

République Algérienne Démocratique et Populaire

Université Abou Bakr Belkaid– Tlemcen

Faculté des Sciences

Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Master en Informatique

Option : Génie Logiciel (GL)

Thème

Développement Odoo pour la gestion de pièces de rechange, pour un client de
SOGESI

Réalisé par :

- BENFEDEL Ahmed
- KAZI AOUL Abdessamad

Présenté le 05 Juin 2016 devant le jury composé de :

- Mme HALFAOUI Amel (Présidente)
- Mr MESSABIHI Mohamed (Encadreur)
- Mme KHITRI Souad (Examinatrice)
- Mr TADLAOUI Mohammed (Examineur)
- Mr BRAHMI Noredine (Maitre de stage)

Année universitaire : 2015-2016

Remerciement

Un très grand merci à :

Notre enseignant et encadreur, Mr Mohamed MESSABIHI qui nous a guidé et soutenu tout au long de la préparation de notre mémoire et pour ses conseils et son écoute.

Mr BRAHMI Nouredine Directeur de SOGESI, de nous avoir donné la possibilité de travailler dans son entreprise, et toute l'équipe SOGESI

Mme HALFAOUI, Mme KHITRI et Mr TADLAOUI pour avoir fait l'insigne honneur d'accepter de faire partie du jury de notre travail.

Tous les enseignants du département d'informatique à qui nous devons notre respect et reconnaissance.

Nous ne saurons oublier de remercier nos parents pour leur contribution, leur soutien et leur patience, nos proches, nos amis et toutes les personnes qui nous ont aidé par leur soutien permanent de près ou de loin durant nos études.

Dédicace

*Je dédie ce travail à celle qui m'a porté avant le monde, mon espoir,
la source de mon bonheur et ma réussite, à ma mère ...*

*A mon père, mon protecteur depuis ma naissance, mon école depuis
mon enfance, la lumière de mon existence...*

Que dieu les garde et les protège.

A mes grandes mères, mes tantes, cousins et cousines

A mes frères Abderezak et Mohammed El Hadi

A mon collaborateur Ahmed BENFDAL

A mes meilleurs amis Yasser, Othmane, Houcine, Brahim

A ma deuxième famille GDG Tlemcen

A tous ceux qui me sont chères.

Abdessamad

Dédicace

Je dédie ce travail à mes parents que dieu les garde et les protège.

A mes sœurs

A mon meilleur ami HAMRA Mohammed Amine

A mes ami(e)s.

A tous ceux qui me sont chères.

A tous ceux qui m'aiment.

A tous ceux que j'aime.

Ahmed

Table des matières

Introduction Générale	3
I.1 Contexte	3
I.2 Présentation de l'organisme d'accueil.....	3
I.2.1 Fiche signalétique SOGESI.....	4
I.2.2 Domaines d'activités.....	5
I.2.3 Prestations et services	5
I.3 Cadre général du projet.....	5
I.4 Problématique	6
I.5 Objectifs du projet.....	6
I.6 Plan Du mémoire	7
Chapitre 1 : Les ERP et Odoo	8
II.1 Introduction.....	8
II.2 Différentes versions d'ERP.....	8
II.3 La Plate-forme Odoo.....	9
II.4 Odoo et ses modules.....	10
II.5 Architecture fonctionnelle et technique d'Odoo	10
II.5.1 Architecture Client / Serveur.....	10
II.5.2 Architecture Modulaire	11
II.5.3 Structure d'un module Odoo	12
II.5.4 Modèle Vue Contrôleur.....	12
II.5.5 Les WorkFlows	13
II.5.6 Création d'un module sous Odoo.....	13
Chapitre 2 : Conception et Réalisation.....	15
III.1 Introduction.....	15
III.2 Outils utilisés.....	15
III.2.1 Gestion de projet avec Trello	15
III.2.2 Modélisation avec Entreprise Architecte	16
III.2.3 Langages : Python & XML	17
III.2.4 Autre : Google Drive, Google Doc	18
III.3 Descriptions des étapes de réalisation de l'application.....	18
III.3.1 Etude de système existant chez l'entreprise cliente	18
III.3.2 Analyse du besoin et Extraction des règles de gestions	24
III.2.3 Diagrammes de classes	32

III.2.4	Une solution pour chaque règle de gestion	34
III.4	Site Web e-commerce	49
III.4.1	Installation du module e-commerce	49
III.4.2	Gestion du catalogue Produits, organisation de la boutique	49
	Conclusion générale.....	54
	Références.....	55
	Liste des figures :	56
	Liste des abréviations.....	57

Introduction Générale

I.1 Contexte

Ce mémoire décrit le travail que nous avons réalisé dans le cadre de la préparation de notre projet de fin d'études de MASTER en informatique option Génie Logiciel (GL) à l'Université de Tlemcen. Le stage s'est déroulé du 1er Février au 26 Mai 2016 au sein de la société SOGESI à Tlemcen.

Notre but principal était de contribuer dans la communauté mondiale de recherche et de développement dans le domaine de la gestion des ressources d'entreprises (ERP), et préparer notre éventuelle insertion professionnelle dans ce domaine.

Pour atteindre cet objectif et durant ces quatre mois, notre mission consistait à déployer une solution Odoo au sein d'une entreprise cliente chez SOGESI en adaptant des modules déjà existants, et en créant de nouveaux modules afin de répondre efficacement aux besoins spécifiques de l'entreprise cliente.

Le but du projet est de faciliter la gestion d'achat, de stock, d'inventaire, de vente et de retour d'articles. Les articles sont principalement des pièces de rechange d'engins. Le projet gère également les clients, les fournisseurs, les bons (de livraison, de réception, facture pro-forma...), et les rapports.

I.2 Présentation de l'organisme d'accueil

SOGESI est une SSII basée en Algérie spécialisée dans les services informatiques et d'ingénierie de proximité, la gestion des entreprises et formations.

Le champ d'intervention de l'Ingénierie Applicative concerne toutes les activités du cycle de conception, de réalisation et d'évolution d'une application, s'appuyant sur les technologies et les solutions les mieux adaptées aux besoins.

Au niveau Conseil, SOGESI accompagne les directions générales et informatiques de grands groupes et PME/PMI dans le cadre de leurs projets de transformations. SOGESI conçoit, développe, teste, maintient et optimise les systèmes d'information des entreprises et des organismes publics. En Algérie, ce savoir-faire de multi spécialiste est unique. Avec son profil de multi-spécialiste, SOGESI compte mettre au service de ses clients une combinaison unique de proximité, d'organisation sectorielle et de solutions de qualité industrielle. [1]

SOGESI tend à répondre aux nouvelles exigences du marché, de plus en plus axées sur la connaissance métier, au-delà de l'expertise technologique.

L'approche sectorielle permet au groupe de proposer de nouvelles offres structurées pour chaque secteur, et de capitaliser sur l'expérience forte de ses collaborateurs en France pour se développer en Algérie et proposer des solutions à des prix très compétitifs.

SOGESI permet également de choisir la prestation qui convient le plus grâce à une flexibilité due à l'organisation des métiers :

- Intégration et déploiement de solutions ERP
- Audit, conseil et gestion de projets
- Formation et délégation de personnel

Les méthodes :

- Approche du projet
- Analyse, étude et critique de l'existant
- Rédaction de cahier des charges
- Proposition de solutions sur mesure
- Intégration, formation et support

Les domaines d'excellence :

- Les solutions de qualité
- Le conseil
- L'assistance aux utilisateurs
- La formation

I.2.1 Fiche signalétique SOGESI

Raison social : « SOGESI » SOciété de GEstion des Solution Informatiques

Adresse : 4, derrar Abderrahmane – Kiffane – 13000 Tlemcen

Date de création : 23/05/2012

Mobile : 00213 5 55 62 12 13

Téléphone bureau : 0021343413374

I.2.2 Domaines d'activités

Intégration de projet et développement d'applications sur mesure ainsi que la mise en place de portails internet, d'applications web et de logiciels sur mesure :

- Dernières technologies Open Source.
- Info design, charte graphique et animations propriétaires.
- Stratégie Web Marketing.

Moyens humains :

SOGESI s'appuie sur des consultants et ingénieurs certifiées dans leurs domaines d'intervention. Ils analysent le besoin en fonction des organisations et assurent la conception, la réalisation et la maintenance de la solution.

I.2.3 Prestations et services

SOGESI offre une large palette de prestations et de services basés sur des composants libres adaptés aux systèmes et aux réseaux des clients. La principale tâche de cette société est d'offrir des solutions sur mesure, en matière de formation et d'assistance, concernant les problématiques relevant des systèmes d'informations.

La gamme de services de SOGESI est articulée autour d'un ensemble d'axes majeurs qui permettent d'accompagner les clients durant toutes les phases d'un projet afin d'en assurer sa réussite.

Formation et conseil : L'offre des formations, techniques et fonctionnelles, permet d'accompagner les organisations qui disposent d'équipes opérationnelles capables de mener à bien des projets. Ces formations peuvent être établies sous forme de transferts de compétences, en phases finales des projets.

I.3 Cadre général du projet

Le système d'information coordonne grâce à l'information les activités de l'organisation, et lui permet ainsi d'atteindre ses objectifs. Il est le véhicule de communication de l'organisation.

Il représente l'ensemble des ressources (humaines, matérielles et logicielles) organisées pour collecter, stocker, traiter et communiquer les informations au sein de l'entreprise. Notre projet de fin d'études s'inscrit dans le cadre d'une solution optimisée de la gestion d'achat, vente, stock et retour d'articles (pièces de rechanges) à l'aide de

l'ERP open source dans lequel on se base pour adapter nos solutions et de développer de nouvelles fonctionnalités.

I.4 Problématique

Pour gérer l'activité globale de la société, chaque département utilise indépendamment des méthodes anciennes et des applications informatiques indépendantes. Ceci génère non seulement une difficulté d'échange de données entre les différents départements, mais également une incohérence de l'information interne.

Avec ces outils, un gérant se retrouve obligé d'effectuer plusieurs tâches qui normalement, avec ces solutions doivent être automatisées, par exemple l'envoi des bons de commande vers le fournisseur qui se fait par téléphone ou papiers en indiquant les produits et la quantité. Les bons de commande sont donc temporaires, et se trouvent dans la plupart des cas supprimés après avoir effectué la réception des produits. Ainsi, il s'avère important de coordonner l'ensemble des activités de la société (comme la gestion des achats, ventes, stocks, etc.) autour d'un même système d'information.

I.5 Objectifs du projet

Notre projet a pour objectif de toucher les axes fondamentaux du domaine d'ERP, l'exemple sur lequel nous avons travaillé était d'automatiser un ensemble de processus d'une entreprise tout en travaillant sur une base de données unique et homogène afin de gagner en matière de productivité et réduire les travaux redondants. Ainsi, chaque maillon de l'organisation apportera sa contribution et la mettra à disposition des autres acteurs de la chaîne.

Notre projet consiste donc à proposer une solution répondant aux besoins fonctionnels de l'entreprise cliente. Cette solution devra, entre autres, assurer :

- L'optimisation de la gestion des processus métiers : gestion des ventes, achats, retour d'articles et gestion facturation.
- Permettre d'effectuer des achats auprès des fournisseurs, où il est possible d'éditer des devis, de créer des bons de commande et de recevoir les produits.
- Permettre une gestion optimale du stock, avec la possibilité de faire les inventaires physiques et de consulter l'état des stocks à tout moment.
- La cohérence et l'homogénéité de l'information interne.

- La gestion des éléments de bases, à savoir les clients, les fournisseurs avec une interface ergonomique qui facilite la navigation entre les fonctionnalités du système.
- Enfin, avoir des statistiques et des rapports concernant ces points cités auparavant.
- La possibilité de développer de nouvelles fonctionnalités.
- La mise à disposition d'un site d'e-commerce aux clients de l'entreprise.

I.6 Plan Du mémoire

La suite de ce mémoire est structurée comme suit :

- Introduction générale : Une introduction générale et présentation du cadre du projet
- Chapitre 1 : Présente la notion des ERP en parlant de leurs principales plateformes, puis nous présentons une étude fonctionnelle et technique de l'ERP Odoo.
- Chapitre 2 : Donne une analyse détaillée de l'application que nous avons réalisé en partant de l'étude de l'existant et du besoin du client, ensuite les différentes étapes de conception et du développement.
- Nous terminerons par une conclusion qui résume notre travail puis quelques perspectives.

Chapitre 1 : Les ERP et Odoo

II.1 Introduction

Les entreprises d'aujourd'hui sont de plus en plus conscientes de l'impact de la bonne gestion des ressources internes pour l'amélioration de leurs performances et productivités dans le marché. Or, les tâches de gestion s'avèrent de plus en plus difficiles et complexes à cause des flux importants de données ainsi que la diversité des processus fonctionnels (achat, vente, GRH...).

Pour affronter ces difficultés, toute entreprise est prête à investir des sommes considérables dans l'implantation de technologies logicielles afin d'améliorer ses services vis-à-vis des clients tout en optimisant la gestion interne de l'entreprise.

Pour le cas du client de SOGESI, la gestion de l'ensemble des activités se fait manuellement ou par le biais des applications informatiques, mais la société souhaite gérer la totalité de ses activités autour d'un même système d'information, à l'aide d'un progiciel de gestion intégré connu sous l'acronyme ERP, c'est dans ce cadre que s'inscrit notre projet de fin d'étude qui a pour objectif de mettre en place une solution de gestion intégrée au sein d'une société cliente de SOGESI.

L'acronyme PGI signifie "Progiciel de Gestion Intégrée" traduit en anglais par Enterprise Resource Planning (ERP). ERP est le terme le plus couramment utilisé. Un ERP est un progiciel qui assure la gestion automatique de l'ensemble des processus d'une entreprise comme la gestion des ressources humaines, la gestion comptable, la gestion des ventes, l'approvisionnement, la production ou encore du e-commerce. [2]

Le principe fondateur d'un ERP est l'utilisation des applications informatiques correspondantes aux divers processus métiers de manière modulaire, en partageant une base de données unique et commune au sens logique. L'autre principe, qui caractérise un ERP, est l'usage de ce qu'on appelle un moteur de Workflow.

Il permet d'acheminer une donnée qui est enregistrée dans le système d'information(SI), vers les autres modules qui en ont besoin. [2]

II.2 Différentes versions d'ERP

Le 20 janvier 2011, OpenERP SA annonçait le lancement de la version 6.0 du logiciel, qui comprend une version à la demande (SaaS). Son approche modulaire permet

aux utilisateurs de commencer avec une application, puis d'ajouter d'autres modules selon leurs besoins. [2]

En décembre 2012, la version 7.0 d'OpenERP est lancée et peut être testée en ligne, téléchargée ou vue en version de démonstration. [3]

Été 2014, Odoo lance la version 8, cette version enrichit principalement le logiciel de nouvelles applications qui font d'Odoo un logiciel allant au-delà d'un ERP. Ces applications sont : Marketing (gestion d'événements, d'enquêtes de satisfactions, campagnes de mails auprès de la CRM,...), CMS (construction d'un site internet - front-end lié au back-end - grâce au déplacement rapide et simple de "blocs" d'éditations), e-commerce (application pour vente en ligne), la figure suivante (Figure1) montre les versions d'ERP avec date de sortie et changement majeurs. [4]

- Première version stable : 2004
- Version stable actuelle : 8.0
- Version avancée : 9.0

Nom du logiciel	Version	Date de lancement	Changements significatifs
Tiny ERP	1.0		
	2.0		
	3.0		
	4.0		
OpenERP	5.0		
	6.0	Octobre 2009	Première publication sous AGPL ⁴
	6.1		
	7.0	Décembre 2012	
Odoo	8.0	Septembre 2014	Support pour le CMS: Construction de site internet, e-commerce, point de vente, vente et business intelligence.
	9.0		
Tryton	3.0		

Figure 1 : Différentes Versions de OpenERP

II.3 La Plate-forme Odoo



Odoo, anciennement OpenERP et Tiny ERP, est à la base un progiciel libre de gestion intégré comprenant de très nombreux modules permettant de simplifier la gestion d'entreprise dans son ensemble. Le logiciel est sous licence AGPL et est utilisé par plus de 2 millions d'utilisateurs à travers le monde.

L'ERP Odoo comporte de nouvelles fonctionnalités liées à des applications front office (CMS, e-Commerce, Blogs, Forums, News, Événements, LiveChat, Job offers,

L'opération client est très simple. Quand un utilisateur exécute une action (sauvegarder un formulaire, ouvrir un menu, imprimer, ...) il envoie cette action au serveur. Le serveur envoie alors la nouvelle action pour s'exécuter côté client. Il y a trois types d'actions :

- Ouvrir une fenêtre (formulaire, listes)
- Imprimer un document.
- Exécuter un wizard.

