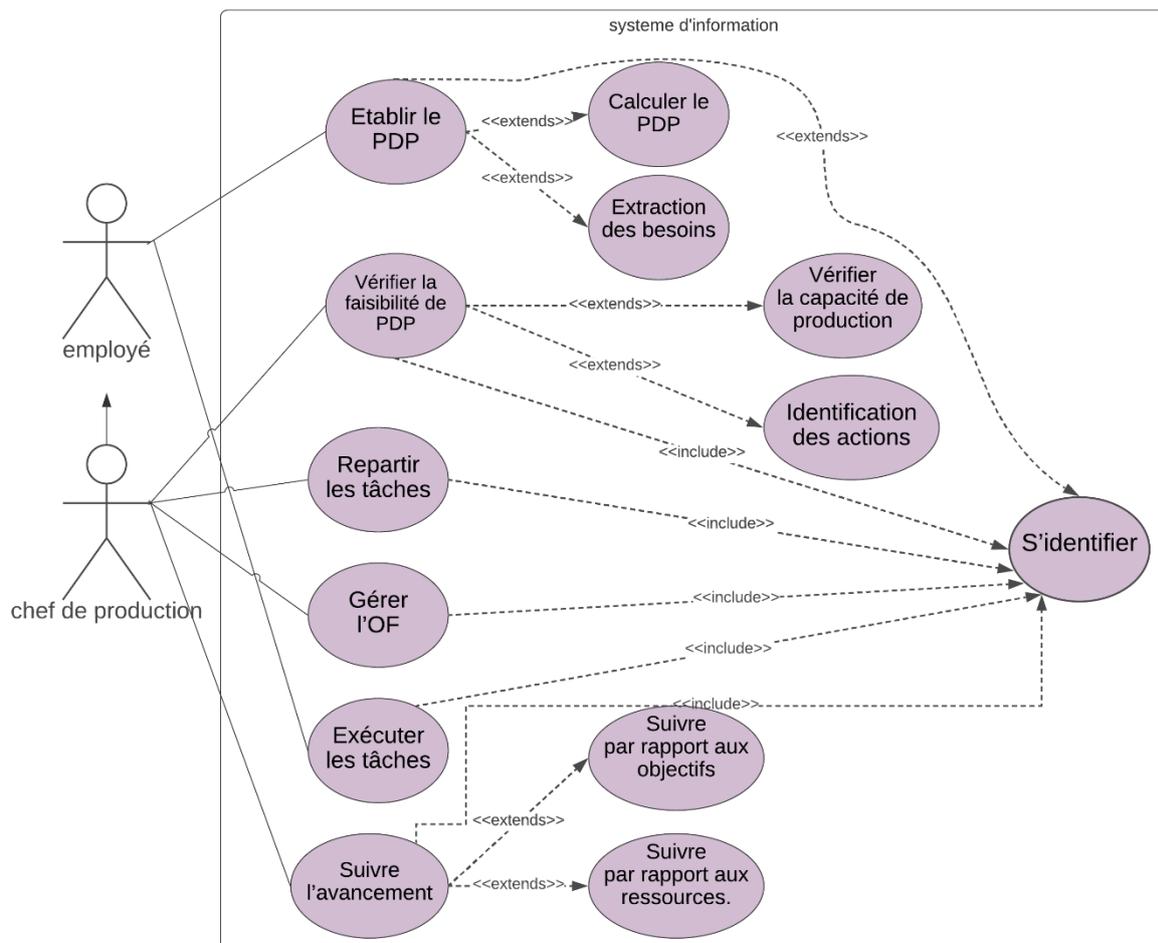


Figure 3.6 : diagramme cas d'utilisation « gestion de la production ».

➤ Les acteurs du système :

- L'employé : il est le carburant qui fait marcher l'entreprise ; sa mission est d'accomplir les tâches qui lui sont affectés.
- Chef de production : il est le responsable de la réalisation des bien et/ou des services, il est responsable de la planification et du suivi.

Les scénarios fournis par ce système :

- Etablir le PDP : définir les besoins en produits réel finis (besoin brut) pour la production en utilisant le PIC ; le PDP est plutôt l'ordre de fabrication quotidien.
- Vérifier la faisabilité du PDP en termes de capacité de production et de ressources disponible.
- Répartir les tâche quotidiennes (tâches a réalisé et tâches de maintenance).
- Gérer les ordres de fabrication c-à-dire crée, valider ou annulé l'OF.
- Suivre l'avancement par rapport aux délais et aux objectifs.

3.1.4. Gestion des achats et approvisionnement

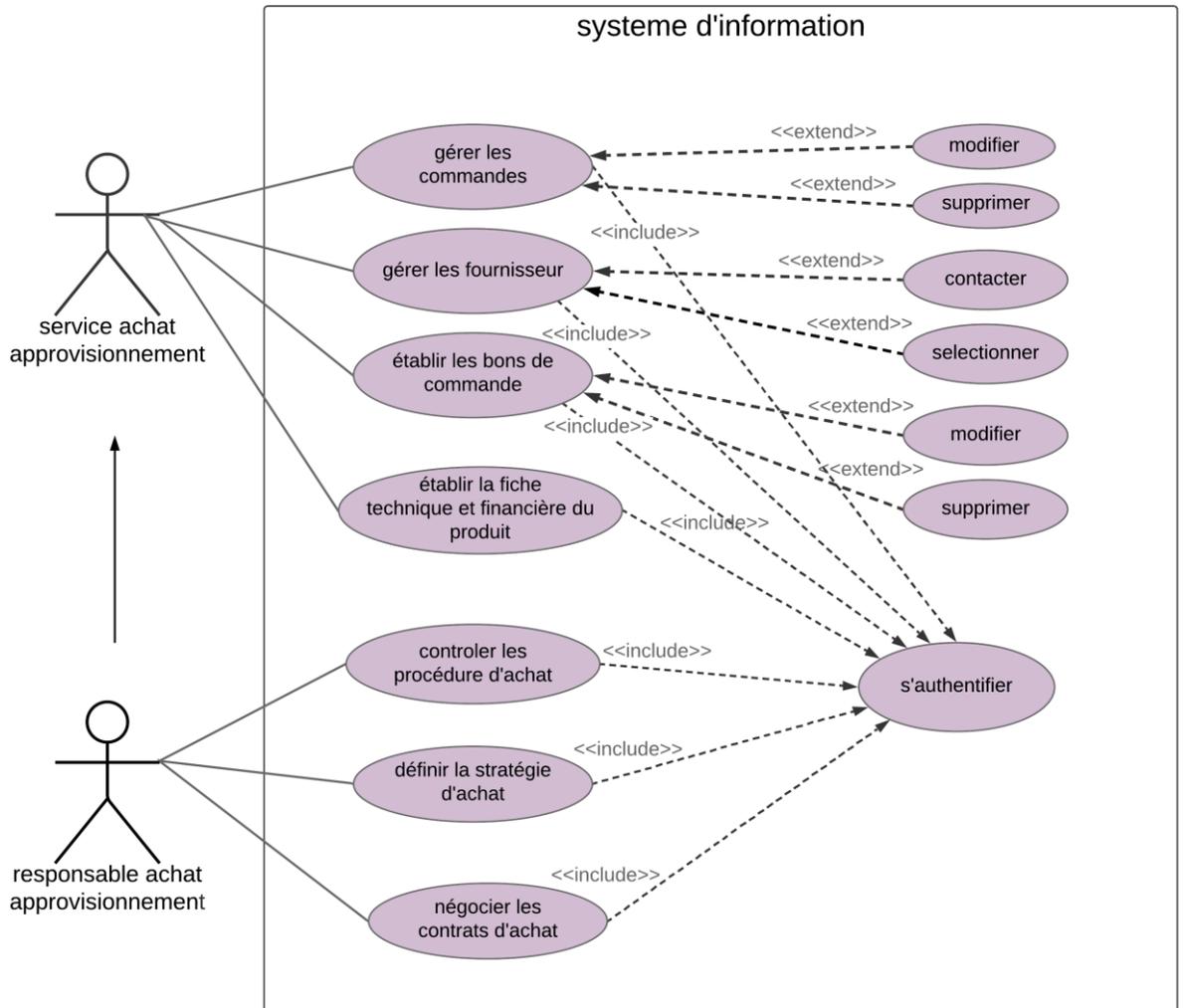


Figure 3.7 : diagramme cas d'utilisation « gestion des achats et approvisionnement ».

Les acteurs de ce système :

- Acteur service d'achat et approvisionnement : cet acteur établit et négocie les achats de tout produit ou service nécessaire à la production de l'entreprise. Il gère les commandes, les fournisseurs aussi établit les fiches techniques et financières des produits.
 - Acteur responsable d'achat et approvisionnement : c'est l'acteur qui définit la stratégie achat en collaboration avec la direction générale
- Les scénarios fournis par ce système :
- Gérer les commandes : consulter les commandes pour définir les achats nécessaires ainsi modifier ou supprimer selon le contact du client avec le commercial.

