

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: Système d'Information et de Connaissances (S.I.C)

Thème

**Proposition d'un module de gestion d'hôtel
et d'un site de réservation en ligne
- sous ODOO-**

Réalisé par :

- **Brahmi Sara**
- **Hammoudi Sihem**

Présenté le 02 Juin 2016 devant le jury composé de MM.

- *Mme Chaouche* (Président)
- *Mme Khitri* (Encadreur)
- *Mr Matallah* (Examineur)
- *Mme Benmansour* (Examineur)
- *Mr Brahmi Noureddine* (Directeur de l'entreprise SOGESI)

Remerciements

*Notre premier remerciement va à Allah Soubhânâho Wa
Tâala.*

*Nous tenons à exprimer nos vifs remerciements et
notre profonde gratitude à Madame Khitri Souad, pour sa
gentillesse, sa disponibilité et sa contribution générale à
l'élaboration de notre travail.*

*Ainsi que l'entreprise Sogesi pour nous avoir accueillis et
guidé durant la période de la réalisation de notre projet.*

*Et puis, un remerciement très chaleureux à nos familles
pour leur soutien aussi moral que financier.*

Ainsi que M. Karâm et D. ramzi pour son assistance.

*Nous tenons également à remercier tous nos enseignants
en particulier Mr. Chouiti et nos amis.*

Dédicace

C'est avec un énorme plaisir, un cœur ouvert et une immense joie, que je dédie ce travail à mes très chers respectueux et magnifique parents et mes grands parents qui m'ont soutenue tout au long de ma vie. Aucune dédicace ne pourrait exprimer mon respect, ma considération et mes profonds sentiments envers eux.

Ainsi à mes Frères Youcef et Mohammed et en particulier mon binôme Brähmi Sarà.

À mes cousins Nadià, Samy, M.Sarah, M.Leila en leur souhaitent du plus profond de mon cœur, beaucoup de joie, de réussite, de prospérité et surtout une santé sans défaillance durant toute leur vie.

Enfin j'exprime les mêmes sentiments à mes respectueux professeurs qui m'ont appris et transmis leur savoir. Je reste reconnaissante durant toute ma vie

Et à mes chers amis Meriem, Imene, Nesrine, Samira, Fatima, Houssein, Amine.

Sihem

Dédicace

Avec un énorme plaisir, un cœur ouvert et une immense joie, que je dédie ce travail à mes très chers respectueux et magnifique parents qui m'ont soutenus tout au long de ma vie ainsi à mes sœurs adorables Hanane et Samira et en particulier mon binôme Hammoudi Sihem.

Ainsi, B.Hakim pour son soutien moral et sa gentillesse sans égale qui m'ont aidé à surmonter les obstacles que j'ai affrontés.

A mes amis Fatima, Samira, Imene, Nesrine, Amine, Morad, Ilyes, Adil, Khelifa, Housseem.

Sara

Introduction générale	4
Chapitre I : Les progiciels de gestion intégrés	
I.1-Introduction	6
I.2- Historique des ERP.....	6
I.3- Définition d'un ERP (Entreprise ressource planning).....	8
I.4- Principe de base d'ERP	8
I.5- Caractéristiques des ERP.....	8
I.6- Le périmètre de gestion qui couvre un ERP.....	9
I.7- Différents environnements de travail d'un ERP	9
I.8- Architecture technique.....	10
I.9- Architecture modulaire	10
I.10- Principaux éditeurs d'ERP	11
I.10.1- Les ERP propriétaires.....	11
I.10.2 - Les ERP Open Source	12
I.11- Points forts d'ODOO.....	13
I.12- Architecture technique d'ODOO.....	13
I.12.1- Architecture client/serveur	13
I.12 .2- Architecture modulaire ODOO	14
I.13- Modèle vue contrôleur (MVC).....	16
I.14- Conclusion	17
Chapitre II : La conception de notre application	
II.1- Introduction	19
II.2- Etat de l'existant.....	19
II.2.1- Besoin de l'entreprise SOGESI	19
II.2.2- Analyse de l'existant.....	19
II.2.3- Adaptation	20
II.3- Processus de fonctionnement de notre application	20
II.4- Outil de conception UML	22

II.4.1- Diagramme des cas d'utilisation.....	22
II.4.2- Diagramme de séquence	23
II.4.2.1- Diagramme de séquence pour la gestion de client	24
II.4.2.2- Diagramme de séquence pour la gestion des chambres :	25
II.4.2.3- Diagramme de séquence pour la gestion de réservation	26
II.4.2.4- Diagramme de séquence de réservation en ligne.....	27
II.4.3- Diagramme d'activité.....	28
II.4.4- Diagramme de classe	28
II.8- Conclusion.....	30

Chapitre III : La réalisation de notre application

III.1- Introduction.....	32
III.2- Environnement de travail.....	32
III.3- Etapes de développement de notre application Dzhotel.....	33
III.3.1- installation d'Odoo 8.0 sous windows.....	33
III.3.2- Mise en place de l'application DzHotel :	36
III.4- Description de notre module.....	39
III.4.1- Gestion des clients	39
III.4.2- Gestion des chambres	40
III.4.3- Gestions des réservations.....	41
III.4.4- Gestions des rapports.....	43
III.5- Développement du site web de la réservation en ligne	44
II.5.1- les outils utilisés :.....	44
III.5.2- L'installation du module website builder :.....	44
III.5.3- Présentation du site.....	45
III.5.3.1- Page d'accueil « Notre hôtel »	45
III.5.3.2- Page Chambres	46
III.5.3.3- Page Demande de réservation	47
III.5.3.4- Page Contacter nous	48

Table des matières

III.6- Conclusion	48
Conclusion générale	50
Références bibliographiques	51
La liste des figures	53

Les organisations évoluent dans un environnement de plus en plus complexe et changeant. Le principal obstacle auquel se heurtent les organisations est la difficulté d'obtenir des données et informations précises et des interfaces convenables entre les différentes fonctions de l'organisation. Le problème vient du fait que l'organisation est basée généralement selon un modèle fonctionnel caractérisé par la multiplication des sous-systèmes d'information. Pour faire face à ce problème l'organisation tente de modifier son fonctionnement par l'intégration des principaux processus de l'organisation elle-même. Et la mise en place d'un système d'information cohérent garantissant l'unicité de l'information et l'accès à celle-ci à partir de toutes les fonctions de l'organisation. Les progiciels ERP (Entreprise Ressources Planning) viennent supporter de telles orientations organisationnelles et stratégiques.