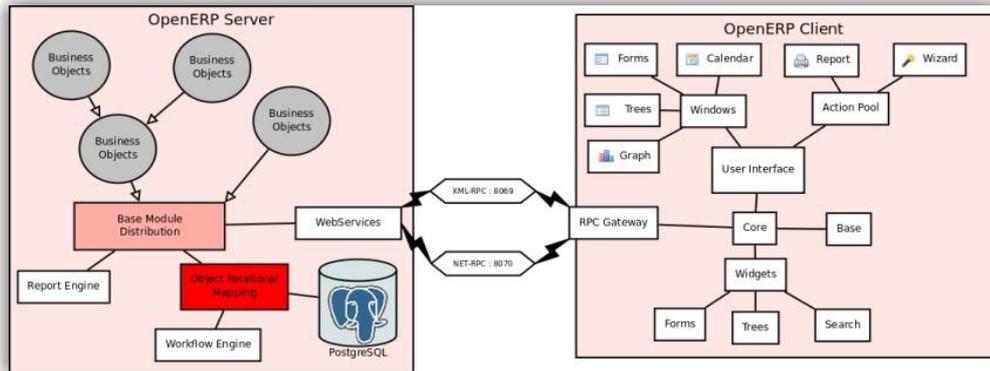


Figure 3 : Architecture Client-Serveur d'open ERP

II.5.2 Architecture Modulaire

Un module OpenERP est la définition, dans le «Framework» OpenERP, d'une gestion informatisée d'un domaine. Cette architecture n'est pas propre à open ERP. Elle est en fait partagée par tous les ERP. Il s'agit de la faculté de construire des applications informatiques de manière modulaire (modules indépendants entre eux) tout en partageant une base de données unique. Ceci apporte une importance significative puisque les données sont maintenant standardisées et partagées. Ce qui élimine les saisies multiples et évite l'ambiguïté des données de même nature. L'architecture modulaire d'open ERP lui permet de couvrir plusieurs domaines illustrés dans la figure ci-dessous : [5]



Figure 4 : Architecture modulaire d'open ERP

II.5.3 Structure d'un module Odoo

La figure suivante montre la structure standard d'un module Odoo, qui contient principalement des vue, des rapports, des wizards, des workflows, et des objets. [7]

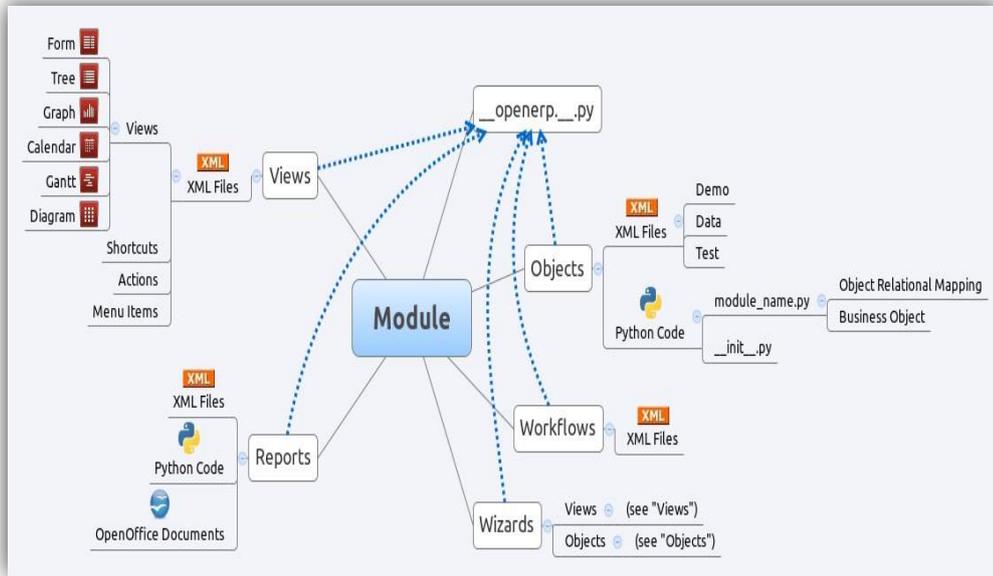
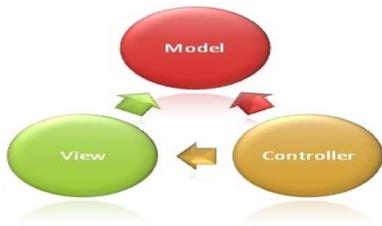


Figure 5 : Structure d'un module OpenERP Python

II.5.4 Modèle Vue Contrôleur



MVC est une architecture de modèles utilisée en génie logiciel. Dans des applications complexes qui présentent des lots de données aux utilisateurs, on souhaite souvent séparer les données (modèle) et l'interface utilisateur (vue), de sorte que les changements à l'interface utilisateur n'affectent pas le traitement des données, et que les données peuvent être réorganisées sans changer l'interface utilisateur. Le MVC résout ce genre de problème en découplant l'accès des données et la logique des applications de la présentation des données et de l'interaction utilisateur, en introduisant un composant intermédiaire : « le contrôleur ». [7]

Dans open ERP, on peut appliquer cette sémantique de Model View Controller avec :

- Model : les modèles sont les objets déclarés dans OpenERP. Ils sont également des tables PostgreSQL.
- View : les vues sont définies en fichiers XML dans OpenERP.

- Contrôler : le contrôleur est les classes Python qui contrôle OpenERP.

OpenERP offre un cadre de développement, c'est à dire des «services» techniques informatiques :

- Un serveur de base de données objet pour représenter et mémoriser les objets de gestion et les rendre accessible via le réseau.
- Un "workflow" qui contrôle l'évolution des objets suivant une procédure.
- Des formulaires et écrans pour l'interaction avec l'utilisateur.
- Des états imprimables des objets. [7]

II.5.5 Les WorkFlows

On appelle workflow la modélisation et la gestion informatique de l'ensemble des tâches à accomplir et des différents acteurs impliqués dans la réalisation d'un processus métier (aussi appelé processus opérationnel ou bien procédure d'entreprise). Le terme de « workflow » pourrait donc être traduit en français par « gestion électronique des processus métier ». [8]

De façon plus pratique, le workflow décrit le circuit de validation, les tâches à accomplir entre les différents acteurs d'un processus, les délais, les modes de validation, et fournit à chacun des acteurs les informations nécessaires pour la réalisation de sa tâche. Pour un processus de publication en ligne par exemple, il s'agit de la modélisation des tâches de l'ensemble de la chaîne éditoriale.

Il permet généralement un suivi et identifie les acteurs en précisant leur rôle et la manière de le remplir au mieux. [8]

Buts de WorkFlows :

- Description de l'évolution du document dans le temps
- Déclenchement automatique d'actions si certaines conditions sont remplies
- Gestion des rôles de l'entreprise et les étapes de validation
- Gestion des interactions entre les différents objets / modules

II.5.6 Création d'un module sous Odoo

Pour créer un module Odoo, il y'a des étapes essentielles à suivre :

- Créer un package python dans le répertoire /addons portant le nom de votre module (requis)

- Créer un fichier de description du module : `__openerp__.py` (requis)
- Créer le fichier Python contenant les modèles (Classes + Méthodes) (requis)
- Créer des fichiers `.XML` pour définir les menus, les vues et les actions (requis)
- Créer des fichiers `.XML` qui téléchargent des données de démonstration
- Créer éventuellement des rapports, des assistants (Wizard) ou des flux de travail (Workflow). [9]

Obligatoires	<code>__init__.py</code>	Initialisation du package Python
	<code>__openerp__.py</code>	La fiche descriptive du module
	<code>module.py</code>	Les classes Python, les objets du module
	<code>module_view.xml</code>	Les vues (Formulaires, listes, graph,...), menus et actions
Optionnels	<code>demo</code>	Données de test et de démonstration
	<code>i18n</code>	Fichiers de traduction – Internationalisation
	<code>report</code>	Définition des rapports
	<code>security</code>	Déclaration des groupes et des droits d'accès
	<code>workflow</code>	Définition des workflow
	<code>wizard</code>	Définition des assistants

Figure 6 : Structure d'un module Odoo

Conclusion : Dans ce chapitre nous avons présenté les ERP en général, puis Odoo en particulier en présentant son architecture, sa logique ainsi que la création d'un module

Chapitre 2 : Conception et Réalisation

III.1 Introduction

Comme nous l'avons déjà signalé dans l'introduction générale, l'une des premières solutions des problématiques de ce projet est d'améliorer et adapter les modules existants d'Odoo, ainsi que développer d'autres modules afin de répondre au maximum de besoins du client concernant la gestion d'achats, stocks, ventes, retour d'article, rapports...

Dans ce chapitre nous allons définir les principales règles de gestions ainsi que leur processus métier. Cette version améliore la version précédente et répond aux exigences demandées et aux problèmes posés au début. Nous décrivons dans cette partie la façon dont le système va fonctionner en lui donnant une forme et une architecture. En fin, nous allons présenter l'utilisation de l'application.

III.2 Outils utilisés

III.2.1 Gestion de projet avec Trello

Trello est un outil en ligne, ergonomique et gratuit. Nous utilisons cet outil pour organiser nos tâches, consigner les informations essentielles et tenir un planning avec tous les membres de notre équipe. Grâce à cet outil nous simplifions le suivi de notre projet.

Ce service en ligne nous aide à mieux organiser nos activités. Nous consignons nos tâches sur des post-it « cards » que nous accrochons sur un panneau « board » en les rangeant dans des listes de gauche à droite :

- Les tâches à faire « To Do » ;
- Les tâches en cours de réalisation « Doing » ;
- Les tâches à tester « To Test » ;
- Les tâches réalisées « Done ».

Les colonnes contiennent autant de post-it que nous le souhaitons. Sur chaque post-it, en plus de renseigner son contenu, nous pouvons :

1. indiquer la liste des points à vérifier pour cette tâche ;
2. épingler des documents bureautique et multimédia ;
3. préciser une date limite,
4. affecter une tâche à un membre de l'équipe

Au fur et à mesure de l'avancement de nos tâches, nous déplaçons les postit d'une colonne à l'autre. Le principe de base est très intuitif et le service regroupe plusieurs options qui peuvent contribuer à l'amélioration de la productivité de l'équipe.

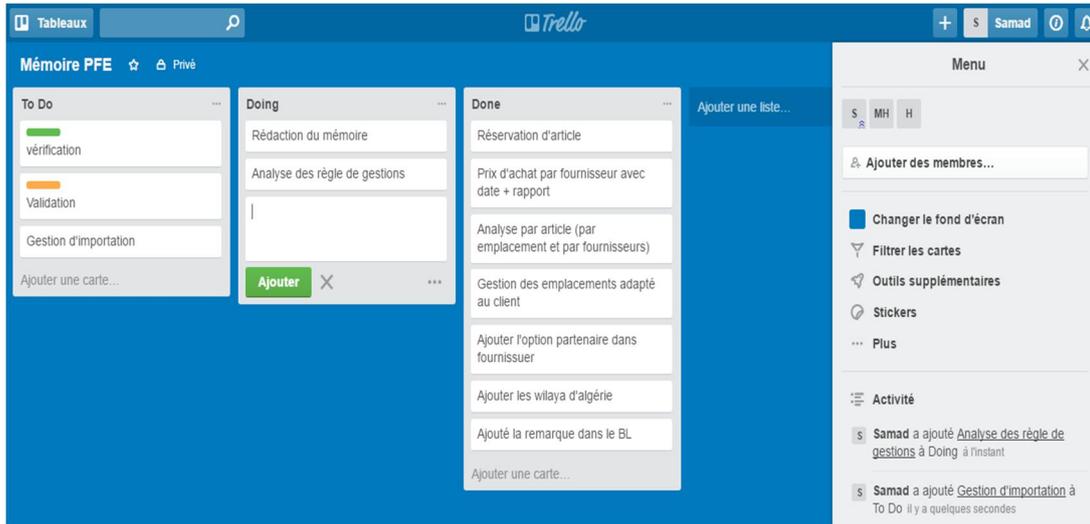


Figure 7 : Interface Trello

III.2.2 Modélisation avec Enterprise Architect

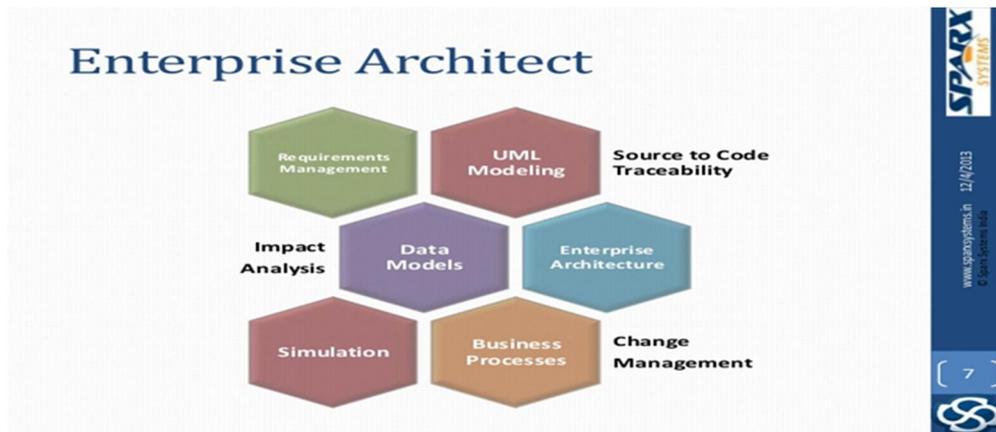


Figure 8 : Structure d'entreprise Architect

Enterprise Architect [10] est un logiciel de modélisation et de conception UML, édité par la société australienne Sparx Systèmes. Couvrant, par ses fonctionnalités, l'ensemble des étapes du cycle de conception d'application, il est l'un des logiciels de conception et de modélisation les plus reconnus. [11]

III.2.3 Langages : Python & XML

- XML :

L'Extensible Markup Language (XML « langage de balisage extensible » en français) est un métalangage informatique de balisage générique qui dérive du SGML. Cette syntaxe est dite « extensible » car elle permet de définir différents espaces de noms, c'est-à-dire des langages avec chacun leur vocabulaire et leur grammaire, comme XHTML, XSLT, RSS, SVG...



Elle est reconnaissable par son usage des chevrons (<>) encadrant les balises. L'objectif initial est de faciliter l'échange automatisé de contenus complexes (arbres, texte riche...) entre systèmes d'informations hétérogènes (interopérabilité). Avec ses outils et langages associés, une application XML respecte généralement certains principes :

- la structure d'un document XML est définie et validable par un schéma ;
- un document XML est entièrement transformable dans un autre document XML. [12]

- Python :



Python est un langage de programmation objet, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions ; il est ainsi similaire à Perl, Ruby, Scheme, Smalltalk et Tcl.

Le langage Python est placé sous une licence libre proche de la licence BSD et fonctionne sur la plupart des plates-formes informatiques, des supercalculateurs aux ordinateurs centraux, de Windows à Unix en passant par GNU/Linux, Mac OS, ou encore Android, iOS, et aussi avec Java ou encore .NET. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser.

Il est également apprécié par les pédagogues qui y trouvent un langage où la syntaxe, clairement séparée des mécanismes de bas niveau, permet une initiation aisée aux concepts de base de la programmation. [13][14][15]

III.2.4 Autre : Google Drive, Google Doc

-  Google Drive : pour le partage de fichier
-  Google Doc : pour collaborer sur un même fichier Word

III.3 Descriptions des étapes de réalisation de l'application

Durant le stage, nous sommes passés principalement par 3 étapes ;

- 1- Compréhension de l'aspect fonctionnel et technique d'Odoo
- 2- Analyse du besoin du client, et extraction des règles de gestions avec l'encadrement du staff de l'entreprise SOGESI
- 3- Réaliser une solution pour chaque règle de gestion :
 - A. Chercher dans les modules existants, les réponses aux besoins (règles) du client
 - B. Adaptation des modules existants selon les besoins
 - C. Création de fonctionnalités et de modules non-existants

III.3.1 Etude de système existant chez l'entreprise cliente

La société cliente repose principalement sur trois divisions qui assurent son fonctionnement normal qui sont : le service des ventes, le service des achats et enfin le service de gestion de stock.

L'ensemble des activités de la société est géré par des traitements manuels ou bien par le biais des applications diversifiées tout en utilisant des supports papiers.

Les besoins de la société s'articulent sur les modules fonctionnels suivants : la gestion des ventes, achats, et stock. A cela, s'ajoutera les modules de gestion de retours et comptabilité.