- Gérer les fournisseurs : contacter les fournisseurs pour l'achat et choisir le meilleur.
- Etablir les bons de commande : après le choix de fournisseur le service d'achat établit le bon de commande.
- Etablir la fiche technique et financière des produits : le service établit un document contient tous les équipements nécessaires pour la réalisation du produit ainsi les prix de ces équipements.
- Définir la stratégie d'achat : c'est définir comment l'entreprise fait ses achats.
- Négocier les contrats d'achat : la négociation est faite avec le fournisseur pour aboutir à des prix idéals.

3.1.5. Gestion commerciale

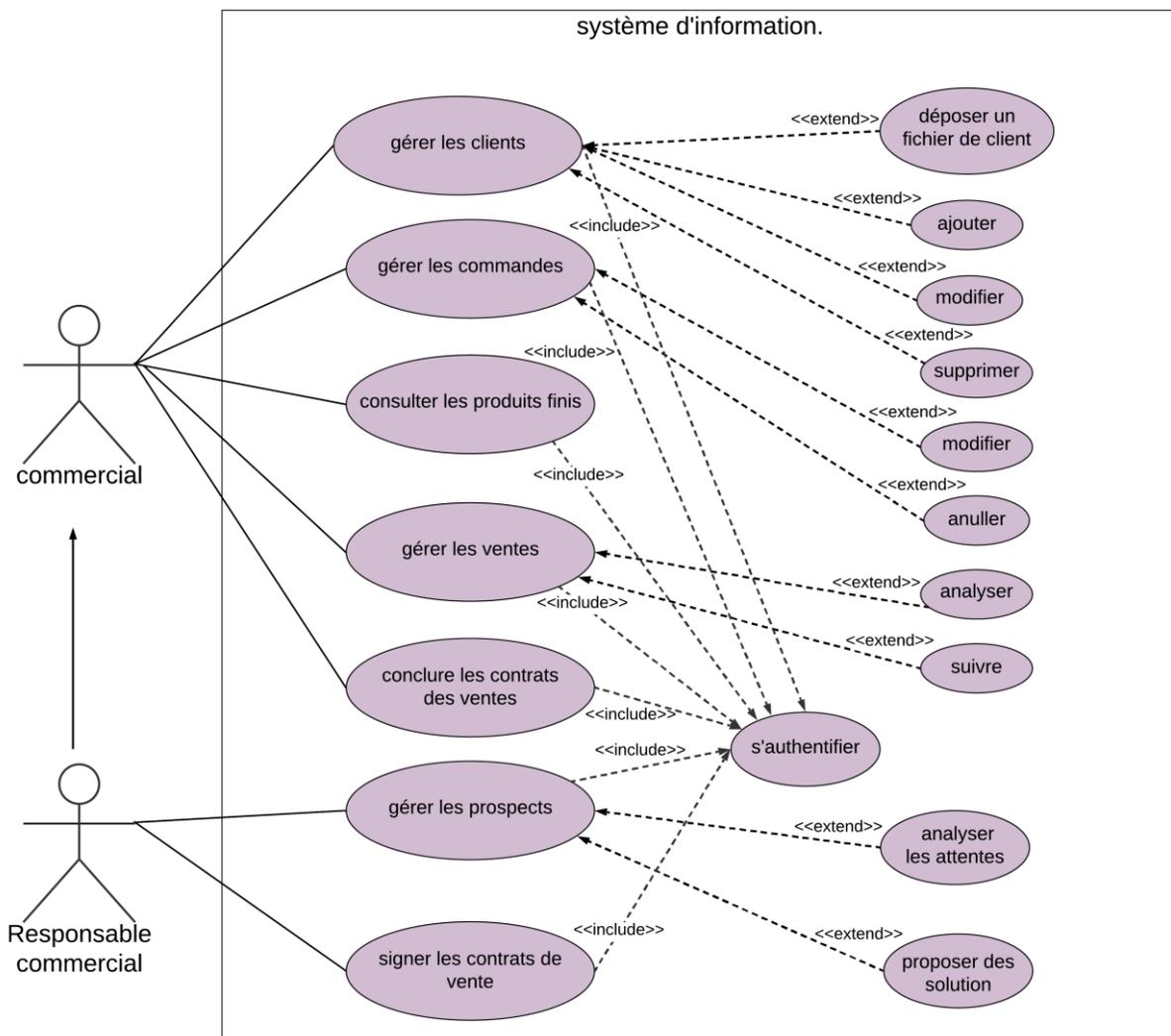


Figure 3.8 : diagramme cas d'utilisation « gestion commerciale ».

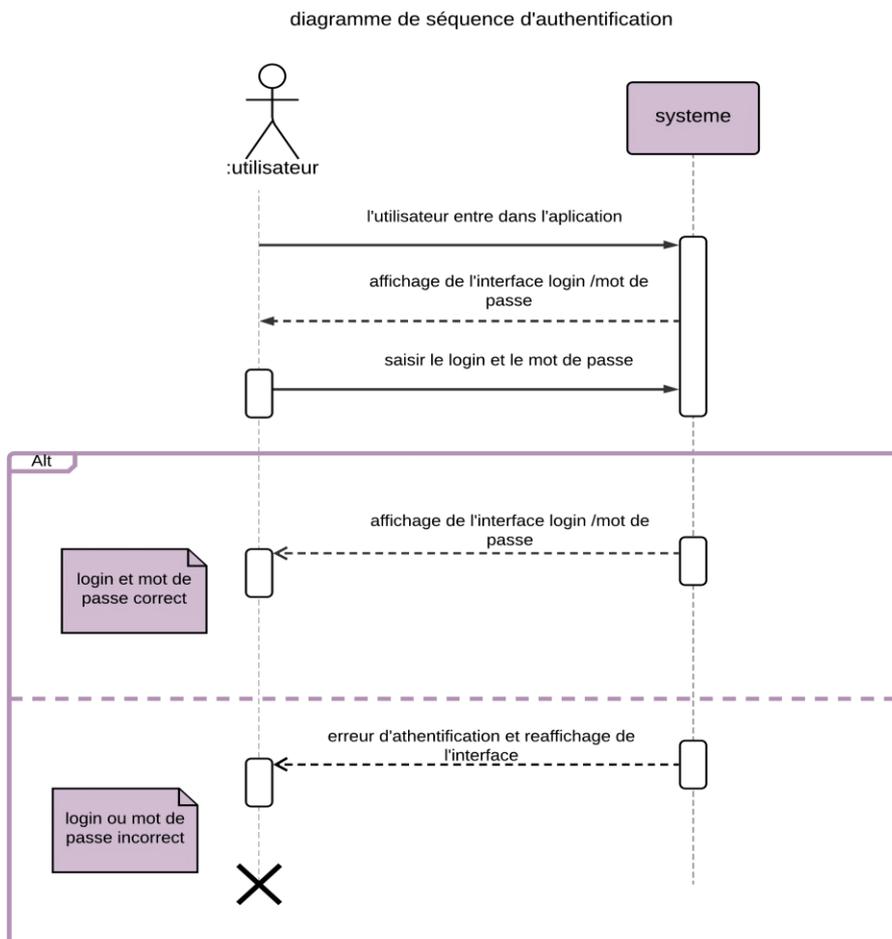
Les acteurs de ce système :

- Acteur service d'achat et approvisionnement : cet acteur établit et négocie les achats de tout produit ou service nécessaire à la production de l'entreprise. Il gère les commandes, les fournisseurs et il établit les fiches techniques et financières des produits.
 - Acteur responsable d'achat et d'approvisionnement : c'est l'acteur qui définit la stratégie achat en collaboration avec la direction générale
- Les scénarios fournis par ce système :
- Gérer les commandes : consulter les commandes pour définir les achats nécessaires ainsi que les modifier ou les supprimer selon le contact du client avec le commercial.
 - Gérer les fournisseurs : contacter les fournisseurs pour l'achat et choisir le meilleur.
 - Etablir les bons de commande : après le choix du fournisseur, le service d'achat établit le bon de commande.
 - Etablir la fiche technique et financière des produits : le service établit un document contenant tous les équipements nécessaires pour la réalisation du produit ainsi que les prix de ces équipements.
 - Définir la stratégie d'achat : c'est définir comment l'entreprise fait ses achats.
 - Négocier les contrats d'achat : la négociation est faite avec le fournisseur pour aboutir à des prix idéals.

3.2. Les diagrammes de séquences du système

Nous avons établi des diagrammes de séquence pour les interactions les plus importantes qui auront lieu entre le système d'information que nous créons et les différents utilisateurs

3.2.1. Diagramme de séquence d'authentification



Chapitre 3.9 : Diagramme de séquence d'authentification.