La problématique de notre travail nous a été proposée en collaboration avec l'entreprise SOGESI spécialisée dans le développement et l'intégration de solutions informatiques notamment l'ERP Oddo (ex. OpenERP). En effet, notre Projet de Fin d'Etudes est autour de l'ERP ODDO V8. Ce dernier propose une approche modulaire qui facilite l'intégration de nouvelles fonctionnalités. Ainsi, nous proposons d'utiliser cet ERP pour fournir une solution de gestion de réservation d'hôtel (en présentiel et en ligne).

Ce mémoire est constitué de trois chapitres. Le premier décrit les ERPS et principalement open source Oddo, le second concerne l'analyse et la conception de notre projet d'hôtel en utilisant le formalisme UML, le dernier concerne le développement, ainsi que les résultats obtenus. Une conclusion générale et perspective est présentée à la fin du mémoire.

Chapitre I

I.1-Introduction

Les entreprises d'aujourd'hui doivent rationaliser leur pilotage vu l'évolution progressive et la compétitivité féroce du marché. Pour ce faire, l'utilisation d'un progiciel de gestion intégrée (PGI) ou **Enterprise Resource Planning (ERP)** est une nécessité car il permet une gestion homogène et cohérente du système d'information (SI) de l'entreprise. Dans ce chapitre nous présentons les ERP ce qui nous conduira à la description de leur architecture.

I.2- Historique des ERP

Dans les années 70, l'informatique a servi à automatiser des procédures. Chaque service avait ainsi son propre Système d'information. Les applications étaient développées indépendamment les unes des autres. Ce qui au final s'avérera problématique pour les entreprises. Ainsi les données sont saisies à plusieurs reprises des systèmes d'information distincts, les conséquences sont [1] :

- erreurs de saisie.
- redondance des données, stockages multiples des mêmes données.
- données incohérentes entre les systèmes.
- données manquantes.
- données non à jour.

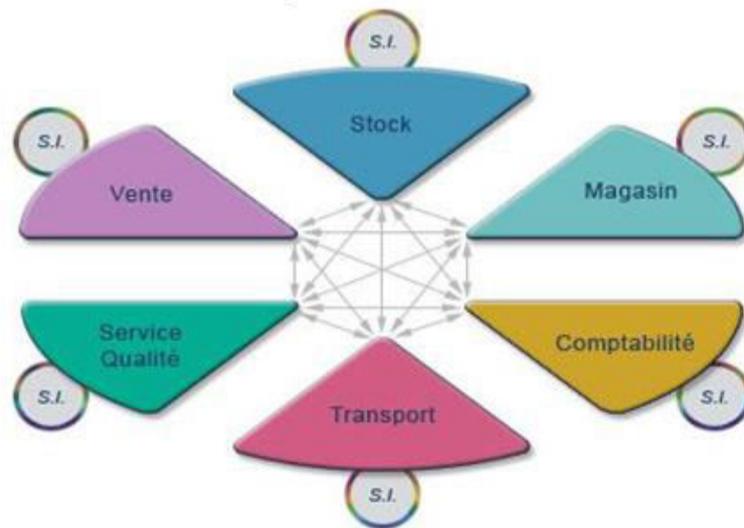


Figure I.1 : Organisation traditionnelle [1]

Pour palier à ces problèmes les responsables SI ont fait développer des interfaces informatiques spécifiques pour faire communiquer les différents systèmes internes entre eux. Mais malgré certains problèmes résolus de nouveaux problèmes apparaissent [1] :

- les mises à jour sont délicate à fournir car elles impactent tout les systèmes interfacé.
- Les coûts et délais de développements répétés à chaque changement d'un des systèmes
- Les dysfonctionnements des interfaces sont fréquent et nécessite des corrections manuelles.

Dans les années 80 l'ERP devient un standard dans les entreprises. Elle adopte ainsi une solution uniformisée pour tous les services ainsi que la centralisation des données dans une base unique. Ce qui permet une saisie unique et une meilleure cohérence des données. La redondance de données entre les différents systèmes d'information est ainsi supprimée. L'environnement de travail est standardisé à tous les niveaux de l'entreprise [1].

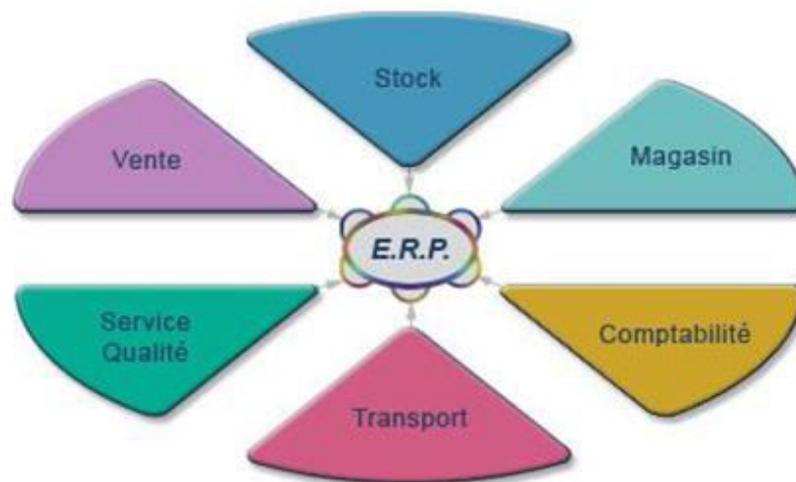


Figure I.2: Organisation avec un ERP [1]

Enfin les ERP sont créés pour être modulaire et flexible afin de répondre aux besoins spécifiques d'une entreprise et à ses besoins futurs.

