Table des matières

Introduction générale 1

Chapitre I : Les ERP (Enterprise Ressources Planning)

I. Introduction 3

II. ERP 3

II.1. Périmètre de gestion couvert par un ERP 3

II.2. Bénéfices d'un ERP 4

II.3. Types d'un ERP 5

II.4. Classement des ERP open source 6

III. Open ERP Odoo 7

III.1. Historique d'Odoo 7

III.1. Evolution de l'Open ERP Odoo 8

III.2. Nouveautés d'Odoo 10 9

III.3. Niveaux architecturaux de L'ERP Odoo 10

III.3.1 Architecture modulaire 10

III.3.2. Architecture technique 10

III.3.3 : Modèle vue contrôleur 11

III.4. Langages et les outils techniques 12

IV. Conclusion 14

Chapitre II : Étude préliminaire & Conception

I. Introduction 15

II. Présentation de l'entreprise cliente (COMO Industries) 15

II.1. Spécifications de l'entreprise cliente 15

III. Présentation du projet 17

III.1. Planification du projet : 17

III.1.1. Phase 1 : Apprentissage 18

III.1.2. Phase 2 : Odoo et le domaine du e-commerce 18

III.1.3. Phase 3 : Conception 18

III.1.4. Phase 4 : Réalisation 18

IV. Analyse de la gestion dans Odoo 20

IV.1 : Ventes et opportunités 20

IV.2 : Achats et inventaire 25

V. Conception 27

V.1. UML 27

V.2. Cas d'utilisation 28

V.3.1. Recherche des articles 29
V.3.2. Gestion du panier 30
V.3.3. Effectuer une commande 31
IV.4. Diagrammes de classe 32
VI. Conclusion 32
Chapitre III : Réalisation
I. Introduction 33
II. Réalisation de transactions de paiement en ligne 33
. II.1. PayPal 33
II.1.1. Comment accepter les paiements PayPal des factures 33
II.1.2. Configuration 33
II.1.3. Création et configuration d'un compte PayPal marchand 34
II.1.4. Configuration 34
II.2. Stripe 39
II.2.1. Comment accepter les paiements Stripe des factures 39
II.2.2. Configuration 39
II.2.3. Création et configuration d'un compte Stripe marchand 39
III. Configurations pour les anneaux de contrôles 42
III.1 Ajout des anneaux de contrôles : 42
IV. Configurations pour les Piges ARI 44
IV.1. Ajout des variantes 44
IV.2. Affichage des variantes coté site web 44
IV.3. Le module product_variant_name 46
IV.3.1. Masquer les prix supplémentaires des caractéristiques des variantes 49
V. Affichage d'une alerte lors de l'ajout dans le panier 50
V.1 Le module como_ecommerce 50
VI. Ajout du commentaire dans le site web 52
VI.1. Module Website_checkout_comment 53
VII. Conclusion 54
Conclusion Générale 55
Bibliographie 56
Liste des figures 57
Liste des tableaux 59
Liste des acronymes 59

Introduction générale

2018 aura été une année riche en événements. Actuellement nous vivons une ère de changements technologiques que nous voyons déjà se concrétiser dans le monde en entendant parler du e-commerce partout et en particulier en Algérie, où on commence à parler sérieusement du commerce électronique qui est devenue un secteur d'activité attractif et dynamique dans le monde.

Au fil des années, la progression des entreprises reste en effervescence afin de répondre aux besoins des clients qui ne cessent d'augmenter à leur tour et deviennent plus exigeants en matière de coût, qualité et temps sans oublier de prendre en considération la concurrence avec d'autres entreprises. Afin de répondre à ces critères ils font appel aux systèmes d'informations qui doivent utiliser des processus de développements logiciels assez stricts.

Suite à ce constat, il était important de côtoyer le monde des ERP, et faisant partie de la communauté mondiale de développement des progiciels intégrés. C'est donc la motivation principale qui nous a incités à se lancer dans la découverte de ce monde. L'opportunité a été accordée par l'entreprise SOGESI, qui nous a proposé un sujet très intéressant avec un stage dans cette entreprise, reconnue pour ses solutions en matière d'ERP.

SOGESI est une Société de Services en Ingénierie Informatique (SSII) Algérienne, qui offre des solutions de gestion des applicatifs et des infrastructures ainsi que le conseil en technologie pour les PME/PMI publics et privés.

La mission de SOGESI au niveau d'intervention de l'ingénierie applicative s'appuie sur la démarche de conception, de réalisation et d´évolution des applications, pour assurer par la suite les phases de test et de maintenance. La grande expérience au long des nombreux projets, lui en permet de présenter les conseils et accompagner ses clients dans la conduite de leurs projets.

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 solutions pérennes proposées par SOGESI sont aptes à répondre aux attentes des clients et aux nouvelles exigences du marché. Parmi les solutions proposées :

Gestion Commerciale : consiste à la gestion des prospections, gestion des achats, CRM et gestion des ventes, gestion des stocks, réalisation des devis et factures, suivi des fournisseurs et des clients…

Gestion de Comptabilité Algérienne : solution stable qui optimise la gestion des transactions commerciales.

Gestion de Projet : a pour but de gérer les différentes tâches quotidiennes en garantissant le respect du temps et de coût. De plus, cette solution permet de détecter les situations critiques.

Gestion des Ressources Humaines et de Paie Algérienne : propose le suivi des employés et assure la conformité de leurs processus avec la réglementation. Elle prend en charge aussi, le calcul de la paie conforme à la réglementation algérienne.

Notre préoccupation est de contribuer dans le développement de l'ERP Odoo et faire les configurations nécessaires pour répondre aux besoins et aux spécifications de l'entreprise cliente. Nous illustrons notre approche à travers l'ajout de quelques modules et la migration de quelques autres modules afin de bien gérer le commerce électronique et répondre aux spécifications. Notre choix s'est porté sur le domaine du e-commerce qui commence à entrer dans l'Algérie, et qui fait partie de la vie quotidienne de la plupart des pays. Odoo traite le e-commerce, mais malheureusement il répond à des besoins standards qui ne satisfont pas tous nos besoins. Nous avons donc choisi l'entreprise française COMO Industries pour étudier et réaliser ce projet et contribuer dans le domaine du e-commerce.

Nous allons présenter dans ce mémoire, les étapes à suivre pour configurer les différents paiements et la migration des modules pour être à jour avec la version actuelle de l'ERP Odoo.

L'organisation de ce mémoire est comme suit :

- **Le premier chapitre :** Nous allons introduire la notion des ERP, ensuite évaluer les différents ERP existants pour justifier le choix de l'ERP Odoo et les définitions de différents outils et technologies utilisés pour la réalisation du travail demandé.
- **Le deuxième chapitre :** Consacré pour la présentation globale de notre projet en premier lieu, puis nous passerons à l'analyse des besoins et les différentes étapes de la conception.
- **Le troisième chapitre :** sera dédié à la mise en œuvre de notre projet.
- Nous terminerons ce mémoire par une conclusion sur l'apport du travail réalisé, et quelques perspectives.

Chapitre I : Les ERP (Enterprise Ressources Planning)

I. Introduction

Les ERP (ou Progiciels de Gestion Intégrés) constituent des solutions pratiques aux entreprises et offrent une amélioration remarquable de leurs performances, ainsi qu'une compétitivité solide et durable. Dans ce chapitre on va présenter les concepts de base des ERP, les différents éditeurs, on détaillera plus particulièrement l'ERP Odoo ainsi que les outils liés à son déploiement

II. ERP

Un ERP (Enterprise Resource Planning) ou également appelé PGI (Progiciel de Gestion Intégré) est un système d'information qui permet de gérer et suivre au quotidien, l'ensemble des informations et des services opérationnels d'une entreprise. Il répond aux caractéristiques suivantes :

- Il émane d'un éditeur unique.
- En cas d'impact d'un module, l'information est mise à jour en temps réel dans l'ensemble des autres modules associés.
- C'est un système qui garantit la piste d'audit : il est facile de retrouver et d'analyser l'origine de chaque information.
- Il peut couvrir l'ensemble du Système d'Information (SI) de l'entreprise (sauf si l'entreprise ne choisit dans un premier temps d'implémenter que certains modules de l'ERP).
- Il garantit l'unicité des informations qu'il contient, puisqu'il n'a qu'une seule base de données au sens logique. [1]

II.1. Périmètre de gestion couvert par un ERP

La vocation d'un ERP est d'homogénéiser le SI de l'entreprise avec un outil unique, capable de couvrir un large périmètre de gestion, c'est-à-dire :

- La gestion des achats.
- Les ventes et la gestion commerciale.
- La gestion comptable : comptabilité client, fournisseur, immobilisations, analytique,…
- Le contrôle de gestion : budgets,…
- La gestion des stocks : logistique, transport,…
- La gestion de production (planification, …)
- La gestion du personnel et de la paie (GRH). [1]

II.2. Bénéfices d'un ERP

Les bénéfices de la mise en place d'un ERP sont les suivants :

Bénéfices économiques

- Réduction des coûts de maintenance par l'automatisation des tâches.
- Réduction des inventaires.
- Identification et quantification des bénéfices.
- Réduction des coûts des matières premières.
- En comptabilité : réduction de la période de fermeture de 80 %.
- Réduction des coûts administratifs de 3.5 % à 1.8 % des ventes. [2]

Bénéfices organisationnels

- Réduction des délais de traitement ou d'acheminement des flux d'information dans l'entreprise.
- Amélioration des processus.
- Accroissement de l'agilité de l'entreprise : en faire plus avec moins (données qualifiées). [2]

Bénéfices humains : Touchent alors des aspects variés, comme l'activité et la performance individuelle du personnel concerné.

Bénéfices globaux :

- Stockage en un seul endroit des données utiles et qualifiées. Il existe ainsi par exemple une seule base client pour les fonctions commerciales et de facturation, dont l'intérêt réside dans la garantie d'absence d'incohérences et la suppression des saisies redondantes.
- Intégration des flux dans l'entreprise étendue, ce qui permet l'accès des ERP par des utilisateurs mobiles et des partenaires externes (via Internet et Extranet).
- Système d'informations en temps réel comme support aux décisions d'affaires. Disparition des problèmes de synchronisation des données qui permet une gestion des budgets avec contrôle en temps réel des engagements.
- Efforts de l'entreprise, focalisés sur la saisie des opportunités plutôt que de déterminer qui a l'information véridique.
- Garantie de la qualité d'exécution des processus supportés par l'ERP. [2]

Rapport-Gratuit.com

II.3. Types d'un ERP

- **Les ERP propriétaires :** Sont édités par des sociétés nécessitant l'achat d'une licence. Voici quelques exemples de cette catégorie les plus utilisés :

- **Les ERP OpenSource :** La plupart de ces solutions sont gratuites, en revanche il faut
  inclure dans le calcul du coût d'acquisition total, les frais de maintenance et d'assistance technique. Voici la liste des principaux ERP Open Source :

II.4. Classement des ERP open source

Voici un tableau représentant le classement des cinq meilleurs ERPs open sources :

Classement	ERP	Caractéristiques
1	**Odoo**	• Odoo est ergonomique, rapide et orienté vers les canaux de vente modernes. • Il permet de piloter également les actions de recrutement de talents. • C'est un ERP pour les TPE (Très Petite Entreprise) et PME (Petite et Moyenne Entreprise) voulant innover et se réinventer.
2	**Dolibarr**	• Dolibarr est le seul de sa catégorie à être conçu pour les petites structures : indépendants, TPE, startups et petites PME. • Il est conçu sur des technologies éprouvées par une communauté de développeurs très dynamique. • Ses modules ne font pas d'impasses fonctionnelles. • Il prend la deuxième position de ce classement pour sa gratuité totale
3	**Axelor**	• Axelor est ergonomique et offre de nombreuses fonctionnalités collaboratives. • Il offre des vues claires permettant de travailler aussi bien sur des tâches opérationnelles que du décisionnel.
4	**Openbravo**	• Openbravo est un classique orienté "retail". • Il convient particulièrement aux PME dans le secteur du commerce de biens grâce à sa gestion fine de tous les canaux de vente, ainsi que des produits mis au catalogue. • Le nombre de contributeurs au projet Openbravo est tout simplement gigantesque et garantit une longue vie à cet ERP Open Source.
5	**Compiere**	• Compiere est un ERP robuste adapté aux niveaux d'exigences les plus élevées. • Son ERP et son CRM fonctionnent en duo parfaitement, et les sous-ensembles métiers sont personnalisables à souhait. • Il est réputé pour son modèle de données reposant sur un dictionnaire de donnés très efficace accessible par l'éditeur et les Web Services. • Cet ERP Open Source s'adresse aux grandes PME qui souhaitent réduire leurs frais de fonctionnement.

Tableau 1: Classement des cinq premiers ERP open source. [3]

Odoo est le grand gagnant du classement avec un nombre d'utilisateurs de 2 millions, lui suit Dolibarr l'ERP pour les petites entreprises et indépendants en 2èm lieu par un nombre d'utilisateurs de plusieurs millions ensuite Axelor, Openbravo et Compiere.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

III. Open ERP Odoo

Odoo, anciennement connu sous le nom d'OpenERP, est un éditeur de logiciels open source fondé en 2004 qui propose une suite complète de modules de gestion d'entreprise entièrement intégrés. C'est le programme de gestion d'entreprise le plus évolutif et le plus installé au monde grâce à ses applications répondant à tous les besoins d'une entreprise, de la gestion et la relation client, à la création de sites web et d'e-commerce, en passant par la production, la gestion d'inventaire, la comptabilité, l'ERP etc., le tout parfaitement intégré. C'est la première fois qu'un éditeur de logiciels parvient à atteindre un tel niveau de fonctionnalités. Il a pour mission d'offrir aux entreprises de différentes tailles et secteurs, une solution professionnelle d'utilisation facile pour tous ses utilisateurs. [4]

III.1. Historique d'Odoo

Odoo, c'est l'histoire d'un belge, **Fabien Pinckaers** qui rêvait d'être le leader du marché du management entrepreneurial, alors il voulait changer le monde des ERP en créant un logiciel de gestion d'entreprise open source.

En 2005, il a développé **Tiny ERP** le progiciel libre de gestion intégré qui a pu à peine prendre sa place avec les géants comme SAP. Les versions lancées sous le nom de Tiny ERP sont 1.0, 2.0, 3.0 et 4.0.

3 ans plus tard, en 2008 **Tiny ERP** a grandi et s'est amélioré en 4 versions. Ce n'est plus **minuscule** tiny ERP qui est devenue plus grande et commence à proposer des offres aux clients multinationales qui ont à gérer des milliers d'employés, donc il a renommé sa société en **Open ERP**, afin d'être crédible devant ces clients en lançant les versions 5.0, 6.0, 6.1 et 7.0.

En 2014, en lançant la version 8 avec plus de 3000 modules, et vu que **Open ERP** est devenu plus qu'un ERP son nom a changé pour devenir **Odoo.** Après ils ont lancé la version 9 puis 10 (version actuelle) et la version 11 qui est encore en cours de développement.

III.1. Evolution de l'Open ERP Odoo

L'ERP Odoo a évolué très rapidement depuis sa création. Le tableau suivant présente quelques versions :

Nom du logiciel	Version	Date de création
Tiny ERP	1.0	Février 2005
	2.0	Mars 2005
	3.0	Septembre 2005
	4.0	Décembre 2006
Open ERP	5.0	
	6.0	Octobre 2009
	6.1	2012
	7.0	Décembre 2012
Odoo	8.0	Septembre 2014
	9.0	Novembre 2015
	10.0	Octobre 2016
	11.0	Octobre 2017

Tableau 2: L'évolution de l'open ERP Odoo depuis sa création

III.2. Nouveautés d'Odoo 10

Odoo 10 introduit une refonte de la gestion de la production avec l'apparition de modules maintenance et qualité. Un certain nombre de nouveautés ont été inspirées par la communauté de développeurs autour d'Odoo, ce qui prouve son importance essentielle pour faire évoluer la solution avec pragmatisme, qualité et rapidité. Parmi ces changements, on cite les améliorations les plus essentielles suivantes :

- **la partie fabrication :** On peut noter la refonte en profondeur de la partie fabrication qui a été réalisée. Il y a maintenant des fonctionnalités comme le plan directeur, qui permet d'optimiser la production en fonction du stock, du délai de réalisation et des commandes enregistrées.

-

- **Module lié à la PLM (Product Lifecycle Management) :** Gestion du cycle de vie des produits qui permet le visionnement des nomenclatures et des gammes.
- **Gestion de la qualité :** Qui commence à être traité dans la version standard et non plus seulement avec les modules communautaires.
- **Partie ressources humaines :** Plus particulièrement la partie note de frais qui est également améliorée.
- **Mode développeur avancé avec nouvelles fonctionnalités :** Le mode développeur s'active désormais via le tableau de bord de configuration (dashlet à droite), il ne s'active plus via « à propos ».
- **Import :** Détection automatique des dates et décimaux en fonction des paramètres régionaux
- **E-commerce :** De nouveaux fournisseurs de paiement tel que Stripe(monde) et PayU(Inde), confirmation de commande avec plusieurs méthodes en fonction du mode de paiement.
- **Gestion de contenu site web :** Éditeur de page réécrit entièrement pour une meilleure ergonomie, Le portail client: gère maintenant les tickets, leads, devis, bons de commande, bons de livraisons et factures et un nouveau gestionnaire de média qui supporte les fichiers binaires, vidéos, images et icones.

III.3. Niveaux architecturaux de L'ERP Odoo

Voici les trois niveaux architecturaux de l'Open ERP Odoo :

III.3.1 Architecture modulaire

Odoo est basé sur le Framework OpenObject qui permet de réunir en un seul projet, sous forme de modules optionnels mais intégrés, toutes les fonctions nécessaires à une entreprise. Il est caractérisé par les points suivants :

- La couche de mapping objet-relationnel (ORM) intégré dans Object Service (OSV) pour éviter le développeur de requêtes SQL.
- La couche de mapping objet-relationnel (ORM) intégré dans Object Service (OSV) pour éviter le développeur de requêtes SQL.
- Model-View-Controller (MVC) qui supporte la notion d'héritage entre vues.
- Génération des rapports (PDF, HTML, ODT).
- Support de traductions dans des fichiers .po

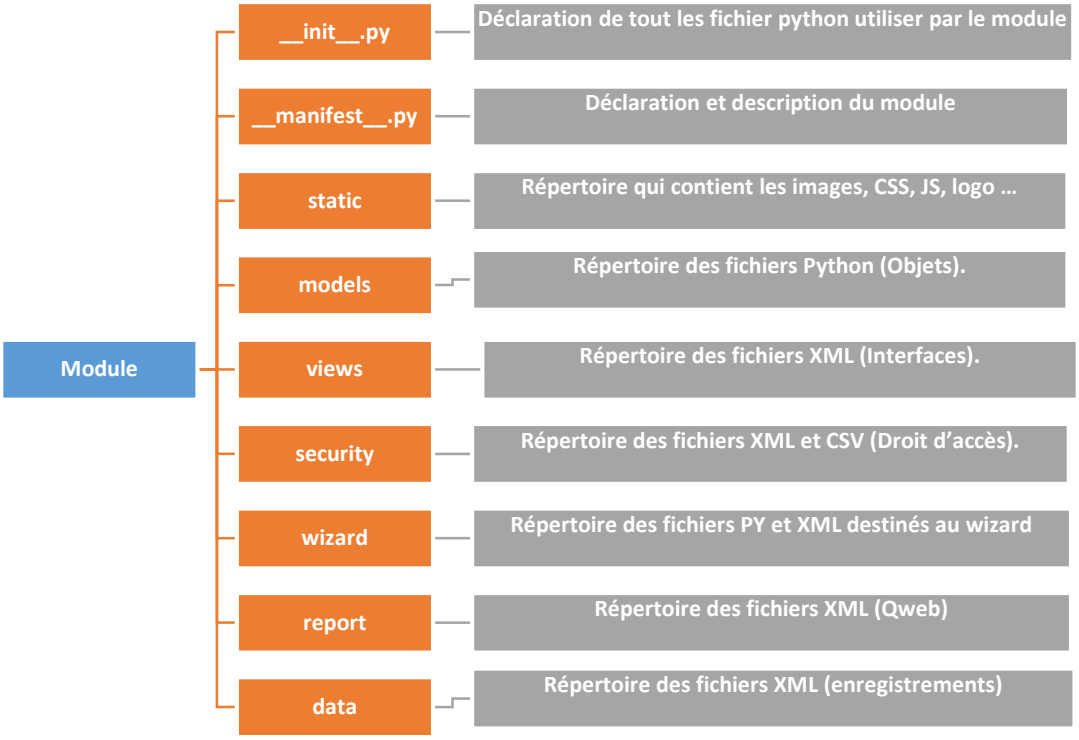


Figure 1: L'architecture modulaire de Odoo

III.3.2. Architecture technique

L'architecture Odoo se compose des 3 composants suivants :

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

ORM. Une autre couche permet les communications entre le serveur et un navigateur Web, la couche Web.