III.3.1.1 Etat de fonctionnement actuel

- Processus achat-stock dans l'entreprise cliente

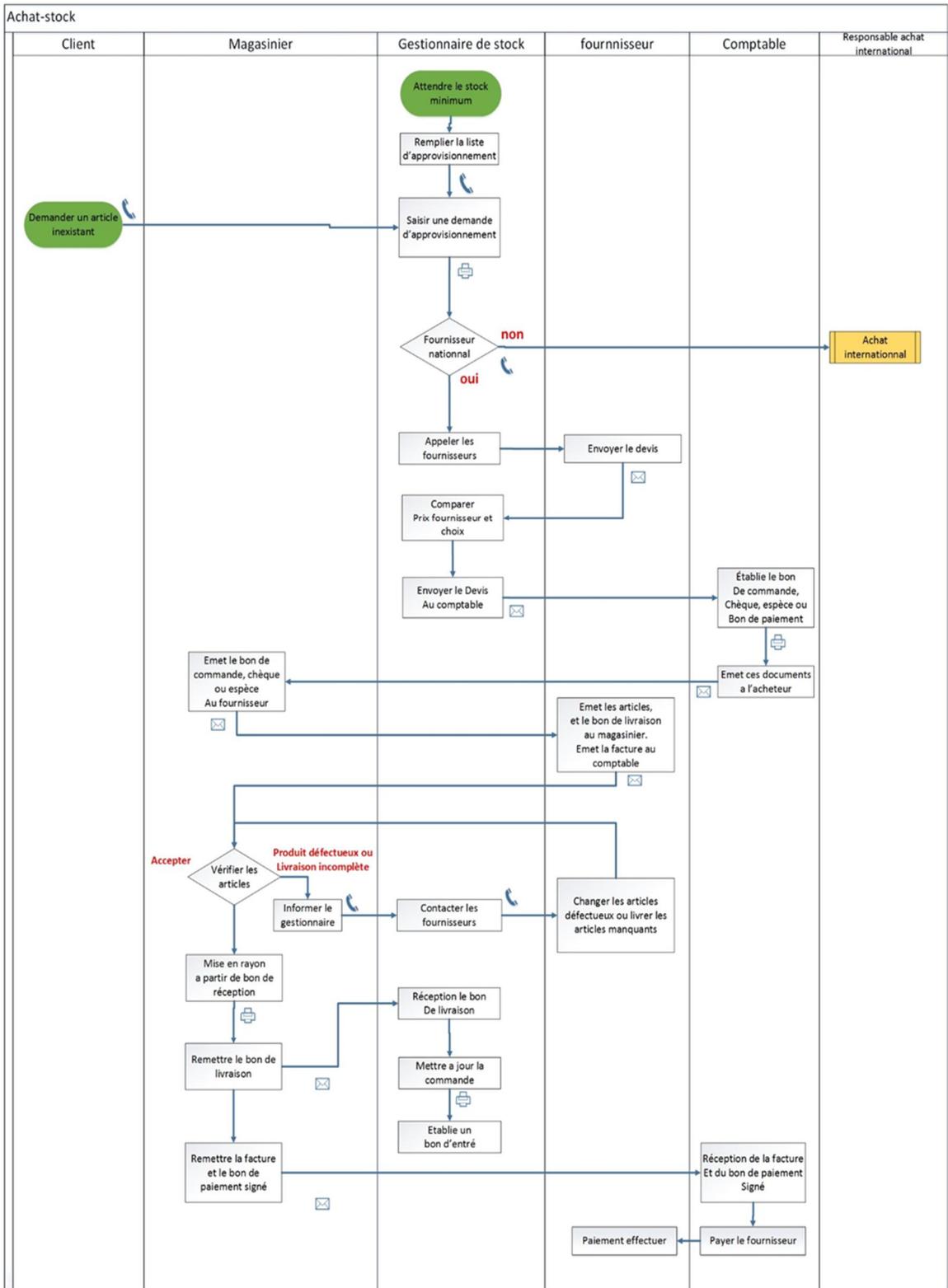


Figure 9 : Diagramme de processus d'achat-Stock

Description

- **Déclenchement de la procédure ACHAT-STOCK :**

La procédure achat stock se déclenche via une commande d'un client ou l'atteinte d'un stock minimum d'un produit.

Avant de rédiger la demande d'approvisionnement, le gestionnaire du stock doit vérifier la disponibilité de l'article dans le stock.

- **Saisir une demande d'approvisionnement :** Le gestionnaire de stock établit une demande d'approvisionnement des pièces suivant la quantité des pièces disponibles en stock ou les quantités des pièces commandées par les clients. La demande d'approvisionnement doit être validée par le responsable financé ou le directeur général. Les informations essentielles saisies sont :

- Les références des pièces.
- Les quantités.

- **Comparer les prix des fournisseurs et choisir les articles :**

Le Gestionnaire de stock contacte les fournisseurs pour avoir des devis sur les articles à commander. Suivant les devis des fournisseurs d'articles, le gestionnaire de stock choisit un fournisseur qui répond aux exigences de l'entreprise (prix, qualité, professionnalisme...).

Dans le cas où l'article doit être fourni seulement par **un partenaire** de l'entreprise (cas de commande urgente livraison par DHL, pièces disponibles que chez le constructeur ou livraison normale), l'action de comparaison des prix ne sera pas nécessaire.

- **Envoyer le bon de commande :**

Après le choix du fournisseur, le gestionnaire de stock envoie le devis au comptable pour qu'il lui fasse un bon de commande (chèque, espèce ou bon de paiement).

Le bon de commande est ensuite envoyé au fournisseur par le magasinier. Ce dernier joue le rôle de l'acheteur.

- **Emettre les articles, BL et Facture :**

Le fournisseur établit un bon de livraison et une facture d'achat après la livraison des articles. Dans certains cas, le magasinier récupère les articles lui-même.

- **Vérifier l'état des articles arrivés :**

Si les articles reçus sont défectueux ou la livraison est incomplète, le magasinier informe le gestionnaire de stock. Ce dernier informe à son tour le fournisseur pour changer les articles défectueux ou compléter la quantité manquante.

Si la livraison est complète et en bon état, le magasinier :

- Stock les articles ;
- Accuse la réception des articles ;
- Envoi du bon de livraison au gestionnaire de stock ;
- Informe le gestionnaire de stock pour mettre à jour la commande et établit un bon d'entrée
- Envoi de la facture au comptable.

- Diagramme de processus de la vente dans l'entreprise cliente

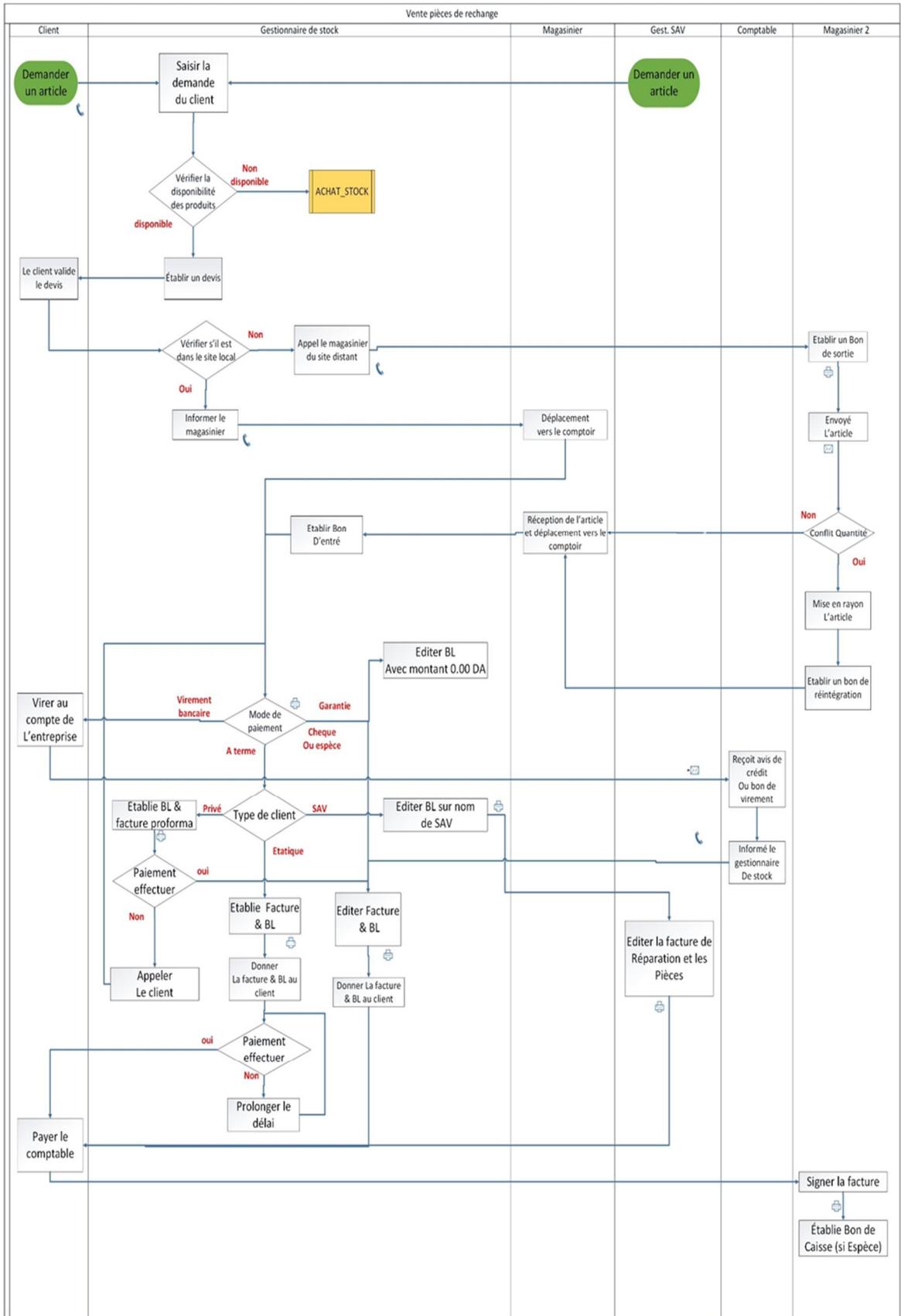


Figure 10 : Diagramme de processus Vente

Description :

- **Saisir la demande du client :**

A partir des demandes reçues par le client ou le SAV, le gestionnaire de stock saisie la demande de clientèle peut éventuellement établir un **devis**.

- **Indisponibilité de l'article ou disponibilité en local :**

Si le client demande un article qui n'est pas disponible, le gestionnaire de stock ajoute cet article à la liste d'approvisionnement pour l'acheter plus tard. En cas où la demande du client est urgente (pour SAV, Caisse engin, ...), le gestionnaire de stock effectue l'achat immédiatement.

Si l'article est disponible mais **réservé** pour un autre client, le gestionnaire de stock contacte le client concerné pour confirmer l'achat de l'article réservé ou d'annuler la réservation.

- **Disponibilité sur site distant :**

Si l'article est dans un autre site, le gestionnaire de stock demande de transférer l'article au magasinier du site où se trouve l'article, établit un bon de sortie et l'envoie avec l'article en question.

Le gestionnaire de stock du site local reçoit l'article et accuse la réception sur le bon de sortie et établit un bon d'entrée.

- **Mode de paiement :**

Il existe 4 types de paiement :

- Chèque ou Espèce : En cas de paiement par chèque ou espèce le gestionnaire de stock établit une facture et bon de livraison signé après avoir reçu le chèque ou l'espèce. Si le paiement est en espèce le comptable établit un bon de caisse.
- Virement bancaire : en cas de paiement par virement bancaire, le client effectue le virement au compte de l'entreprise. Lorsque le comptable reçoit un avis de crédit ou un bon de virement, il informe le gestionnaire de stock pour qu'il lui fasse un Bon de livraison et une facture signé.

- Garantie : ce mode de paiement est réservé au service après-vente. Dans le cas où un client à une garantie sur certain pièces de son engin. Le gestionnaire de stock livre au service après-vente un bon de livraison avec un montant 0.
- Paiement à terme : si le client veut effectuer un paiement à terme (client important ou fidèle) son paiement s'effectue selon la nature du client (privé, étatique ou service après-vente, étant donné que le service après-vente est considéré comme un client ordinaire)

Critique :

On remarque clairement que les processus actuels sont complexes, et implique un nombre assez important d'acteurs, ce qui implique une redondance des données avec la possibilité de non-cohérence de données.

III.3.2 Analyse du besoin et Extraction des règles de gestions

Après l'analyse du cahier des charges et plusieurs réunions avec le staff de SOGESI, nous avons déduit plusieurs règles de gestions, et nous avons réalisé jusqu'à présent 13 règles (entre paramétrage, adaptation, et création de modules OdoO), les règles sont les suivantes :

- 1- Possibilité d'indiquer qu'un client est un partenaire
- 2- Réservation d'articles à la commande
- 3- Ajout de la notification « Marchandises ni reprises ni échangées après 48h »
- 4- Avoir une liste de prix d'achat par fournisseur, avec rapport
- 5- Analyse par emplacement et par articles (analyse du stock)
- 6- Ajouter la liste des Wilaya d'Algérie
- 7- Ajouter une liste d'approvisionnement
- 8- Possibilité de remise fixe et remise en % dans la commande
- 9- Possibilité de créer des succursales et de leur associer des Showrooms
- 10- Ajouter des codes pour le client et pour le fournisseur
- 11- Gérer le retour d'articles (sachant qu'il y a des articles non retournables)
- 12- Ajouter tous les champs non existant dans les fiches (client, fournisseur, société)
- 13- Création des rapports avec toutes les informations de l'entreprise cliente

III.3.2.1 Diagrammes de processus métier

- Processus métier d'achat :

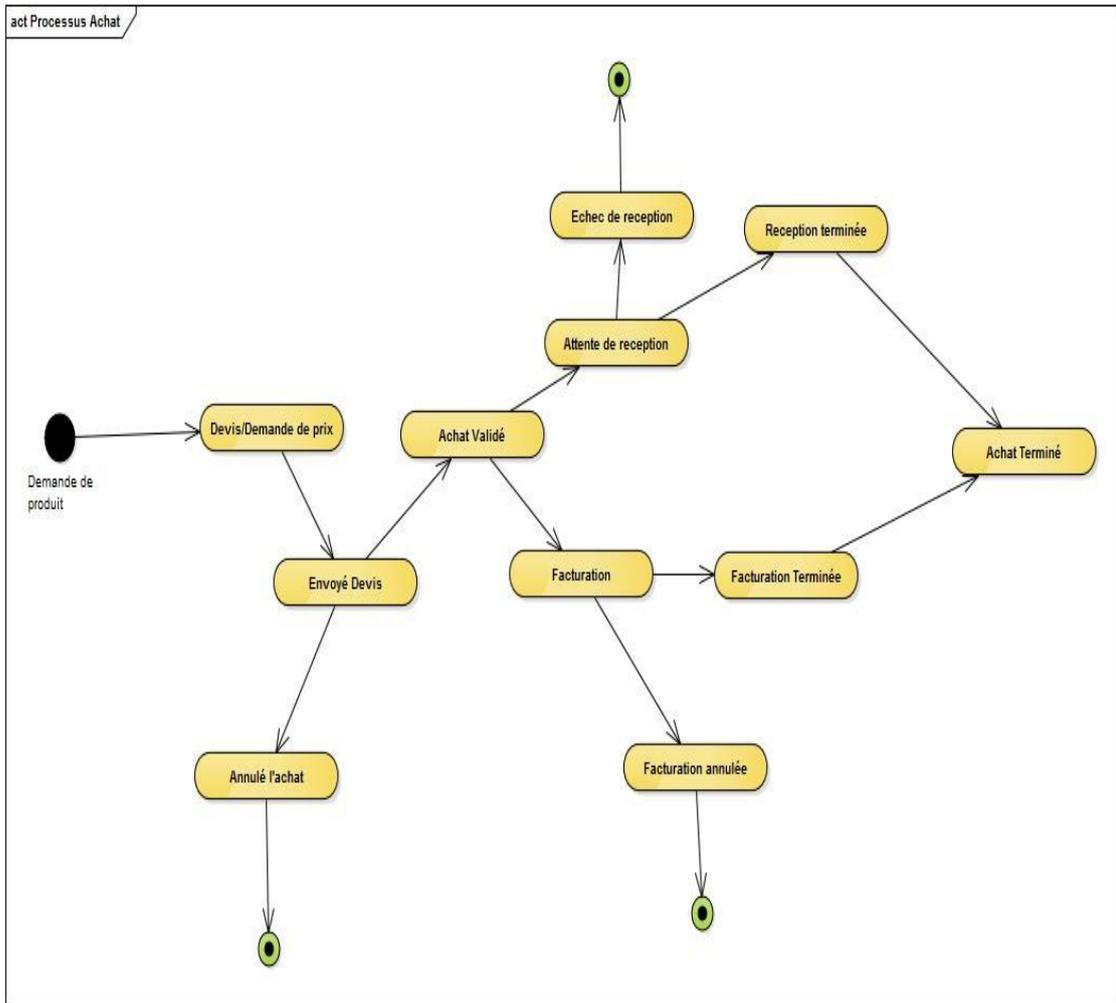


Figure 11 : Diagramme du processus d'achat

Description :

Une fois l'entreprise a besoin d'un produit le responsable, déclenche une opération d'achat qui commence par la création d'un devis pour la demande de prix envoyer au fournisseur, et après la validation d'achat par le responsable, la demande de prix doit être converti en une commande d'achat validée. Cette dernière passe, par la suite, par le service de facturation et de réception. L'achat se termine une fois elle est réceptionnée et facturée.