I.3- Définition d'un ERP (Entreprise ressource planning)

Enterprise Resource Planning (ERP ou Progiciel de Gestion intégré) [2] est un logiciel qui permet de gérer l'ensemble des processus d'une entreprise, en intégrant l'ensemble des fonctions comme la gestion des ressources humaines, la gestion comptable et financière, la relation client, les achats, la gestion des stocks

Un progiciel **ERP** induit généralement une réponse rapide aux besoins de l'entreprise. L'ERP est basé sur la construction des applications informatiques de l'entreprise sous forme de **modules indépendants**. Ces modules partagent une **base de données commune**, permettant la communication de données entre les applications.

I.4- Principe de base d'ERP

Le principe qui caractérise un ERP [2] est l'usage de ce qu'on appelle un moteur de **work flow**. C'est un système permettant d'automatiser un flux d'information au sein d'une organisation, il permet lorsqu'une donnée est entrée dans le système d'information, de la propager dans tous les modules qui en ont l'utilité, selon une programmation prédéfinie.

I.5- Caractéristiques des ERP

- Permet d'éviter la redondance d'information entre différents SI de l'entreprise) [2].
- Une modification sur un module provoque une mise à jour en temps réel des autres modules liés [2].
- Un ERP est un **outil multilingue et multidevise**, il est donc adapté au marché mondial, en particulier aux multinationales [2].
- Un ERP facilite l'audit en cas de dysfonctionnement, permettant d'identifier facilement le ou les modules concernés [2].
- Un ERP peut suffire à couvrir la totalité des besoins de l'entreprise en termes de système d'information (la nature modulaire de l'ERP permet également de l'implémenter progressivement, module par module, selon les besoins) [2].

- Cohérence et homogénéité des informations [2].
- Une meilleure maîtrise des stocks [2].
- Une meilleure coordination des services [2].
- Intégrité et unicité du système d'information [2].
- Communication interne et externe facilitée par le partage d'un même système d'information [2].

I.6- Le périmètre de gestion qui couvre un ERP

La vocation d'un ERP est d'homogénéiser le Système d'Information de l'entreprise avec un outil unique qui est capable de couvrir un large périmètre de gestion, c'est-à-dire [3] :

- La gestion des **achats**
- La gestion des **ventes**
- La **gestion comptable** : comptabilité client, fournisseur...
- Le **contrôle de gestion**
- La **gestion de production** (planification, ...)
- La **gestion des stocks** (logistique)

Un ERP est subdivisé en modules qui répondent chacun à un des domaines de gestion listés ci-dessus. On dit aussi que l'ERP est constitué de modules fonctionnels, chacun couvrant un périmètre de gestion de l'entreprise [3].

I.7- Différents environnements de travail d'un ERP

Un ERP contient généralement trois environnements de travail [3] :

- Un « **environnement de développement** » qui permet d'adapter le progiciel standard à des besoins spécifiques de l'entreprise.
- Un « **environnement de test** » dit encore environnement de recette qui permet de réaliser des simulations. Ces simulations permettent de tester de nouveaux

paramétrages et de vérifier le fonctionnement correct du progiciel par rapport à un processus de gestion donné (une vente, un achat, une sortie de stock, ...)

- Un « **environnement de production** » qui correspond au progiciel utilisé par les gestionnaires de l'entreprise au quotidien.

Le travail en environnement de test est préalable au passage à l'environnement de production

I.8- Architecture technique

La figure 3 ci-dessous nous démontre l'architecture technique d'un ERP : L'ERP est donc sur serveur. La majorité des ERP sont couplés à une base de données ORACLE. De plus, les ERP sont compatibles packs Office, en particulier pour PowerPoint et Excel. En effet, le premier étant utile pour personnaliser les bureaux ERP en fonction de l'entreprise et le second pour effectuer les imports/exports de données. Enfin, les ERP sont aussi compatibles avec des outils de reporting. Le reporting étant utilisé en particulier pour le module de gestion relation client [4].

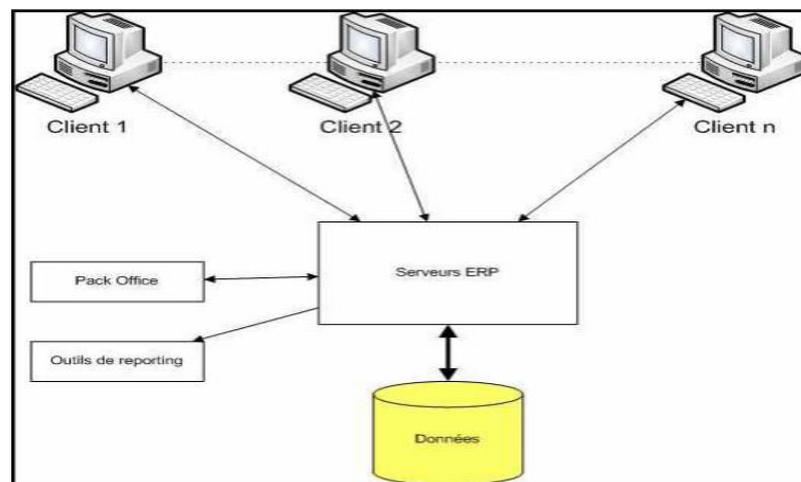


Figure I.3 : Architecture technique des ERP [4]

I.9- Architecture modulaire

Un ERP est un ensemble dont toutes les parties fonctionnent les unes avec les autres d'où l'ergonomie et l'unicité des informations et donc la cohérence du SI.

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

ERP permettent d'être sûr de la compatibilité des modules entre eux, ils fonctionnent ensemble [4].