- **Client Web :** Qui fournit l'interface utilisateur, il s'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'actions d'utilisateur, il envoie des requêtes pour modifier les données sur le serveur. [5]

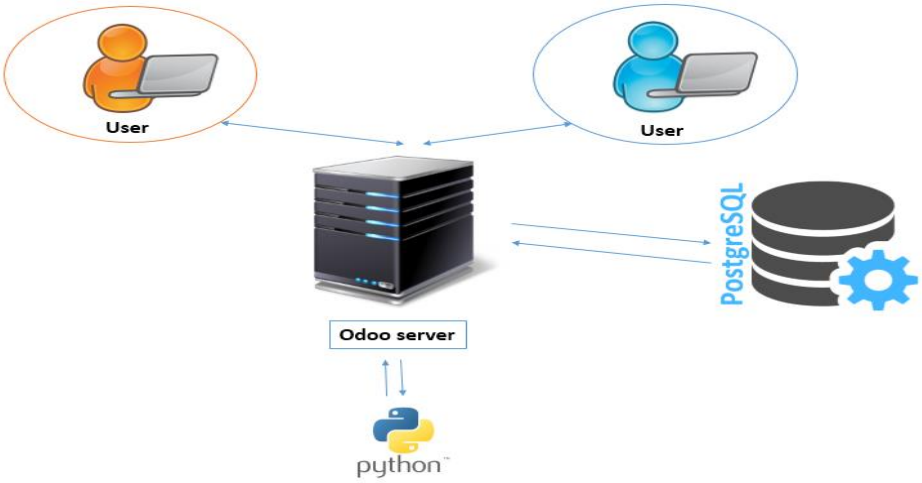


Figure 2: Architecture technique de Odoo

III.3.3 : Modèle vue contrôleur

L'ERP Odoo adopte une architecture MVC. Cette derniére est un modèle de conception qui décrit une architecture d'application informatique en la décomposant en 3 parties : modèle, vue et contrôleur.

- **Modèle** : Chaque objet déclaré dans Odoo correspond à un modèle, il est mappé à une table dans PostgreSQL.
- **Vue** : Est l'ensemble des fichiers XML dans Odoo.
- **Contrôleur** : Sont des classes Python qui gèrent indiquent la partie contrôleur.

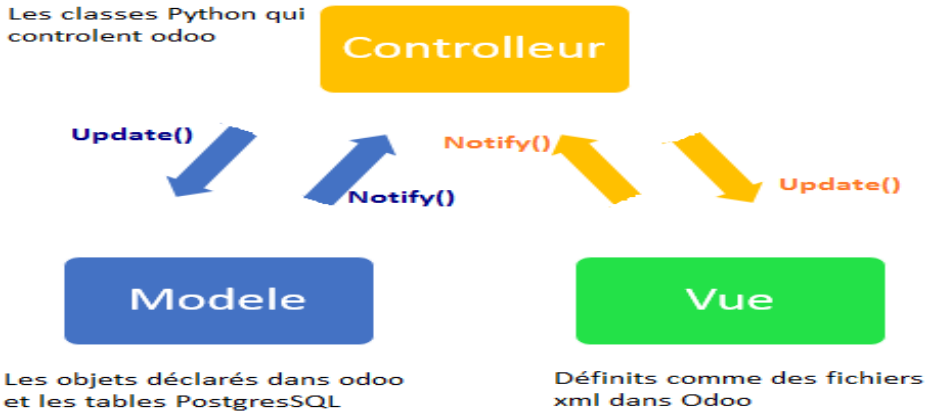


Figure 3: Modèle – Vue – Contrôleur

III.4. Langages et les outils techniques

Odoo est basé sur des technologies modernes en utilisant des langages dynamiques tels que Python et Bootstrap et le gestionnaire de bases de données relationnelles PostgresSQL, dont voici les langages et les outils utilisés :

III.4.1 : Python

Python est un langage de programmation de haut niveau conçu pour être facile à lire et simple à mettre en œuvre. C'est un logiciel libre , ce qui signifie qu'il est gratuit, même pour des applications commerciales. Python peut s'exécuter sur des systèmes Mac, Windows et Unix et a également été porté sur des machines virtuelles Java et .NET. Python est considéré comme un langage de script, comme Ruby ou Perl et est souvent utilisé pour créer des applications Web et du contenu Web dynamique.[6]

III.4.2 : XML

XML est un langage de balisage générique qui permet de structurer des données afin qu'elles soient lisibles aussi bien par les humains que par des programmes de toute sorte. Il est souvent utilisé pour faire des échanges de données entre un programme et un serveur ou entre plusieurs programmes. HTML (HyperText Markup Language). Comme ces derniers, il s'agit d'un langage orienté texte et formé de balises qui permettent d'organiser les données de manière structurée. [7]

III.4.2 : Bootsrap

Bootstrap est un framework frontal gratuit pour un développement web plus rapide et plus facile. IL inclut des modèles de conception HTML et CSS pour la typographie, les formulaires, les boutons, les tableaux, la navigation, les modaux, les carrousels d'images et bien d'autres, ainsi que des plugins JavaScript optionnels. IL donne également la pos sibilité de créer facilement des conceptions réactives. [8]

III.4.3 : QWeb

Qweb est le principal texturant moteur utilisé par Odoo pour rédiger les rapports . C'est un moteur de template XML utilisé principalement pour générer des fragments et des pages HTML. Les directives de modèle sont spécifiées en tant qu'attributs XML préfixés avec t-. [9]

III.4.4 : PostgresSQL

PostgreSQL est un système de gestion de bases de données relationnelles objet (ORDBMS) basé sur POSTGRES, Version 4.2, développé à l'université de Californie au département des sciences informatiques de Berkeley. POSTGRES a lancé de nombreux concepts rendus ensuite disponibles dans plusieurs systèmes de bases de données commerciales. PostgreSQL est un descendant open-source du code original de Berkeley. Il supporte SQL92 et SQL99 tout en offrant de nombreuses fonctionnalités modernes : requêtes complexes, clés étrangères, triggers, vues, intégrité des transactions, contrôle des accès concurrents (MVCC ou multiversion concurrency control). [10]

III.4.5 : JavaScript

JavaScript désigne un langage de développement informatique, et plus précisément un **langage de script orienté objet**. On le retrouve principalement dans les pages Internet. Il permet, entre autres, d'introduire sur une page web ou HTML des petites animations ou des effets. Il existe de nombreux frameworks JavaScript orientés vers les interfaces web (ou "orientés client"). Les trois plus connus sont **Jquery**, AngularJSet **React** (qui, lui, est né chez Facebook). [11]

IV. Conclusion

Odoo se situe en tête de classement des ERP actuellement, il se considère comme le leader grâce aux solutions qu'il présente satisfaisant la plupart des entreprises en terme fonctionnel et organisationnel. Bien que cet ERP englobe la majorité des domaines et en particulier le e-commerce, chaque entreprise a besoin de les adapter selon ces besoins qui ne peuvent pas être satisfaits seulement en ajoutant le module standard du e-commerce. De ce fait, notre travail consiste à effectuer les configurations nécessaires pour la réalisation de transactions de paiement en ligne et la migration des modules répondant aux besoins exigés.

Rapport-Gratuit.com

Chapitre II :
Étude préliminaire
&
Conception

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

I. Introduction

Odoo intègre un module e-commerce qui facilite la mise en place d'une boutique en ligne sous forme de plateforme, mais ça reste limité pour répondre à tous les besoins spécifiés par les clients. Donc c'est un travail qui demande une étude préalable du projet, et définir sa structure afin de préparer sa mise en œuvre.

Dans ce chapitre, nous allons présenter l'entreprise cliente et ces spécifications, ensuite le contexte dans lequel le projet s'est déroulé, puis nous allons ajouter une présentation de l'analyse du système, et établir enfin une étude conceptuelle à l'aide des diagrammes UML.

II. Présentation de l'entreprise cliente (COMO Industries)

COMO Industries, est une société qui se situe à St Avold en France, elle fournit à l'industrie française et européenne des produits de mécanique de précision de haute qualité fabriqués à ces ateliers de Courbevoie. Elle s'est spécialisée au fil du temps dans les produits de niche. Elle est le fabriquant ou revendeur exclusif sur la France.

COMO, c'est également la réalisation en sous-traitance de pièces de mécanique générale ou de précision élaborées sur-mesure pour ces clients ou encore des pièces décoratives pour des architectes, décorateurs d'intérieur responsables de la rénovation de grands magasins, hôtels, bureaux, ou habitants de luxe. [12]

II.1. Spécifications de l'entreprise cliente

La société commercialise des calibres à broche de diamètre (mm) xx, y, z avec x : en millimètres, y : dixième (1/10) de millimètres et z : centième (1/100) de millimètres.

1. Si on regarde http://www.comomecanique.com/shop/product/piges-ari-manche-alu-2693 qui est l'ancien site, on pourra choisir la taille de la jauge au format x, y, z avec les variantes, et si on ajoute un produit dans le panier, on obtient :

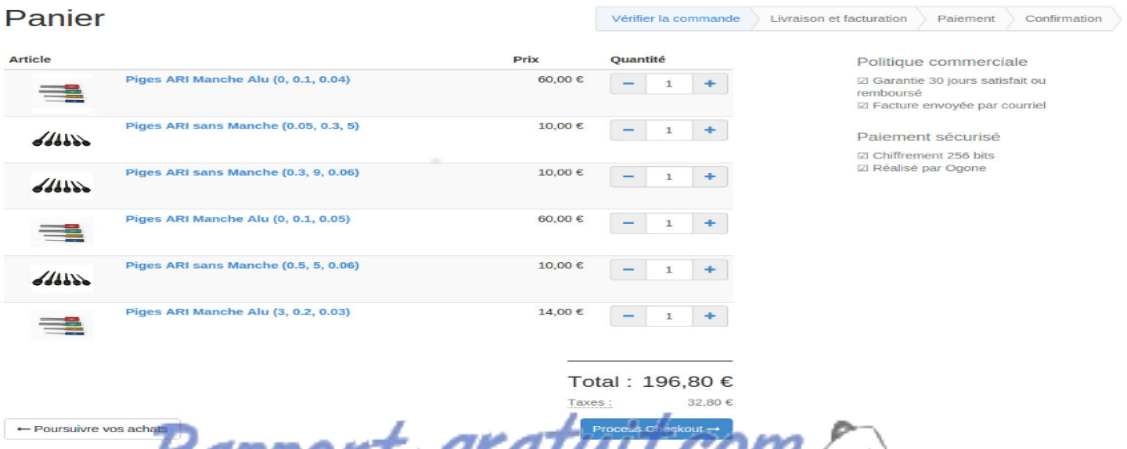


Figure 4: L'ajout des produits dans le panier dans la version8

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Donc on a besoin de vendre les piges selon une taille spécifiée, comme ceci :Pige ARI Manche Alu 0,14 mm, Pige ARI sans Manche 5.35 mm, Pige ARI sans manche 9.36 mm, etc.

2. Le format demandé pour ce produit spécifique doit être utilisé partout : sur le panier, sur les devis, les bons de commande, les factures, les rapports ...

   Voici la liste des variantes pour les 2 types de piges vendus :

   - Piges type P avec manche : son diamètre (mm) varie de 0 à 20.
   - Piges type S sans manche : son diamètre (mm) varie de 0 à 6.

   Et pour les 2 types, on a les dixièmes et centièmes suivantes :

   - Dixième (mm) : 1/10ème millimètre varie de 0.0 à 0.9.
   - Centième (mm) : 1/100ème millimètre varie de 0.00 à 0.09.

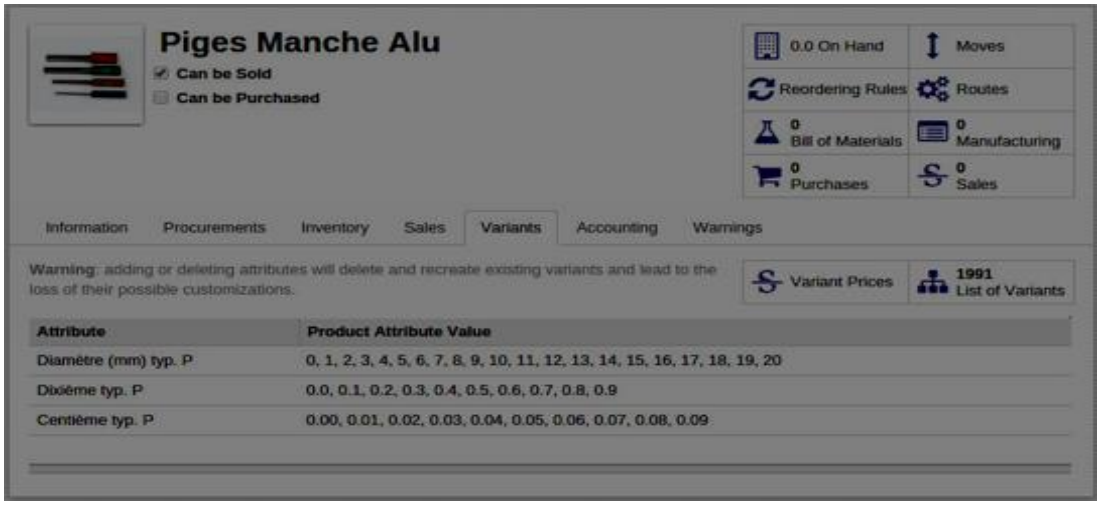


Figure 5: Les variantes des piges

3. Le programme doit également générer automatiquement les références internes pour chaque variante, les références internes sont au format S ou P suivit d'un zéro(0), puis 2 chiffres pour le diamètre en mm, 1 chiffre pour 1/10ème de millimètre et 1 chiffre pour 1/100ème de millimètre. Comme suit :

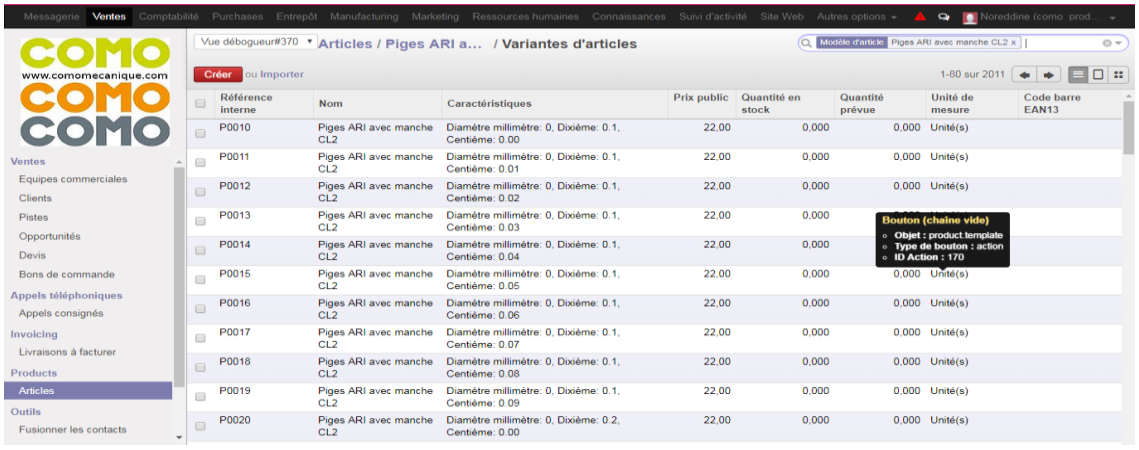


Figure 6: La génération automatique des références internes dans la version 8

La référence interne indique la mesure de la pige, exemple : P0134 signifie une pige d'un diamètre (1,34 mm).

4. Le client ne vend pas uniquement des piges, il faut donc ajouter un champ qui s'appelle **Préfix** sur la page de modèle du produit et, si le préfix est défini sur S ou P, les références des piges doivent être générées automatiquement.

5. Si le champ préfix est vide, le programme ne génère rien, et il ne remplace pas la chaîne par défaut qui est générée pour chaque variante.

6. Entrer les tarifs des anneaux de contrôles qui se vendent par jeux de 15 anneaux avec un prix qui varie selon un intervalle de nombre d'anneaux vendu

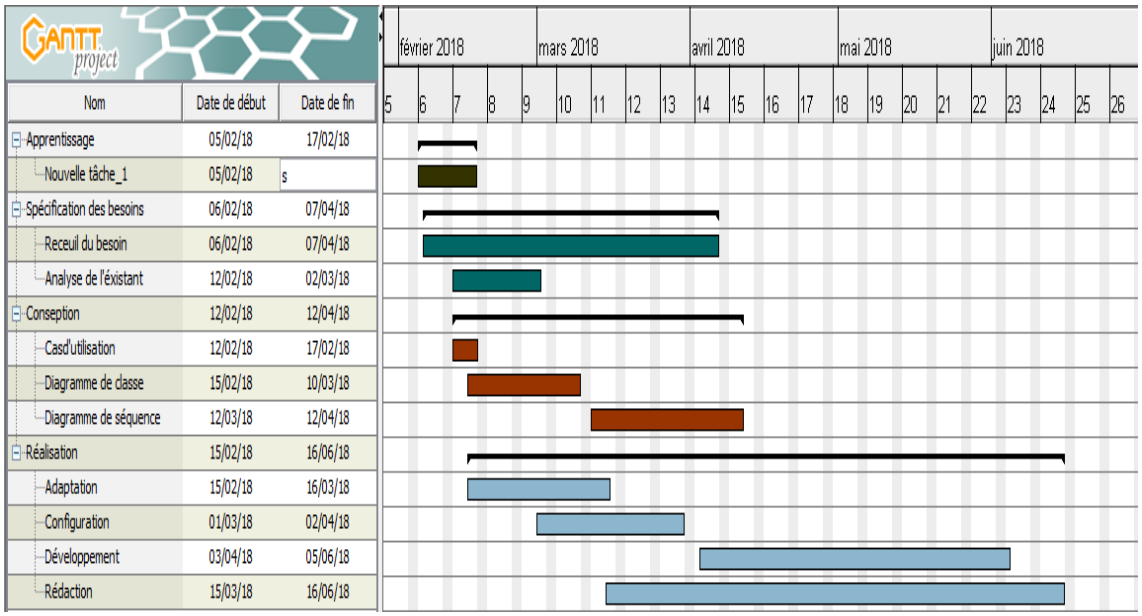
7. Réalisation des transactions de paiement en ligne : PayPal et Stripe.

III. Présentation du projet

III.1. Planification du projet :

Les projets informatiques représentent des systèmes très complexes, et pour un bon déroulement de projet il a fallu choisir un outil de gestion de projet, donc on a choisi « Gantt Project » qui est un logiciel open source pour bien structurer les taches et les fonctionnalités à réaliser.

Afin de planifier le travail, le projet a été découpé en plusieurs phases se résumant dans le diagramme de Gantt suivant :



Nom	Date de début	Date de fin
Apprentissage	05/02/18	17/02/18
Nouvelle tâche_1	05/02/18	s
Spécification des besoins	06/02/18	07/04/18
Receuil du besoin	06/02/18	07/04/18
Analyse de l'éxistant	12/02/18	02/03/18
Conseption	12/02/18	12/04/18
Casd'utilisation	12/02/18	17/02/18
Diagramme de classe	15/02/18	10/03/18
Diagramme de séquence	12/03/18	12/04/18
Réalisation	15/02/18	16/06/18
Adaptation	15/02/18	16/03/18
Configuration	01/03/18	02/04/18
Développement	03/04/18	05/06/18
Rédaction	15/03/18	16/06/18

Figure 7: Diagramme de Gantt

III.1.1. Phase 1 : Apprentissage

Avant de commencer la mise en œuvre du projet, il a fallu améliorer nos compétences dans le domaine de l'open ERP Odoo. On a déjà vu un TP dans le même contexte, mais la difficulté résidait dans la pauvreté de la documentation fournie, et en particulier pour basculer d'une version à une autre, cependant l'aide a été trouvé en tirant des connaissances de l'équipe de la société SOGESI.