- Processus métier vente

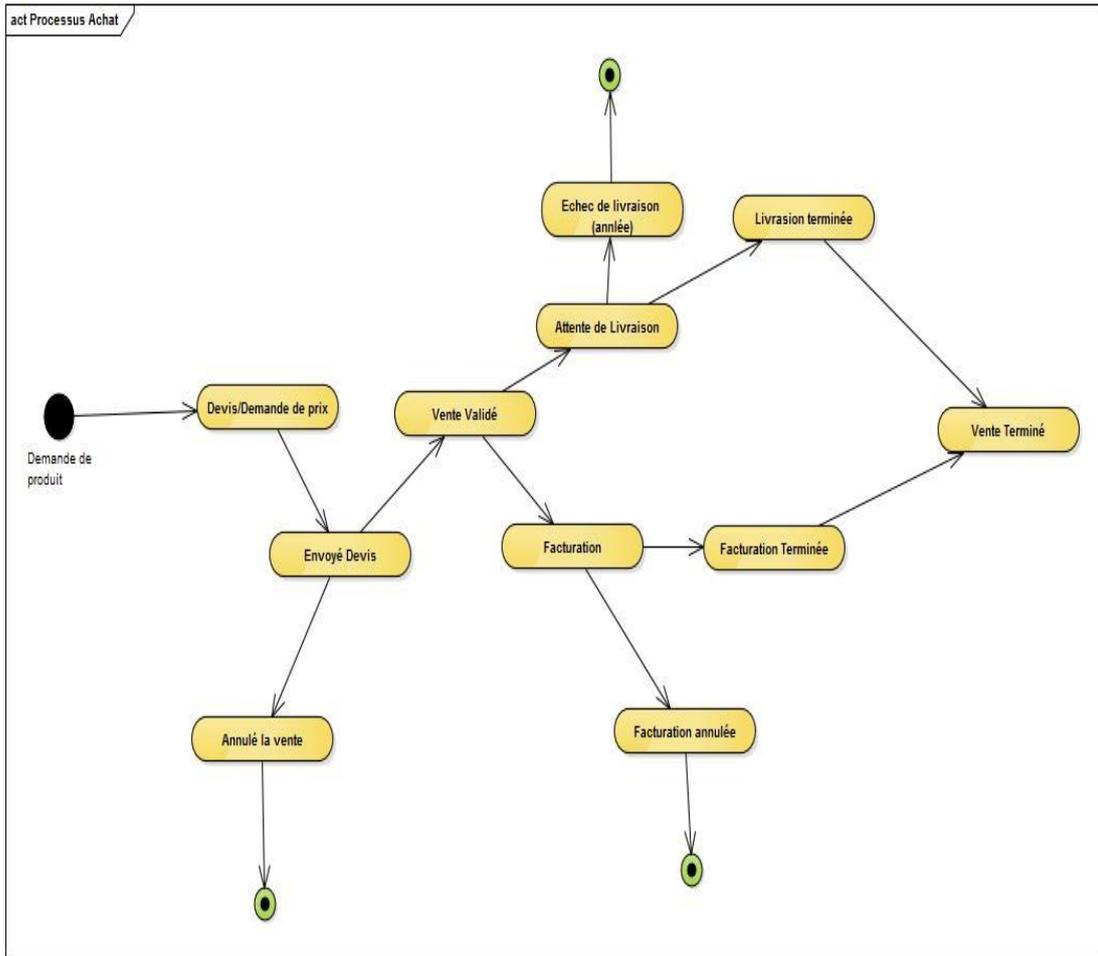


Figure 12 : Diagramme du processus de vente

Description :

Une fois le commercial reçoit un bon de commande de la part d’un client, il crée un devis qui doit être envoyé par la suite à ce client par email pour vérifier les informations afin de la convertir en une commande de vente validée. Pour ensuite passer par le service de facturation et de livraison. La vente se termine une fois elle est livrée et facturée.

Remarque :

Dans Odoo, l’achat et la vente ont la même logique, ç’est à dire dans le cas **d’achat** on est **client** chez un **fournisseur**, et dans le cas de **vente** on est **Fournisseur** pour un **client** ; comme le montre le schéma illustratif suivant :

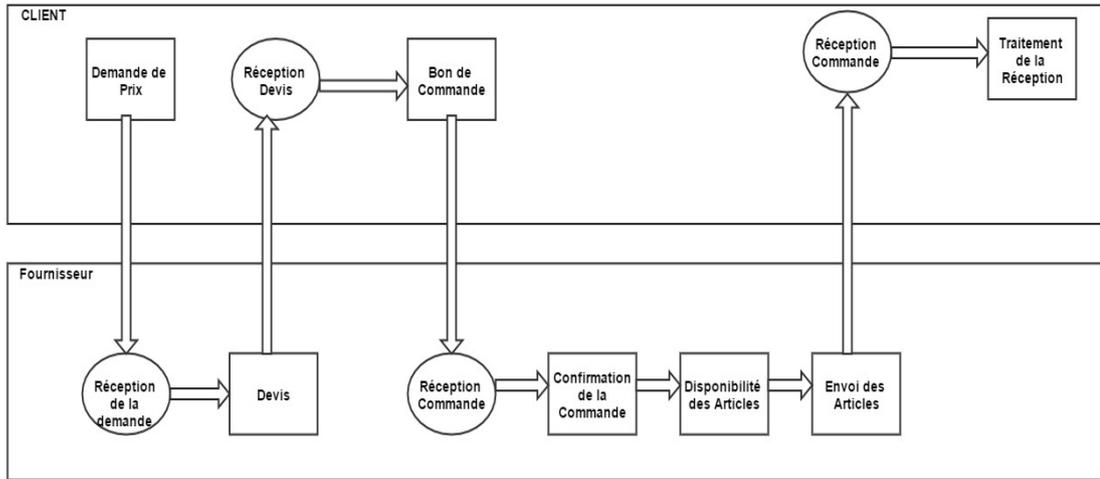


Figure 13 : Schéma illustratif de l'achat-Vente

III.3.2.2 Diagrammes de cas d'utilisations

- Diagramme de cas d'utilisation dans le cas de vente

La gestion des ventes consiste à gérer l'ensemble d'activités des commerciaux, et assurer le suivi des commandes des clients tout en gérant les factures et les dossiers des clients.

Les utilisateurs sont tenus d'utiliser le système dans la réalisation de leurs tâches qui sont accordées.

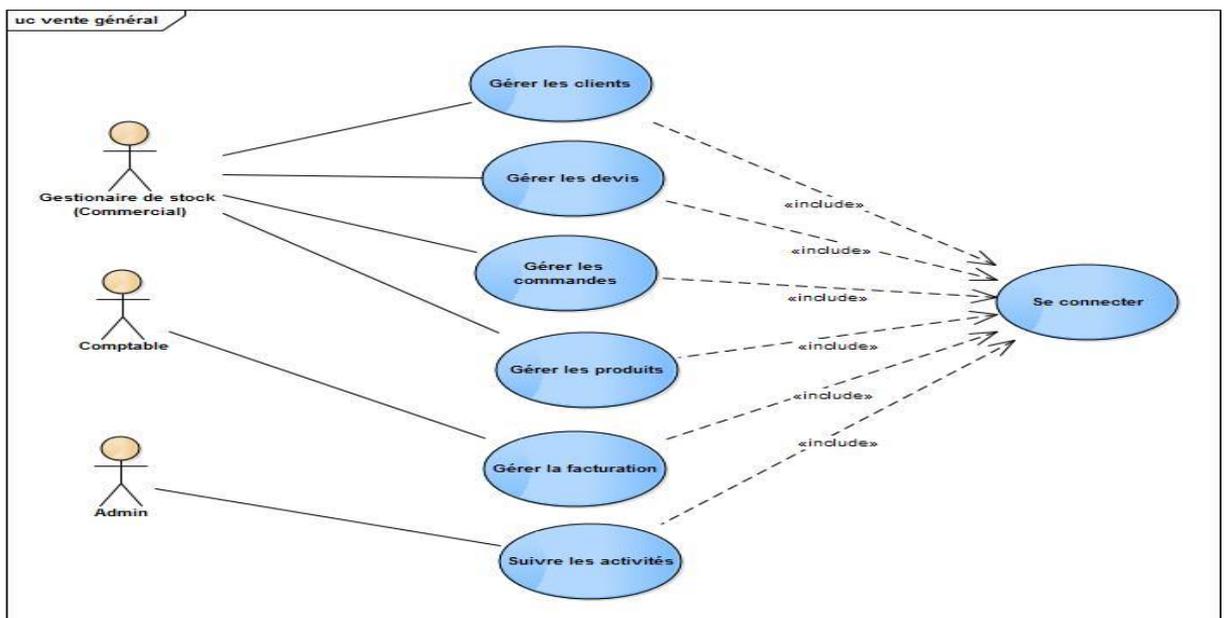


Figure 14 : Diagramme cas d'utilisation (vente en générale)

Ce diagramme regroupe les fonctionnalités globales que doit assurer le système vis-à-vis la gestion des ventes, et par la suite on va détailler ce diagramme pour bien éclaircir l'utilité de chaque cas d'utilisation.[16]

Description des cas d'utilisation

- **Gérer les clients** : ce cas d'utilisation permet aux acteurs de créer et mettre à jour les fiches liées aux clients, comme il doit assurer le suivi des clients, comme il permet d'associer les contacts au client si le client est une société.
- **Gérer les commandes** : permet de gérer et de faire le suivi des commandes vis-à-vis au client avec les rapports imprimés.
- **Gérer la facturation** : ce cas assure la création et la mise à jour des factures, comme il permet de faire le suivi des factures jusqu'à la dernière étape qui est le paiement.
- **Gérer les produits** : comporte la création des produits ou bien les mettre à jour. Aussi, ce cas d'utilisation permet de visualiser la variation des stocks et de signaler les besoins vis-à-vis les produits matériels non disponibles.
- **Suivi d'activité** : contient l'ensemble des tableaux de bords qui font l'analyse des ventes selon les indicateurs (comme le chiffre d'affaire par mois, le nombre des ventes par client, analyse des factures).

- Diagramme de séquence des ventes

Description des interactions entre différents acteurs du système :

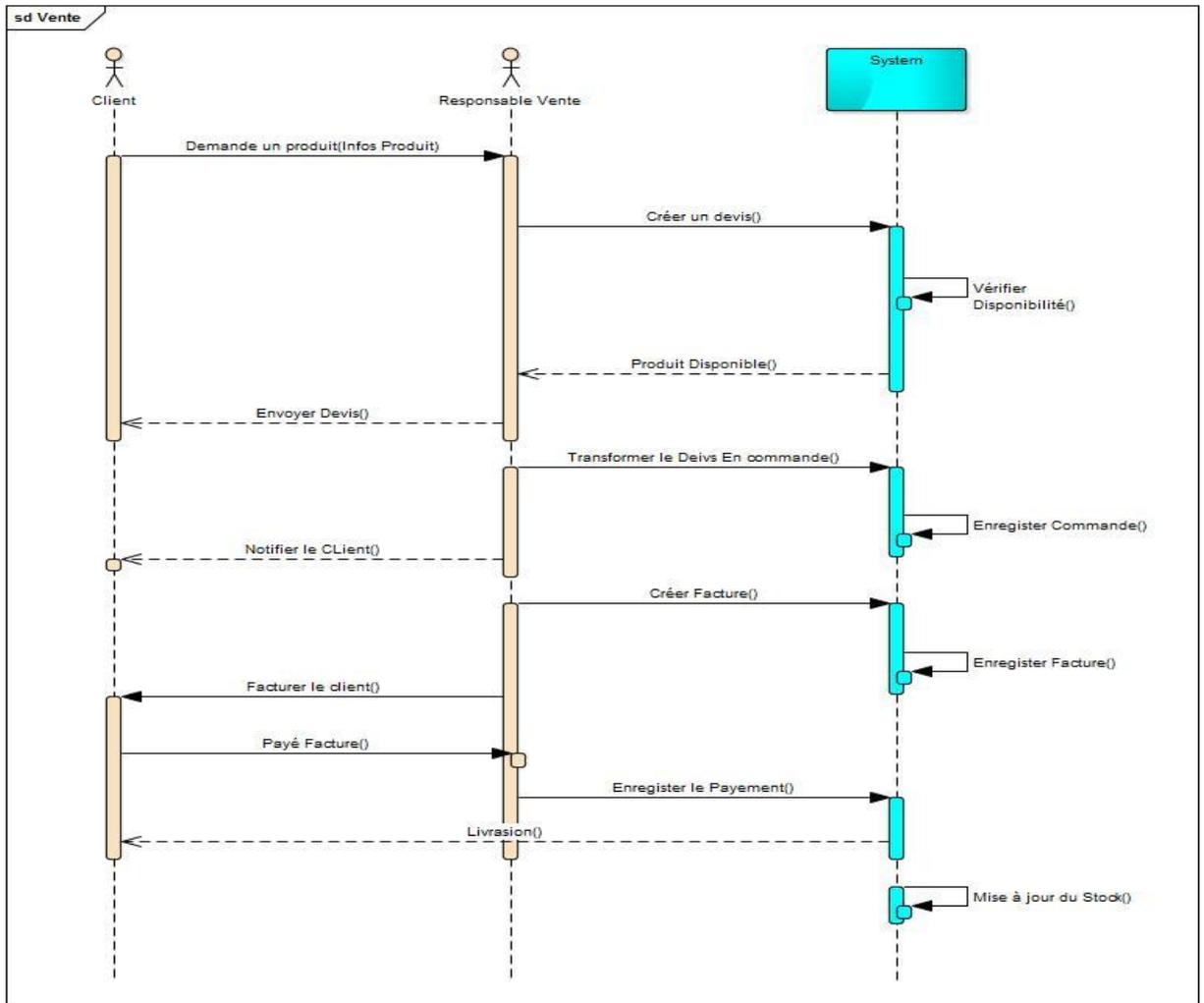


Figure 15 : Diagramme de séquence Vente

Les scénarios :

Dans le processus de vente on a l'acteur responsable qui interagi avec le système directement.

- Un client qui a besoin d'un produit doit interagir avec le responsable de vente pour établir un devis
- Après l'établissement du devis le système doit vérifier l'état du stock
- Après l'enregistrement du devis le responsable valide la commande et notifie le client pour confirmer la commande
- Finalement générer la facture qui sera payé par le client

- Diagramme de cas d'utilisation dans le cas de l'achat-stock

La gestion des achats consiste à gérer les produits entrants et sortants dans le stock comme elle assure la gestion des achats des produits.

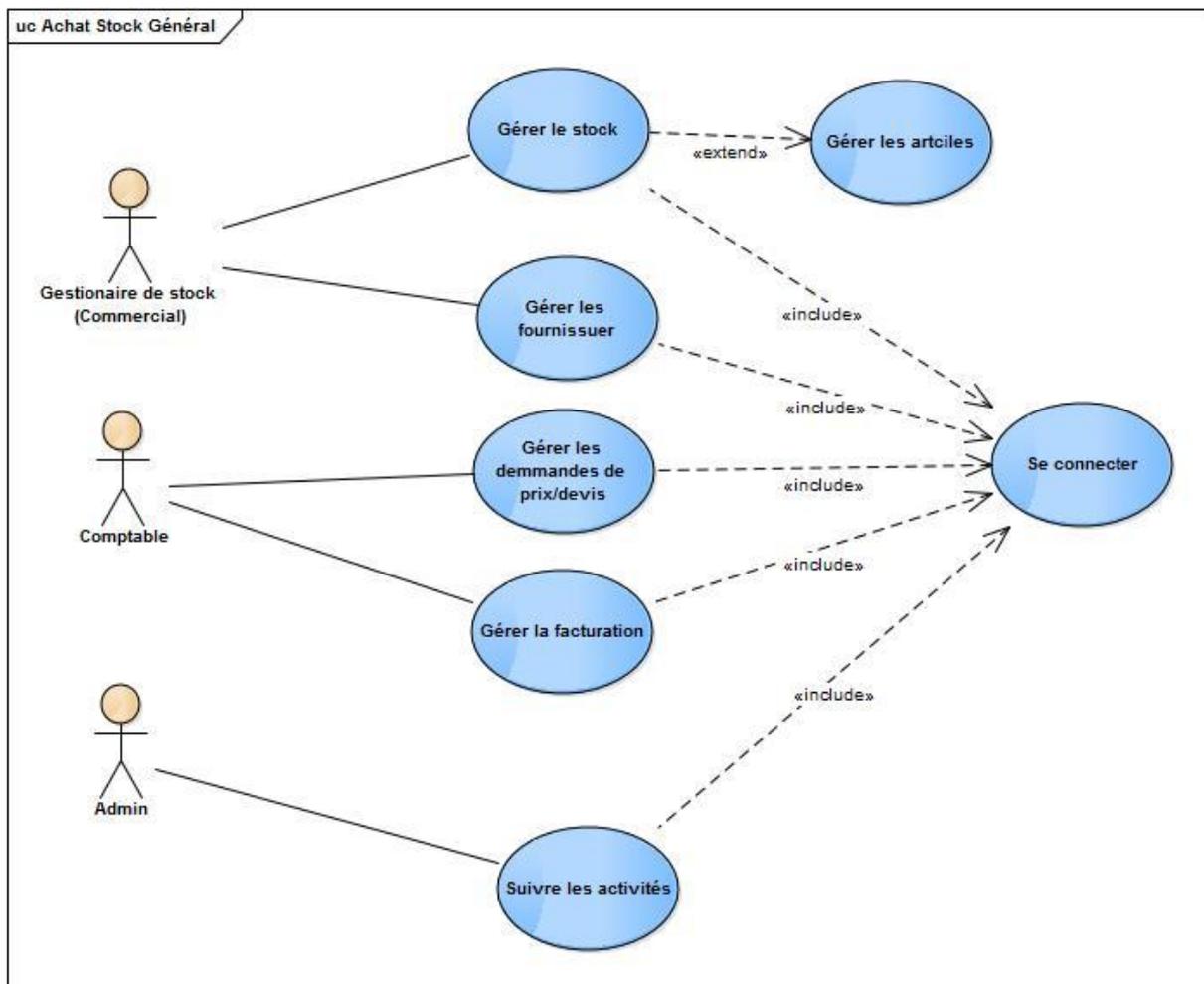


Figure 16 : Diagramme cas d'utilisation (Achat-Stock)

Ce diagramme regroupe les fonctionnalités globales que doit assurer le système vis-à-vis la gestion des achats et par la suite, on va détailler ce diagramme pour bien montrer l'utilité de chaque cas d'utilisation.