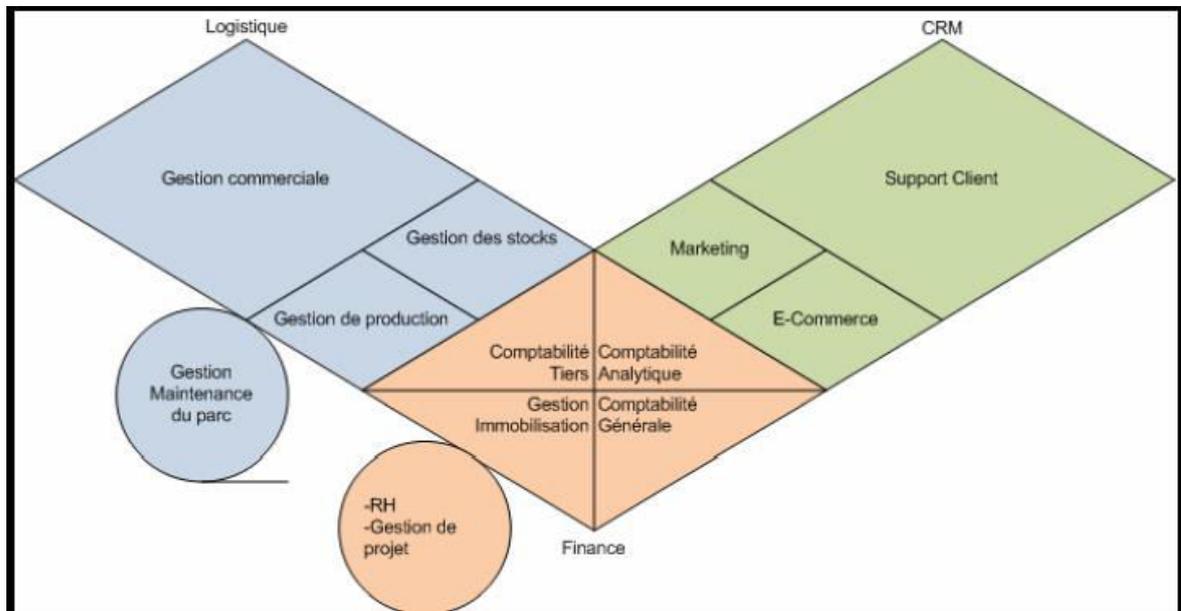


Figure I.4 : Architecture modulaire des ERP [4]

I.10- Principaux éditeurs d'ERP

On distingue deux sortes d'ERP : les ERP propriétaires et les ERP Open Source [4].

I.10.1- Les ERP propriétaires

Aujourd'hui, il existe de nombreux ERP propriétaires mais seuls quelques éditeurs internationaux restent vraiment dans la course.

Les trois premiers sont [4] :

SAP (Business One) : SAP est le leader mondial des ERP, est une application client/serveur. Ses modules couvrent l'ensemble des fonctions de gestion de l'entreprise et chaque module couvre des besoins complets de gestion .

Oracle (JD Edwards) : est un progiciel de gestion intégré. Anciennement appelé People EnterpriseOne et vendu par J.D. Edwards puis par Peoplesoft. J.D. Edwards a été racheté par PeopleSoft puis par Oracle. Le produit est depuis renommé "Oracle

JDEdwards EnterpriseOne". Il est composé de plusieurs modules plus ou moins indépendants .

Microsoft Dynamics ERP(NAVISION) : est un progiciel de gestion intégrée (ERP/PGI), conçu pour les structures de 20 à 500 employés, sociétés autonomes et filiales de groupes, des secteurs de l'industrie, du négoce et des services.

I.10.2 - Les ERP Open Source

Les principaux progiciels Open Source sont:

Aria: il couvre tous les domaines fonctionnels internes nécessaires pour gérer l'entreprise. Il a été développé à base d'un ERP open source appelé Nola, et son environnement est PHP MYSQL [4] .

Compiere: est un progiciel de gestion intégré (PGI) et gestion de la relation client (GRC) à source ouvert : open source pour les petites et moyennes entreprises (PME) dans la distribution et le service. L'application est fournie sous double licence GPL et propriétaire. Les sources peuvent être adaptées aux besoins du client. Le support technique et la documentation sont payants. Son origine est Jorg Jank [4].

ERP5: est un progiciel de gestion intégré (ERP) libre, son origine est Nexedi, grâce à son intégration avec Nuxeo CPS, c'est aussi un système de gestion de contenu. son environnement sont le Python et le Zope [4].

Fisterra: est un PGI sous licence PGL, le premier client pour cet ERP été la société espagnole Auto Arte, son origine est Igalia, et son environnement est GNOME2 developement platform Psotgresql [4].

OpenERP: anciennement TinyERP, est un progiciel libre de gestion intégré comprenant des modules de gestion des ventes, des relations clients, des projets, des entrepôts, de la production, de la comptabilité et des ressources humaines. Son environnement est le Postgresgl, XML [4].

Odoo [5] : (anciennement OpenERP) est un progiciel de gestion d'entreprise (ERP) destiné à intégrer l'ensemble des données opérationnelles et de gestion de l'entreprise

dans une base de données unique, accessible par une interface web. Cette base de données centrale est associée à une couche fonctionnelle très innovante qui met en relation des informations d'origines diverses et assure un déroulement efficace des processus transversaux de création de valeur ajoutée de l'entreprise

- ❖ Pour la réalisation de notre projet nous avons choisit le dernier ERP ODOO V8 parce qu'il gère toutes les grandes fonctions de l'entreprise. Il propose des applications spécialisées qui facilitent la gestion et améliorent la productivité.

I.11- Points forts d'ODOO

Odoo présente plusieurs avantages qui sont [6]:

Gestion et création de site web : Odoo V8 est à ce jour le seul ERP qui propose un CMS intégré. Cette solution est particulièrement efficace pour les projets d'e-boutique, de portail clients, de site d'annonces spécialisées.