III.1.2. Phase 2 : Odoo et le domaine du e-commerce

Cette phase consistait à intégrer le e-commerce dans l'ERP Odoo, ou on a commencé par le recueil des modules intégrés par l'entreprise cliente pour sa gestion, en ajoutant le e-commerce.

Après l'installation et l'analyse fonctionnelle de ces modules, on a bien remarqué que le système disponible est loin de répondre à tous les besoins du client, d'où notre contribution parvienne pour l'amélioration du coté commercialisation de l'open ERP Odoo par la configuration et la migration d'un ensemble de modules qui répondent aux besoins spécifiques de l'entreprise COMO Industries.

III.1.3. Phase 3 : Conception

Après avoir recensé les besoins de l'entreprise COMO Industries et recevoir ces spécifications, nos réflexions se sont concentrées sur la recherche de solutions idéales qui permettent de satisfaire cette dernière.

Tous d'abord, on a commencé à étudier les modules intégrés dans Odoo pour le commerce électronique et les transactions de paiement en ligne, puis on a entamé la phase de conception de notre projet en employant les modèles UML à l'aide de l'outil de modélisation de conception « Entreprise Architect ».

III.1.4. Phase 4 : Réalisation

L'open ERP Odoo offre beaucoup d'avantages, non seulement pour les utilisateurs, mais ainsi pour les développeurs, en leur donnant la main pour modifier les modules de base pour satisfaire leurs besoins ou développer leurs propres modules, comme ils peuvent migrer les modules pour passer d'une version à une autre.

- <u>Configuration</u> : dans la phase réalisation, la première mission était de configurer les comptes et les modules de transaction de paiement en ligne.

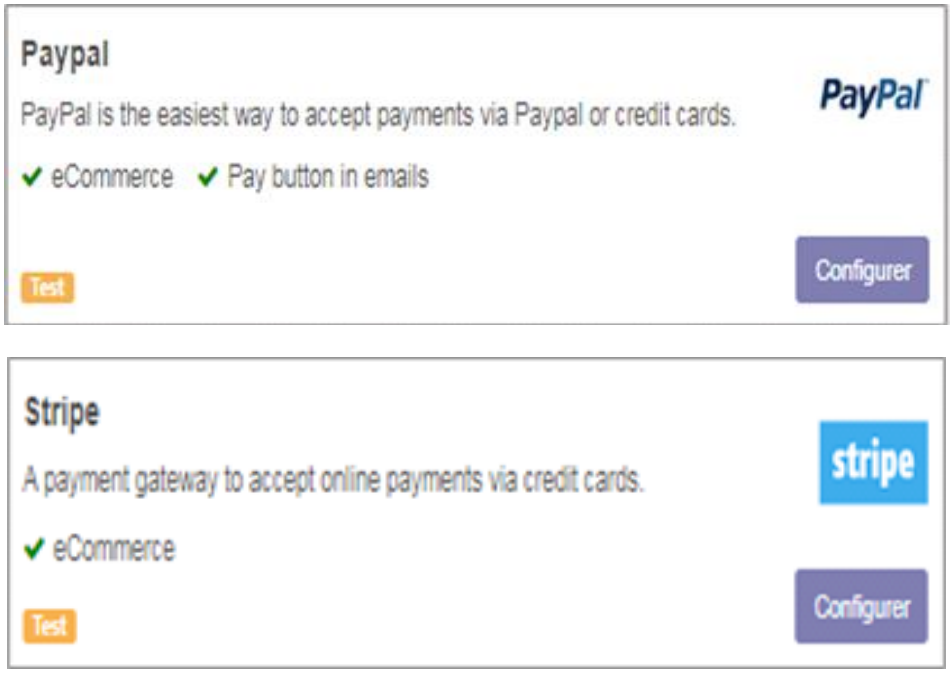


Figure 8: les intermédiaires de paiement à configurer

- Migration : L'ERP Odoo a évolué en plusieurs versions depuis 2005, ainsi pour passer d'une version à une autre, il faut migrer tous les modules développés pour satisfaire les besoins spécifiques du client, par conséquence, l'entreprise COMO Industries a demandé de mettre à jour sa version, et de passer de la version 8 vers la version 10. La réalisation de ce travail a demandé une migration de ces modules afin de satisfaire ces besoins.

```
<template id="optional_products_inherit" inherit_id="website_sale_options.modal" name="Optional products inherit
">
    <xpath expr="//table/tbody/tr[@class='js_product']/td[last()]" position="replace">
        <td>
            <input type="hidden" class="js_optional_same_quantity" t-attf-name="optional-add-#{option_inc}"
            t-att-value="total_bom_products"/>
            <a href="#" class="js_add"><strong>Add to Cart</strong></a>
            <span class="js_remove hidden">
                <span class="js_item"><t t-esc="total_bom_products"/> Item</span><span class="js_items hidden">
                5 Items</span><br/>
                <a href="#" class="js_remove"><small>Remove from cart</small></a>
            </span>
        </td>
    </xpath>
</template>
```

Figure 9: Modification du site web dans la version8

```
<template id="optional_products_inherit" inherit_id="website_sale_options.modal" name="Optional products inherit
">
    <xpath expr="//table/tbody/tr[@class='js_product']/td/input[@t-attf-name='optional-add-#{option_inc}']"
    position="replace">
        <input type="hidden" class="js_optional_same_quantity" t-attf-name="optional-add-#{option_inc}"
        t-att-value="total_bom_products"/>
    </xpath>

    <xpath expr="//table/tbody/tr[@class='js_product']/td/span[@class='js_remove hidden']" position="replace">
            <span class="js_remove hidden">
                <span class="js_item"><t t-esc="total_bom_products"/> Item</span><span class="js_items hidden">
                5 Items</span><br/>
                <a href="#" class="js_remove"><small>Remove from cart</small></a>
            </span>

    </xpath>
</template>
```

Figure 10: Migration de la Même modification dans le site web à la version10

IV. Analyse de la gestion dans Odoo

L'ERP Odoo est basé sur une interface adaptative et flexible que n'importe quelle entreprise commerciale peut utiliser. Il est possible de le configurer grave à sa flexibilité pour correspondre à nos besoins, et grâce à ses modules, l'enregistrement des opérations et toutes les activités financières est très facile en seulement quelques clics.

IV.1 : Ventes et opportunités

Afin de de montrer les fonctionnalités de ces modules, nous allons suivre le scénario suivant, commençant par les opportunités :

Pour créer une opportunité il faut passer par créer un contact :

Figure 11: Création d'un contact dans Odoo

Une fois le contact est créé, on remarque qu'aucune opportunité, aucun rendez-vous prévu, ni ventes ou factures. Alors, pour créer notre opportunité qui se trouve dans le CRM, on clique sur le bouton opportunités :

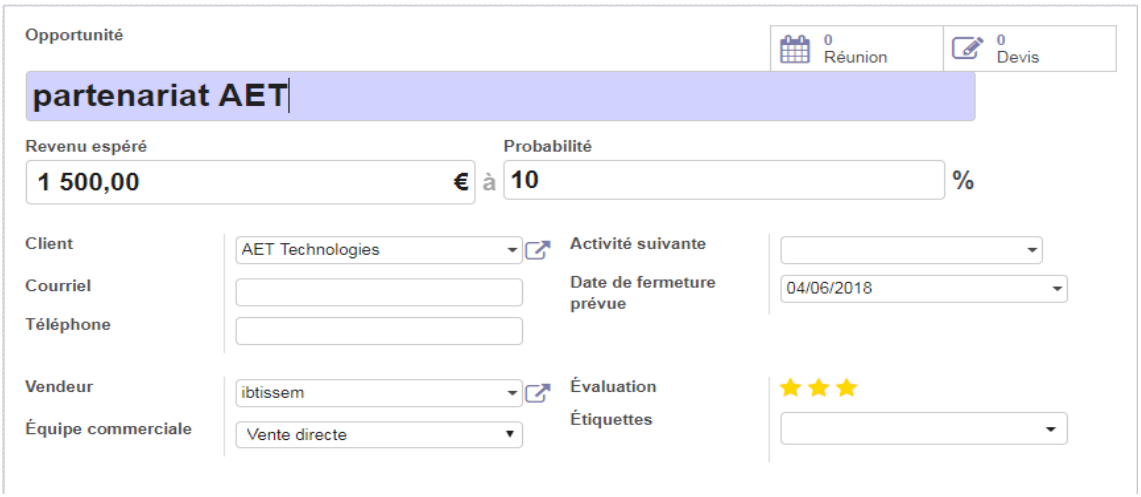


Figure 12: Création d'une opportunité

Nous voyons bien en haut qu'on peut ajouter des réunions ou des devis pour cette opportunité. Une fois nous avons nos opportunités, nous pourrons passer les opportunités d'une colonne à une autres selon le stade où celle-ci est arrivée (qualifié, proposition, gagné… etc.).

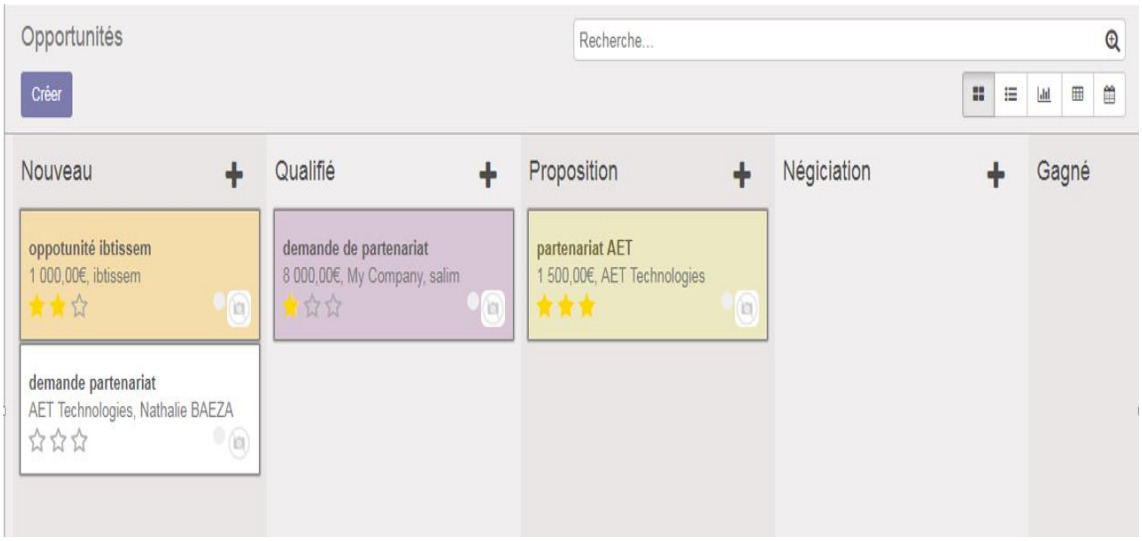


Figure 13: Les opportunités selon leurs étapes

Après la création des opportunités, on peut procéder à la création de devis à la demande du client, en cliquant directement sur «Nouveau devis » qui est crée automatiquement. Par la suite, on peut lui associé une date fin de validité, une condition de règlement, un mode de livraison et le nombre d'articles souhaité par le client, comme suit :

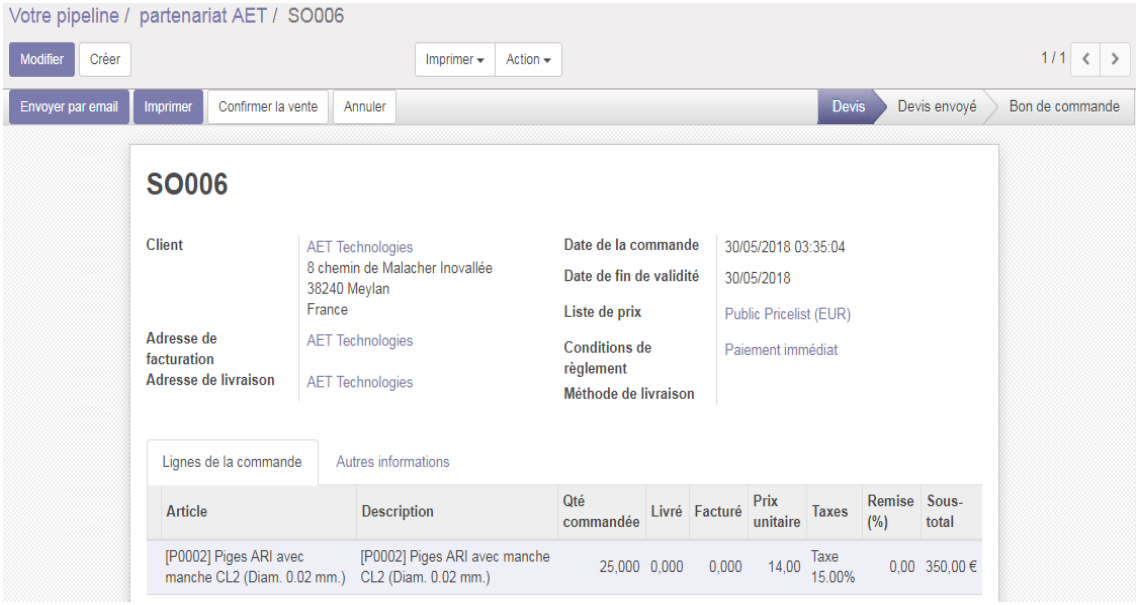


Figure 14: Devis de commande créé

Une fois le devis est créé, il peut être imprimé et envoyé au client. Si on va l'envoyer par mail, en cliquant sur « Envoyer par mail », on voit le mail avec un petit Template déjà fait, qui peut être configuré et personnalisé. Un format PDF du devis peut être généré aussi.

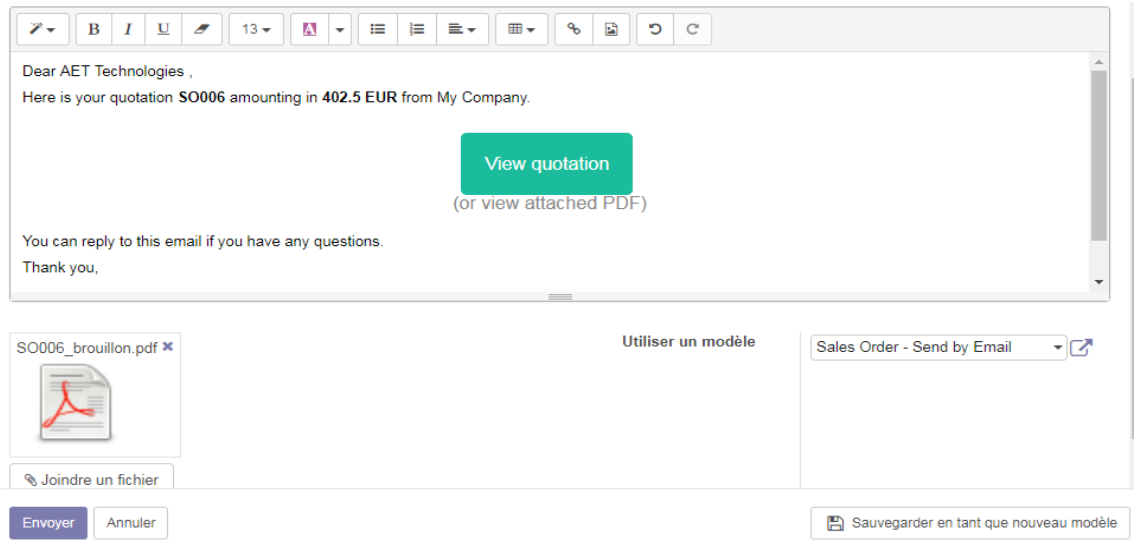


Figure 15: Le devis à envoyer

Si la vente est confirmée, le devis passe de l'état devis à un bon de commande :

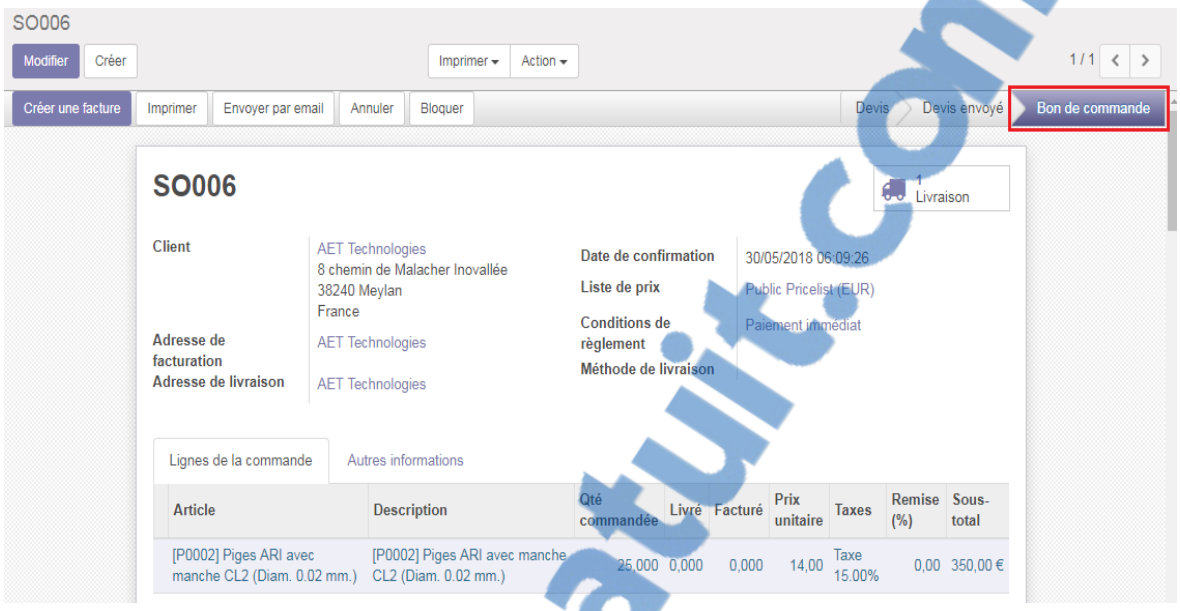


Figure 16: Création de bon de commande

Une fois le bon de commande généré, on peut créer une facture en cliquant sur « Créer facture », en sélectionnant les éléments à facturer

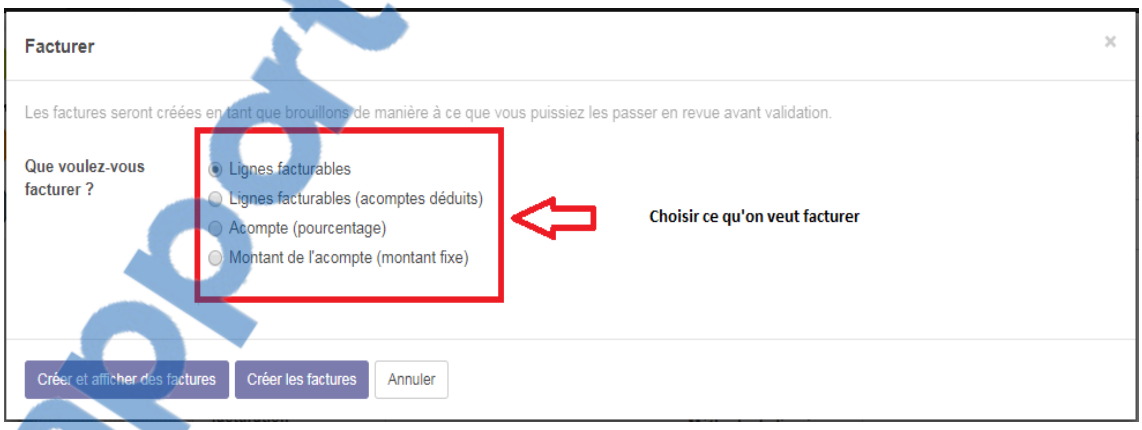


Figure 17: Création de la facture

Après l'avoir créé, la facture est en stade brouillon, alors elle peut être modifiée librement. Une fois, cliquer sur valider, la facture va passer en statut ouvert, qui veut dire qu'on peut plus modifier les informations de cette dernière. La facture aussi peut être imprimée et envoyée directement, ou envoyée par mail en suivant les mêmes étapes suivies pour l'envoi du devis.