- L'utilisateur entre dans l'application et essaie de se connecter.
- Le système affiche l'interface login et mot de passe.
- Si le login et mot de passe d'utilisateur sont correcte le système affiche l'espace correspond.
- Si le login ou mot passe sont incorrecte le système réaffiche l'interface.

3.2.2. Diagramme de séquence de la gestion de vente

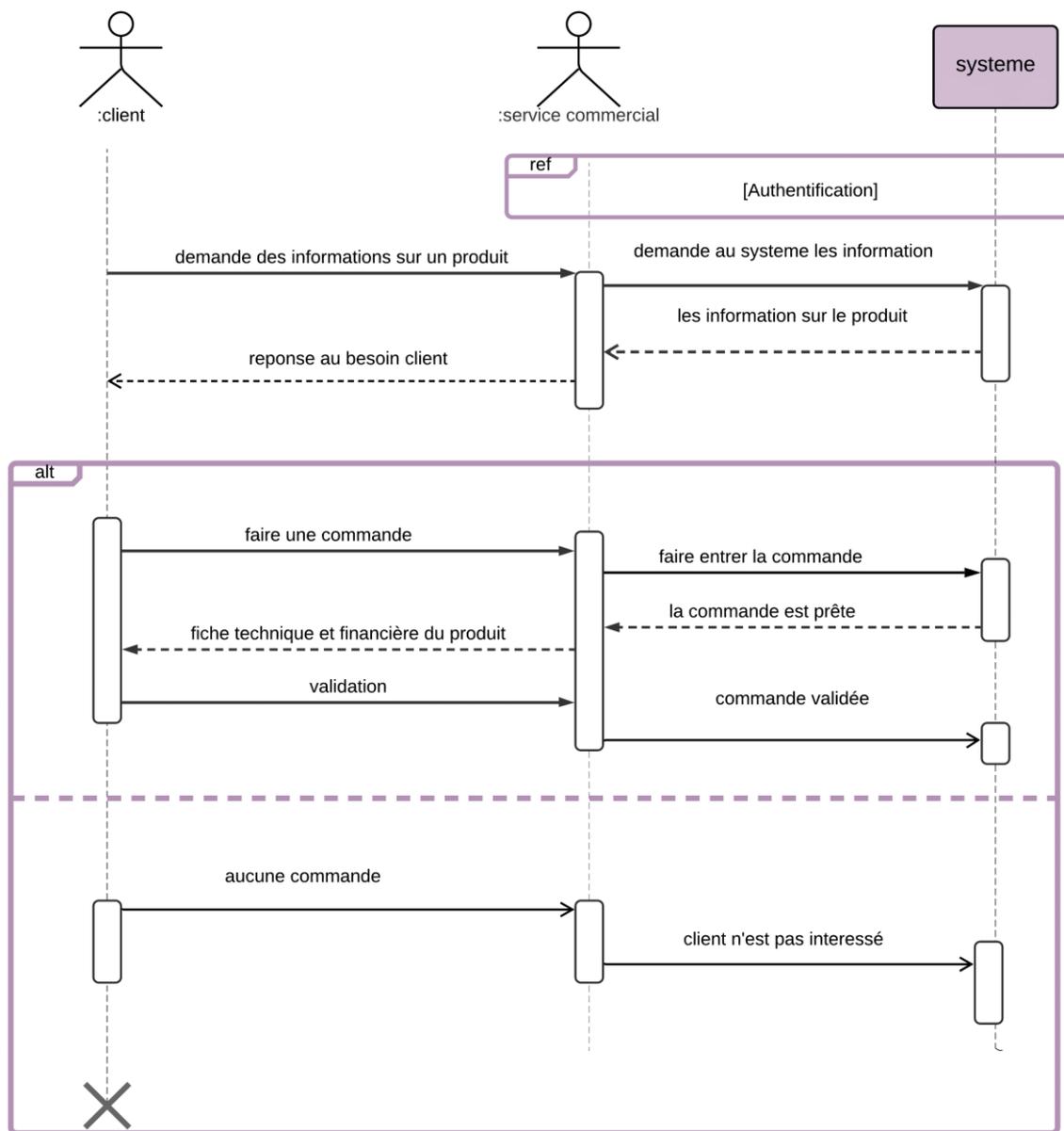


Figure 3.10 : Diagramme de séquence de la gestion de vente.

- Le service responsable de la gestion de vente c'est le service commercial.
- Le système demande au service de s'authentifier pour pouvoir entrer dans l'application.
- Si un client demande au commercial des informations sur un produit, le service demande ces informations au système.
- Dans le cas où un client passe une commande, le commercial passe la commande au système.

- Le service envoie au client la fiche technique et financière.
- Le client valide sa commande.
- Si le client ne passe pas une commande alors client n'est pas intéressé.

3.2.3. Diagramme de séquence de la gestion des prévisions

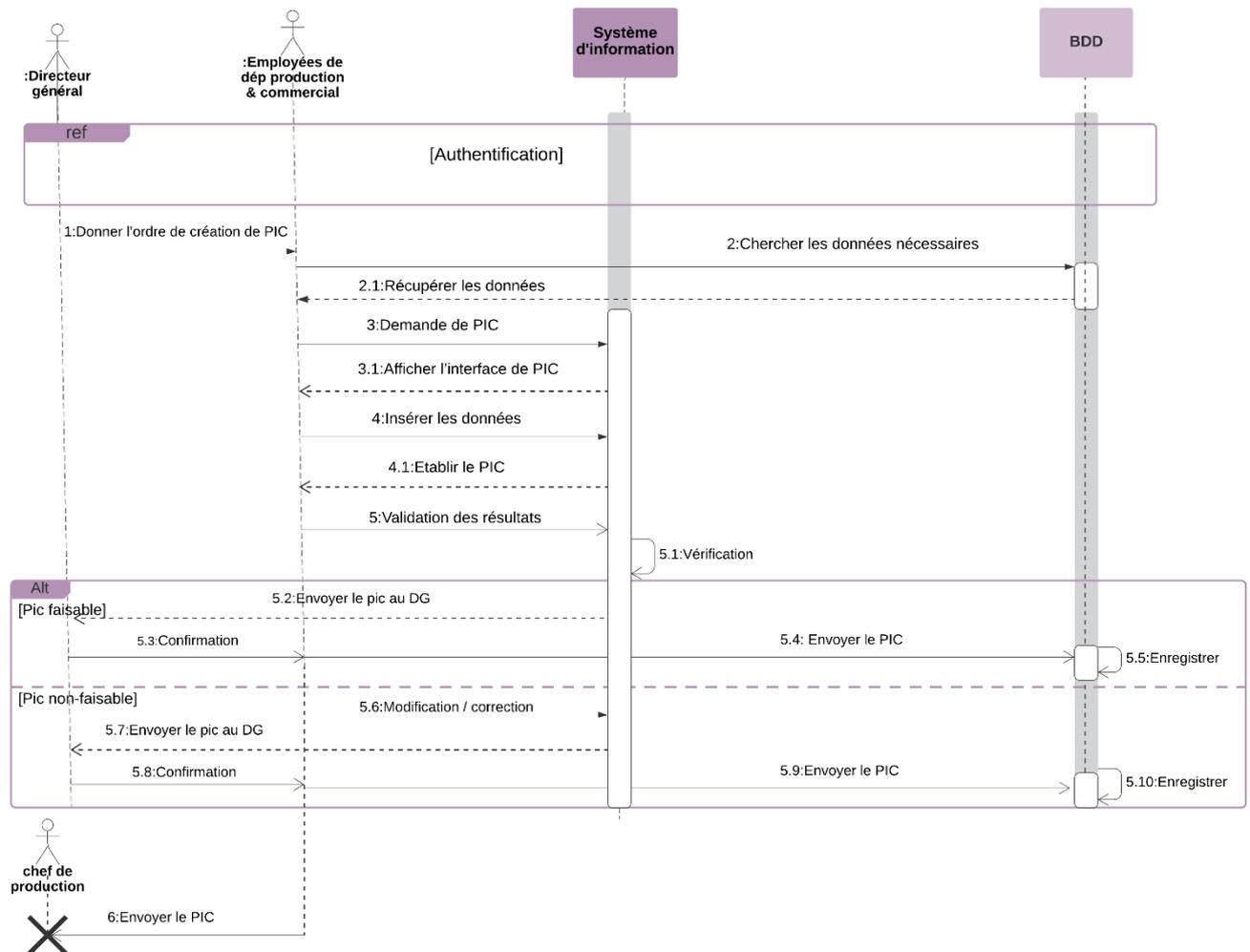


Figure 3.11 : diagramme de séquence de la gestion des prévisions.