Gestion Commerciale : Odoo V8 couvre toutes les fonctions de la gestion commerciale (CRM – Devis – Commande – Facturation – Comptabilité) et va plus loin avec la dématérialisation des devis et factures. Fini les envois de PDF, avec Odoo il suffit d'envoyer un lien et l'interlocuteur fera le reste .

Marketing : Gérez toutes les campagnes de prospection et de suivi de portefeuille de sondage clients avec Odoo V8. Allez plus loin avec les modules spécialisés pour la gestion des campagnes Google Adwords de vos sites web.

Ressources Humaines : Odoo V8 assure la gestion du recrutement via le site web jusqu'à la gestion administrative des employés. Il permet à chaque collaborateur de s'abonner à un événement et d'être notifié. Le chat et le réseau social d'entreprise sont autant de fonctionnalités qui facilitent le travail quotidien des utilisateurs.

I.12- Architecture technique d'ODOO

I.12.1- Architecture client/serveur

Open ERP repose sur une architecture client/serveur [7]. Le serveur et le client communiquent via le protocole XML-RPC. C'est un simple protocole qui permet au

client de faire des appels aux Procédures. Une fois la fonction est appelée, ses arguments et ses résultats sont envoyés par le Protocole http, eux-mêmes sont encodés par le langage XML

ODOO est couplé à une base de données PostgreSQL. De plus, il est compatible au pack Open Office, et aussi avec des outils de reporting pour produire des rapports en PDF ou en HTML.

La logique d'openERP est entièrement du côté serveur. La tâche du client se résume à demander les données (formulaire ou listes) au serveur et de les renvoyer. Avec cette approche, presque tout le développement est fait du côté serveur. Ce qui rend OPENERP plus simple au développement et à la maintenance

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

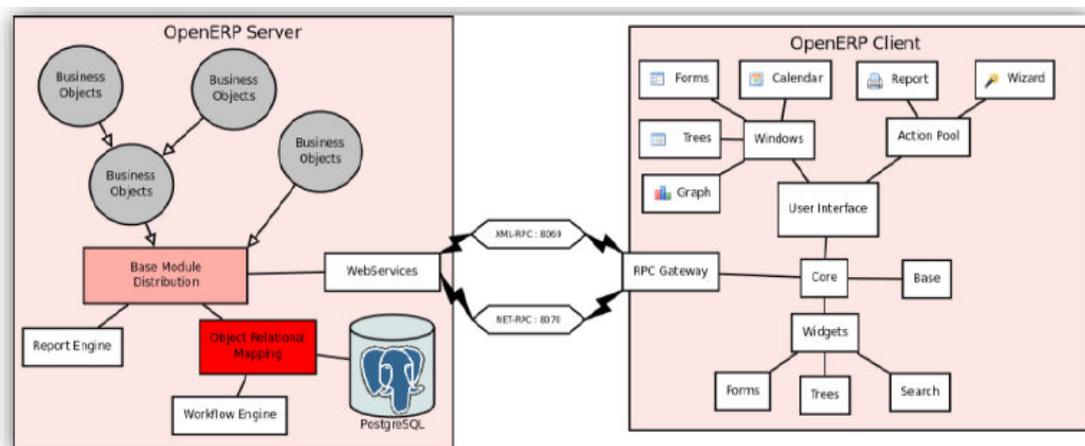


Figure I.5: Architecture client/serveur [7]

I.12 .2- Architecture modulaire ODOO

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 [7].

Tous les modules d'ODOO sont créés au-dessous du répertoire **odoo/server/openerp/addons**

4 fichiers sont obligatoires :

__init__.py

Permet d'importer les fichiers python qui doivent être chargé en mémoire. Le contenu c'est :

Import module

__openerp__.py

Le fichier qui permet de décrire le module. Le contenu c'est :

name: le nom du module

version: la version du module

category: la catégorie dans laquelle est classé un module

author: l'auteur du module

description: la description complète du module

depends: les dépendances du module

data: les fichier XML a charger

installable: si votre module est installable ou non

module.py

Ce fichier va contenir les classes (objets Odoo) permettant de persister les données dans la base de données via la couche ORM de Odoo

data.xml

Ce fichier contient la description de l'interface du module :

Les menus: la déclaration des éléments de l'interface

Les actions: Une action est un événement déclenché suite à un click. Pour déclarer une action, il faut utiliser le modèle "ir.actions.act_window"

Les vues (views): Les vues sont un moyen pour représenter les objets sur le navigateur du client. Plusieurs vues peuvent être déclarées pour le même objet et peuvent être utilisées suivant leur priorités. On peut distinguer plusieurs types de vues :

- Formulaire (form views) : Pour afficher chaque élément dans un formulaire et ainsi le modifier ou le supprimer
- Arbre (tree views et list views) : Permet de lister les différents éléments
- Kanban :représente les données sous forme de colonnes
- Calendar : Permet d'afficher le(s) calendrier (s)

Après la création de ces fichiers il faut :

- ❖ Redémarrer le service
- ❖ Mettre a jour la listes des modules **configuration/ mettre a jour la listes des modules**

Et puis installer le module.

I.13- Modèle vue contrôleur (MVC)

MVC [4] est une architecture de modèles utilisée en génie logiciel. Dans des applications complexes qui présentent des lots de données aux utilisateurs, nous souhaitons 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 ».

Dans open ERP, nous pouvons appliquer cette sémantique de Model View Controller avec :

- **Model** : les modèles sont les objets déclarés. Ils sont également des tables PostgreSQL.

- **View** : les vues sont définies en fichiers XML.
- **Controller** : le contrôleur est les classes Python qui contrôlent ODOO.