Rapport-Gratuit.com

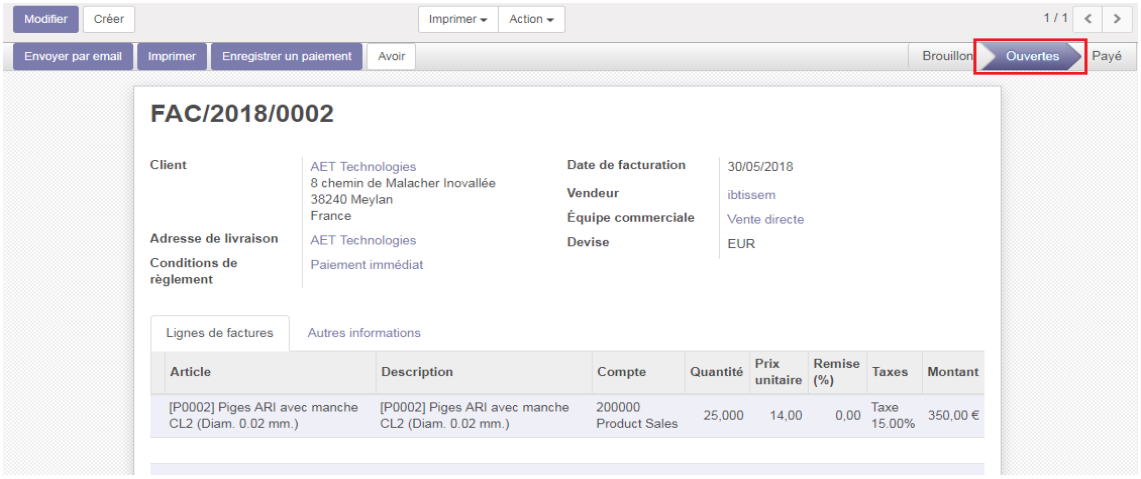


Figure 18: Facture ouverte pour le paiement

Si notre client veut s'enregistrer un paiement, alors il faut cliquer sur « Enregistrer un paiement », puis spécifier la somme qu'il veut payer en choisissant le mode de paiement adéquat :

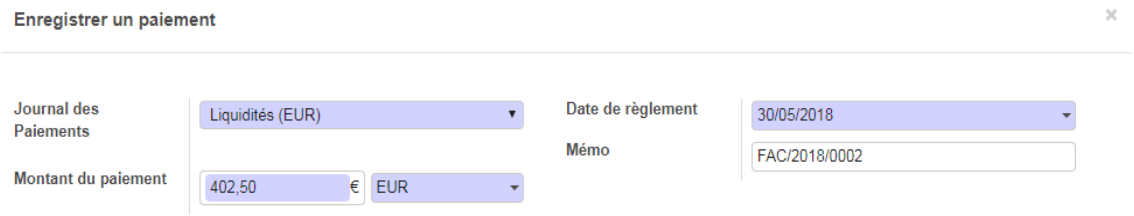


Figure 19: L'enregistrement d'un paiement

Une fois notre facture est payée, la vente est terminé, on peut passer donc notre opportunité au stade gagné en disant que cette opportunité est gagnée.

Figure 20: passer notre opportunité au stade gagné

IV.2 : Achats et inventaire

Nous allons suivre un scénario pour pouvoir effectuer nos achats :

Afin d'effectuer des achats, nous aurons besoin des produits. Dans la fiche d'un produit, on peut indiquer des informations générales : s'il peut être vendu ou acheté, son type (stockable, consommable ou s'il s'agit d'un service), sa catégorie, son prix de vente et son cout, et on peut aussi indiquer d'autre informations supplémentaires, comme son fournisseur, des variantes et des informations de facturation.

Figure 21: La fiche d'un article

On enchaine avec le menu achats : pour faire une commande auprès d'un fournisseur, On opte pour l'onglet « Demande de prix » et on crée une demande en choisissant le fournisseur, l'article et sa quantité :

Figure 22: La création d'une demande de prix

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

On peut envoyer notre demande de prix par courriel, ou l'imprimer et l'envoyer. Après avoir reçu une réponse du fournisseur, on confirme la commande en cliquant sur « Confirmer la commande », et automatiquement deux boutons se rajoutent à droite :

- Si on opte pour Expédition, ça va nous permettre de recevoir l'article dans notre stock, donc on valide et on aura cette quantité en stock :

Figure 23: Expédition d'un article entré en stock

- La partie Factures fournisseurs, va afficher automatiquement notre fournisseur et les articles commandés au préalable. Après avoir sauvegardé et validé la facture fournisseur, elle passe de l'état brouillon à l'état ouverte :

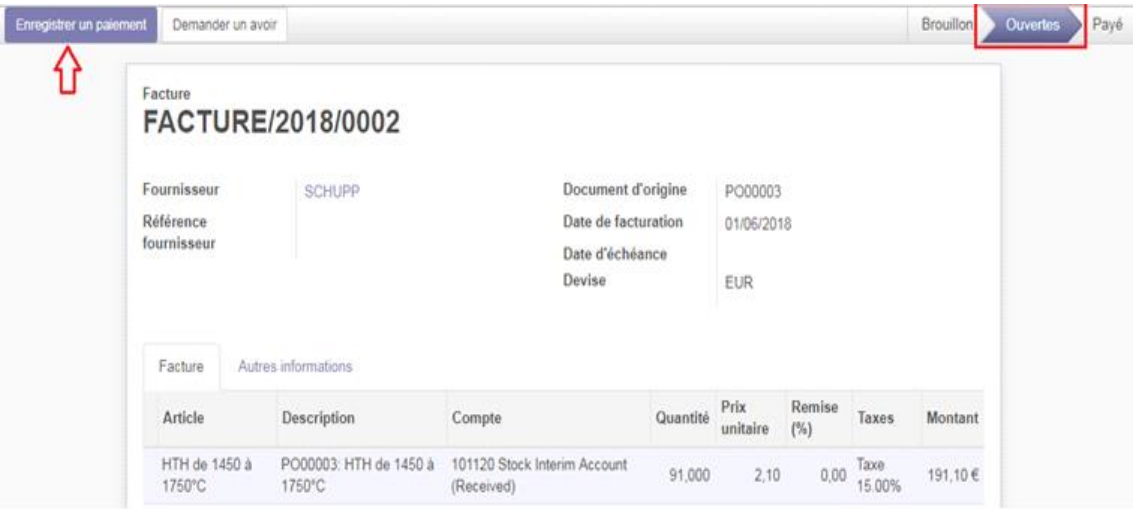


Figure 24: Création d'une facture fournisseur

Par la suite, on peut enregistrer un paiement en cliquant sur « Enregistrer un paiement », ou on peut choisir d'enregistrer le montant total ou partiel comme l'enregistrement d'un paiement dans la vente.

Une fois l'enregistrement de paiement est validé, la facture passe à état payé. Donc nous avons passé un flot d'achat classique avec une demande de prix et une commande fournisseur avec un paiement de la facture.

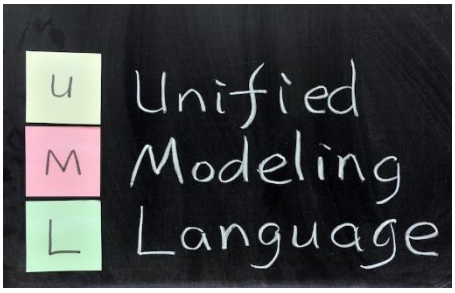
V. Conception

La conception est une phase cruciale dans le cycle de vie d'un système ; elle consiste à créer une représentation simplifiée de la réalité via une modélisation abstraite, de façon que le système soit fidèle aux demandes des futurs utilisateurs.

Pour cela, nous avons utilisé le standard de modélisation UML, vu qu'il permet de définir les aspects fonctionnels, structurels et dynamiques du système étudié.

V.1. UML

UML (Unified Modeling Language ou langage unifié de modélisation) est un langage graphique destiné à la modélisation de systèmes et des processus. UML est un langage basé sur l'approche objets, celle qui a d'abord conduit à la création des langages de programmation comme Java, C++ ou Smalltalk.



Aujourd'hui, UML est promu par l'OMG (Object Management Group), un consortium de plus de 800 sociétés et universités actives dans le domaine des technologies de l'objet. [13]

UML inclut treize diagrammes différents, en nous concernant, nous n'avons utilisé que ce dont nous avons réellement besoin, à savoir :

- Le diagramme de cas d'utilisation : est utilisé pour la modélisation des besoins fonctionnels des utilisateurs.
- Le diagramme de séquence : pour détailler les scénarios possibles des acteurs avec le système.
- Le diagramme de classe : Pour définir les composantes du système final et faciliter l'implémentation orienté objet.

V.2. Cas d'utilisation

Les cas d'utilisation décrivent sous la forme d'actions et de réactions, le comportement du système étudié du point de vue des utilisateurs. Ils définissent les limites du système et ses relations avec son environnement. [13]

Dans la suite de cette section, nous allons présenter les cas d'utilisations les plus pertinents :

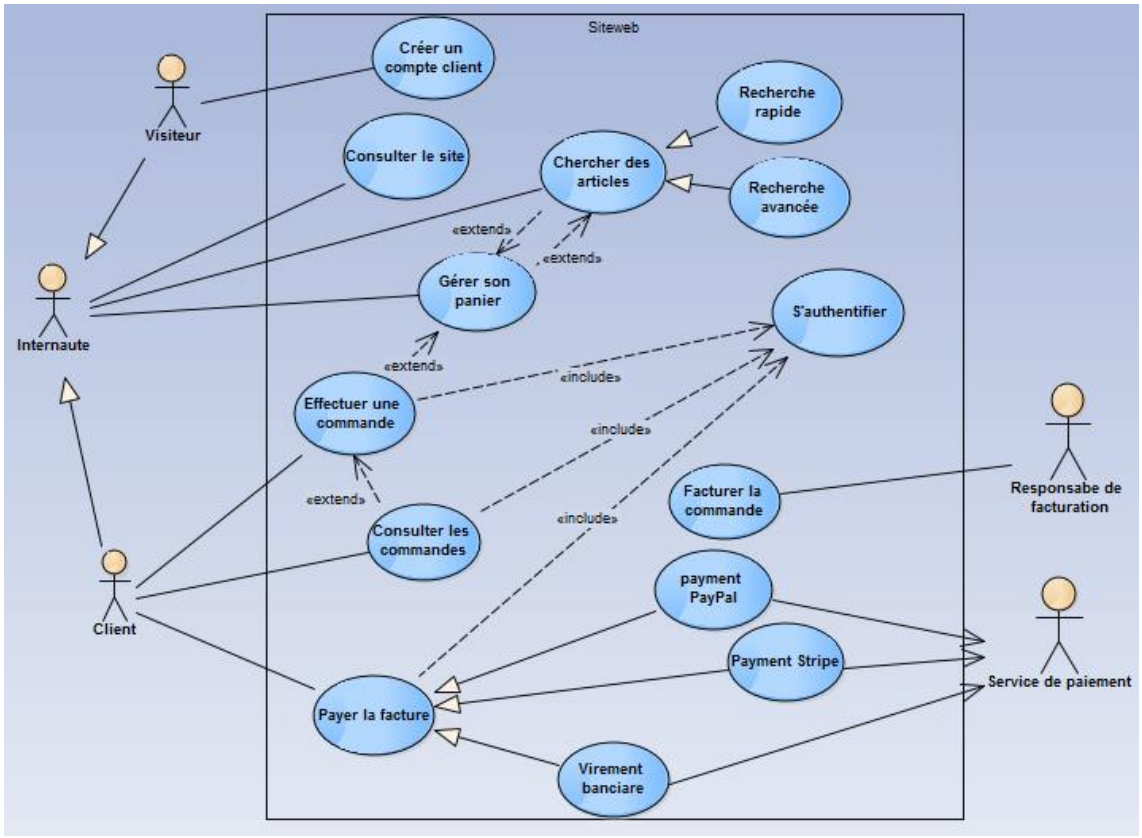


Figure 25: Cas d'utilisation d'effectuer une vente

Ce diagramme définit l'ensemble de cas d'utilisations dans le site web pour effectuer une commande. La relation « include » indique que l'authentification est obligatoire pour réaliser les différentes tâches, et la relation « extend » indique une réalisation optionnelle des tâches. Les acteurs impliqués sont les suivants :

- L'internaute : acteur général qui peut être client ou visiteur.
- Le visiteur : inconnu du site web, mais il peut néanmoins consulter le site, rechercher des articles et gérer son panier.
- Le client : il est déjà connu par le site grâce à un compte, peut seul effectuer une commande et suivre les étapes requises.
- Le responsable de facturation : administrateur qui gère les ventes.
- Service de paiement : sont les services de paiement tel que PayPal et Stripe.

V.3. Diagrammes de séquence

Le diagramme de séquence décrit les interactions entre un groupe d'objets en montrant, de façon séquentielle, les envois de message qui interviennent entre les objets. Le diagramme peut également montrer les flux de données échangées lors des envois de message. [13]

V.3.1. Recherche des articles

Etapes	Description
Résumé	**Acteurs :** internaute. **Titre :** rechercher des articles dans le site web.
Pré conditions	Les articles disponibles pour pouvoir effectuer une recherche.
Scénario nominal	• L'internaute recherche les articles. • Le système lui affiche les articles trouvés. • L'internaute sélectionne un article. • Le système lui affiche la fiche détaillée de l'article. • L'internaute ajoute l'article au panier.
Scénario alternatif	**2.a.** Aucun article trouvé.

Tableau 3: Recherche des articles

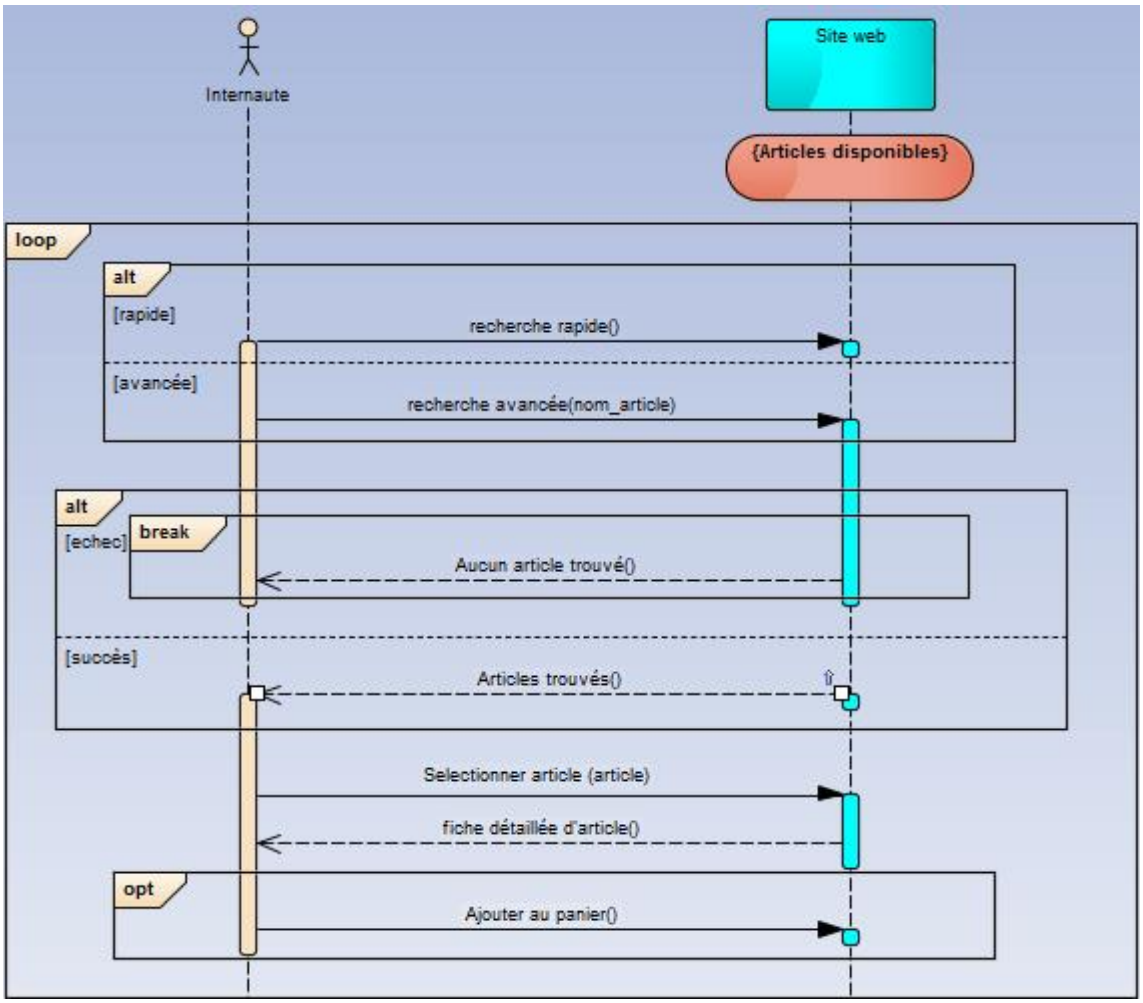


Figure 26: Diagramme de séquence de rechercher des articles

V.3.2. Gestion du panier

Etapes	Description
Résumé	**Acteurs :** L'internaute. **Titre :** Gérer son panier.
Pré conditions	Panier disponible.
Scénario nominal	• L'internaute modifie la quantité d'articles. • L'internaute supprime un article. • L'internaute accède au paiement.
Scénario alternatif	• Si la quantité égale à 0, le system supprime la ligne de commande. • L'internaute retourne à la boutique.

Tableau 4: Gérer son panier

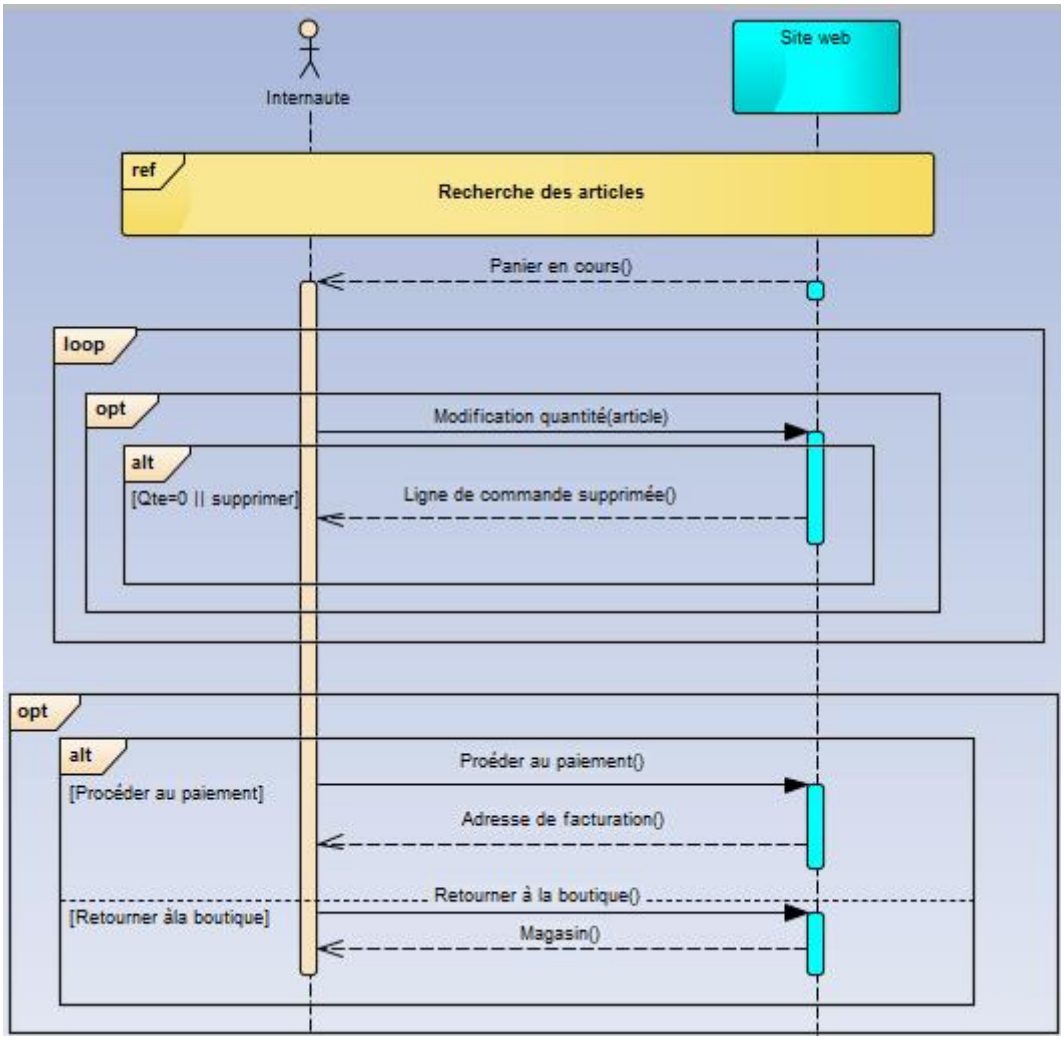


Figure 27: Diagramme de séquence de la gestion du panier

V.3.3. Effectuer une commande

Etapes	Description
Résumé	**Acteurs :** Client, visiteur. **Titre :** Effectuer une commande.
Pré conditions	Panier non vide. L'internaute doit être un client pour pouvoir effectuer une commande.
Scénario nominal	• Le client s'authentifie. • Le client confirme la commande. • Le système affiche la confirmation de commande. • Le client sélectionne le mode de paiement. • Le client paye la facture.
Scénario alternatif	**1.b.** Le visiteur crée un compte. **2.b.** Le client retourne à la boutique. **3.b.** Le client ajoute une autre adresse pour la facturation.