Description des cas d'utilisation :

- **Gérer le stock** : ce cas permet le contrôle des produits entrants et sortants et la mise à jour du stock si par exemple un produit est livré il doit être mis à jour automatiquement, comme il doit le mettre à jour si un produit est acheté.
- **Gérer les fournisseurs** : permet de créer des dossiers des fournisseurs et de les mettre à jour une fois il y a une notification à ajouter.

- **Gérer les demandes d'achat** : permet de faire le suivi des demandes d'achat qu'on peut les créer, mettre à jour et les analyser.
- **Gérer la facturation** : ce cas assure la création et la mise à jour des factures, comme il permet de faire le suivi des factures.
- **Gérer les produits** : comporte la gestion des produits. Aussi, ce cas d'utilisation permet de visualiser la variation des stocks et de signaler les besoins vis-à-vis les produits non disponible.
- **Suivi d'activité** : contient l'ensemble des tableaux de bords qui font l'analyse des produits selon les indicateurs.

- Diagramme de séquence des achats :

Après le déclenchement d'insuffisance d'un article dans le stock ou la demande d'un article non existant le processus d'achat e déclenche, on peut présenter le diagramme de séquence suivant :

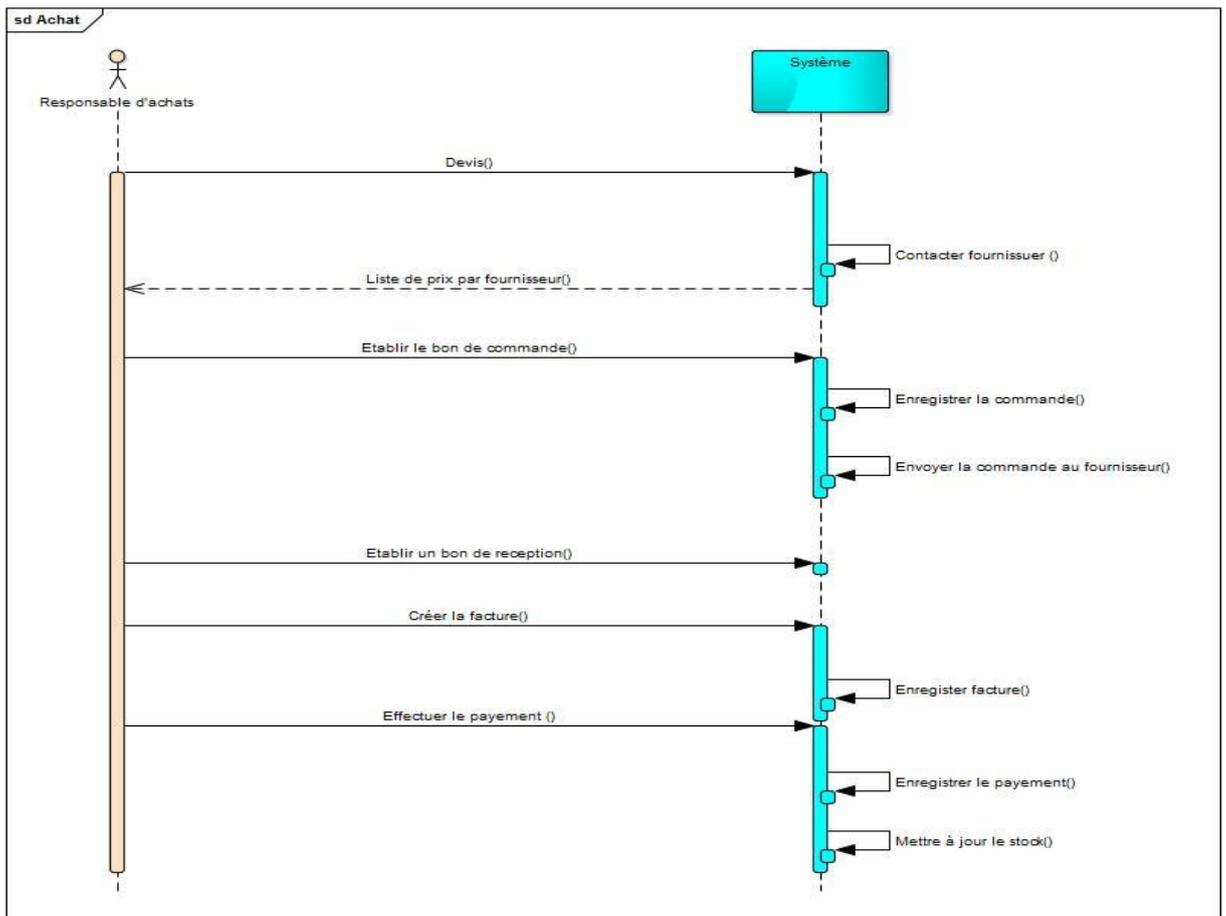


Figure 17 : Diagramme de séquence Achat

Les scénarios :

Dans le processus d’achat, on a l’acteur responsable des achats qui interagit avec le système central (ERP).

- Après le déclenchement d’insuffisance d’un article dans le stock ou la demande d’un article non existant le processus d’achat e déclenche, le responsable d’achat demande une liste des prix
- Après la consultation des prix des produits par les entités de l’entreprise le responsable établit le bon de commande.
- Après la validation de commande, le système doit traiter les données et enregistrer la commande.
- Le responsable crée le bon de réception de commande.
- Après la réception de la commande le système génère la facture finale a payé.
- Après le payement et la réception le stock sera mis à jour

III.2.3 Diagrammes de classes

Pour répondre aux règles de gestion, on adapter et créer plusieurs fonctionnalités, les diagrammes suivant décriront les classes majeures d’Odoo et les classes qu’on a créées (avec attributs et méthodes essentiels).

- Diagramme de classe « Approvisionnement » :

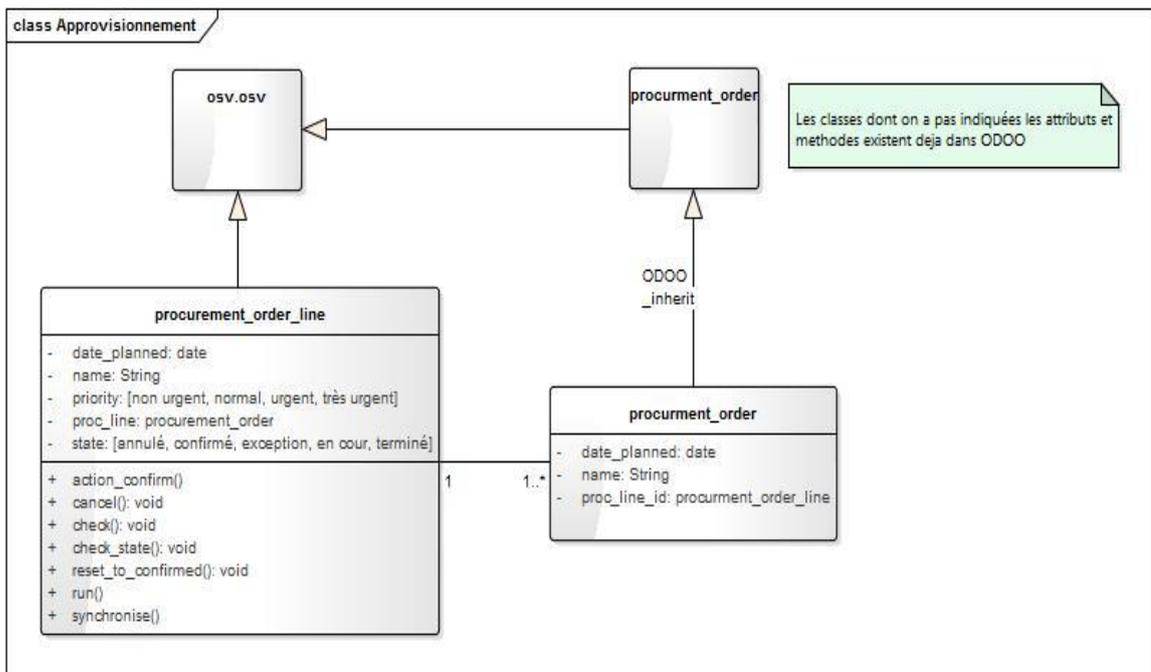


Figure 18 : Diagramme de classe « Approvisionnement »

- Diagramme de classe « formulaire » :

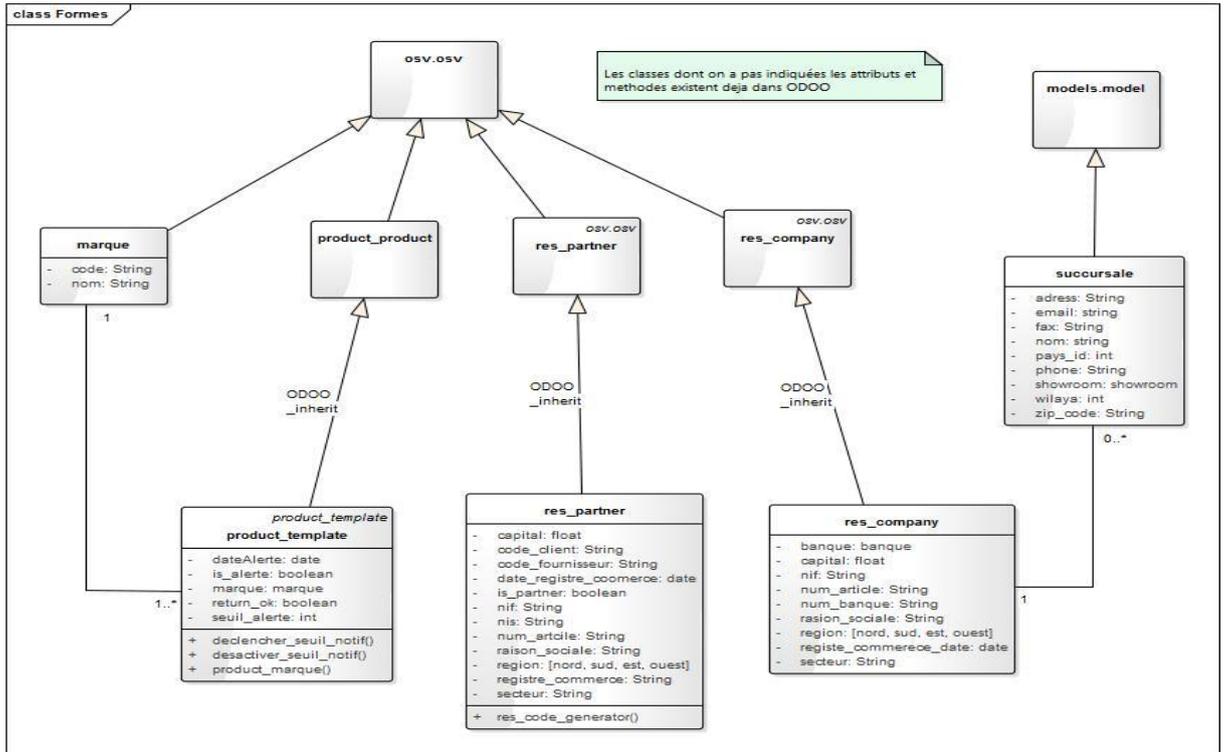


Figure 21 : Diagramme de classe « Formes »

III.2.4 Une solution pour chaque règle de gestion

Dans cette section nous allons voir l’application réelle qui répond aux règles de gestion identifiées dans la section III.3.2

1- Possibilité d’indiquer qu’un client est un partenaire :

L’entreprise à des traitements spéciaux pour ces partenaires, comme les prix et les devis, etc.



Figure 22 : Une partie de la fiche client

2- Réserveation d'articles à la commande :

Cette réserveation est très importante car un produit commandé par un client ne peut pas être vendu pour un autre.

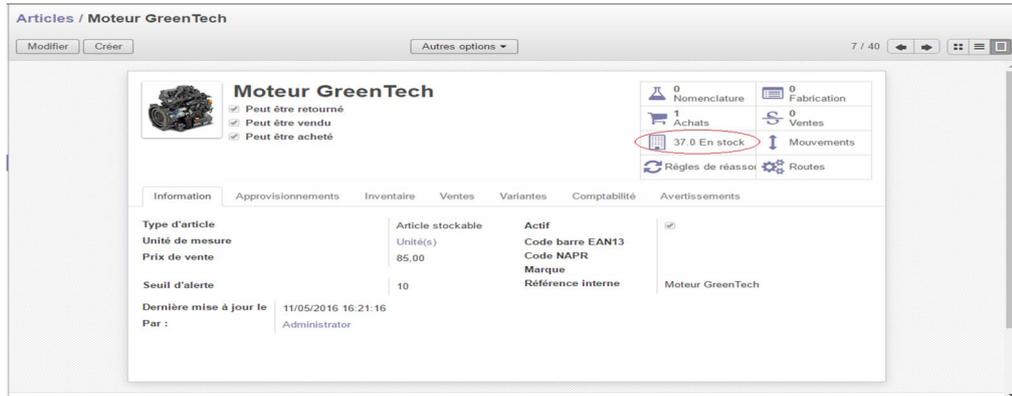


Figure 23 : Etat du Stock

Si on commande une certaine quantité de ce produit, elle sera automatiquement réservée, et le stock sera calculé selon cette valeur tant que la commande est valide.

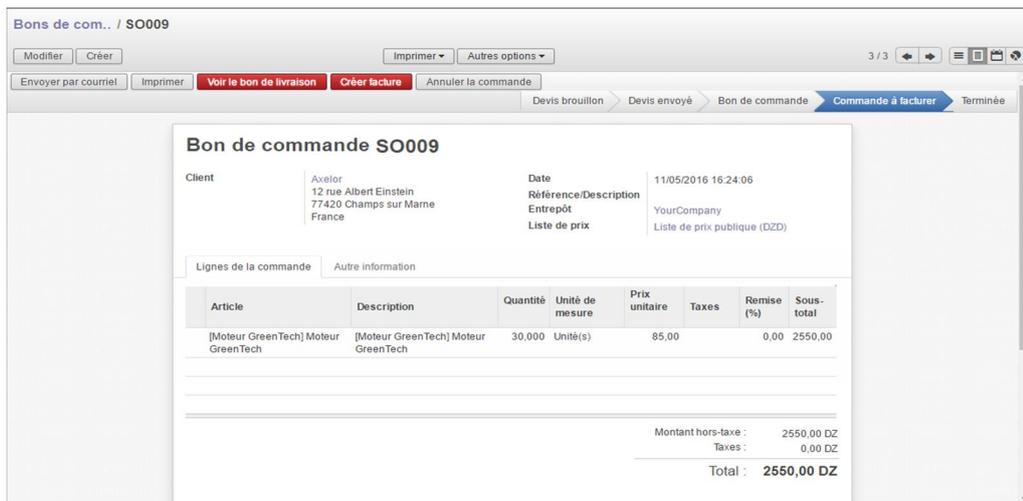


Figure 24 : Commande validée

Si on passe une commande avec une quantité supérieure au stock disponible (avec réserveation), on aura un message d'erreur comme le montre la figure suivante :

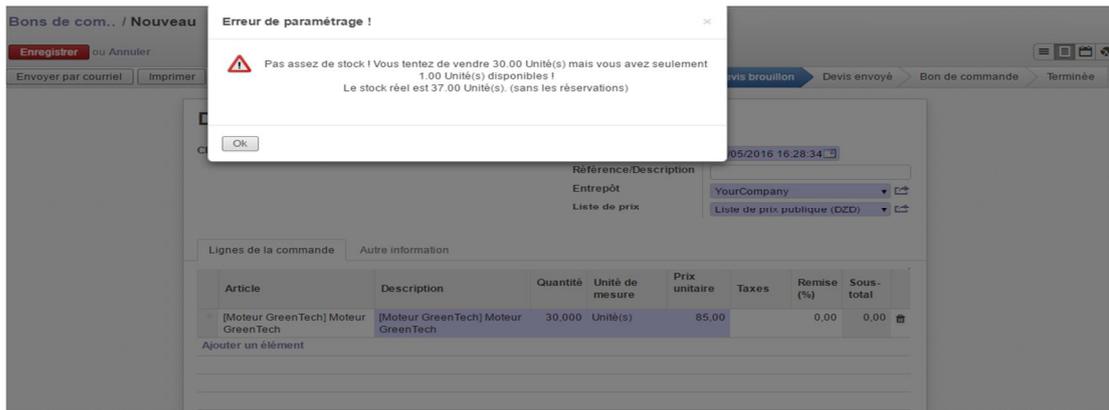


Figure 25 : Alerte de quantité réservée

3- Ajout de la notification « Marchandises ni reprises ni échangées après 48h » :