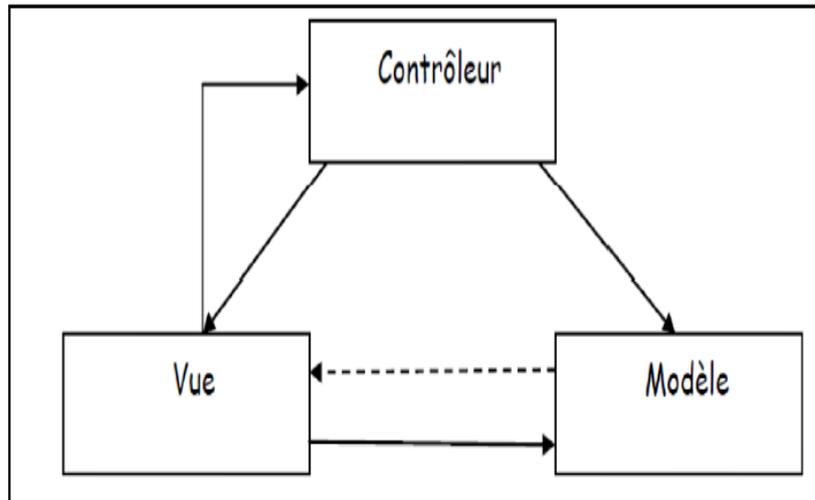


Figure I.6 : Modèle vue contrôleur [4]

I.14- Conclusion

Nous avons présenté dans ce chapitre l'ERP en tant qu'une technologie d'intégration fonctionnelle qui permet à l'entreprise d'instaurer une gestion intégrée notamment, ODOO. Le chapitre suivant convient de construire la conception de notre travail.

Chapitre II

II.1- Introduction

La conception d'un projet est une phase très importante pour définir les objectifs et les fonctionnalités de notre application. Dans ce chapitre nous détaillons la conception après un état de l'existant.

II.2- Etat de l'existant

II.2.1- Besoin de l'entreprise SOGESI

Le besoin initial de l'entreprise SOGESI c'était la création d'un module qui gère la gestion des hôtels :

Le module doit être capable de gérer :

- ❖ Les réservations des clients sur site via le web.
- ❖ Les chambres et leurs disponibilités.
- ❖ Les options offertes par l'hôtel.
- ❖ Le personnel.

Nous avons commencé à schématiser notre travail pour pouvoir faire une conception initial de notre projet. Cependant l'entreprise nous a proposé d'améliorer des modules de gestion d'hôtel et de restauration déjà existants (sur **OpenErp7**) au lieu de créer un nouveau module.

Les modules déjà existant sur la version 7 d'OpenERP s'avèrent insuffisants aux besoins d'une gestion performante d'hôtel. Il fallait ajouter des nouvelles fonctionnalités proposé par l'entreprise puis faire une adaptation pour les installer sur ODOO version 8 pour réaliser la réservation en ligne via un site web ce qui n'est pas accessible sur la version 7.

II.2.2- Analyse de l'existant

Pour mieux comprendre et maîtriser le fonctionnement des codes sources existants il nous a fallu une étude approfondie de trois semaines. Nous avons réussi, par la suite, à réaliser quelques modifications et nous avons intégré les fonctionnalités suivantes :

- ❖ Un planning pour visualiser rapidement la situation des réservations.
- ❖ Un lien avec le module restauration au niveau du module Hôtel. C'est-à-dire qu'on devra être capable de créer une réservation resto à partir du

module hôtel et que cette réservation (une fois validée) sera visible dans la liste des réservations dans le module restaurant. Comme ça, l'administrateur ne sera pas obligé à chaque fois de quitter le module hôtel pour aller sur le module Resto.

- ❖ La liste des réservations des tables sous forme de planning (Vue calendrier dans Openerp).

Nous avons eu des difficultés pour terminer les modifications et nous avons pensé de faire une adaptation de ces modules sur ODOO pour réaliser le site web.

II.2.3- Adaptation

Concernant la réalisation de l'adaptation nous avons rencontré des difficultés durant l'installation. Nous étions contraintes d'installer les modules sans les fichiers data.xml pour une mise en marche. En plus la majorité des fonctionnalités du module ne rependaient pas.

Cette adaptation demande beaucoup de temps pour la réaliser Comme la durée restante pour le projet de fin d'étude est limité par le temps nous avons opté pour la création d'un nouveau module de gestion d'hôtel qui reprend au besoin initial demandé par l'entreprise.

II.3- Processus de fonctionnement de notre application

La figure ci-dessous représente les différentes étapes de fonctionnement de notre module :