Tableau 5: Effectuer une commande

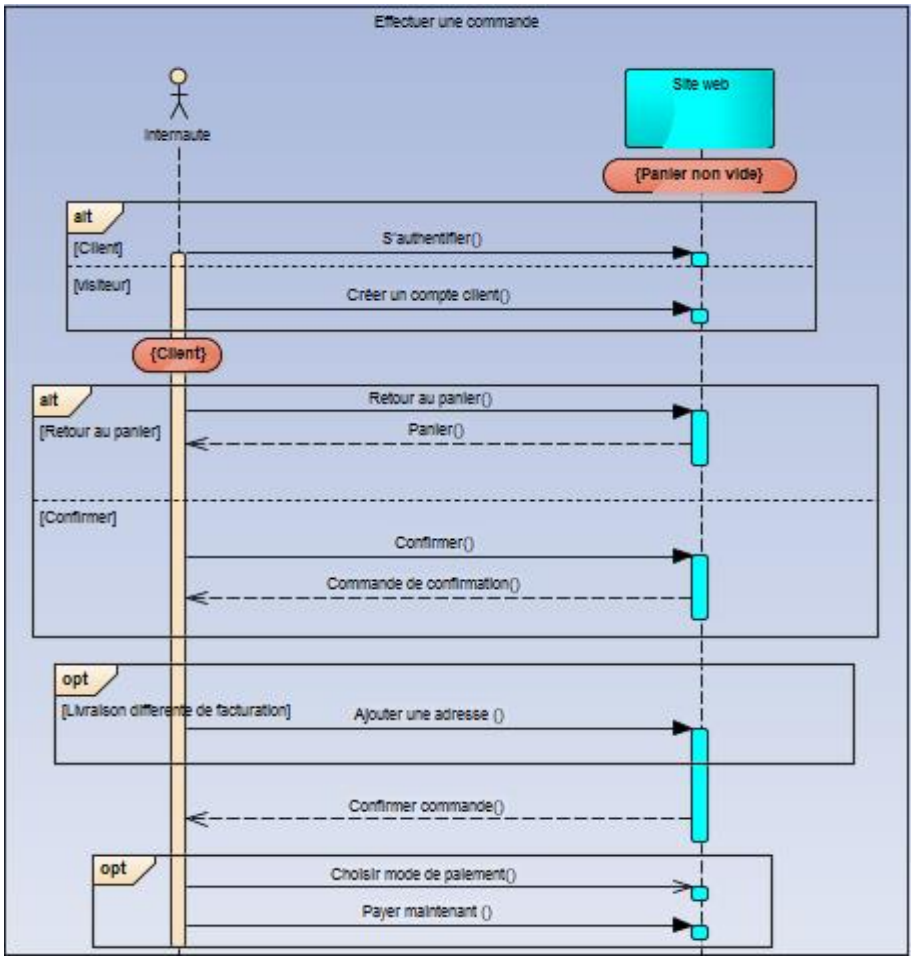


Figure 28: Diagramme de séquence d'effectuer une commande

Les acteurs pour UML, sont les entités externes qui interagissent avec le système. Dans notre système, il s'agit d'un administrateur qui gère les achats et les fournisseurs, les ventes et les clients et le stock.

IV.4. Diagrammes de classe

Il représente les classes intervenant dans le système. Le diagramme de classes est une représentation statique des éléments qui composent un système et leurs relations. Chaque application qui va mettre en œuvre le système sera une instance des différentes classes qui le compose. [14]

Le diagramme ci-dessus, présente les différentes classes qui participent dans le système. On a une relation de composition qui indique que la destruction de l'objet composé bon_de_commande implique la destruction de l'objet composant ligne_de_commade, et la même chose pour facture et ligne_de_facture.

VI. Conclusion

Dans le chapitre précédent, l'étude préliminaire et les éléments de base nécessaires pour la réalisation de note PFE, ont été décrites. Nous avons commencé par un bref survol du travail en introduisant l'entreprise cliente et ces spécifications. Ensuite nous avons présenté les différentes phases du projet qui sont : l'étude et l'analyse du système afin de pouvoir mettre les solutions qui conviennent, ainsi que modéliser la conception à l'aide des diagrammes UML.

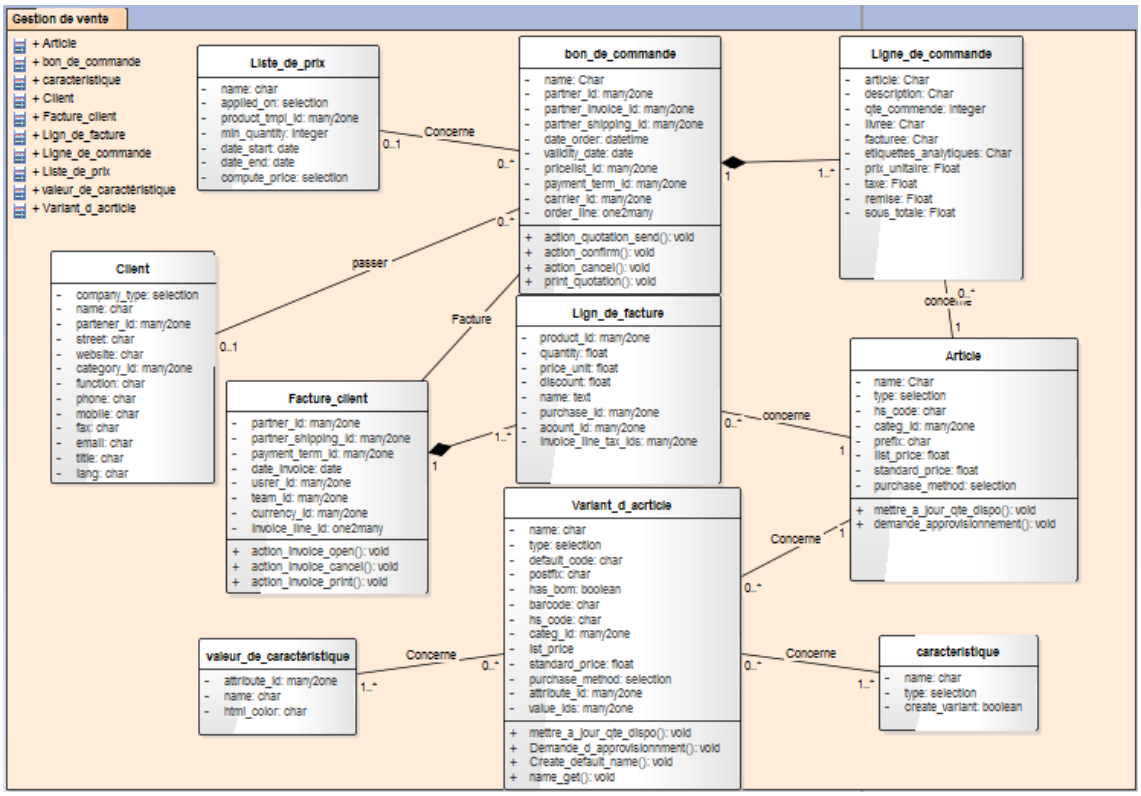


Figure 29: diagramme de classe de gestion de vente

Chapitre III : Réalisation

Rapport-Gratuit.com

I. Introduction

Après analyse de l'existant, nous avons commencé par concevoir une solution abstraite. Pour pouvoir la concrétiser et développer notre solution Odoo, il a fallu faire des configurations, écrire et réécrire des programmes en utilisant un certain nombre de langages différents, en tirant profil de la complétude du langage python et XML et JavaScript.

Dans ce qui suit, nous allons dérouler en expliquant les étapes suivies pour faire les configurations nécessaires et le développement de notre ERP Odoo.

II. Réalisation de transactions de paiement en ligne

Pour valider ces transactions en ligne, l'entreprise COMO exige de faire les configurations des intermédiaires de paiement **PayPal** et **Stripe**. Pour le faire, nous allons suivre les étapes suivantes :

II.1. PayPal

PayPal est un service de paiement en ligne pour régler des achats et recevoir ou transférer de l'argent sur Internet. Une fois que vous avez votre compte, vous pouvez l'utiliser même si vous n'avez pas d'argent sur PayPal. Le service va faire le pont entre votre banque et le vendeur et met à votre disposition gratuitement son système de paiement sécurisé. [15]

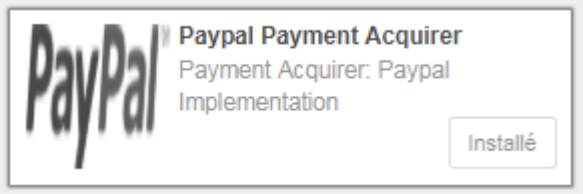
II.1.1. Comment accepter les paiements PayPal des factures

Le module « Intermédiaire de paiement PayPal » permet d'accepter les paiements par PayPal, pour les commandes clients, les factures, ou les commandes en ligne.

PayPal prend en charge deux modes de fonctionnements, un compte marchand qui redirige l'utilisateur vers PayPal pour faire le paiement, et un compte PayPalPayFlow qui ne redirige pas l'utilisateur vers PayPal et le paiement peut être fait en arrière-plan. Mais à ce jour, Odoo ne supporte que la méthode compte marchand PayPal.

II.1.2. Configuration

Pour accepter les paiements PayPal il faut commencer par installer le module « Intermédiaire de paiement PayPal ».Une fois installé, on le trouve dans Comptabilité => Configuration => Intermédiaires de paiement :



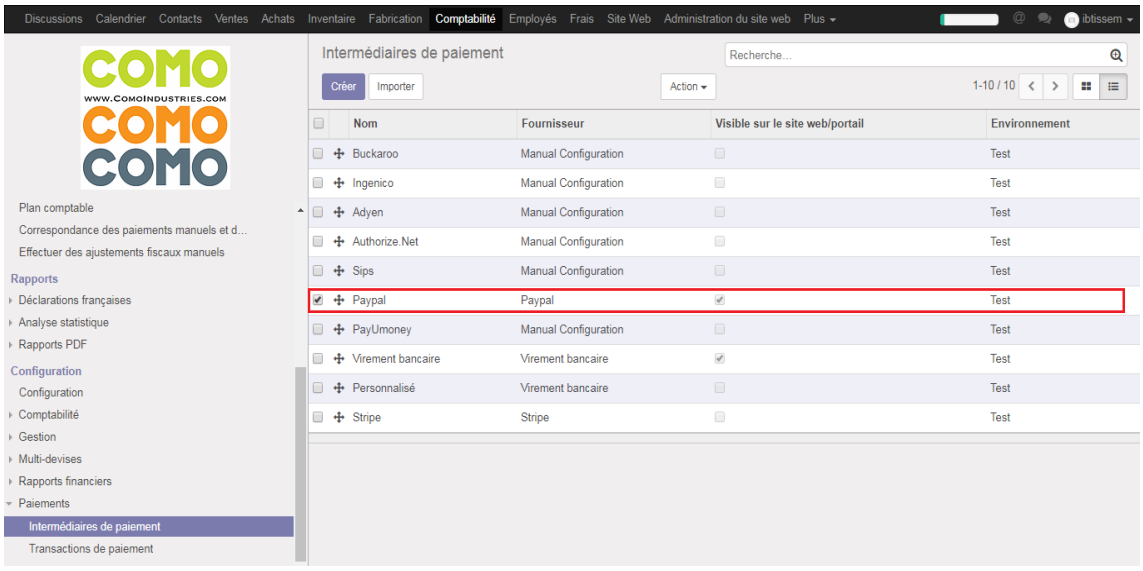
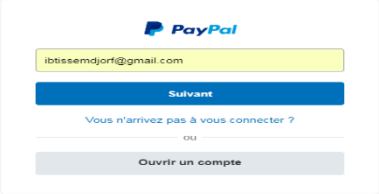


Figure 30: Le chemin pour arriver à l'intermédiaire de paiement PayPal dans Odoo

PayPal par défaut est configuré pour fonctionner dans un environnement de test pour pouvoir tester le chemin complet de la commande au paiement. Mais pour un fonctionnement dans un environnement correct il faut créer un compte marchand.

II.1.3. Création et configuration d'un compte PayPal marchand

Il faut commencer par créer un compte PayPal Business (marchand) ou un compte personnel puis on le transforme en compte PayPal marchand.



II.1.4. Configuration

1. Une fois notre compte est un compte marchand, on accède au menu Préférences => Préférences et paramètres => Mes ventes => Préférences de site marchand :

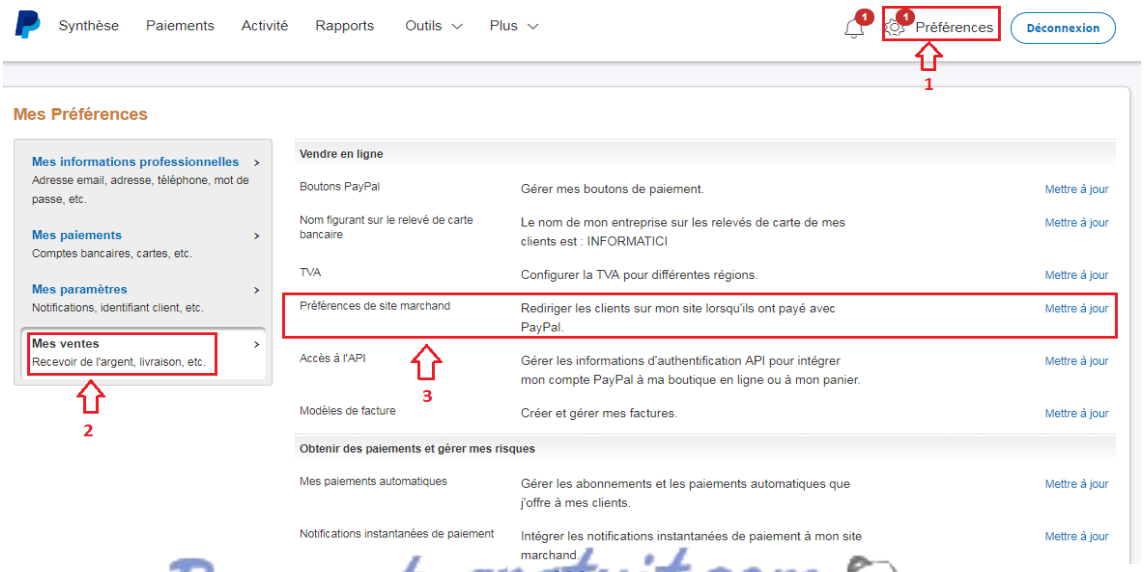


Figure 31: Les préférences du site web

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

2. Ensuite, il faut activer le renvoi automatique pour les paiements et mettre à jour l'URL de renvoi de notre compagne par : http ://c10.sushi-toppings.w1037.openerp-online.fr/shop/confirmation.

Figure 32: configuration du revoie automatique et l'url de renvoie

Ces paramètres redirigent notre acheteur vers notre page web de confirmation de la commande :

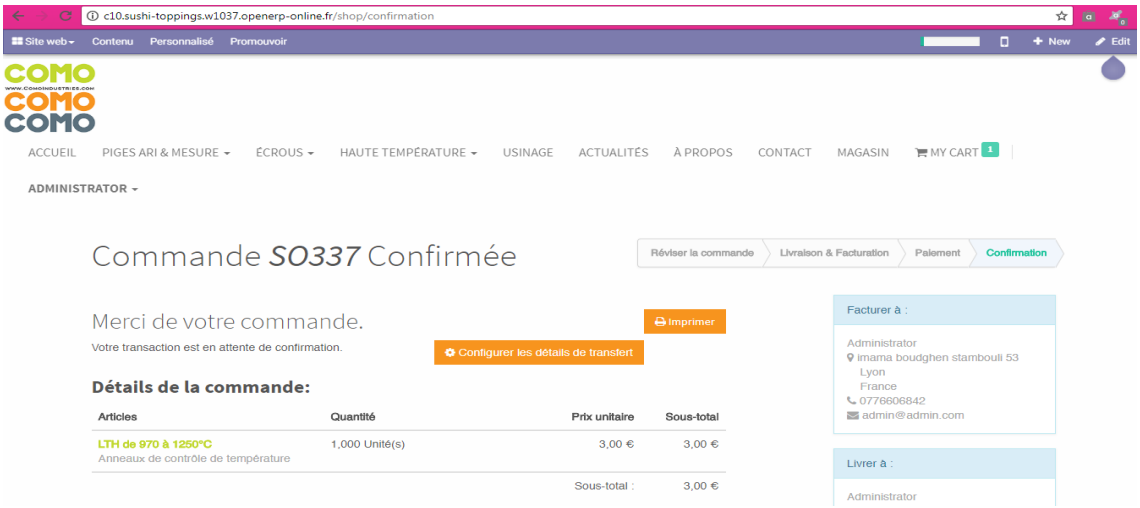


Figure 33: Page web de confirmation de la commande

3. Pour que les clients puissent payer avec PayPal sans avoir de compte PayPal, le paramètre « Compte PayPal facultatif » doit être réglé sur ON, qui va permettre aux utilisateurs de choisir l'option de payer par carte sans la nécessité d'avoir un comptePayPal.

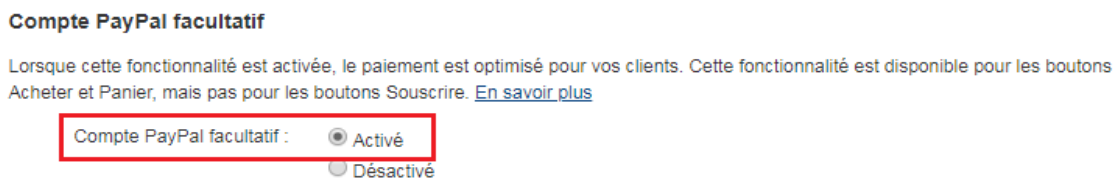


Figure 34: Activation d'un compte PayPal facultatif

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

4. Apres « Préférences de site marchand », on accède à « Notifications instantanées de paiement **».**
5. Les messages de notification instantanée de paiement sont envoyés par PayPal aux URL spécifiées ci-dessous, et pour les recevoir il faut activer les messages de notification instantanée IPN :

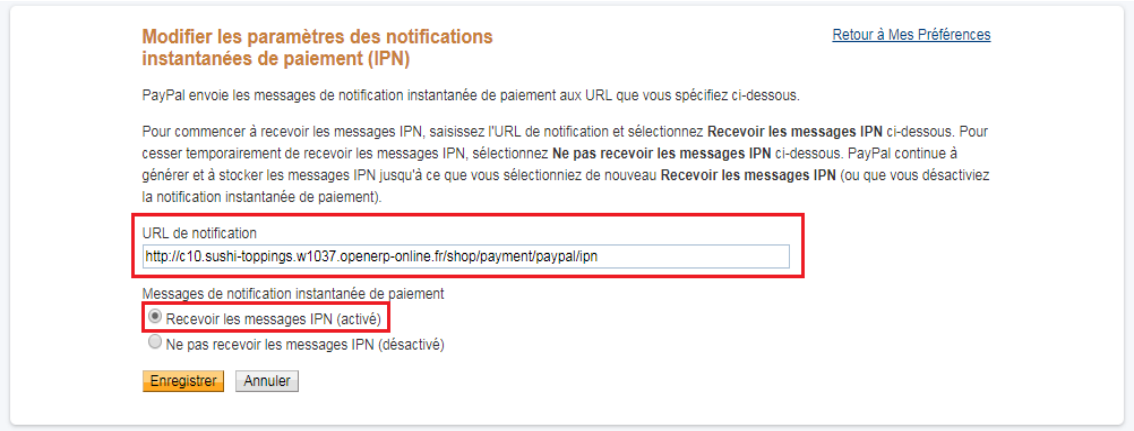


Figure 35: Modifications des paramètres de notifications instantanées de paiement IPN

6. Après terminer la modification et la configuration des préférences du site web et les notifications instantanées, nous allons définir l'encodage UTF-8 en allant à Préférences=>Mes ventes=>Autres outils de vente=>Encodage de langage des boutons PayPal =>Autres options.