- Tout le personnel responsable de la réalisation du plan doit s'identifier dans le système
- Le directeur donne l'ordre de création de PIC à les employés de service de production et le service commercial.
- Ensuite, les données requises pour réaliser ce plan sont restaurées à partir de la base de données du système
- Ces informations sont ensuite saisies dans l'interface de PIC du système.
- Le système résultera le plan en fonction des données saisies.
- Une série de vérifications aura lieu ; ces vérifications visent à contrôler les points suivants

- Est-il réalisable avec les ressources disponibles (humaines, matériels et temporelles) ?
- Est-il bénéfique pour l'entreprise ?
- Enfin le directeur général validera l'ensemble du travail pour l'enregistrer dans la base de données et l'envoyer au service de production.

3.2.4. Diagramme de séquence de la gestion de production

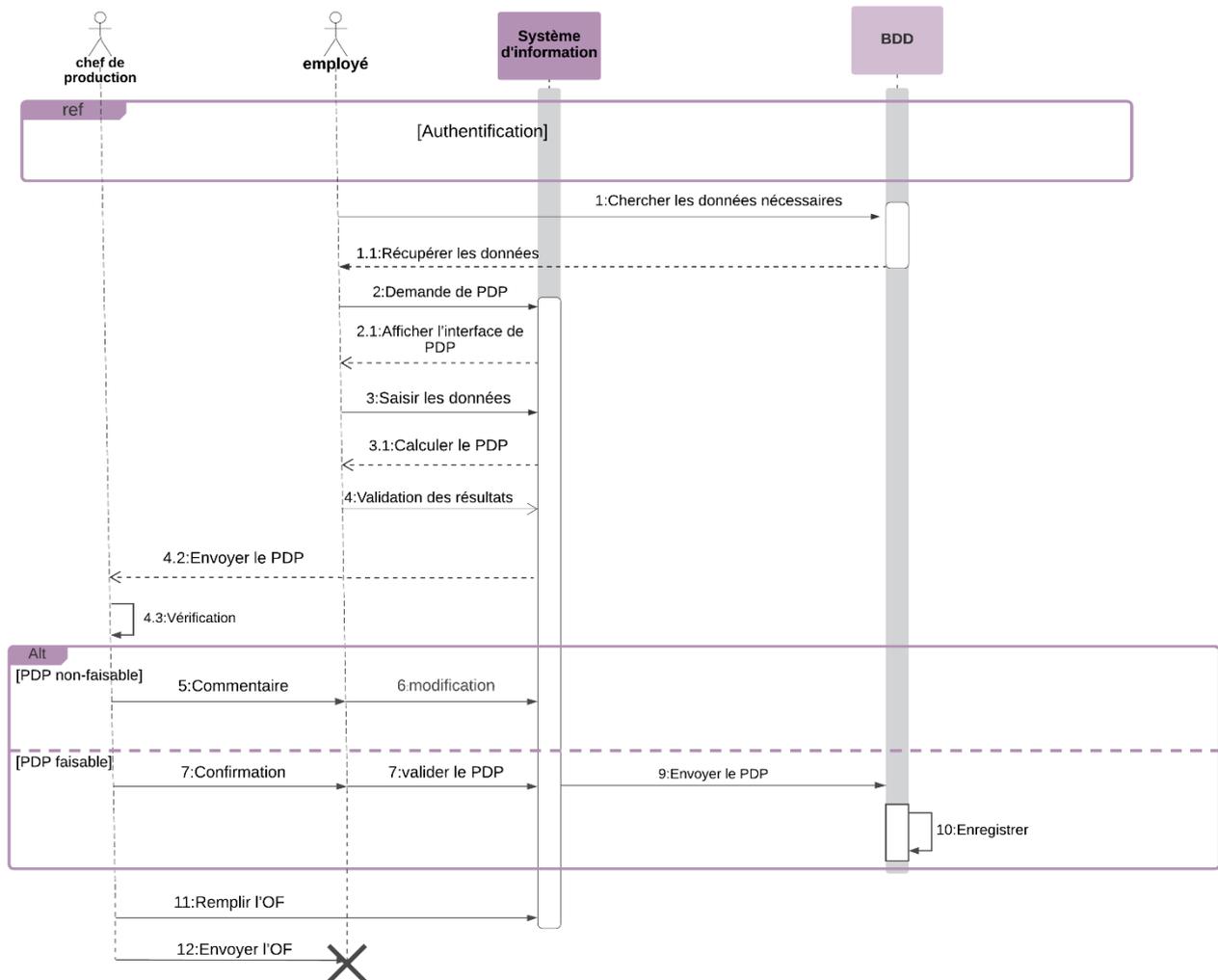
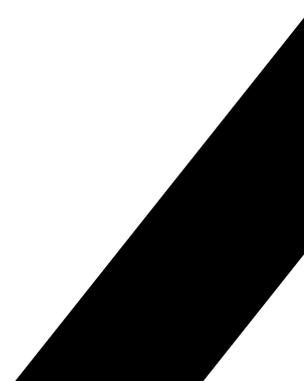


Figure 3.12 : Diagramme de séquence de la gestion de production.

- Tout le personnel responsable de la réalisation du PDP doit s'identifier dans le système.
- Les données requises pour réaliser ce plan (les commandes fermes, les résultats de PIC) sont restaurées à partir de la base de données du système.
- Ces informations sont ensuite saisies dans l'interface de PDP du système.
- Le système résultera le plan en fonction des données saisies.
- Enfin la vérification de faisabilité et la validation.

4. Conclusion :

Informatiser le système de gestion de production de l'entreprise Aquatec Biotechnologie facilitera certainement la tâche aux employés et permettra d'avoir une bonne prévision des produits à travers les données enregistrées. Nous avons présenté les diagrammes de la conception de notre système, les détails de l'implémentation seront présentés dans le chapitre suivant.



Chapitre 04 : Réalisation et implémentation du module gestion de la production

1. Introduction

Ce chapitre est consacré à la phase implémentation du module gestion de production en utilisant l'ERP Odoo. Nous présentons en ce qui suit l'environnement du travail ainsi que les différentes interfaces du module réalisé.

2. Environnement de travail

Dans cette partie, nous exposerons l'environnement nécessaire au développement de notre projet.

2.1. Environnement matériel

Pour réaliser ce projet en toute souplesse et sans heurts, nous avons choisi ce matériel, qui présente les caractéristiques suivantes :

- Processeur : Intel(R) Core (TM) i5 CPU @2.50GHz @2.70GHz.
- Mémoire installer (RAM) : 4.00 GO.
- Système : système d'exploitation 64 bits, processeur x64.

2.2. Environnement logiciel

Pendant le travail sur notre projet principal, nous avons déployé de nombreux gadgets et logiciels.

- PostgreSQL : un moteur open source de base de données relationnelle, qui utilise et étend le langage SQL combiné pour stocker et échelonner en toute sécurité les charges de données les plus complexes.
- Sublime text : un éditeur de code source multiplateforme de type shareware avec une interface de programmation d'applications Python. Il supporte nativement de nombreux langages de programmation et de balisage.
- Lucidchart : une plateforme web propriétaire qui permet aux utilisateurs de collaborer pour dessiner, réviser et partager des tableaux et des diagrammes.

3. Le codage

Comme nous l'avons déjà dit, le codage de ce logiciel a été fait avec une combinaison de langage python et de xml.

Nous verrons donc dans cette partie un aperçu général du code avec lequel nous avons créé notre module.

3.1. Structure de module « manufacturing Aquatec »

Tous les modules dans Odoo sont localisés dans le dossier « \server\odoo\addons », ils ont un architecteur spécifique qui va comme suit :

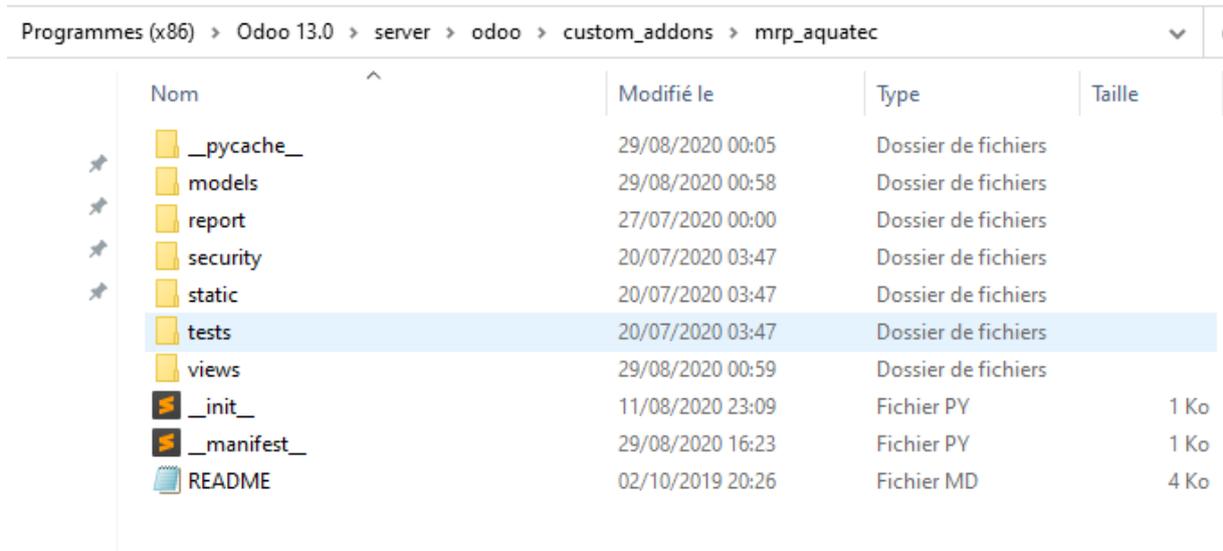


Figure 4.1: structure de module mrp_aquatec

- Fichier `__init__.py` : contenant des instructions pour importer divers fichiers Python dans le module.