Tous les clients de l'entreprise doivent savoir que la marchandise est ni reprise ni échangée après 48H, cette notification doit être ajoutée dans le bon de livraison, comme dans la figure suivante :

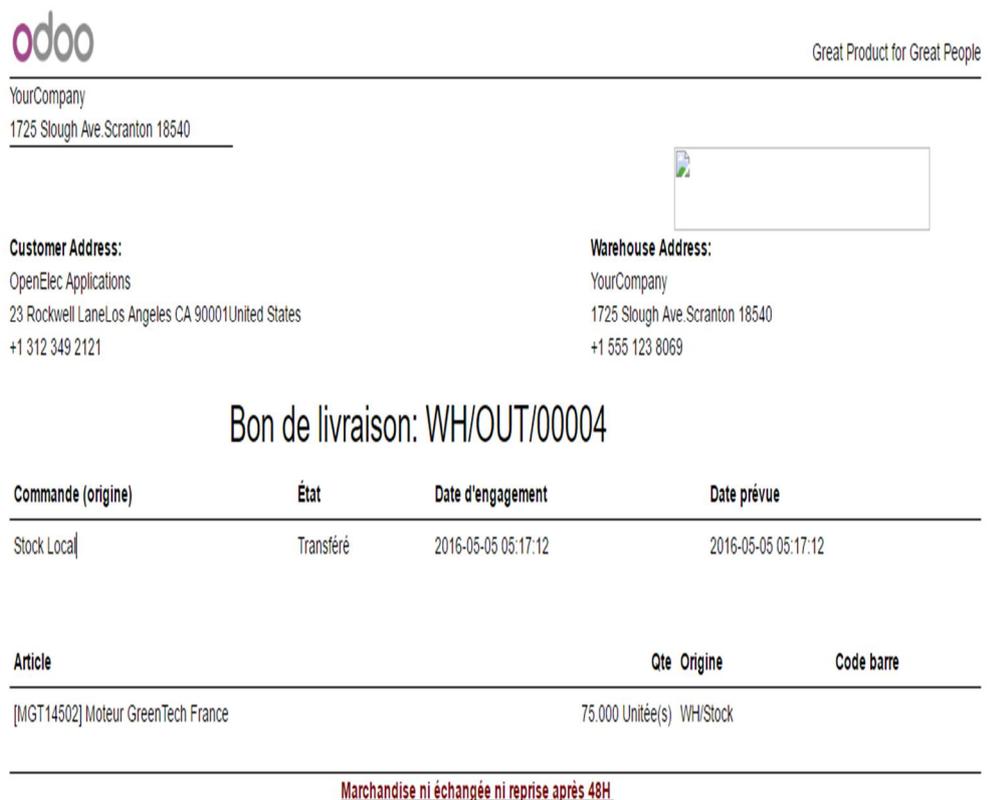


Figure 26 : Bon de livraison avec Notification

4- Avoir une liste de prix d'achat par fournisseur, avec rapport :

L'entreprise cliente souhaite avoir une liste de prix d'achat par fournisseurs, de plus la possibilité d'imprimer un rapport détaillé. Le principe de cette liste est qu'à chaque confirmation d'une commande d'achat, les prix pour chaque fournisseur sont automatiquement enregistrés dans la base de donn e.

On a donc cr e une classe « purchase_order_history » indiqu e dans la figure 28

```
class purchase_order_history(models.Model):
    _name = 'purchase.order.history'
    _description = "Historique des prix d'achat par fournisseur"
    _order = 'product_id desc, supplier_id desc'
    product_id = fields.Many2one('product.product', 'Product', required=True)
    supplier_id = fields.Many2one('res.partner', 'Fournisseur', required=True)
    product_qty = fields.Float('Quantit ')
    prix_unit = fields.Float('Prix unitaire')
    date = fields.Date("Date d'achat")
    origin = fields.Many2one('purchase.order', 'Bon de commande')
```

Figure 27 : La classe “purchase_order_history”

L'ORM traduit donc la classe en une table dans la BDD (Figure 29).

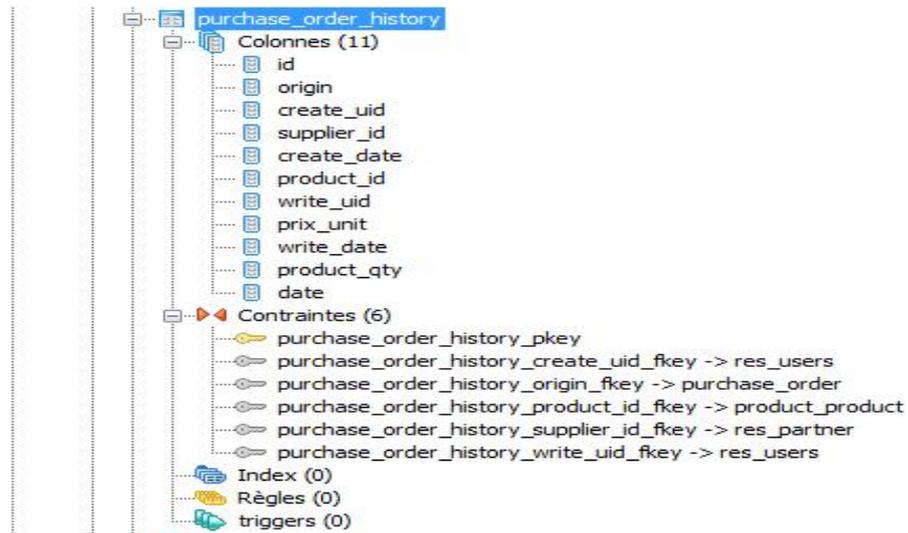


Figure 28 : La table purchase_order_history

La liste s'affiche sous forme d'un arbre, créée en xml comme suit :

```
<tree string="Liste de prix d'achat" create="false">
  <field name="product_id"/>
  <field name="supplier_id"/>
  <field name="product_qty"/>
  <field name="prix_unit"/>
  <field name="origin"/>
  <field name="date"/>
</tree>
```

Figure 29 : Un arbre en XML

Les enregistrements sont visibles dans le menu Achats ---> Liste de prix d'achat (Figure 31). Les lignes sont par défaut regroupées par l'article et le fournisseur.

Groupe	Product	Fournisseur	Quantité	Prix unitaire	Bon de commande	Date d'achat
▼ Alternateur (1)			10,00	1500,00		
▼ PIECE DR, MachinePro (1)			10,00	1500,00		
	Alternateur	PIECE DR, MachinePro	10,00	1500,00	PO00001	11/05/2016
▼ Compresseurs spiro-orbitaux (1)			15,00	2000,00		
▼ PIECE DR, MachinePro (1)			15,00	2000,00		
	Compresseurs spiro-orbitaux	PIECE DR, MachinePro	15,00	2000,00	PO00001	11/05/2016
▼ Courroies (1)			20,00	500,00		
▼ PIECE DR, MachinePro (1)			20,00	500,00		
	Courroies	PIECE DR, MachinePro	20,00	500,00	PO00001	11/05/2016
▼ Moteur GreenTech (France) (1)			50,00	500000,00		
▼ Moral tale3 (1)			50,00	500000,00		
	Moteur GreenTech (France)	Moral tale3	50,00	500000,00	PO00002	11/05/2016

Figure 30 : Liste de prix d'achat par fournisseur

On peut aussi accéder à chaque ligne séparément grâce à une forme créé en XML comme suit :

```

<form string="Liste de prix d'achat" create="false">
  <sheet>
    <group>
      <div>
        <field name='product_id' readonly="1"/>
      </div>
    </group>
    <group>
      <field name='supplier_id' readonly="1" />
    </group>
    <group>
      <group>
        <field name='product_qty' readonly="1"/>
        <field name='prix_unit' readonly="1"/>
      </group>
      <group>
        <field name='date' readonly="1"/>
        <field name='origin' readonly="1"/>
      </group>
    </group>
  </sheet>
</form>

```

Figure 31 : Formulaire en XML

La figure suivante (Figure 33) indique comment s’affiche le formulaire.

Carburateur			
Fournisseur	Audi SpareParts Algeria		
Quantité	20,00	Date d'achat	23/05/2016
Prix unitaire	600,00	Bon de commande	PO00001

Figure 32: Formulaire d'une ligne d'achat

Le client peut imprimer la liste et même spécifier l’article, le fournisseur ou bien période d’achat (Figure 35).

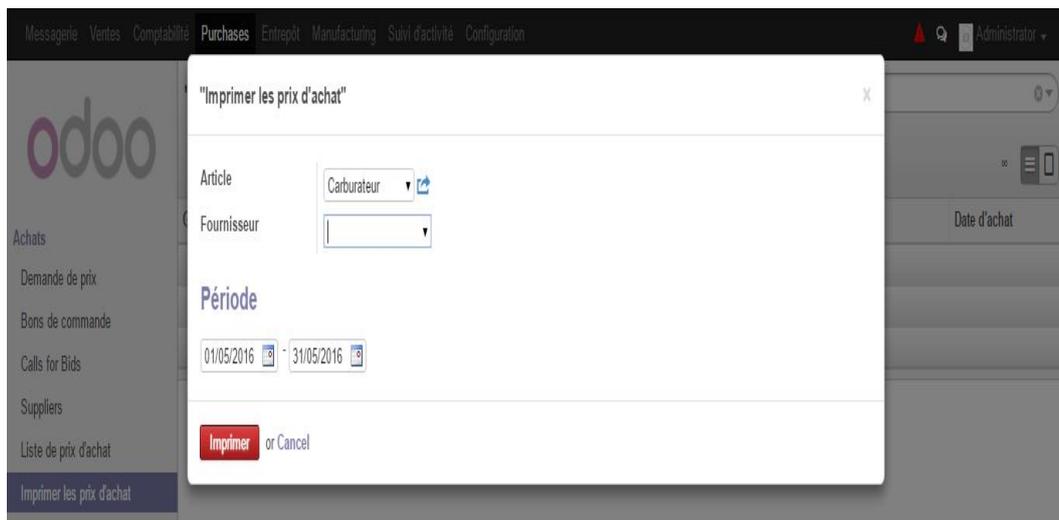


Figure 33 : Impression de la liste des prix d’achat par fournisseur

CONCESSIONNAIRE D'ENGINS DE TRAVAUX PUBLICS ET BATIMENTS

Direction Générale:

- Tél :
 - Fax :
 - Email : info@yourcompany.com
 - SiteWeb : http://www.yourcompany.com

Liste de prix d'achat

Article : Carburateur

Période : Du 01/05/2016 Au 31/05/2016

N°	Article	Fournisseur	Quantité	Prix unitaire	Date	Bon de commande
1	Carburateur	EnginDZ	50,0	300,0	23/05/2016	PO00002
2	Carburateur	Audi SpareParts Algeria	20,0	600,0	23/05/2016	PO00001

Figure 34 : Rapport liste prix d'achat / Fournisseur

5- Analyse par emplacement et par articles (analyse du stock) :

L'entreprise doit pouvoir consulter l'état du stock, ainsi d'effectuer une analyse par emplacement et par articles.

La figure 36 montre une analyse par emplacement, c'est-à-dire afficher l'emplacement et les articles stockés dans cet emplacement

Articles / Moteur Gree... / Stock actuel	Emplacements internes x	Emplacement x			
Groupe	Article	Quantité	Emplacement	Date d'entrée	Valeur d'inventaire
▼ WH/Stock/Étagère 1 (2)		37,00			2960,00
	[Moteur GreenTech] Moteur GreenTech	6,00	WH/Stock/Étagère 1	05/05/2016 04:17:05	480,00
	[Moteur GreenTech] Moteur GreenTech	31,00	WH/Stock/Étagère 1	05/05/2016 04:17:05	2480,00

Figure 35 : Analyse par emplacement

La figure 37 montre une analyse par article, c'est-à-dire afficher l'article et les différents emplacements dans les quels cet article est stocké

Quants					
<input type="text" value="Article"/> <input type="text" value="Quantité"/> <input type="text" value="Emplacement x"/>					
Groupe	Article	Quantité	Emplacement	Date d'entrée	Valeur d'inventaire
▶ Alternateur (8)		25,00			0,00
▶ Compresseurs spiro-orbitaux (1)		15,00			0,00
▶ Courroies (1)		20,00			0,00
▼ Moteur GreenTech (France) (1)		50,00			0,00
▼ 50,00 (1)					0,00
▼ WH/Stock (1)		50,00			0,00
	<input type="checkbox"/> Moteur GreenTech (France)	50,00	WH/Stock	11/05/2016 16:00:51	0,00
▶ Pompe à Eau (1)		15,00			0,00

Figure 36 : Analyse par articles

6- Ajouter la liste des Wilaya d'Algérie :

Pour plus de faciliter, nous avons ajouté la liste des wilayas de l'Algérie

Partenaire

Adresse

Slogan de la société

Site Web

YourCompany

Tlemcen

Tlemcen

Nord-ouest

Algérie

Société cilente

http://www.client.c

Wilaya ▼ 13000

- Adrar
- Chlef
- Laghouat
- Oum El-Bouaghi
- Batna
- Béjaia
- Biskra
- Chercher plus...
- Créer et modifier...

Tél.

Fax

Courriel

Taxe ID

registre de la société

modifié le

S

pital

N° Compte bancaire

Figure 37 : Liste de Wilaya d'Algérie

7- Ajouter une liste d’approvisionnement :

Nous avons ajouté la possibilité de créer une liste d’approvisionnement, qui contient plusieurs approvisionnements et les lance en une seule fois à travers une classe « procurement_order_line » qui est en relation « one to many » avec la classe « procurement_order » (Figure 39)

```
class procurement_order_line(osv.osv):
    _name = 'procurement.order.line'
    _description = "ligne d'approvisionnement"
    _order = "id desc"
    _columns = {
        'ref': fields.char("Référence de liste", required = True),
        'date_plan': fields.datetime('Date planifiée', required=True, select=True, track_visibility='onchange'),
        'proc_line': fields.one2many('procurement.order', 'proc_line_id', required = True),
        'priority': fields.selection(PROCUREMENT_PRIORITIES, 'Priorité', required=True, select=True, track_visibility='onchange'),
        'state': fields.selection([
            ('cancel', 'Cancelled'),
            ('confirmed', 'Confirmed'),
            ('exception', 'Exception'),
            ('running', 'Running'),
            ('done', 'Done')
        ], 'Status', required=True, track_visibility='onchange', copy=False),
    }
```

Figure 38 : La classe 'procurement_order_line'

Pour créer une liste d’approvisionnement il suffit d’aller dans le menu Entrepôt - --> Liste d’approvisionnement.

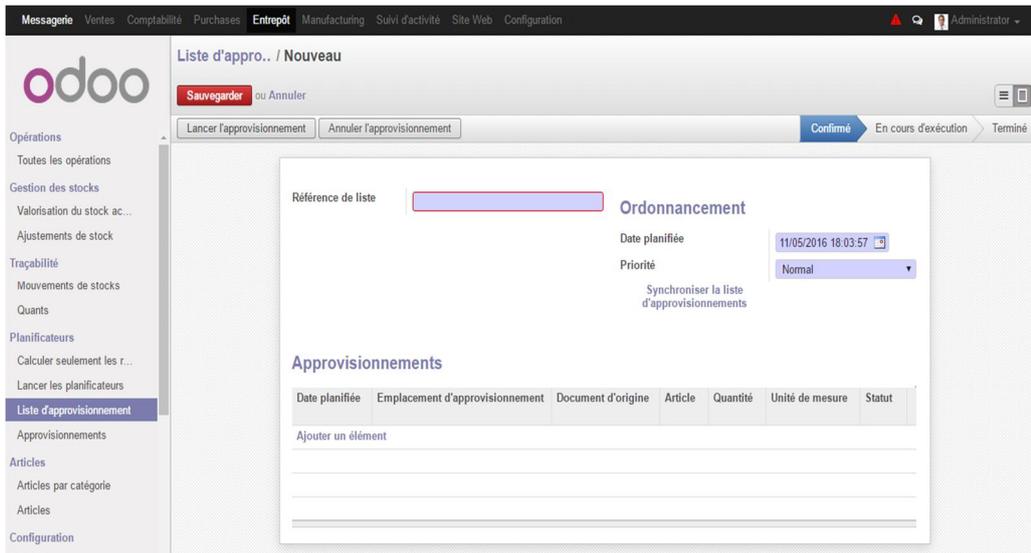


Figure 39 : Liste d’approvisionnement

On peut ajouter des approvisionnements en cliquant sur ajouter un élément, on obtient donc le formulaire suivant (Figure 41) :

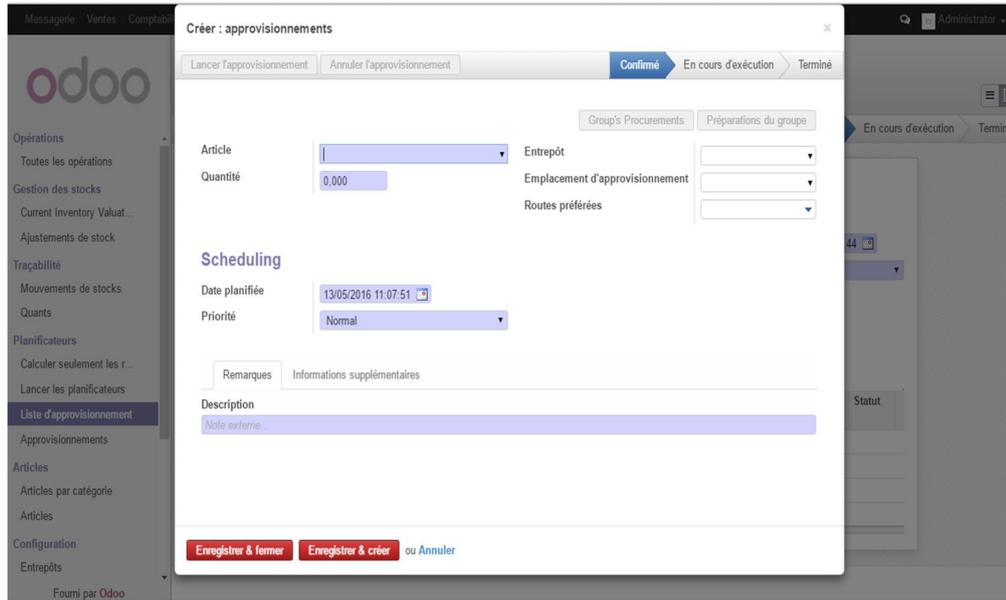


Figure 40 : Création d'un approvisionnement

8- Possibilité de remise fixe et remise en %, les droits de timbre et autres TVA

L'entreprise est libre de faire des remises à ces clients, soit une remise fixe, ou en pourcentage par rapport au total, comme le montre les figures 42 et 43.