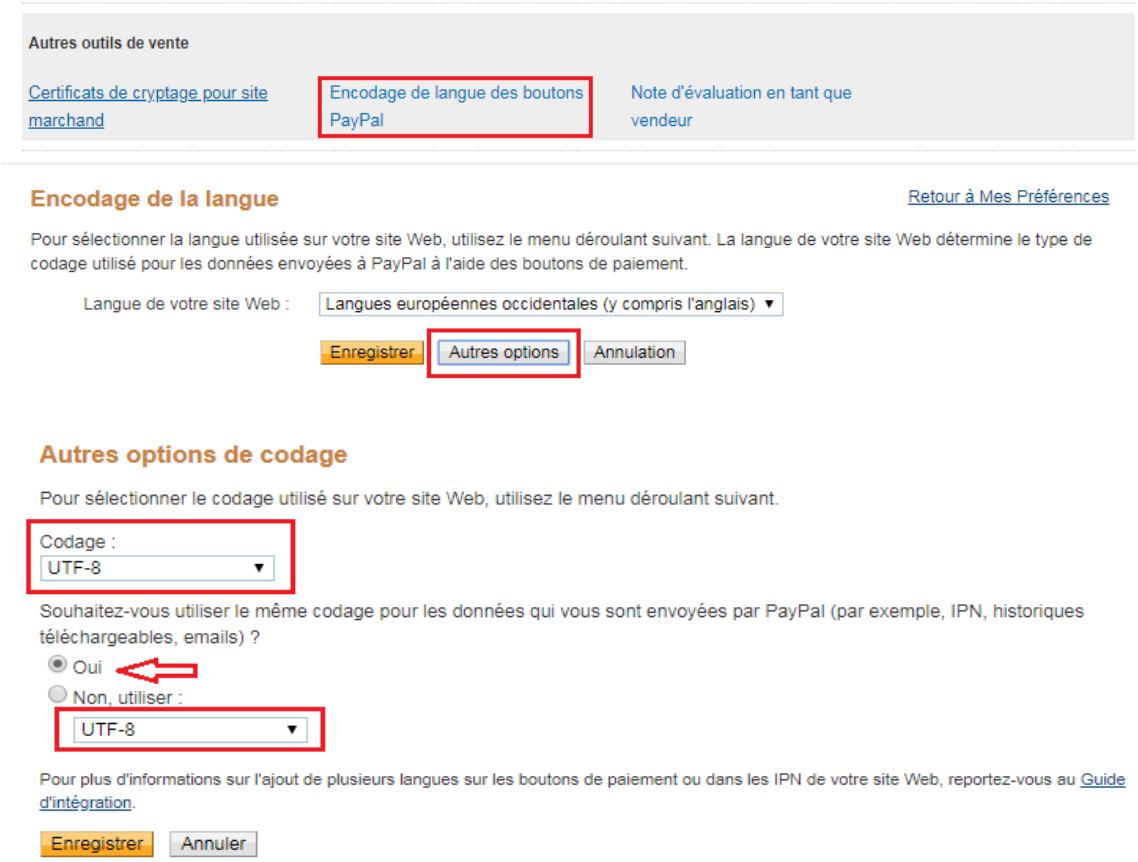


Figure 36: Sélectionner le codage utilisé sur le site web

7. Une fois la configuration terminée, l'identifiant de compte marchand qui sera utilisé dans Odoo, est récupéré à partir de « Mes informations personnelles ».

Figure 37: Récupération d' l'identifiant de compte marchand et l'adresse mail

8. Finalement, on va configurer dans Odoo à partir des informations récupérées ci-dessus en modifiant les paramètres de l'intermédiaire de paiement PayPal :

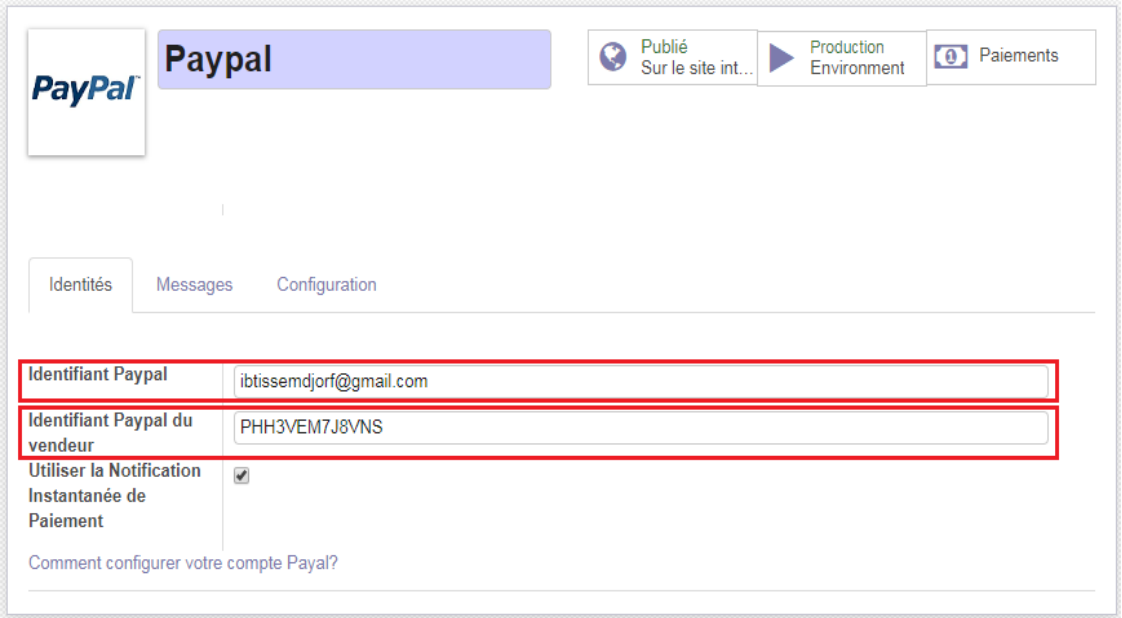


Figure 38: Configuration de l'intermédiaire de paiement PayPal

9. Après avoir terminé la configuration du compte Paypal et l'intermédiaire de paiement PayPal à Odoo, nous allons pouvoir payer avec PayPal en sélectionnant PayPal comme mode de paiement qui va nous rediriger vers PayPal.

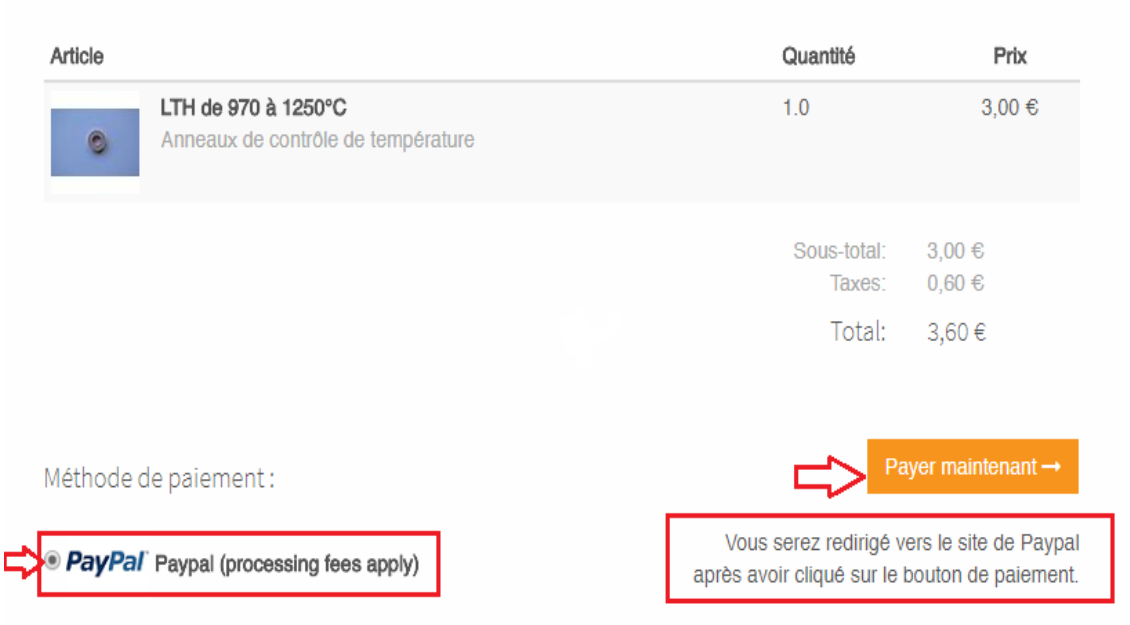


Figure 39: choisir PayPal comme méthode de paiement

Après avoir cliqué sur « payer maintenant » il vous redirige vers le site du paypal.

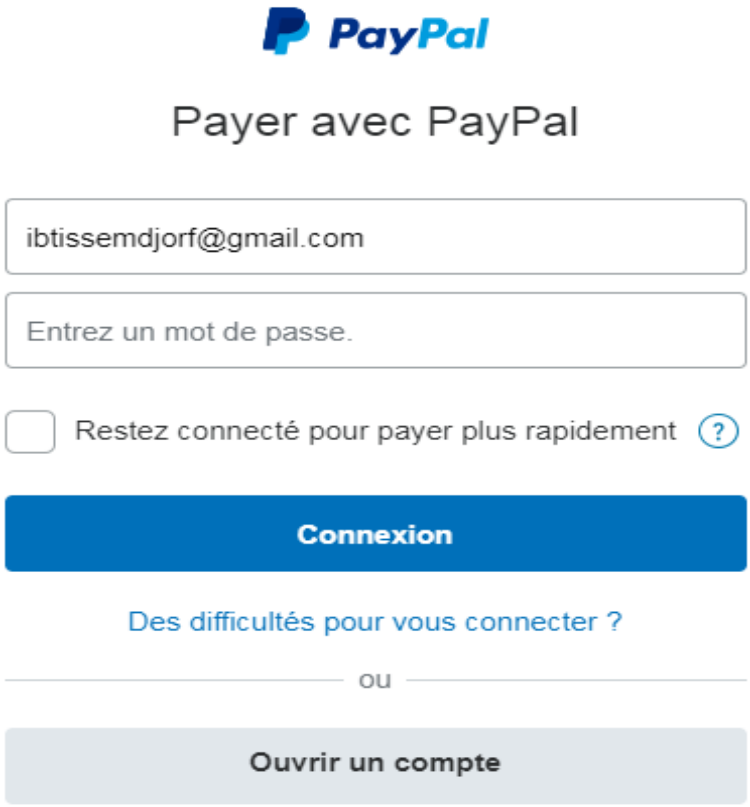


Figure 40: Redirection vers PayPal pour se connecter à son compte

.

II.2. Stripe

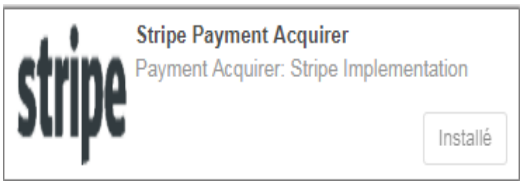
Stripe est un nouveau service facilitant le paiement en ligne. Ce moyen de paiement est rapidement devenu une référence en matière de paiement e-commerce. De nombreuses grandes entreprises l'utilisent aussi comme : Twitter, Facebook, Pinterest, Shopify, et pleins d'autres. Il est rapide et sécurisé, ainsi vous n'aurez ni à vous inscrire ni à quitter notre site pour régler votre commande (contrairement à PayPal). Il utilise l'algorithme de chiffrement de niveau 1 de la norme PCI (le niveau le plus élevé de certification). [16]

II.2.1. Comment accepter les paiements Stripe des factures

Le module « Intermédiaire de paiement Stripe » permet d'accepter les paiements par Stripe, pour les commandes client, les factures, ou les commandes en ligne.

II.2.2. Configuration

Pour accepter les paiements Stripe, il faut commencer par installer le module « Intermédiaire de paiement Stripe ».



Comme PayPal après l'avoir installé, on le trouve dans Comptabilité => Configuration => Intermédiaires de paiement :

Stripe est configuré dans Odoo pour fonctionner dans un environnement de test pour pouvoir faire le test du chemin de paiement de la commande. Comme PayPal, Stripe fonctionne aussi à partir d'un compte marchand.

II.2.3. Création et configuration d'un compte Stripe marchand

1. On commence par créer un compte Stripe.
2. Ensuite, il faut cliquer sur « API keys », puis «reveal test key token » pour récupérer la clé secrète et la clé publiée :

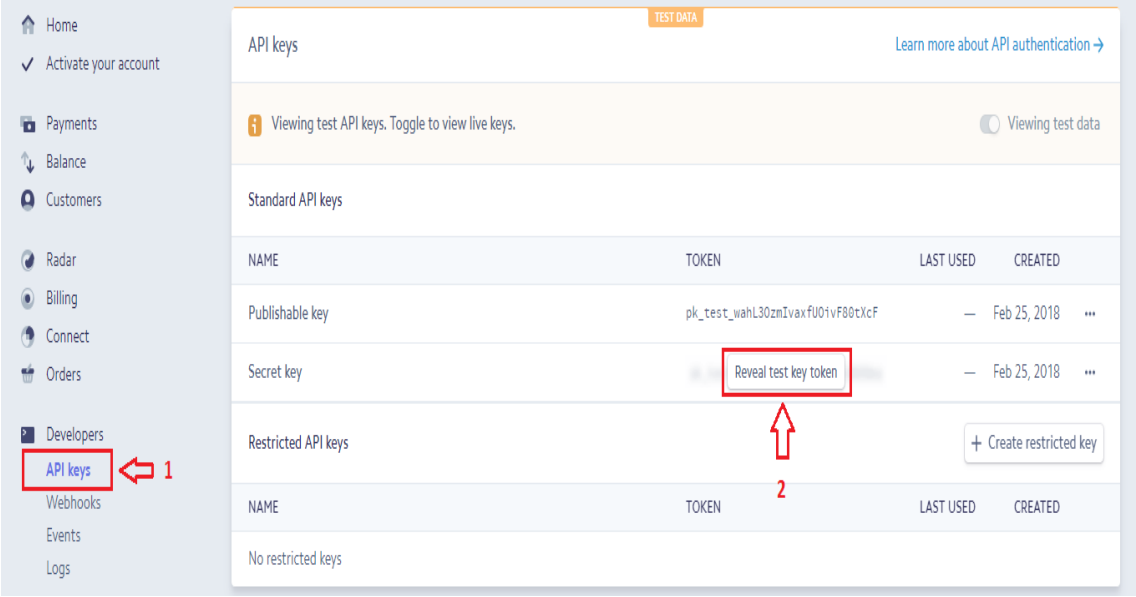


Figure 41: Récupération des clés API

3. Une fois on a choisi « reveal test key token », la clé secrète et la clé publiée vont être récupérées pour la configuration d'Odoo .

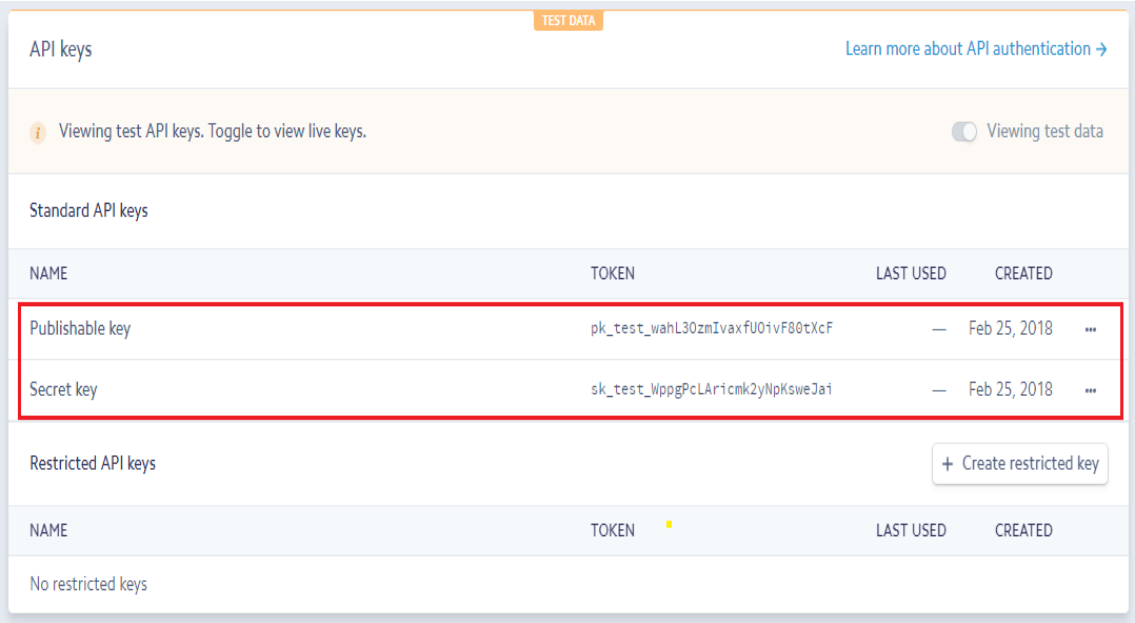


Figure 42: La clé secrète et la clé publiée du compte Stripe

4. Enfin, on récupère ces clés et on configure l'intermédiaire de paiement Stripe dans Odoo pour qu'il fonctionne.

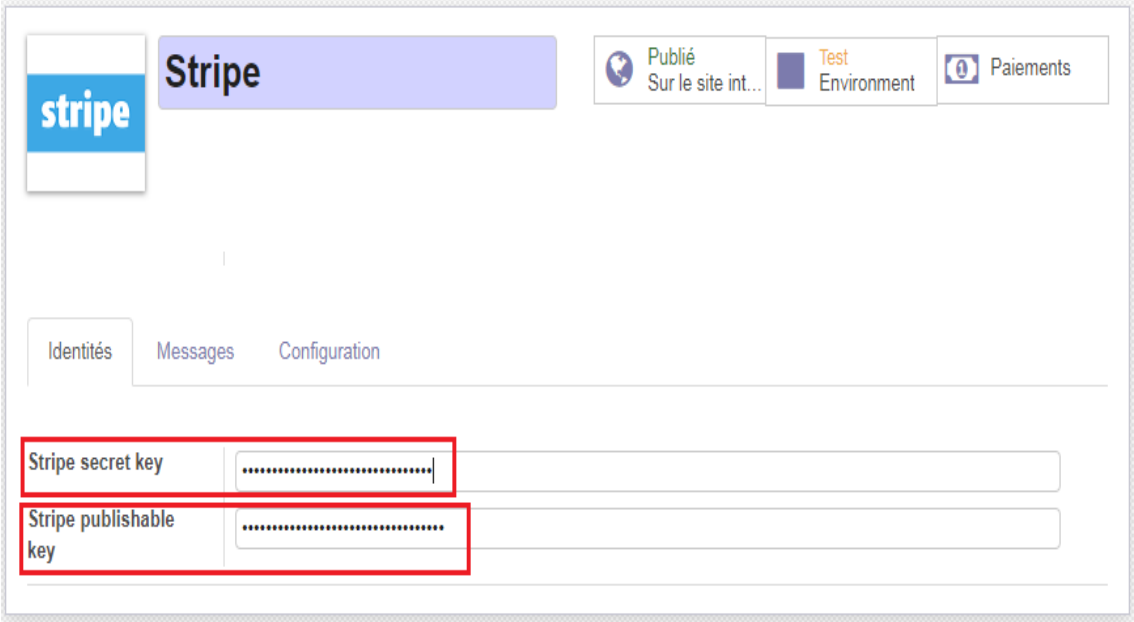


Figure 43: configuration de l'intermédiaire de paiement Stripe

Remarque : pour un fonctionnement de paiement des commandes clients par Stripe correcte, il faut bien s'assurer que les clés sont en mode live et non pas en mode.