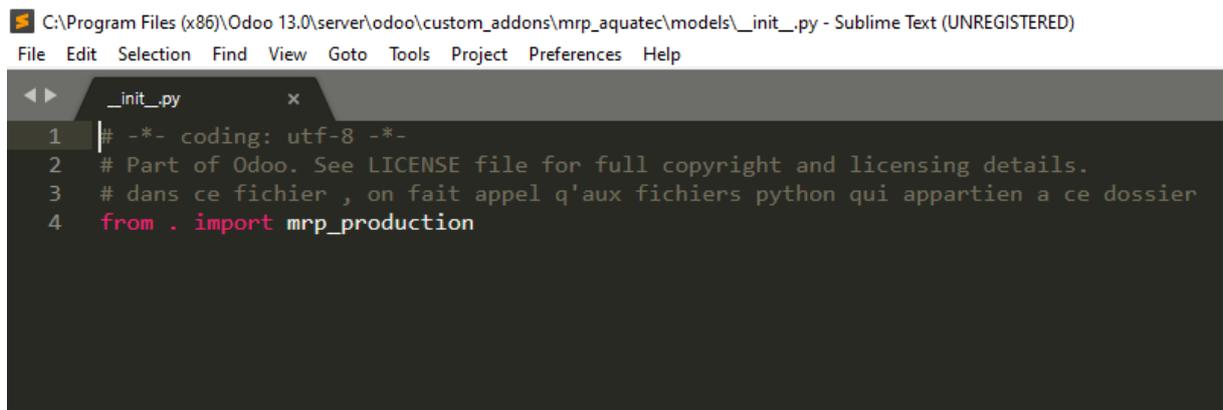
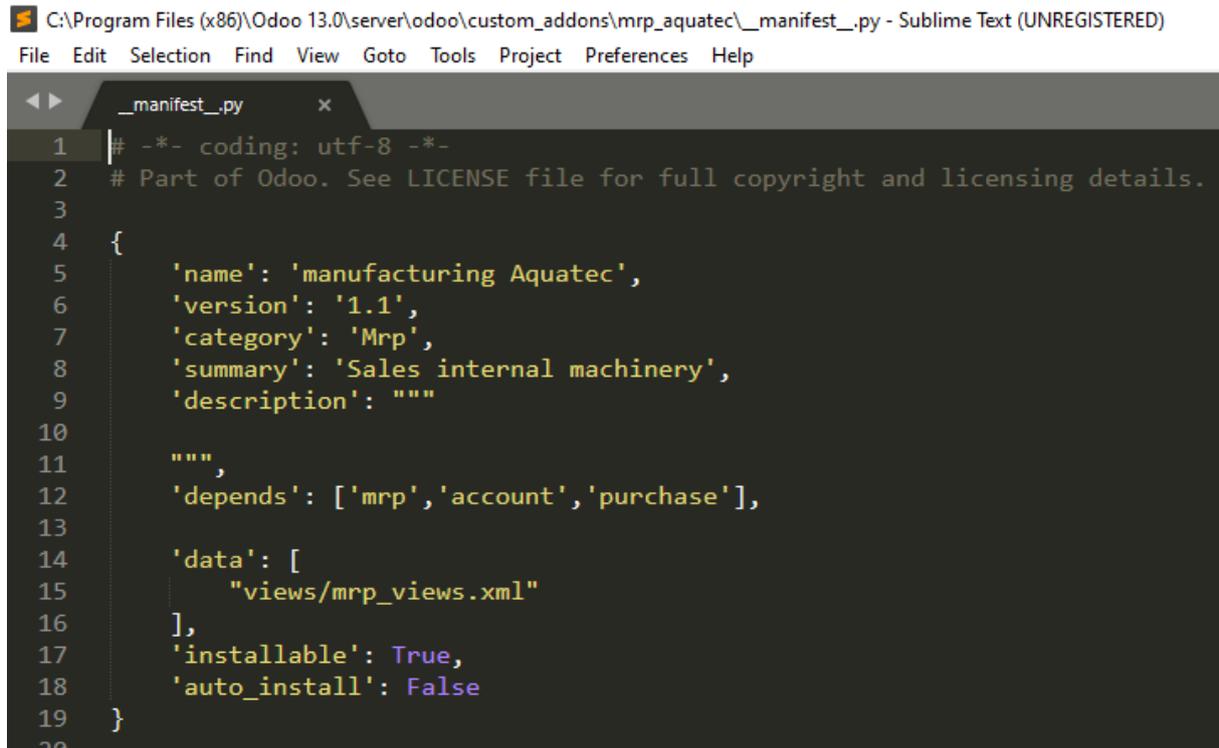


Figure 4.2: fichier init de module

- Fichier `__Manifest__.py` : contient une description détaillée du module

Name	Le nom du module
Summary	Un bref résumé du module (facultatif)
Description	Description du module (facultatif)
Author	L'auteur du module
Website	Le site web de module
Category	La catégorie dans laquelle le module va être placé
Version	La version du module
Depends	Les modules par défaut dont dépend ce module
Data	Listes des fichiers xml contenant les interfaces
Demo	Base de données
Installable	True or False, indique si le module est installable ou non

Tableau 4.1 : composantes d'un fichier manifest.

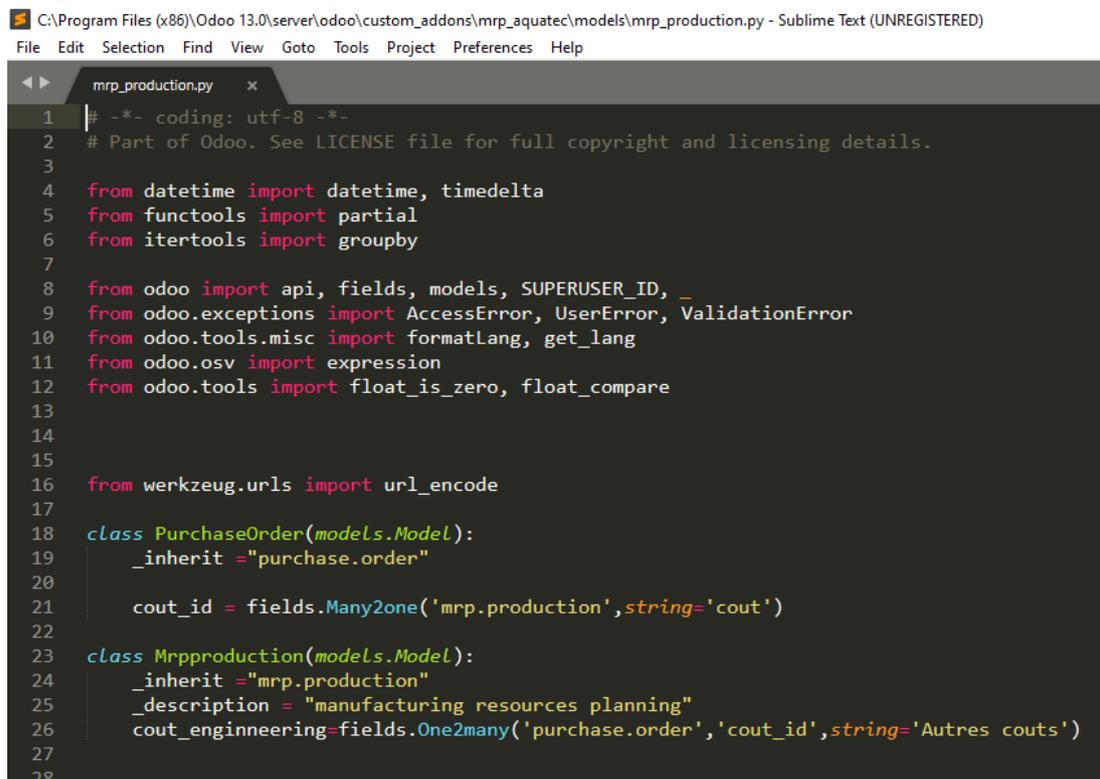


```
C:\Program Files (x86)\Odoo 13.0\server\odoo\custom_addons\mrp_aquatec\_manifest_.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

_ _manifest_.py x
1 # -*- coding: utf-8 -*-
2 # Part of Odoo. See LICENSE file for full copyright and licensing details.
3
4 {
5     'name': 'manufacturing Aquatec',
6     'version': '1.1',
7     'category': 'Mrp',
8     'summary': 'Sales internal machinery',
9     'description': ""
10
11     """
12     'depends': ['mrp','account','purchase'],
13
14     'data': [
15         "views/mrp_views.xml"
16     ],
17     'installable': True,
18     'auto_install': False
19 }
20
```