Facture Brouillon

Client: Messabihi
8, rue Amir Abdel Kader, La marine
Saïda 200 20000
Algérie

Date de facture: []
Type de remise: Pourcentage
Type de vente: []

Position fiscale: []

Modalité de paiement: Liquidités (DZD)

Pièce comptable: []

Lignes de facture

Article	Description	Quantité	Unité de mesure	Prix unitaire	Remise	Taxes	Montant
Moteur AUDI 8 cylindre	Moteur AUDI 8 cylindre	1,000	Unité(s)	700000,00	10,00	20.0	630000,00

Sous-total : 630000,00 DZ
Remise HT(%): 10,00
Montant de remise : 63000,00 DZ
Taxes : 126000,00 DZ
Autre TVA : 20.0
Montant autre TVA : 126000,00 DZ
Timbre : 8820,00 DZ
Total : 700560,00 DZ

Figure 41 : Remise, timbre et autre TVA dans la facture

The screenshot shows the 'Demande de prix /' (RFQ) form in Odoo. The form includes fields for 'Fournisseur', 'Référence du fournisseur', 'Liste de prix', 'Date de commande' (11/05/2016 18:19:29), and 'Deliver To' (Your Company). Below these are tabs for 'Articles', 'RFQ & Bid', and 'Livraisons & Factures'. A table with columns 'Article', 'Description', 'Date prévue', 'Compte analytique', 'Quantité', 'Prix unitaire', 'Taxes', and 'Sous-total' is present, with an 'Ajouter un élément' button. On the right, a summary section shows: 'Montant hors taxe : 0,00 DZ', 'Remise % : 0,00', 'Montant de remise : 0,00 DZ', 'Taxes : 0,00 DZ', and 'Total : (mise à jour) 0,00 DZ'. A sidebar on the left contains navigation options like 'Achats', 'Bons de commande', and 'Articles'.

Figure 42 : Bon de commande avec remise en pourcentage

9- Possibilité de créer des succursales et de leur associer des Showrooms :

L'entreprise à un certain nombre de succursales et de Showrooms, dont elle doit avoir la possibilité de les gérés.

La figure 44 montre les informations de l'entreprise, avec la possibilité d'ajouter un succursale (figure 45) et Showroom (figure 46).

The screenshot shows the 'Sociétés' (Companies) form in Odoo. It features a logo upload icon and a 'Nom de la société' field with a dropdown menu showing 'PIECE DR', 'SARL par exemple', and 'SARL par exemple'. A checkbox 'Est un partenaire ?' is also present. Below the company name, there is a note: 'Cliquez pour définir le logo de votre société.' with an arrow pointing to the logo icon. Further down, there are fields for 'Capital' (0,00) and 'N° Compte bancaire' (20 Caractères max). The form is divided into sections: 'Succursale' with a table for 'Titre', 'Wilaya', and 'Tél'; and 'Comptes bancaires' with a table for 'Numéro de compte', 'Nom de la banque', 'Afficher sur les rapports', and 'Titulaire du compte'. At the bottom, there are 'Enregistrer' and 'Annuler' buttons.

Figure 43 : Informations de la société

Créer : Succursale x

Succursale

Email

Tél

Fax

Ville	Tél	Fax
Ajouter un élément		

ou [Annuler](#)

Figure 44 : Ajout d'une succursale

Créer : Showroom x

Adresse

Adresse 2

Ville

Tél

Fax

ou [Annuler](#)

Figure 45 : Ajout d'un Showroom

10- Ajouter des codes pour le client et pour le fournisseur :

L'entreprise a plusieurs clients et fournisseurs, de plus de leurs informations un code de type (CL XXX) sera attribué au client, et un autre de type (Fr XXX) au fournisseur

Dans l'exemple de la figure 47, c'est la fiche d'un client qui est au même temps un fournisseur, donc il a deux code (C10009 code client) et (Fr0009 code fournisseur).

Dans la figure 48 c'est le cas d'un client simple, donc il a un seul code (C1007 code client).

The screenshot shows the profile for 'SARL ENGIN-Pro'. The client code 'C10009' and supplier code 'Fr0009' are circled in red. The interface includes a top navigation bar with icons for 'Facturé', 'Ventes', 'Fact. Fournisseur', 'Achats', 'Ecritures comptat', and 'Contrats'. The main area is divided into 'Adresse' and 'Site Web' on the left, and a list of fields on the right including 'Poste occupé', 'Tél.', 'Tél. portable', 'Fax', 'Courriel', 'Civilité', 'NIF', 'A.I', 'NIS', and 'Capital'. The 'Capital' field is set to '0,00'. Below the main area are tabs for 'Notes internes', 'Achats-Ventes', 'Comptabilité', and 'Avertissements'. At the bottom, the 'Vendeur' section has 'Client' and 'Fournisseur' checked, also circled in red.

Figure 46 : Exemple d'un client qui est fournisseur (avec code)

The screenshot shows the profile for 'SARL MCQ'. The client code 'C10007' is circled in red. The interface includes a top navigation bar with icons for 'Facturé' (17000,00) and 'Ventes' (1). The main area is divided into 'Adresse' and 'Site Web' on the left, and a list of fields on the right including 'Poste occupé', 'Tél.', 'Tél. portable', 'Fax', 'Courriel', 'Civilité', 'NIF', 'A.I', 'NIS', and 'Capital'. The 'Capital' field is set to '0,00'. Below the main area are tabs for 'Notes internes', 'Achats-Ventes', 'Comptabilité', and 'Avertissements'. At the bottom, the 'Vendeur' section has 'Client' checked and 'Fournisseur' unchecked, with the 'Client' checkbox circled in red.

Figure 47 : exemple d'un client simple (avec code)

11- Gérer le retour d'articles (sachant qu'il y a des articles non retournables) :

L'entreprise veut gérer le retour d'articles, sachant qu'il n'y a pas de retour dans le cas des articles non retournables, ou articles hors garantie.

Comme le montre la figure suivante (figure 49), un message d'erreur s'affiche pour un indiquer que les articles sont non retournable ou hors garantie

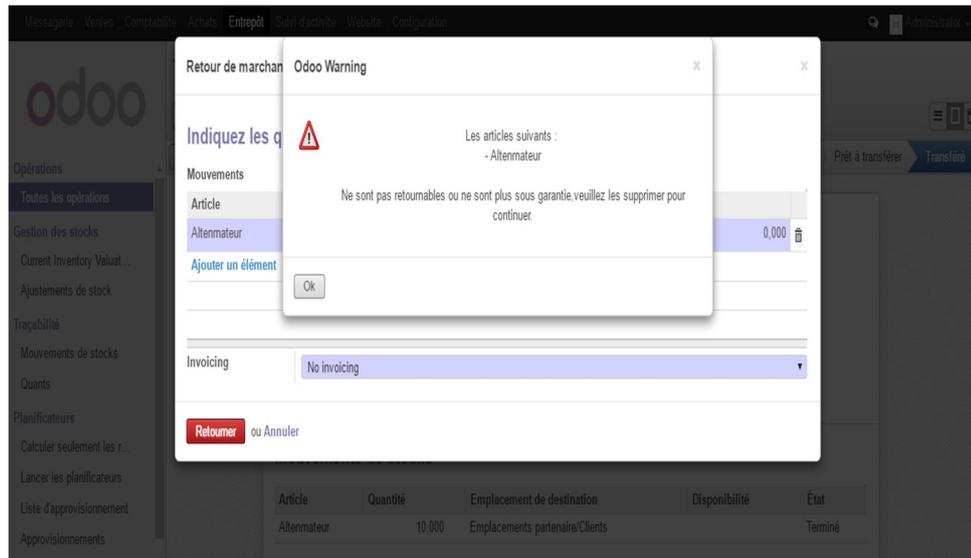


Figure 48 : Alerte d'article non retournable

12- Ajouter tous les champs non existant dans les fiches (client, fournisseur,...Etc) :

Prenons l'exemple de la fiche fournisseur standard d'Odoo, elle contient des informations générales, mais le client a beaucoup d'autres informations concernant le fournisseur

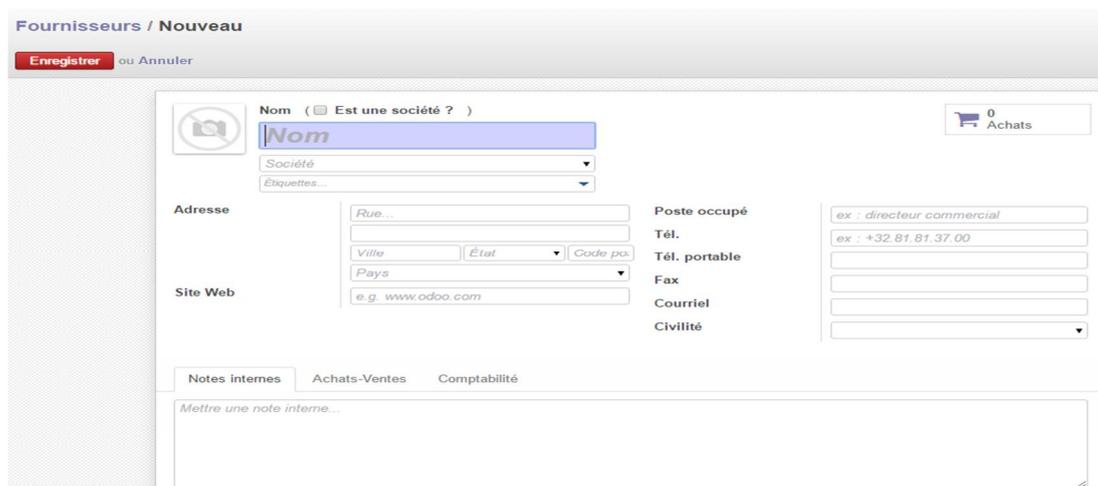


Figure 49 : Fiche fournisseur Standard

Pour cela nous avons ajouté toutes les informations manquantes, comme le montre la figure 51.

Fournisseurs / Nouveau
Enregistrer ou Annuler

Nom (Est une société ?)
Nom
 SARL par exemple

Société
 Étiquettes...

Adresse
 Rue...
 Commune Wilaya Code po
 Région Secteur
 Pays

Site Web
 e.g. www.odoo.com

Poste occupé
 Tél.
 Tél. portable
 Fax
 Courriel
 Civilité
 NIF
 Numéro d'article
 NIS
 Capital
 N° de registre commerce
 Modifié le

ex : directeur commercial
 ex : +32.81.81.37.00

max 15 chiffres
 max 11 chiffres
 max 15 chiffres
 0,00
 RC
 Date de RC

Notes internes Achats-Ventes Comptabilité Avertissements

Figure 50 : Fiche fournisseur après modifications

13- Création des rapports avec toutes les informations de l'entreprise cliente :

L'entreprise gère plusieurs rapports, comme les bons de livraisons, factures...

Nous avons donc créé ces rapports, voici l'exemple de la facture :

Facture Brouillon du 2016-05-13

Règlement/Réf:
 Espèce

Client: [C10092]
 Abu Bakr Belkaid, Messabih Mohamed
 B, rue Amir abdel Kader, La marine
 Saida 200 13000
 Algérie

R.C : 1236202565 N.I.S : 102365402145896
 N.I.F : 142360123564785 A.J : 10232565478

N°	Code	Désignation	UM	Qte	P.U HT	TVA	Montant HT
1		Moteur AUDI Origine France	Unité(s)	1,000	160000,00	17	160000,00 DZ
Total HT							160000,00 DZ
Remise (15.0%)							24000,00 DZ
TVA							0,00 DZ
Timbres							1920,00 DZ
Total TTC							137600,00 DZ

Arrêtez la présente Facture à la somme de :
 Un cent trente-sept mille, six cent d'ad

Figure 51 : Rapport de facture personnalisée selon le besoin du client

III.4 Site Web e-commerce

Pour pouvoir vendre les articles disponibles en ligne, et mettre le processus d'achat disponible via un site web, il existe un module site web e-commerce très puissant sous Odoo, mais il faut paramétrer notre application pour que les articles soient visible et peuvent être vendu en ligne.

Pour cette partie il faut suivre les étapes suivantes :

III.4.1 Installation du module e-commerce

Pour l'installation il faut se rendre dans le menu configuration, puis local module, puis e-Commerce (comme le montre la figure 53).

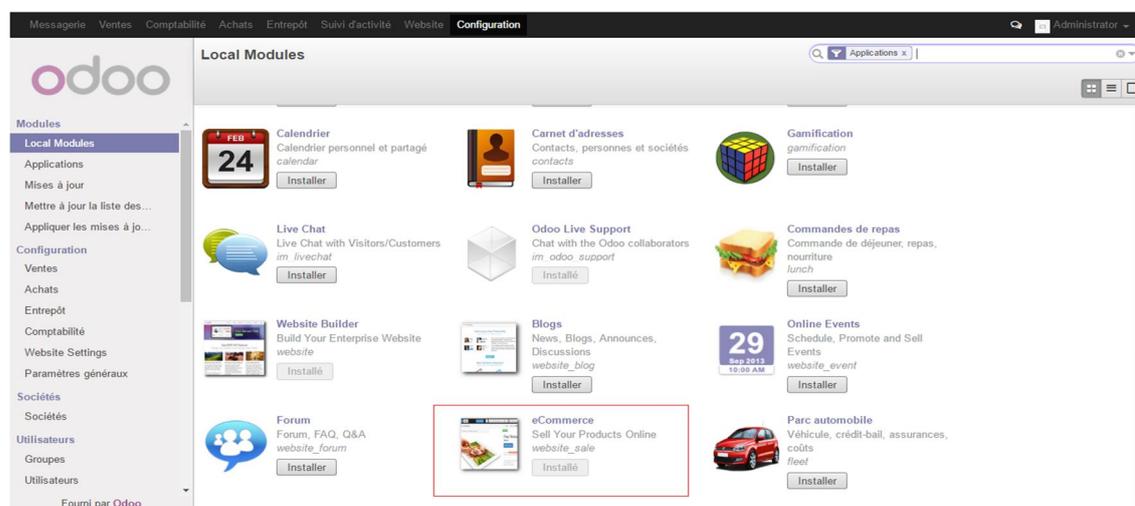


Figure 52 : Installation du module e-commerce

III.4.2 Gestion du catalogue Produits, organisation de la boutique

1- Les produits :

La première des choses à faire et la définition des informations de base : Libellé, image, code, Fournisseur, prix d'achat...etc

Ensuite les paramètres liés au site Web e-Commerce dans onglet "Ventes" où on doit définir :

- Catégorie d'affichage
- Produits alternatifs : produits de remplacement si l'article n'est pas disponible
- Accessoires : produits associés à ce produit
- Style d'affichage : en mode vignette ou grande image

- Séquence : permet de définir l'ordre d'affichage dans la catégorie
Le bouton « Not Published » ou « Published » permet de rendre l'article disponible sur le site ou non .

The screenshot shows a product management interface for 'Moteur Green Tech'. At the top, there is a product image and a title 'Moteur Green Tech'. Below the title, there are two checkboxes: 'Peut être vendu' and 'Peut être acheté', both of which are checked. To the right, there is a summary table with the following data:

0 Achats	0 Ventes
0.0 En stock	1 Mouvements
Règles de réassor	Routes

Below this, there are several tabs: 'Information', 'Approvisionnements', 'Inventaire', 'Ventes', 'Variantes', 'Comptabilité', and 'Avertissements'. The 'Ventes' tab is currently selected. Underneath, there are sections for 'Conditions de vente' (with fields for 'Garantie' set to 0.00 mois and 'Délai de livraison au client' set to 7.00 jours), 'Website' (with a dropdown for 'Public Category' set to 'Others', a 'Disponibilité sur le site' dropdown set to 'Published', and other fields for 'Alternative Products', 'Accessory Products', 'Styles', and 'Sequence' set to 6), and 'Description pour les devis' (with a text area for 'Note à afficher sur les devis...').