Une fois les configurations faites, on pourra choisir Stripe comme méthode de paiement des commandes de clients juste en le sélectionnant.

Figure 44: Payer une commande en utilisant Stripe comme mode de paiement

III. Configurations pour les anneaux de contrôles

Comme objectif, nous avons à vendre les anneaux de contrôles et faire les configurations dans la base de test de l'entreprise cliente COMO.

III.1 Ajout des anneaux de contrôles :

On a ajouté les piges de contrôles UTH, STH, MTH, ZTH, HTH, LTH et ETH qui sont vendus par jeux de 15 anneaux. Le tableau suivant représente les prix unitaires des articles :

DESIGNATION	1 jeu de 15 anneaux	PU de 15 à 90 anneaux	PU de 105 à 585 anneaux	PU de 600 à 1200	PU au-delà
ZTH	4	2.2	1.58	1.05	0.96
UTH	3	1.95	1.31	0.95	1.78
ETH	3	1.95	1.31	0.95	0.78
LTH	3	1.95	1.31	0.95	0.78
STH	3	1.95	1.31	0.95	0.78
MTH	3.33	2.1	0.99	0.99	0.95
HTH	3.33	2.1	0.99	0.99	0.95

Tableau 6: Tarif des anneaux de contrôle

Puisque nous n'avons pas la possibilité d'ajouter ces prix dans des intervalles, il fallait ajouter une liste de prix(PTCR) pour ces anneaux, et les éléments de cette liste vont avoir la quantité minimum de chaque intervalle. Le tableau suivant présente ces quantités :

DESIGNATION	1 jeu de 15 anneaux	PU de 15 à 90 anneaux	PU de 105 à 585 anneaux	PU de 600 à 1200	PU au-delà
QTE minimum	15	16	91	586	1 201
ZTH	4	2.2	1.58	1.05	0.96
UTH	3	1.95	1.31	0.95	1.78
pETH	3	1.95	1.31	0.95	0.78
LTH	3	1.95	1.31	0.95	0.78
STH	3	1.95	1.31	0.95	0.78
MTH	3.33	2.1	0.99	0.99	0.95
HTH	3.33	2.1	0.99	0.99	0.95

Tableau 7: Tarif des anneaux avec la quantité minimum de chaque intervalle

Rapport-Gratuit.com

Les éléments de la liste de prix PTCR représente chacune un prix unitaire d'un intervalle selon sa quantité minimum. La figure suivante représente le prix unitaire de l'anneau de contrôle ZTH. En choisissant une quantité dans l'intervalle [16, 90], la quantité minimum est de 16 anneaux :

Figure 45: Prix unitaire de l'anneau de contrôle ZTH dans l'intervalle [16 90]

Voici les prix de tous les anneaux selon leurs intervalles de quantités vendues :

Éléments de la liste de prix

Applicable à	Quantité min.	Date de début	Date de fin	Prix
HTH de 1450 à 1750°C	1 201			0.95 EUR
ZTH de 560°C à 660°C	1 201			0.96 EUR
MTH de 1340 à 1520°C	1 201			0.95 EUR
STH de 1130 à1400°C	1 201			0.78 EUR
LTH de 970 à 1250°C	1 201			0.78 EUR
HTH de 1450 à 1750°C	586			0.99 EUR
ZTH de 560°C à 660°C	586			1.05 EUR
MTH de 1340 à 1520°C	586			0.99 EUR
STH de 1130 à1400°C	586			0.95 EUR
LTH de 970 à 1250°C	586			0.95 EUR
HTH de 1450 à 1750°C	91			1.55 EUR
ZTH de 560°C à 660°C	91			1.58 EUR
MTH de 1340 à 1520°C	91			1.55 EUR
STH de 1130 à1400°C	91			1.31 EUR
LTH de 970 à 1250°C	91			1.31 EUR
HTH de 1450 à 1750°C	16			2.1 EUR
ZTH de 560°C à 660°C	16			2.2 EUR
MTH de 1340 à 1520°C	16			2.1 EUR
STH de 1130 à1400°C	16			1.95 EUR
LTH de 970 à 1250°C	16			1.95 EUR
HTH de 1450 à 1750°C	15			3.33 EUR
ZTH de 560°C à 660°C	15			4.0 EUR
MTH de 1340 à 1520°C	15			3.33 EUR
STH de 1130 à1400°C	15			3.0 EUR
LTH de 970 à 1250°C	15			3.0 EUR

Figure 46: Tarif des anneaux selon la quantité vendu

IV. Configurations pour les Piges ARI

Notre objectif pour les piges ARI, est de les vendre selon des variantes (caractéristiques), ainsi les acheteurs peuvent commander des piges en précisant la taille du diamètre qui leur convient.

IV.1. Ajout des variantes

Pour pouvoir ajouter des variantes, il faut commencer par cocher la case « Les articles peuvent avoir plusieurs attributs, définissant des variantes (exemple : taille, couleur, ...) » dans la configuration des ventes.

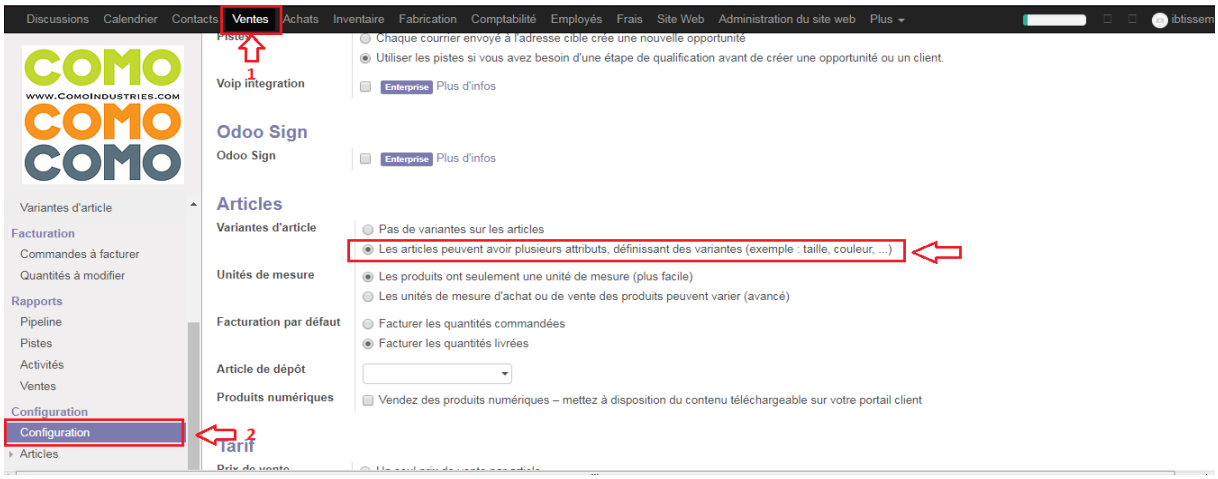


Figure 47: Configuration des ventes pour avoir des variantes d'article

IV.2. Affichage des variantes coté site web

En ajoutant les variantes, l'article qui les contient s'affiche dans le site avec toutes les variantes comme ceci :

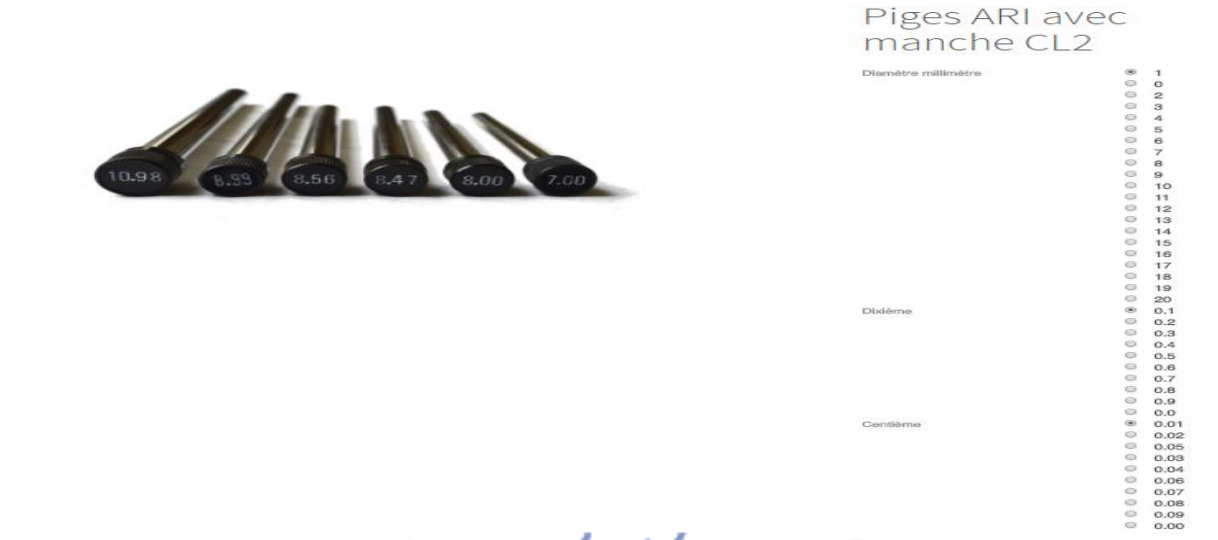


Figure 48: L'affichage des variantes d'articles de type Radio

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

L'entreprise COMO, veut que l'affichage de ces caractéristiques s'affiche sous forme de sélections, donc il fallait les configurer en choisissant le type sélection au lieu radio.

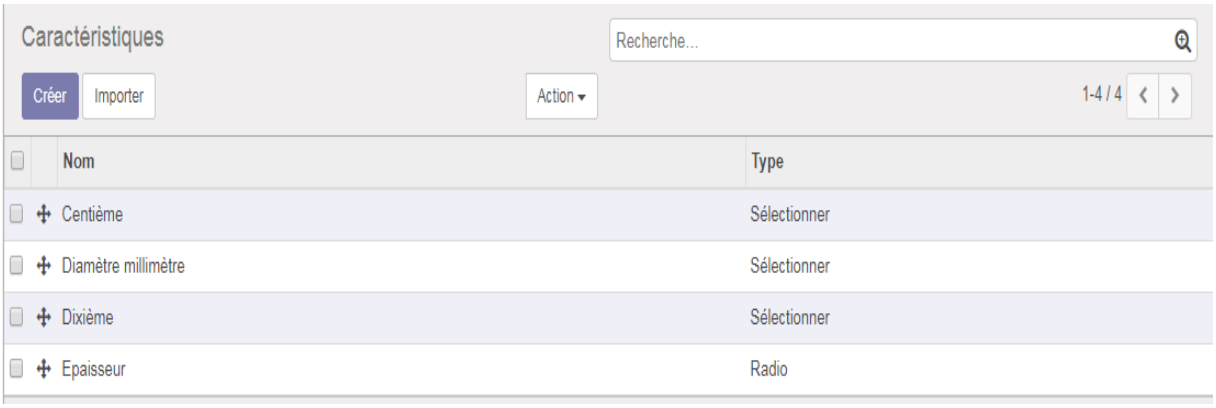


Figure 49: configuration de type des caractéristiques

Après avoir configuré les caractéristiques des variantes, l' affichage se présente comme ceci :

Figure 50: L'affichage des variantes d'article sous forme de sélections

IV.3. Le module product_variant_name

Ce module a comme objectif, la génération automatique des références internes des variantes après la création de article avec l'ajout d'un postfix qui définit l'article selon ces variantes, et la modification dans le site en masquant le prix des variantes.

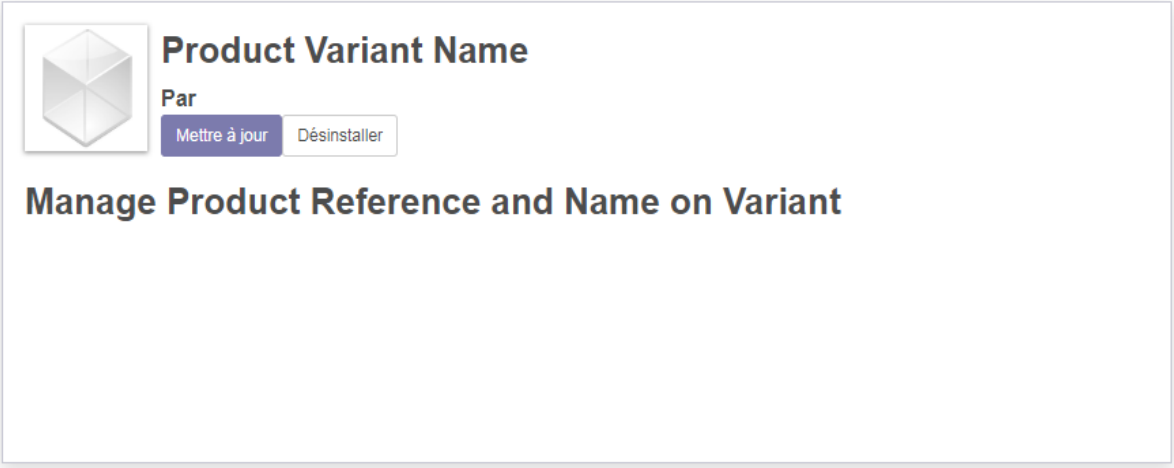


Figure 51: Le module qui gère les références internes et leurs supplémentaires

IV.3.1. Génération automatique des références internes :

Afin de générer automatiquement la référence interne des variantes, il fallait ajouter un champ « Préfix » pour les articles qui va être suivi d'un zéro, puis, le diamètre millimètre, puis la dixième, ensuite la centième. Ce préfix s'ajoute dans la vue des articles.

Nous avons aussi ajouté le champ « Postfix » pour définir les variantes :

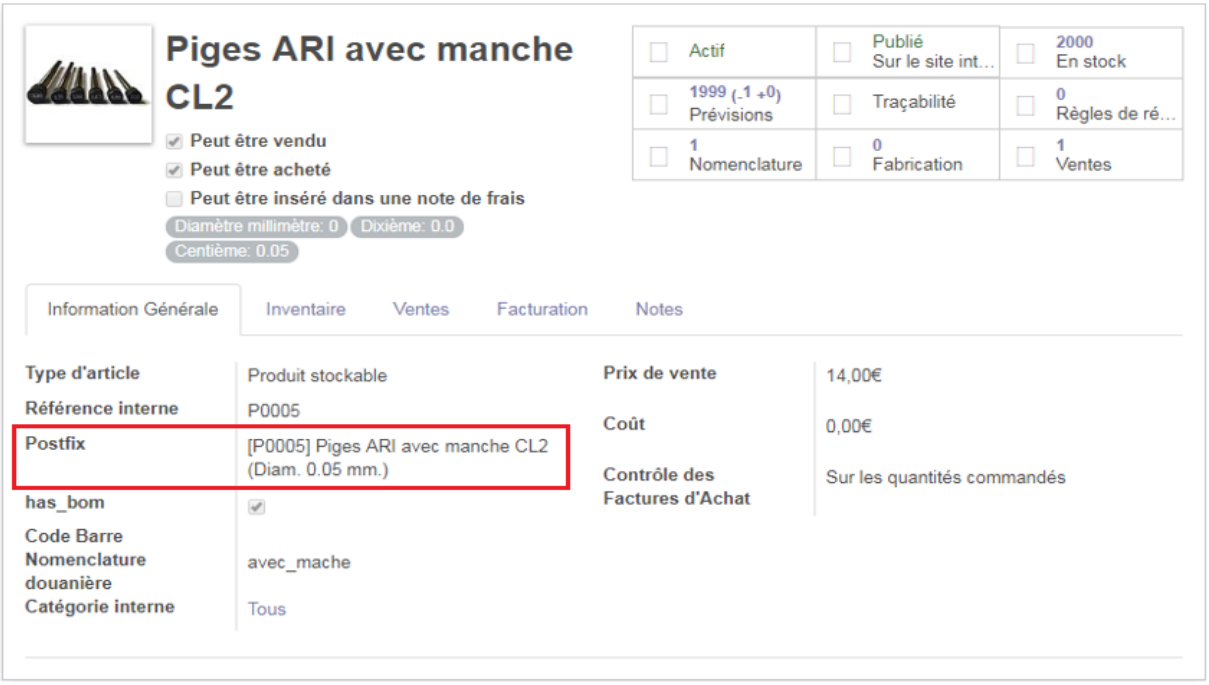


Figure 52 : L'ajout du postfix dans la vue variante

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Figure 53: L'ajout du préfix dans la vue article

```python
class product_template(models.Model):
    _inherit = "product.template"
    prefix = fields.Char(string="Prefix")
```

Figure 54: Code python de l'ajout de l'objet préfix .

```xml
<!--ajout de préfix danx la vue de variantes articles -->
<record id="prefix_product_template" model="ir.ui.view">
    <field name="name">product.product.template.prefix</field>
    <field name="model">product.template</field>
    <field name="inherit_id" ref="product.product_template_only_form_view"/>
    <field name="arch" type="xml">
        <field name = "categ_id" position='after'>
            <field name ="prefix"/>
        </field>
    </field>
</record>
```

Figure 55: Extrait du code XML de l'ajout du préfix dans la vue article

```python
#Génération du code des variantes des articles
@api.model
def create(self, vals):
    pdt_id = super(product_product, self).create(vals)
    res={}
    if not vals.get("default_code"):
        if vals.get("product_tmpl_id") :
            prefix = self.env["product.template"].browse(vals.get("product_tmpl_id"))
            if prefix.prefix:
                if prefix.prefix.upper() == 'S' or prefix.prefix.upper() == 'P':
                    value = 0
                    attr_values = [float(vals.name) for vals in pdt_id.attribute_value_ids]
                    attr_values.sort(reverse=True)
                    if attr_values:
                        value = sum(attr_values)
                        value1 = str(value).replace(".",'')
                        if len(value1) == 2:
                            value1 = "0"+value1+"0"
                        elif len(value1) == 3 and attr_values[0]<10:
                            value1 = "0"+value1
                        elif len(value1) == 3 and attr_values[0]>9:
                            value1 = value1+"0"
                        res["default_code"] = "%s%s" %(prefix.prefix.upper(),value1)
                        res["postfix"] = "[%s] %s (%s %.2f %s)" %(res["default_code"], prefix.name, "Diam.", value, "mm.")
                        pdt_id.write(res)
    return pdt_id
```

Figure 56: Code Python de la génération automatique de la référence interne et le Postfix des variantes

Référence interne	Nom	Caractéristiques	Prix de vente	Quantité en stock	Quantité prévue	Code Barre
P0000	Piges ARI avec manche CL2	Diamètre millimètre: 0, Centième: 0.00, Dixième: 0.0	14,00	2 000,000	2 000,000	
P0001	Piges ARI avec manche CL2	Diamètre millimètre: 0, Dixième: 0.0, Centième: 0.01	14,00	2 000,000	2 000,000	
P0002	Piges ARI avec manche CL2	Dixième: 0.0, Diamètre millimètre: 0, Centième: 0.02	14,00	3 000,000	2 975,000	
P0003	Piges ARI avec manche CL2	Diamètre millimètre: 0, Centième: 0.03, Dixième: 0.0	14,00	1 000,000	1 000,000	
P0004	Piges ARI avec manche CL2	Dixième: 0.0, Centième: 0.04, Diamètre millimètre: 0	14,00	1 000,000	1 000,000	
P0005	Piges ARI avec manche CL2	Dixième: 0.0, Diamètre millimètre: 0, Centième: 0.05	14,00	2 000,000	1 999,000	
P0006	Piges ARI avec manche CL2	Centième: 0.06, Dixième: 0.0, Diamètre millimètre: 0	14,00	0,000	0,000	
P0007	Piges ARI avec manche CL2	Centième: 0.07, Dixième: 0.0, Diamètre millimètre: 0	14,00	0,000	0,000	
P0008	Piges ARI avec manche CL2	Centième: 0.08, Diamètre millimètre: 0, Dixième: 0.0	14,00	0,000	0,000	
P0009	Piges ARI avec manche CL2	Dixième: 0.0, Centième: 0.09, Diamètre millimètre: 0	14,00	0,000	0,000	
P0010	Piges ARI avec manche CL2	Diamètre millimètre: 0, Dixième: 0.1, Centième: 0.00	14,00	0,000	0,000	
P0011	Piges ARI avec manche CL2	Dixième: 0.1, Diamètre millimètre: 0, Centième: 0.01	14,00	0,000	0,000	
P0012	Piges ARI avec manche CL2	Centième: 0.02, Diamètre millimètre: 0, Dixième: 0.1	14,00	0,000	0,000	
P0013	Piges ARI avec manche CL2	Dixième: 0.1, Centième: 0.03, Diamètre millimètre: 0	14,00	0,000	0,000	
P0014	Piges ARI avec manche CL2	Diamètre millimètre: 0, Dixième: 0.1, Centième: 0.04	14,00	0,000	0,000	

Figure 57: Les références internes générées automatiquement dans le système

IV.3.1. Masquer les prix supplémentaires des caractéristiques des variantes

Les variantes peuvent avoir un prix supplémentaire pour chaque caractéristique qui se rajoute au prix de l'article et qui s'affiche dans le site web devant les variantes.