Figure 4.3: fichier manifest de module

- `Nom_du_module.py` : le fichier qui renferme tous les objets.



```
C:\Program Files (x86)\Odoo 13.0\server\odoo\custom_addons\mrp_aquatec\models\mrp_production.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

mrp_production.py x
1 # -*- coding: utf-8 -*-
2 # Part of Odoo. See LICENSE file for full copyright and licensing details.
3
4 from datetime import datetime, timedelta
5 from functools import partial
6 from itertools import groupby
7
8 from odoo import api, fields, models, SUPERUSER_ID, _
9 from odoo.exceptions import AccessError, UserError, ValidationError
10 from odoo.tools.misc import formatLang, get_lang
11 from odoo.osv import expression
12 from odoo.tools import float_is_zero, float_compare
13
14
15
16 from werkzeug.urls import url_encode
17
18 class PurchaseOrder(models.Model):
19     _inherit = "purchase.order"
20
21     cout_id = fields.Many2one('mrp.production', string='cout')
22
23 class Mrpproduction(models.Model):
24     _inherit = "mrp.production"
25     _description = "manufacturing resources planning"
26     cout_engineering = fields.One2many('purchase.order', 'cout_id', string='Autres couts')
27
28
```

Figure 4.4: exemple d'un module

Nous avons apporté quelques modifications pour personnaliser le module de production sur Odoo.

On a ajouté un nouveau champ nommé « cout » et un autre champs nome « autres couts » qui représente les couts supplémentaires de la production

Fichier views : contient tous les fichier Xml (les vues, les menus, les actions...).

- Les vues : définissent comment le champ doit être affiché dans le système.

```
C:\Program Files (x86)\Odoo 13.0\server\odoo\custom_addons\mrp_aquatec\views\mrp_views.xml - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

mrp_views.xml x
1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3     <record id="mrp_production_inherit_view_form" model="ir.ui.view">
4         <field name="name">mrp.production.inherit</field>
5         <field name="model">mrp.production</field>
6         <field name="inherit_id" ref="mrp.mrp_production_form_view"/>
7         <field name="arch" type="xml">
8             <xpath expr="//page[2]" position="after">
9                 <page string='Autres couts'>
10                    <field name="cout_engineering"/>
11                </page>
12            </xpath>
13        </field>
14    </record>
15 </odoo>
16
17
18
19
```

Figure 4.5: exemple de vude mrp production.

- Les actions : Les actions définissent le comportement du système en réponse aux actions des utilisateurs.

```
32 <record id="production_order_server_action" model="ir.actions.server">
33     <field name="name">Mrp: Plan Production Orders</field>
34     <field name="model_id" ref="mrp.model_mrp_production"/>
35     <field name="binding_model_id" ref="mrp.model_mrp_production"/>
36     <field name="binding_view_types">list</field>
37     <field name="groups_id" eval="[(4, ref('mrp.group_mrp_routings'))]"/>
38     <field name="state">code</field>
39     <field name="code">records.button_plan()</field>
40 </record>
41
```

Figure 4.6: exemple d'une action

4. L'implémentation

Après avoir programmé le module et toute son interface, il ne reste plus qu'à implémenter notre module nommé "manufacturing Aquatec" dans l'éditeur gratuit Odoo 13.

- Interface des applications de Odoo :

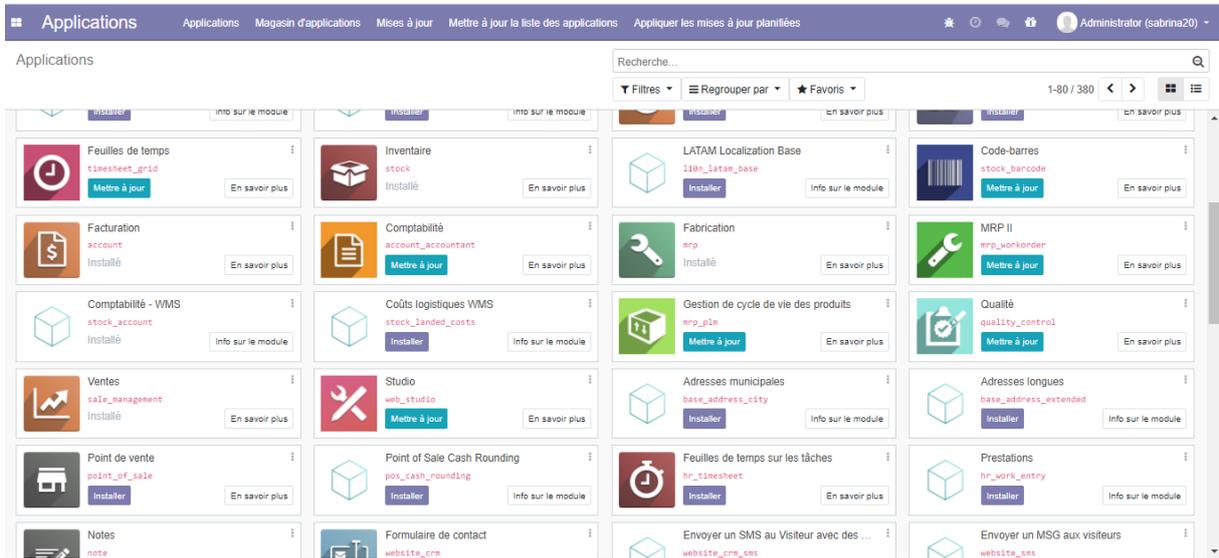


Figure 4.7 : Interface des applications de Odoo

- Installation de module :

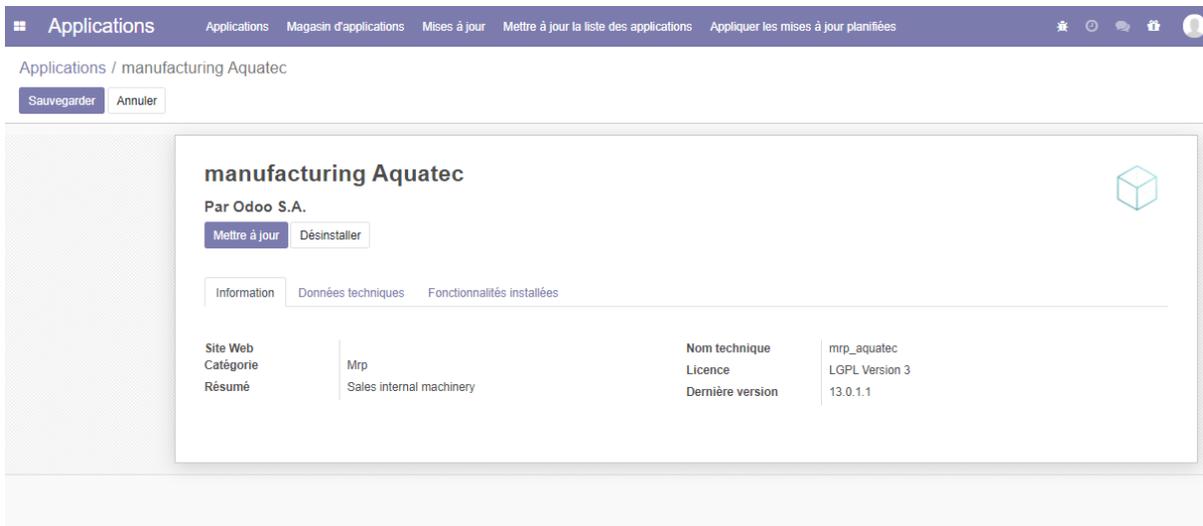


Figure 4.8 : module de production installé

5. La manipulation

Maintenant que le module est installé et configuré, nous pouvons commencer à faire les opérations pour lesquelles le module a été conçu.