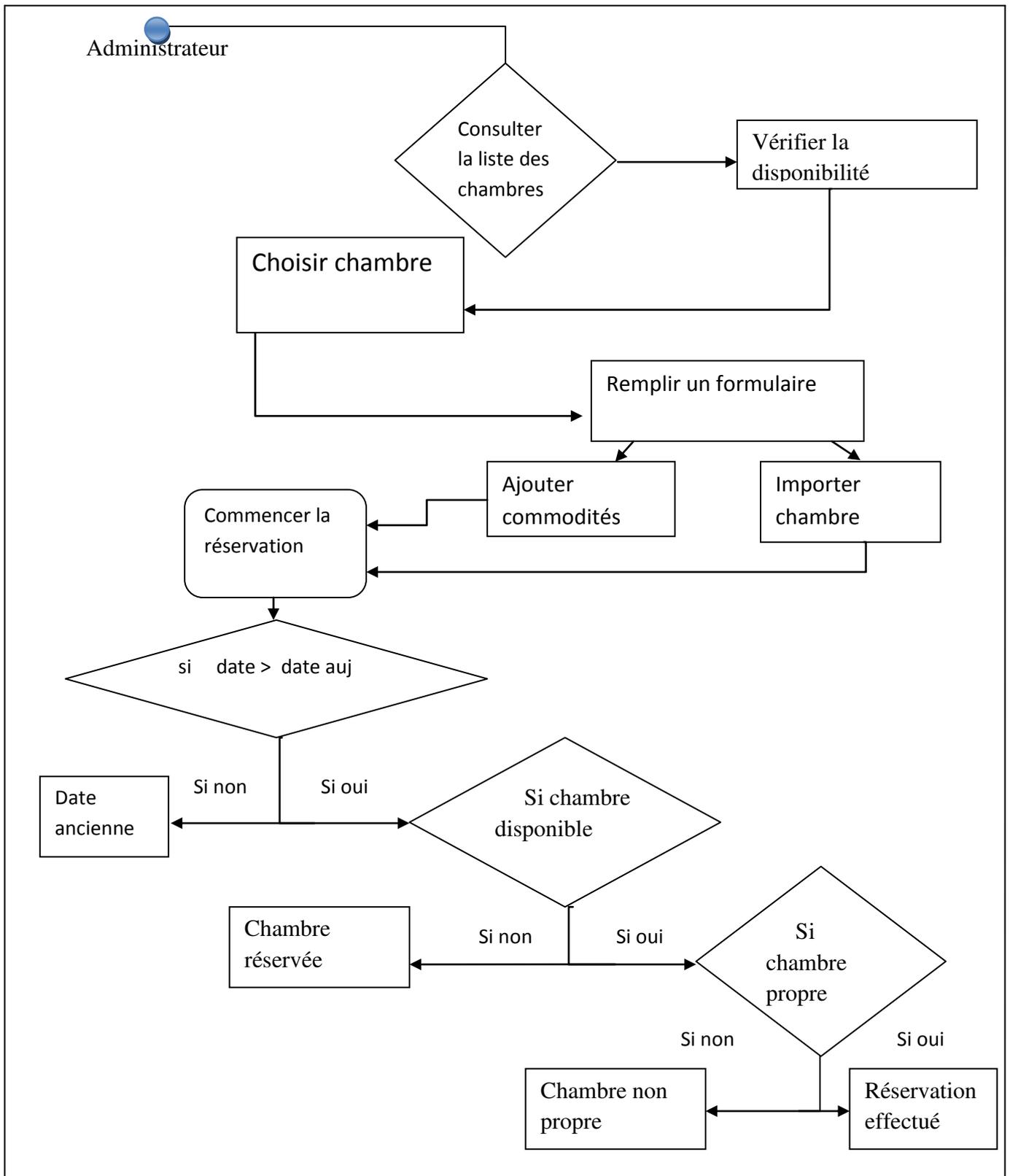


Figure II.1 : processus de fonctionnement de notre application

II.4- Outil de conception UML

UML [8] est une notation permettant de modéliser un problème de façon standard. Ce langage est né de la fusion de plusieurs méthodes existantes auparavant, et est devenu désormais la référence en termes de modélisation objet.

Nous avons choisi UML comme langage de conception vu sa simplicité et sa puissance à l'abstraction des objets réels. Parmi les facteurs de puissance, UML présente plusieurs diagrammes statiques et dynamiques facilitant la tâche de conception et offrant une vue globale sur la composition et le fonctionnement de notre logiciel.

II.4.1- Diagramme des cas d'utilisation

Le diagramme de cas d'utilisation représente la structure des fonctionnalités nécessaires aux utilisateurs du système [9].

A partir de l'étude préliminaire, nous avons pu dégager le diagramme des cas d'utilisation

Présenté dans la figure ci-dessous qui gère :

La gestion des clients : ce cas d'utilisation permet aux administrateurs de créer et mettre à jour la liste des clients.

La gestion des réservations : permet de gérer les réservations. L'administrateur peut créer, modifier et supprimer une réservation

La gestion des chambres : permet de gérer les listes des chambres, les types, les prix et les commodités.

La gestion des rapports : l'administrateur peut imprimer :

- Un rapport de réservation qui indique le nom du client, la date d'arrivée, la date départ, ainsi le numéro de la chambre et le montant à payer.