Figure 58: Le prix de l'article variant selon le supplémentaire des caractéristiques

L'entreprise COMO ne veut pas afficher ce supplémentaire devant les variantes, pour la réalisation de cette tâche, il fallait ajouter le commentaire dans la vue du site web et la vue du devis dans le système en ajoutant une fonction python qui récupère le commentaire du site web vers la vue devis.

```
<!-- cacher la différence de prix qui s'affiche aprés les variants dans la vue du site web-->
<template id="product_variants_inherit" inherit_id="website_sale.variants">
    <xpath expr="//span[@class='badge']" position="replace"/>
    <xpath expr="//select/t/option/span[@t-if='value_id.price_extra']" position="replace"/>
</template>
```

Figure 59: Extrait du code XML pour masquer le supplémentaire

V. Affichage d'une alerte lors de l'ajout dans le panier

Lorsque le client ajoute un article dans son panier, une alerte s'affiche en lui donnant la main de modifier la quantité, et de choisir entre procéder au paiement, ou retourner au magasin

Figure 60: Alerte qui indique un produit à ajouter dans le panier

V.1 Le module como_ecommerce

Pour la réalisation de ce travail, nous avons ajouté le module « **como_ecommerce** », en ajoutant l'alerte en utilisant le langage de programmation JavaScript.

```
$(document).ready(function () {
    var url = window.location.href;
    if (url.search('added') > 0)
        $('.js_sale').prepend("<div id ='myAlert' class ='alert alert-danger' width='50%'>
            <a href = '#' class = 'close' data-dismiss = 'alert'>&times;</a>
            <strong><center>Success : Product has been added to your shopping cart.</center></strong></div>");
});
```

Figure 61: Code JavaScript de l'ajout d'une alerte

```
<?xml version="1.0" encoding="UTF-8"?>
<odoo>
    <data>
        <template id="assets_frontend" inherit_id="website.assets_frontend" name="Shop">
          <xpath expr="." position="inside">
              <script type="text/javascript" src="/como_ecommerce/static/src/js/website_sale.js"></script>
          </xpath>
        </template>
    </data>
</odoo>
```

Figure 62: Code XML pour ajouter l'alerte en JavaScript dans le site web

```
class website_sale(http.Controller):

    @http.route(['/shop/cart/update'], type='http', auth="public", methods=['POST'], website=True)
    def cart_update(self, product_id, add_qty=1, set_qty=0, **kw):
        cr, uid, context = request.cr, request.uid, request.context
        request.website.sale_get_order(force_create=1)._cart_update(product_id=int(product_id), add_qty=float(
        product_tmpl_id = request.registry['product.product'].read(cr, SUPERUSER_ID, int(product_id), ['produ
        product_name = product_tmpl_id and product_tmpl_id[1].replace(" ", "-").replace(".", "-") + "-" + str(
        return request.redirect("/shop/product/" + product_name.lower())
# vim:expandtab:tabstop=4:softtabstop=4:shiftwidth=4:
```

Figure 63: Extrait du code Python qui gère l'alerte

VI. Ajout du commentaire dans le site web

Après avoir procéder au paiement, et avant de confirmer la commande le client peut ajouter un commentaire qui sera enregistré dans le devis du client dans le système

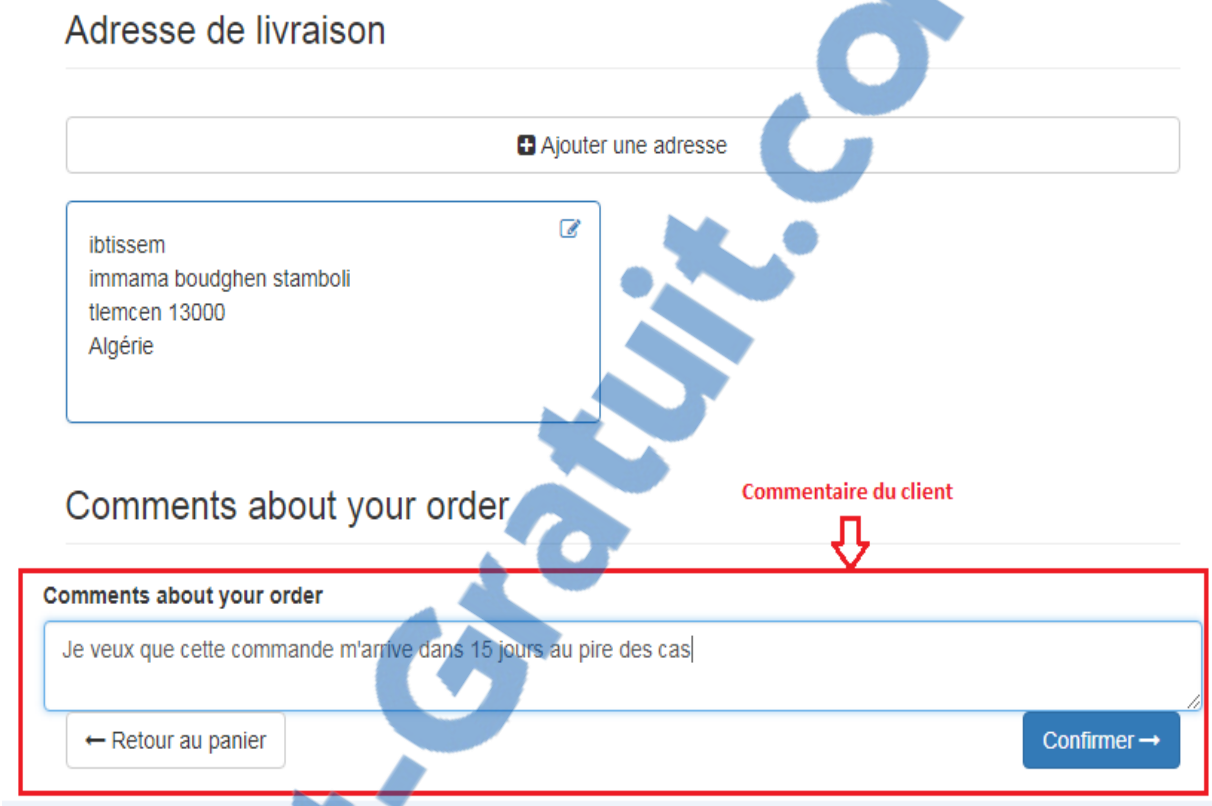


Figure 64: L'ajout du commentaire client sur sa commande par le site web

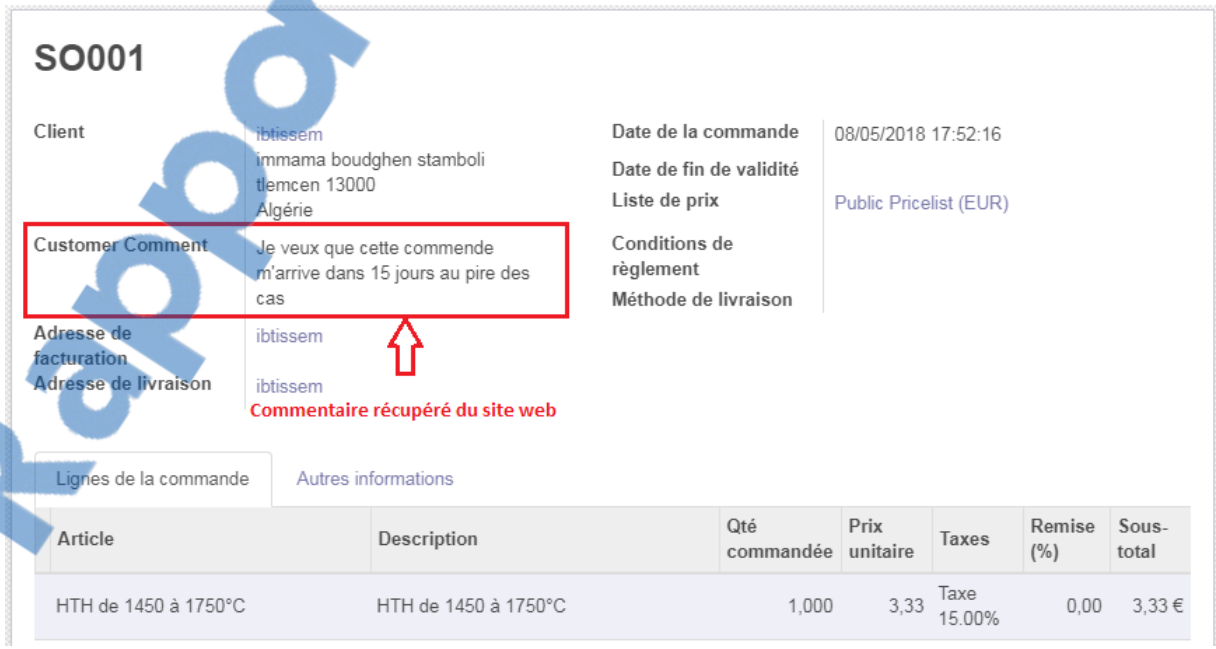


***Figure 65: Le commentaire du client dans le* site web récupéré dans le devis**

Rapport-Gratuit.com

VI.1. Module Website_checkout_comment

Le module qui gère le commentaire du client dans le site web et dans le système est « **website_checkout_comment** », il qui donne la main au client d'ajouter son commentaire concernant la commande avant de la confirmer.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<odoo>
    <data>
        <template id="website_sale_customer_comment" name="Customer Comment" inherit_id="website_sale.checkout">
            <xpath expr="//div[@class='row']/div[12]" position="after">
                <div t-attf-class="form-group col-lg-12">
                    <label class="control-label" for="customer_comment">Comments about your order</label>
                    <textarea type="text" name="customer_comment" class="form-control" t-att-value="
                    checkout.get('customer_comment')" placeholder="No comments about your order."/>
                </div>
            </xpath>
        </template>
    </data>
</odoo>
```

Figure 66: Code XML qui ajoute le champ du commentaire dans le site web

```python
class CheckoutComment(website_sale):

    def checkout_values(self, data=None):
        cr, uid, context, registry = request.cr, request.uid, request.context, request.registry
        sale_order_obj = registry.get('sale.order')
        if data:
            current_order = request.website.sale_get_order(context=context)
            sale_order_obj.write(cr, SUPERUSER_ID, [current_order.id],
                                 {'customer_comment': data.get('customer_comment', None)},
                                 context=context)
        return super(CheckoutComment, self).checkout_values(data)
```

Figure 67: Fonction Python qui récupère le commentaire du client dans le site web et l'ajouter dans la vue devis dans le système

VII. Conclusion

Dans ce chapitre nous avons présenté le système final obtenu, en essayant de satisfaire la majorité des besoins. Plus précisément nous avons expliqué en détail l'analyse faite pour le système et les différentes étapes à suivre pour faire les configurations et intégrer les données en plus des migrations exigées pour passer d'une version à une autre.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Conclusion Générale

Le présent travail a été réalisé dans le cadre du projet de fin d'étude au sein de l'entreprise SOGESI. Le but était de découvrir le E-commerce, mettre en pratique nos connaissances et apporter une solution aux différents problèmes auxquels était confronté notre client, l'entreprise COMO Industries.

La première partie, était consacrée à la présentation des ERP, et un focus plus particulier sur l'ERP Odoo objet de notre étude.

Dans la deuxième partie, nous avons commencé par la présentation de l'entreprise cliente et ces spécifications, puis nous avons entamé la conception en utilisant le langage de modélisation UML. Le dernier volet était l'implémentation de la solution, en intégrant les données, et en utilisant le langage python, xml, javascript et d'autres technologies.

Le travail en entreprise m'a permis de me développer sur le plan humain, professionnel, et technique. J'ai acquis des compétences dans plusieurs domaines, tel que le travail en équipe, la planification des tâches, la gestion d'un projet et la recherche de solutions fiables et conviviales.

Il est important de souligner que j'ai pu couvrir la majorité des fonctionnalités qui n'existait pas dans l4open ERP Odoo, et atteint les objectifs fixés qui était validés par le client qui se sont les suivantes :

- Réalisation de transaction de paiement en ligne avec PayPal et Stripe
- Intégration de données en ajoutant les prix des anneaux de contrôles dans des listes de prix, et en ajoutant les variantes d'articles.
- Génération automatique des références internes des variantes.
- L'ajout d'une alerte lors de l'ajout d'un produit dans le panier.
- L'ajout d'un commentaire dans le site web et le récupérer dans le devis dans l'application dans les ventes.

Cependant, nous pouvons toujours améliorer notre travail en ouvrant la voie vers de nouvelles perspectives. A court terme, j'envisage le déploiement du e-commerce et les solutions trouvées dans des entreprises Algériennes. Je souhaite également contribuer à l'intégration et l'adaptation de l'ERP Odoo dans d'autres secteurs d'activités tels que : l'enseignement, le tourisme, le secteur de santé, etc.

Bibliographie

[1] http://www.cegid.com/fr/faq/quest-quun-erp/ Copyright 2018.

[2] http://nathalie.diaz.pagesperso-orange.fr/html/erpqualite.htm / Consulté le 15 septembre 2017.

[3] https://www.appvizer.fr/magazine/operations/erp/top-5-erp-gratuit-open-source/ Consulté le 01er mars 2017

[4] https://www.celge.fr/editeurs/odoo-logiciel-de-gestion-dentreprise-crm-erp-facturation-comptabilite-gpao-cms-ecommerce/ copyright 2009-2018.

[5] http://www.odoo.gotodoo.com/basics-odoo-architecture/ Consulté le 2 juin 2018

[6] https://techterms.com/definition/python Consulté /Consulté le 7 octobre 2017.

[7] Ludovic Rolond, Structurez vos données avec XML/ Consulté janvier 2014.

[8] https://www.w3schools.com/bootstrap4/bootstrap_get_started.asp/ Copyright 1999-2018

[9] https://www.odoo.com/documentation/10.0/reference/reports.html/ Consulté Décembre 2017.

[10] https://docs.postgresql.fr/7.4/preface.html/ Consulté 2015.

[11] https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203585-javascript/ Mis à jour le 23 avril 2018.

[12] https://www.comomecanique.com

[13] Fien Van Der Heyade, Laurent Debrauwer, UML 2 Initiation: exemples et exercices,corrigés, 2ième édition,Eni,2012,192 pages.

[14] http://www.uml-sysml.org/diagrammes-uml-et-sysml/diagramme-uml/diagramme-de-classe/ Copyright 2000-2018.

[15] https://fr.softonic.com/articles/paypal-questions-reponses/ Consulté le 25 juillet 2014.

[16] https://support.john-steel.com/hc/fr/articles/213809109-Stripe-qu-est-ce-que-c-est-/ Consulté en 2018.

Liste des figures

Figure 1: L'architecture modulaire de Odoo 10
Figure 2: Architecture technique de Odoo 11
Figure 3: Modèle – Vue – Contrôleur 11
Figure 4: L'ajout des produits dans le panier dans la version8 15
Figure 6: La génération automatique des références internes dans la version 8 16
Figure 8: les intermédiaires de paiement à configurer 19
Figure 9: Modification du site web dans la version8 19
Figure 10: Migration de la Même modification dans le site web à la version10 20
Figure 11: Création d'un contact dans Odoo 20
Figure 12: Création d'une opportunité 21
Figure 13: Les opportunités selon leurs étapes 21
Figure 14: Devis de commande créé 22
Figure 15: Le devis à envoyer 22
Figure 16: Création de bon de commande 23
Figure 17: Création de la facture 23
Figure 18: Facture ouverte pour le paiement 24
Figure 19: L'enregistrement d'un paiement 24
Figure 20: passer notre opportunité au stade gagné 24
Figure 21: La fiche d'un article 25
Figure 22: La création d'une demande de prix 25
Figure 23: Expédition d'un article entré en stock 26
Figure 24: Création d'une facture fournisseur 26
Figure 25: Cas d'utilisation d'effectuer une vente 28
Figure 26: Diagramme de séquence de rechercher des articles 29
Figure 27: Diagramme de séquence de la gestion du panier 30
Figure 28: Diagramme de séquence d'effectuer une commande 31
Figure 29: diagramme de classe de gestion de vente 32
Figure 30: Le chemin pour arriver à l'intermédiaire de paiement PayPal dans Odoo 34
Figure 31: Les préférences du site web 34
Figure 32: configuration du revoie automatique et l'url de renvoie 35
Figure 33: Page web de confirmation de la commande 35
Figure 34: Activation d'un compte PayPal facultatif 35

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Figure 35: Modifications des paramètres de notifications instantanées de paiement IPN 36
Figure 36: Sélectionner le codage utilisé sur le site web 36
Figure 37: Récupération d' l'identifiant de compte marchand et l'adresse mail 37
Figure 39: choisir PayPal comme méthode de paiement 38
Figure 41: Récupération des clés API 39
Figure 42: La clé secrète et la clé publiée du compte Stripe 40
Figure 43: configuration de l'intermédiaire de paiement Stripe 40
Figure 44: Payer une commande en utilisant Stripe comme mode de paiement 41
Figure 45: Prix unitaire de l'anneau de contrôle ZTH dans l'intervalle [16 90] 43
Figure 46: Tarif des anneaux selon la quantité vendu 43
Figure 48: L'affichage des variantes d'articles de type Radio 44
Figure 49: configuration de type des caractéristiques 45
Figure 50: L'affichage des variantes d'article sous forme de sélections 45
Figure 52 : L'ajout du postfix dans la vue variante 46
Figure 53: L'ajout du préfix dans la vue article 47
Figure 54: Code python de l'ajout de l'objet préfix 47
Figure 55: Extrait du code XML de l'ajout du préfix dans la vue article 47
Figure 56: Code Python de la génération automatique de la référence interne et le Postfix des variantes 48
Figure 58: Le prix de l'article variant selon le supplémentaire des caractéristiques 49
Figure 59: Extrait du code XML pour masquer le supplémentaire 49
Figure 61: Code JavaScript de l'ajout d'une alerte 50
Figure 63: Extrait du code Python qui gère l'alerte 51
Figure 64: L'ajout du commentaire client sur sa commande par le site web 52
Figure 66: Code XML qui ajoute le champ du commentaire dans le site web 53
Figure 67: Fonction Python qui récupère le commentaire du client dans le site web et l'ajouter dans la vue devis dans le système 53

Liste des tableaux

Tableau 1: Classement des cinq premiers ERP open source. 6

Tableau 2 : L'évolution de l'open ERP Odoo depuis sa création.....7

Tableau 3: Recherche des articles..... 29

Tableau 4: Gérer son panier..... 30

Tableau 5: Effectuer une commande 31

Tableau 6: Tarif des anneaux de contrôle 42

Tableau 7: Tarif des anneaux avec la quantité minimum de chaque intervalle 42

Liste des acronymes

ERP : Entreprise Ressource Planing

UML : Unified Modelling Language

AGPL : Affero GNU General Public Licence

PME : Petite Moyenne Entreprise

TPE : Très Petite Entreprise