5.1. Insertion les cordonnées de l'entreprise

The screenshot shows a web form for entering company information. At the top, the company name 'AQUATEC Biotechnologie' is displayed in a blue box. Below this, there are several sections for data entry:

- Contact:** AQUATEC Biotechnologie
- Adresse:** Z.I. Chetouane, Desserte N°3 13000 Tiemcen. (with sub-fields for Rue 2..., Tiemcen, État, and Code postal)
- Téléphone:** 043 27 23 62.
- Email:** aquatec@groupekherpouche.com
- Site Web:** http://gkgroupe.com/gk/aquatec/index.html
- N° TVA:** (empty field)
- Registre de la société:** (empty field)
- Devise:** DZD (selected from a dropdown)
- Favicon de la société:** (selected from a dropdown)

Figure 4.9 : Les cordonnées d'entreprise.

5.2. Ajouter un utilisateur

The screenshot shows the user management interface. The breadcrumb trail is 'Configuration / Utilisateurs / chef de production'. There are buttons for 'Modifier' and 'Créer', and an 'Action' dropdown. The user 'chef de production' is shown with email 'adress01@gmail.com'. Below the user name, there are tabs for 'Droits d'accès' and 'Préférences'. The 'Type d'utilisateur' is set to 'Utilisateur interne'. At the top right, there are statistics: 25 Groupes, 444 Droits d'accès, and 38 Règles sur l... The user status is 'Jamais connecté'.

Figure 4.10 : interface d'ajout un utilisateur

Nous avons ajouté comme utilisateurs tous les acteurs que nous avons mentionnés dans la phase de conception avec l'attribution des droits d'accès.

5.3. Ajouter un article

Comme le montre la figure ci-dessous, pour ajouter un produit, nous devons entrer les informations clés de ce dernier, comme le nom du produit, sa référence, son prix de vente, la taxe qui s'y applique et son coût.

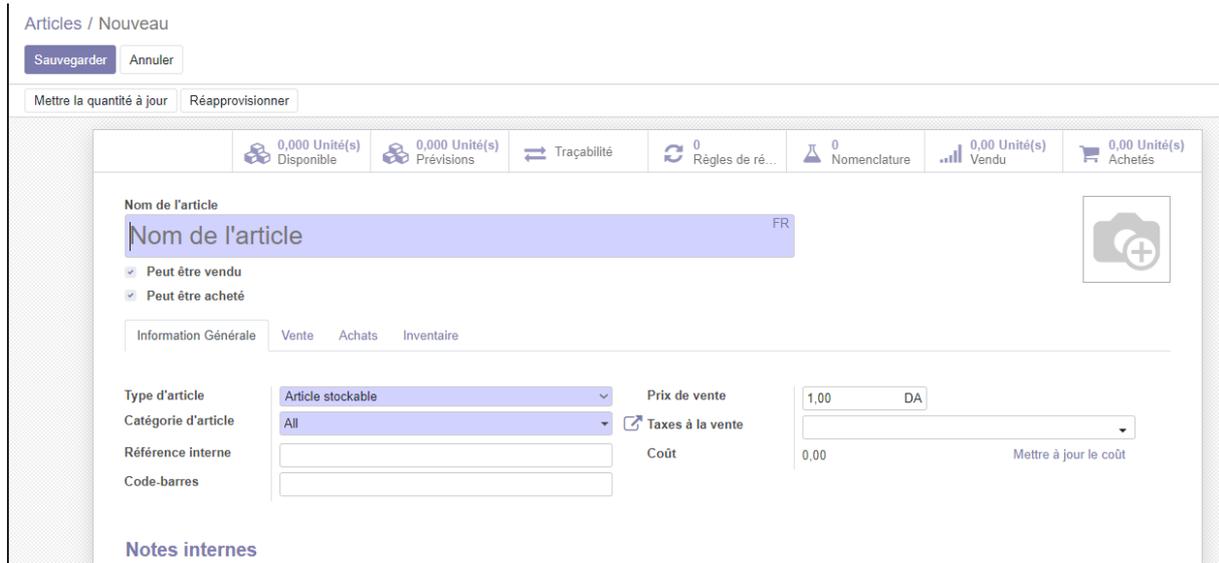


Figure 4.11 : Interface d'ajout d'un article

Pour chaque article on a des outils de gestion par exemple :

- Le réapprovisionnement :

Dans cette interface on définit la quantité min et max de ce produit.



Figure 4.12 : interface de réapprovisionnement

- Les prévisions :

A travers les statistiques des ventes et achats le système estime les besoins en termes de métiers premier à acheter les commandes et les ventes (le PIC), afin de calculer le PDP (les quantités des produits finis à produire).

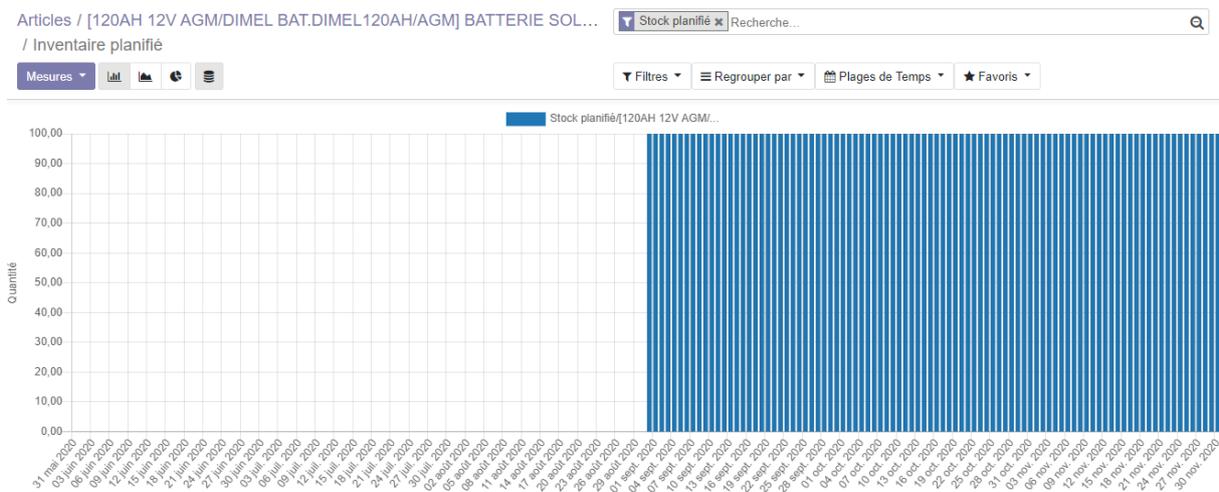


Figure 4.13 : interface de prévision d'un article

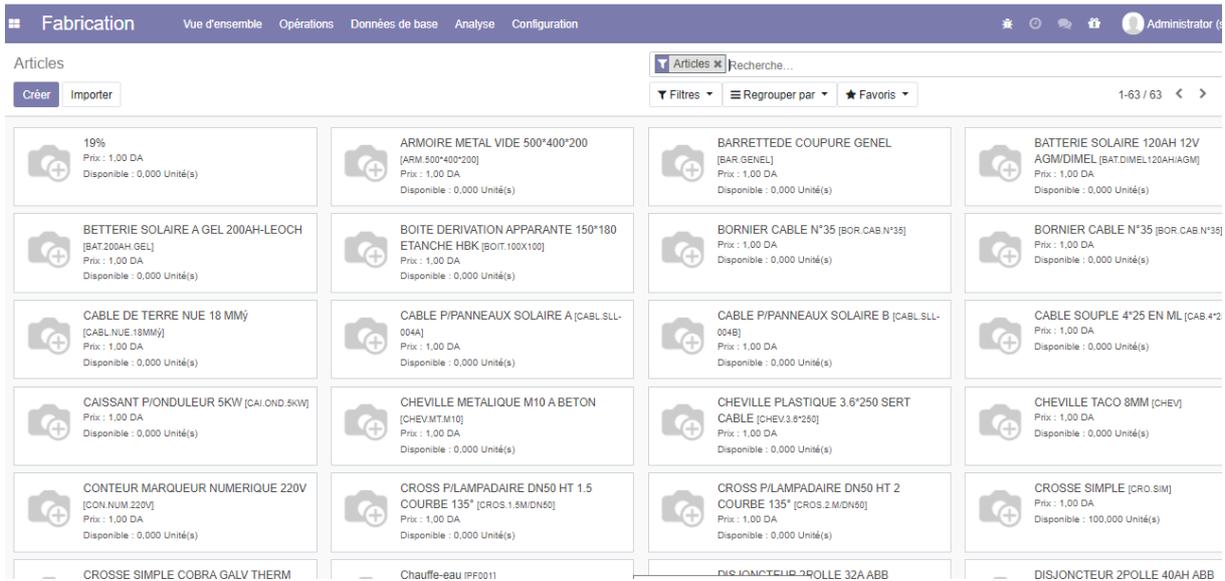


Figure 4.14 : Liste des articles ajoutés

- Cette interface représente la liste des articles de l'entreprise Aquatec qui se trouve dans le magasin et le détail de chaque article (quantité, prix, disponibilité...).
- Pour voir les modifications que nous avons apportées à notre module on donne l'une des interfaces de ces articles.