Figure 53 : Exemple d'article accessible via le site e-commerce

2- Les variantes :

Certains produits possèdent plusieurs variantes, en fonction de tailles, de couleurs ou d'options différentes...*etc*

Il faut alors créer les variantes, par exemple des variantes de l'origine du produit : chaque produit est disponible en plusieurs origines (Allemagne, Chine, ...*etc.*)

Pour ajouter une variante, il faut aller dans l'onglet « Variantes », puis cliquer sur "Ajouter un élément", et créer l'attribut s'il n'existe pas.

Puis affecter les variantes disponibles pour chaque attribut, ou les créer si elles n'existent pas

Comme le montre l'exemple dans la figure 55 (ajout des variantes en back office) et figure 56 (affichage des variantes dans le site web).

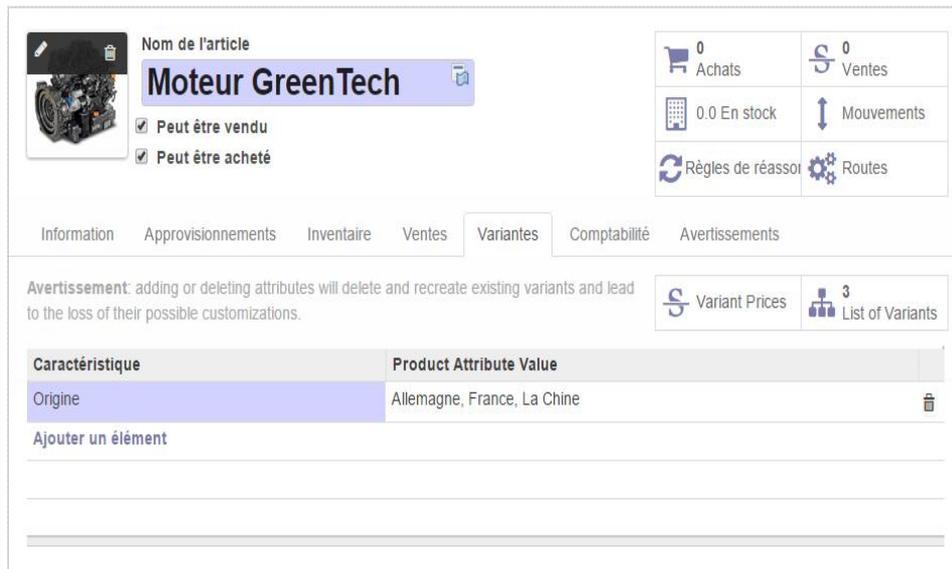


Figure 54 : Ajout de variante en back office

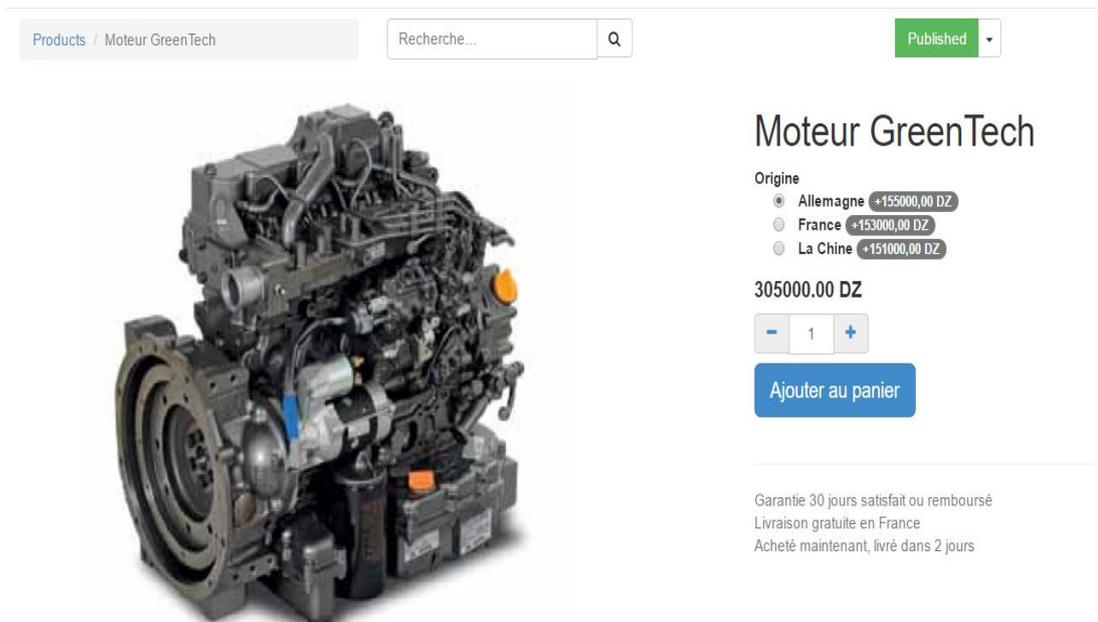


Figure 55 : Ajout de variante en front-office

3- L'organisation de la boutique

Odoo crée automatiquement une page nommée /Shop, cette page affiche l'ensemble des produits de la boutique, avec une pagination et possibilité de modification d'emplacement, comme le montre la figure suivante (figure 57).

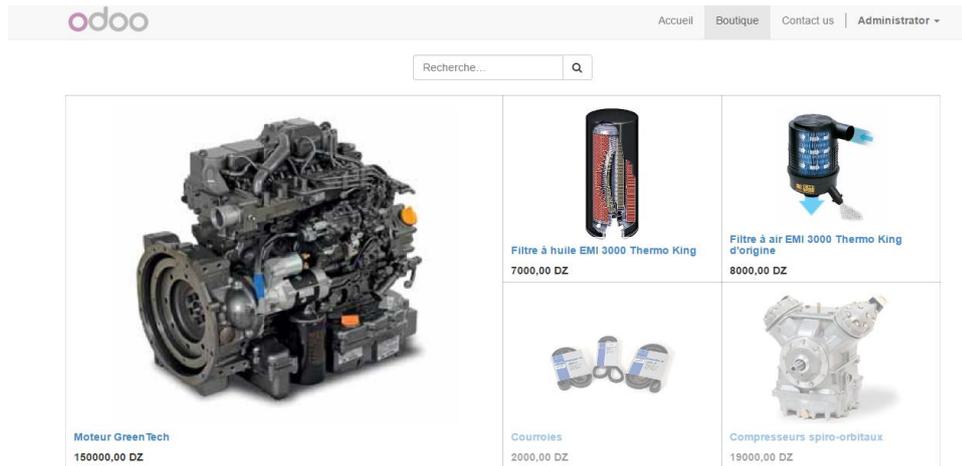


Figure 56 : Affichage des articles sur le site Web

4- La gestion des tarifs :

La première chose à faire est d'autoriser la gestion des tarifs de vente, ce qu'Odoon appelle les listes de prix de vente.

Pour cela, il faut utiliser le **menu Configuration / Ventes** et cocher l'option "**Utilisez les listes de prix** pour adapter votre prix aux clients".

Ensuite, il faut utiliser le menu "**Ventes / Listes de prix**" pour gérer les tarifs.

Le principe est de créer des modèles de tarification en fonction des catégories de produits, des clients, des quantités commandées.

Il est possible de définir un ordre de priorité entre les règles et une règle peut se baser sur une autre et la compléter (une sorte d'héritage de règles).[16]

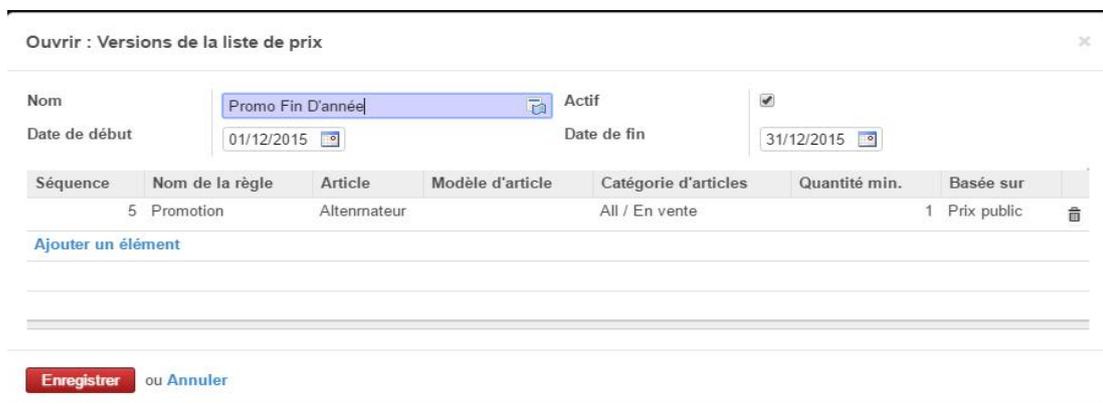


Figure 57 : Ajout d'une liste de prix temporaire

Conclusion :

Dans ce chapitre nous avons vu la majorité des règles de gestion développées pour répondre aux besoins du client, ce qui nous a permis de toucher les axes fondamentaux du développement dans les ERP en générale, et Odoo en particulier.

Nous avons cherché dans les modules existants, puis adapter l'existant s'il ne répond pas efficacement au besoin, et finalement nous avons développé d'autres modules non-existants.

Enfin, nous avons proposé le paramétrage d'un site dans lequel les visiteurs et les clients de l'entreprise pourront voir les produits disponibles avec leurs prix. *Etc.*

Conclusion générale

Ce stage de fin d'études a été une occasion pour nous pour une immersion dans le monde des ERP et de l'entreprise avec plus de responsabilités.

Dans notre projet de fin d'études, le travail a été réalisé au sein de la société SOGESI. Dans ce cadre, nous avons travaillé sur un projet réel et un besoin bien spécifique d'une entreprise cliente de la société SOGESI.

Pour ce faire, nous avons commencé par l'analyse des besoins, Par la suite, nous nous sommes basés sur une étude technique et fonctionnelle sur les différentes solutions existantes dans Odoo afin d'arriver à une solution qui respecte les exigences de la société.

La phase suivante consistait à paramétrer les modules d'Odoo avec les modules fonctionnels de la société. Or, la société exige des fonctionnalités non prise en charge par Odoo. Donc, pour ce développement spécifique, on a débuté par une analyse conceptuelle de ces modules et par la suite nous avons développé l'ensemble des vues et des objets. Finalement, nous avons implémenté ces modules au sein d'Odoo afin d'avoir un système adapté pour la société cliente pour assurer sa gestion globale.

D'un point de vue personnel, travailler dans le monde des ERP et plus précisément sur un ERP open source tel que Odoo nous a permis d'acquérir une large partie des technologies notamment le langage python, et de rejoindre une communauté mondiale de la recherche et le développement, afin de faciliter l'intégration d'une telle solution dans de nombreux secteurs d'activités.

Par ailleurs, nous avons acquis une expérience dans le domaine professionnel, tout en essayant de contribuer au développement de l'entreprise et proposer ce que nous avons appris durant notre formation universitaire.

Comme perspective, nous allons continuer sur la même démarche avec paramétrage, et la création de modules tels que le module de l'importation qui va assurer la gestion de leurs achats de l'étranger, puis l'installation et le test de l'application directement chez le client, et une formation pour le personnel technique de l'entreprise cliente. Nous souhaitons également contribuer à l'intégration d'Odoo dans d'autres secteurs d'activité tels que l'éducation et l'enseignement supérieur, le tourisme, le transport. *Etc*

Références

- [1] <http://www.sogesi-dz.com/> consulté le 10/03/2016
- [2] https://fr.wikipedia.org/wiki/Progiciel_de_gestion_int%C3%A9gr%C3%A9 consulté le 13/04/2016
- [3] *Karl Hellec*, **Open ERP : Découverte et mise en œuvre**, août 2013
- [4] *Geoffrey S. Gardiner and Fabien Pinckaers*, **OpenERP: A modern approach to integrated business management based on a free Open Source software système**, janvier, 2011.
- [5] <http://myopenerp.blogspot.com/2009/11/architecture-technique-et-modulaire.html> consulté le 17/04/2016
- [6] *Smile*, **Livre blanc ERP Open Source version 1.2**, 2008.
- [7] *Greg Moss*, **Working with Odoo**, 2015
- [8] <https://ics.utc.fr/c2m/DOCS/L2e/html/co/Workflow.html> consulté le 17/04/2016
- [9] *Daliel Reis*, **Odoo developpement essentials**, Avril 2015
- [10] https://fr.wikipedia.org/wiki/Enterprise_Architect consulté le 8/05/2016
- [11] *Peter Doomen*, **Fifty Enterprise Architect Tricks**, 2014
- [12] https://fr.wikipedia.org/wiki/Extensible_Markup_Language consulté le 8/05/2016
- [13] [https://fr.wikipedia.org/wiki/Python_\(langage\)](https://fr.wikipedia.org/wiki/Python_(langage)) consulté le 20/03/2016
- [14] *Tarek ZIADÉ*, **Programmation python** Edition Eyrolles, janvier 2006.
- [15] *Gérard Swinnen*, **Apprendre à programmer avec Python3** Edition Eyrolles, février 2012.
- [16] *Pascal Roques*, **UML 2 par la pratique**, Septembre 2009

Liste des figures :

Figure 1 : Différentes Versions de OpenERP	9
Figure 2 : Principales applications logicielles Odoo.....	10
Figure 4 : Architecture Client-Serveur d'open ERP	11
Figure 5 : Architecture modulaire d'open ERP.....	11
Figure 6 : Structure d'un module OpenERP Python.....	12
Figure 7 : Structure d'un module Odoo	14
Figure 8 : Interface Trello	16
Figure 9 : Structure d'entreprise Architect.....	16
Figure 10 : Diagramme de processus d'achat-Stock.....	19
Figure 11 : Diagramme de processus Vente.....	22
Figure 12 : Diagramme du processus d'achat	25
Figure 13 : Diagramme du processus de vente	26
Figure 14 : Schéma illustratif de l'achat-Vente	27
Figure 15 : Diagramme cas d'utilisation (vente en générale)	27
Figure 16 : Diagramme de séquence Vente	29
Figure 17 : Diagramme cas d'utilisation (Achat-Stock)	30
Figure 18 : Diagramme de séquence Achat	31
Figure 19 : Diagramme de classe « Approvisionnement »	32
Figure 20 : Diagramme de classe « Commande ».....	33
Figure 21 : Diagramme de classe « Facture ».....	33
Figure 22 : Diagramme de classe « Formes ».....	34
Figure 23 : Une partie de la fiche client.....	34
Figure 24 : Etat du Stock.....	35
Figure 25 : Commande validée	35
Figure 26 : Alerte de quantité réservée	36
Figure 27 : Bon de livraison avec Notification	36
Figure 28 : La classe "purchase_order_history"	37
Figure 29 : La table purchase_order_history.....	37
Figure 30 : Un arbre en XML	38
Figure 31 : Liste de prix d'achat par fournisseur	38
Figure 32 : Formulaire en XML.....	39
Figure 33: Formulaire d'une ligne d'achat.....	39
Figure 34 : Impression de la liste des prix d'achat par fournisseur.....	39
Figure 35 : Rapport liste prix d'achat / Fournisseur.....	40
Figure 36 : Analyse par emplacement.....	40
Figure 37 : Analyse par articles	41
Figure 38 : Liste de Wilaya d'Algérie	41
Figure 39 : La classe 'procurement_order_line'	42
Figure 40 : Liste d'approvisionnement	42
Figure 41 : Création d'un approvisionnement.....	43
Figure 42 : Remise, timbre et autre TVA dans la facture	43
Figure 43 : Bon de commande avec remise en pourcentage	44

Figure 44 : Informations de la société.....	44
Figure 45 : Ajout d'une succursale	45
Figure 46 : Ajout d'un Showroom	45
Figure 47 : Exemple d'un client qui est fournisseur (avec code).....	46
Figure 48 : exemple d'un client simple (avec code)	46
Figure 49 : Alerte d'article non retournable.....	47
Figure 50 : Fiche fournisseur Standard	47
Figure 51 : Fiche fournisseur après modifications	48
Figure 52 : Rapport de facture personnalisée selon le besoin du client	48
Figure 53 : Installation du module e-commerce.....	49
Figure 54 : Exemple d'article accessible via le site e-commerce.....	50
Figure 55 : Ajout de variante en back office	51
Figure 56 : Ajout de variante en front-office	51
Figure 57 : Affichage des articles sur le site Web.....	52
Figure 58 : Ajout d'une liste de prix temporaire.....	52

Liste des abréviations

Abréviation	Désignation
SOGESI	SOciété de Gestion et Solution Informatique
UML	Unified Modeling Language
MVC	Model View Controller
ORM	Object Relational Mapping
SGBD	Système de Gestion de Base Données
PDR	Pièce De Rechange
PGI	Progiciel de Gestion Intégré
ERP	Entreprise Ressource Planning
PME	Petites et Moyennes Entreprises
SAV	Service Après-Vente