5.4. Insertion de nomenclature

On a pris comme exemple le nomenclature de produit fini « kite solaire 150w ».

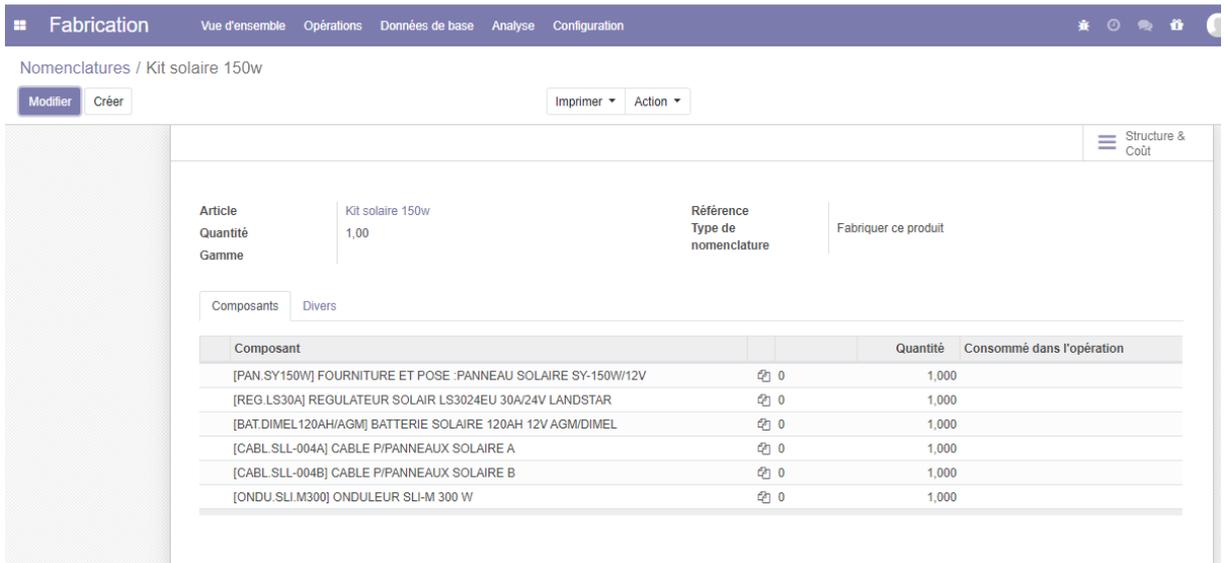


Figure 4.15 : nomenclature de kite solaire 150w

5.5. Lancement un ordre de fabrication

La figure ci-dessous montre l'ordre de fabrication d'une kite solaire 150w.

Ordres de fabrication / WH/MO/00001

Modifier Créer Imprimer Action 1/1 < >

WH/MO/00001

Article	kite solaire 150w	Date limite	16/10/2020 12:21:01
Quantité à produire	3,000 Mettre à jour	Date planifiée	14/10/2020 11:19:46 au 14/10/2020 12:19:46
Nomenclature	kite solaire 150w	Responsable	chef de production
		Source	

Composants Produits finis

Article	A consommer	Réservé	Consommé
[SY-150W/12V] PANNEAU SOLAIRE	3,000	3,000	0,000
[LANDSTAR REG.LS30A] REGULATEUR SOLAIRE LS3024EU 30A/24V	3,000	3,000	0,000
[ONDU.SLI.M300] ONDULEUR SLI-M 300 W	3,000	3,000	0,000
[CABL.SLL-004B] CABLE P/PANNEAUX SOLAIRE B	3,000	3,000	0,000
[CABL.SLL-004A] CABLE P/PANNEAUX SOLAIRE A	3,000	3,000	0,000
[120AH 12V AGM/DIMEL BAT.DIMEL120AH/AGM] BATTERIE SOLAIRE	3,000	3,000	0,000

Figure 4.16: ordre de fabrication.

5.6. Déconstruction d'un ordre de fabrication

Commandes de déconstruction / UB/00001

Modifier Créer Action 1/1 < >

Deconstruction Brouillon Fait

UB/00001

Article	kite solaire 150w	Ordre de fabrication	WH/MO/00002
Nomenclature	kite solaire 150w	Société	Aquatec biotechnologie
Quantité	1,00		

Figure 4.17 : interface de déconstruction d'un ordre de fabrication

6. Conclusion

Le logiciel développé pour l'entreprise Aquatec Biotechnologie a pour objectif d'informatiser le système de gestion de production de cette dernière. Ceci permettra certainement une gestion efficace évitant les erreurs dues à l'utilisation du papier et permettant de garder la traçabilité des informations.

Conclusion générale

“ Ce rapport s’inscrit dans le cadre de notre projet de fin d’étude à l’école supérieure en sciences appliqués de Tlemcen ESSAT et au sein de l’entreprise « Aquatec Biotechnologie Groupe Kherbouche ».

Notre travail durant le stage à l’entreprise c’était l’analyse et la spécification des besoins pour une gestion efficace de la production des différent produits et l’optimisation des flux.

En ce sens, nous avons travaillé dans ce projet sur la mise en place d’un progiciel de gestion intégré PGI (ERP) précisément un ERP open source Odoo pour le développement d’un module de gestion de production (prévisions des demandes clients et planifier les processus de la production) tout en respectant les besoins de l’entreprise.

Après le choix précis des langages et outils de modélisation de notre solution, nous avons entamé la conception qui présente une partie très importante pour la réussite du projet. Cette étape comporte les diagrammes d’UML accompagné par les descriptions textuelles. Ces diagrammes représentent une description graphique du fonctionnement du système conçu.

Par la suite nous avons passé à la réalisation de notre solution proposée, nous avons développé et implémenté le module de la gestion de la production. Les langages utilisés c’est python et xml. Le travail et ces langages se trouvent dans l’environnement du travail Odoo version 13.

L’implémentation de notre module nécessite d’autre module qui doivent être mise en place au parallèle se sont la gestion de stock et le commercial ce qui nous a permis de développer et d’implémenter ces modules aussi pour la réussite du module en question.

Finalement, l’étude effectuée et l’étape du pratique ont donnés un résultat et une solution au besoin de la société. De plus, c’est une chance pour l’entreprise pour continuer à l’amélioration du système par le déploiement de la solution ainsi de penser au développement et l’implémentation des solution ERP pour le reste de ses services pour obtenir une meilleure

”
gestion pour tous ses fonctions.

Résumé

Les enjeux actuels et futurs et l'évolution progressive et la compétitivité féroce dans le marché imposent aux entreprises de s'ouvrir sur les nouvelles technologies et l'informatisation qui sont devenues l'une des tendances les plus connues aujourd'hui. En effet, la gestion automatisée peut être un élément capital de rentabilisation d'une entreprise.

Pour ce faire, nous allons mettre en place un progiciel de gestion pour l'entreprise Aquatec Biotechnologie. Certes il existe plusieurs solutions sur le marché, mais pour notre cas le choix adéquat c'est les ERP's qui sont révolutionnés d'une façon impressionnante.

L'objectif de ce travail qui a été fait au sein de l'entreprise Aquatec est d'apporter à son organisme un levier de croissance primordiale par le développement d'un ERP pour la gestion du stock.

Abstract

The current and future stakes and the progressive evolution and fierce competitiveness in the market require companies to open up to new technologies and computerization, which have become one of the best-known trends today. Indeed, automated management can be a key element in making a company profitable.

To do so, we are going to set up a management software package for the company Aquatec Biotechnology. Of course, there are several solutions on the market, but in our case the most suitable choice is ERPs (enterprise resource planning) which are revolutionized in an impressive way.

The objective of this work that was done at Aquatec Biotechnology is to bring to its organization a primary growth lever by developing an ERP for production management.

ملخص

التحديات الحالية والمستقبلية والتطور التدريجي والقدرة التنافسية الشرسية في السوق تجبر الشركات على الانفتاح على التقنيات الجديدة والحوسبة، التي أصبحت واحدة من أفضل الاتجاهات المعروفة اليوم. في الواقع، يمكن أن تكون الإدارة الآلية عنصرًا حاسمًا في جعل الأعمال التجارية مربحة.

و لهذا سنقوم باعداد برنامج لادارة شركة Aquatec biotechnologie، بالتأكيد هناك العديد من الحلول في السوق، لكن الحل الأمثل لمشروعنا هذا هو نظام تخطيط موارد المؤسسة (بالإنجليزية: Enterprise Resource Planning)، والذي يشهد تطورًا كبيرًا في الآونة الأخيرة.

هو تزويد منظمتهما برافعة نمو رئيسية من خلال تطوير برنامج Aquatec الهدف من هذا العمل الذي تم تنفيذه داخل شركة تخطيط موارد المؤسسات لإدارة الإنتاج.