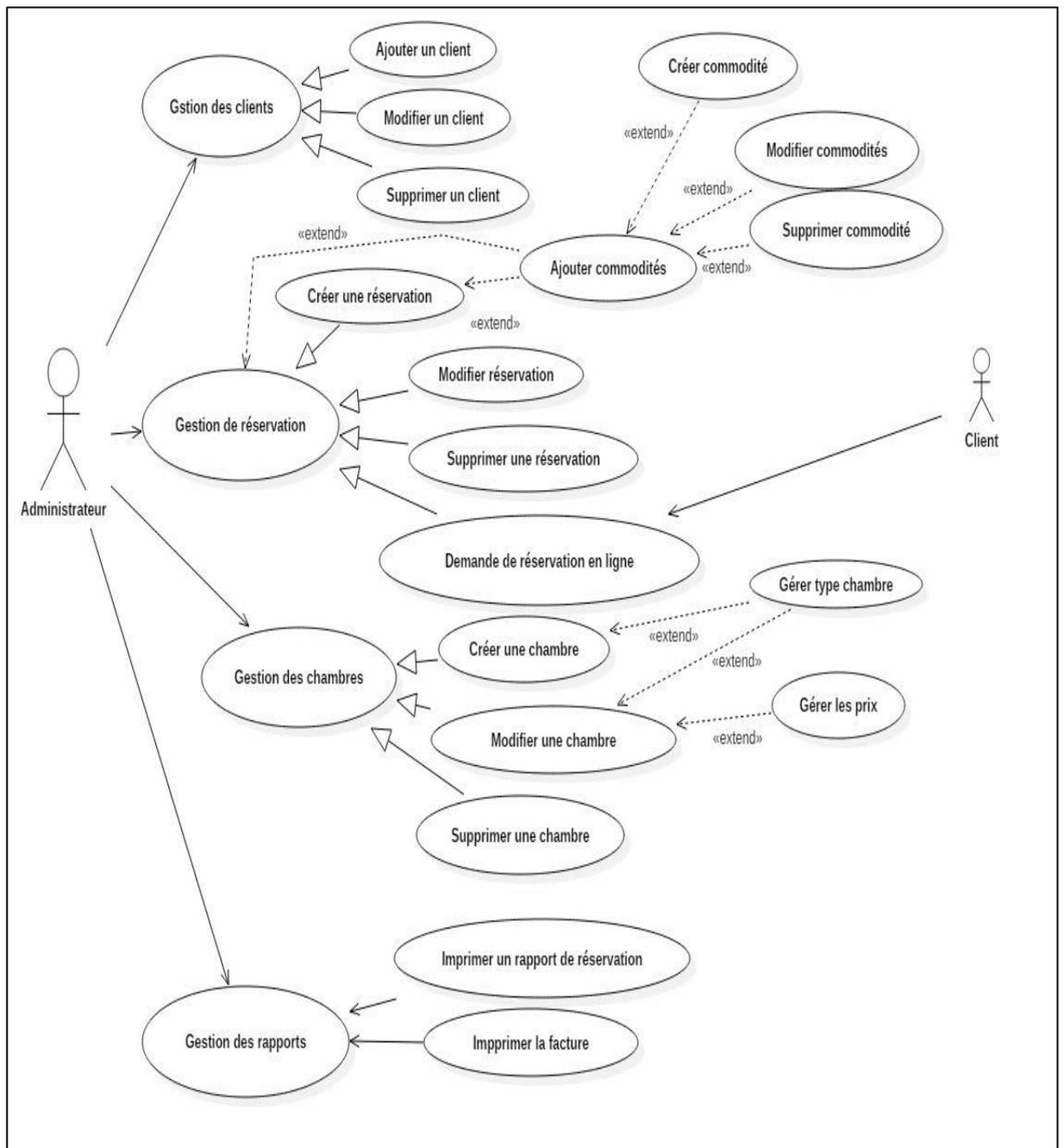
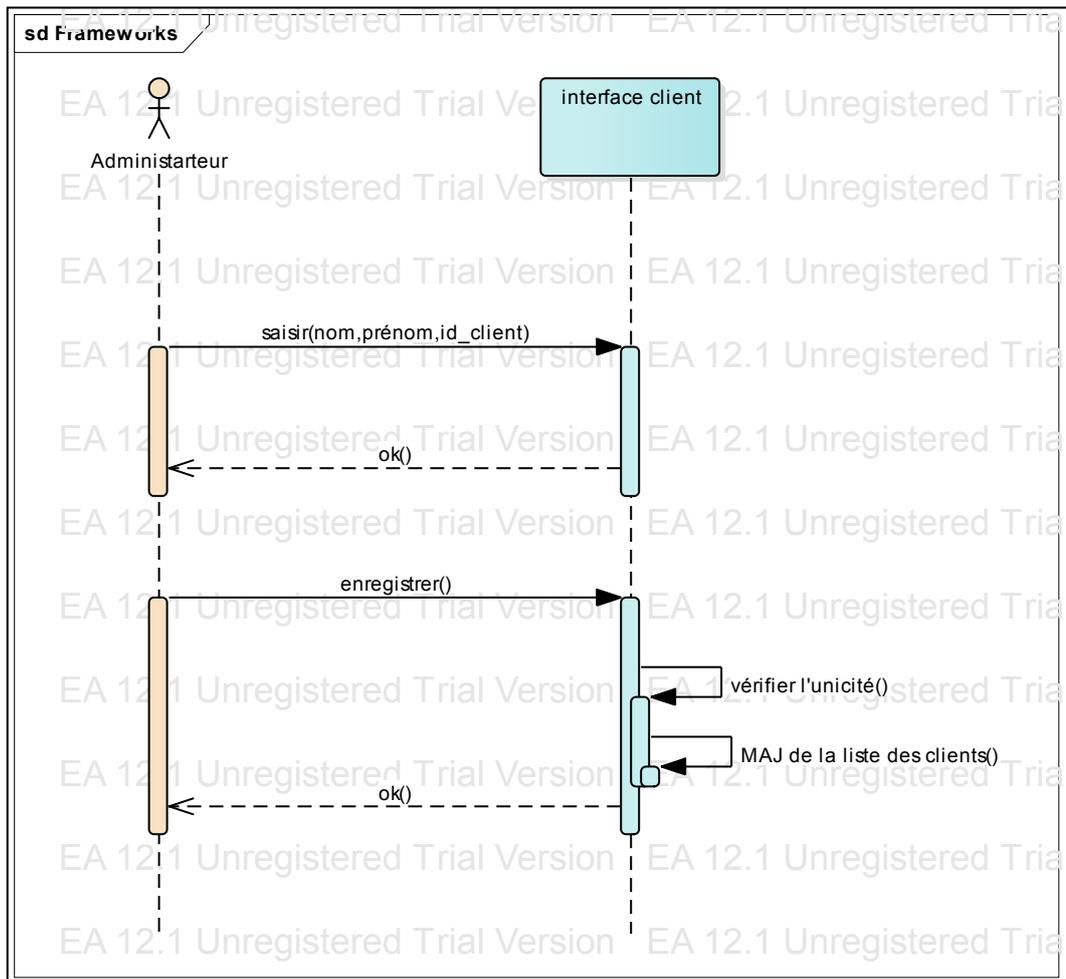


Figure II.2 : Diagramme de cas d'utilisation globale

II.4.2- Diagramme de séquence

Un diagramme de séquence est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont. Les diagrammes de séquence sont organisés en fonction du temps [10]. Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction du moment où ils prennent part dans la séquence de messages.

II.4.2.1- Diagramme de séquence pour la gestion de client**Figure II.3** : diagramme séquence gestion client

- L'administrateur gère les clients, il peut créer des clients en remplissant un formulaire contenant toute les informations nécessaire demandées par l'hôtel.
- Le client est défini par un numéro d'identité unique.

II.4.2.2- Diagramme de séquence pour la gestion des chambres :

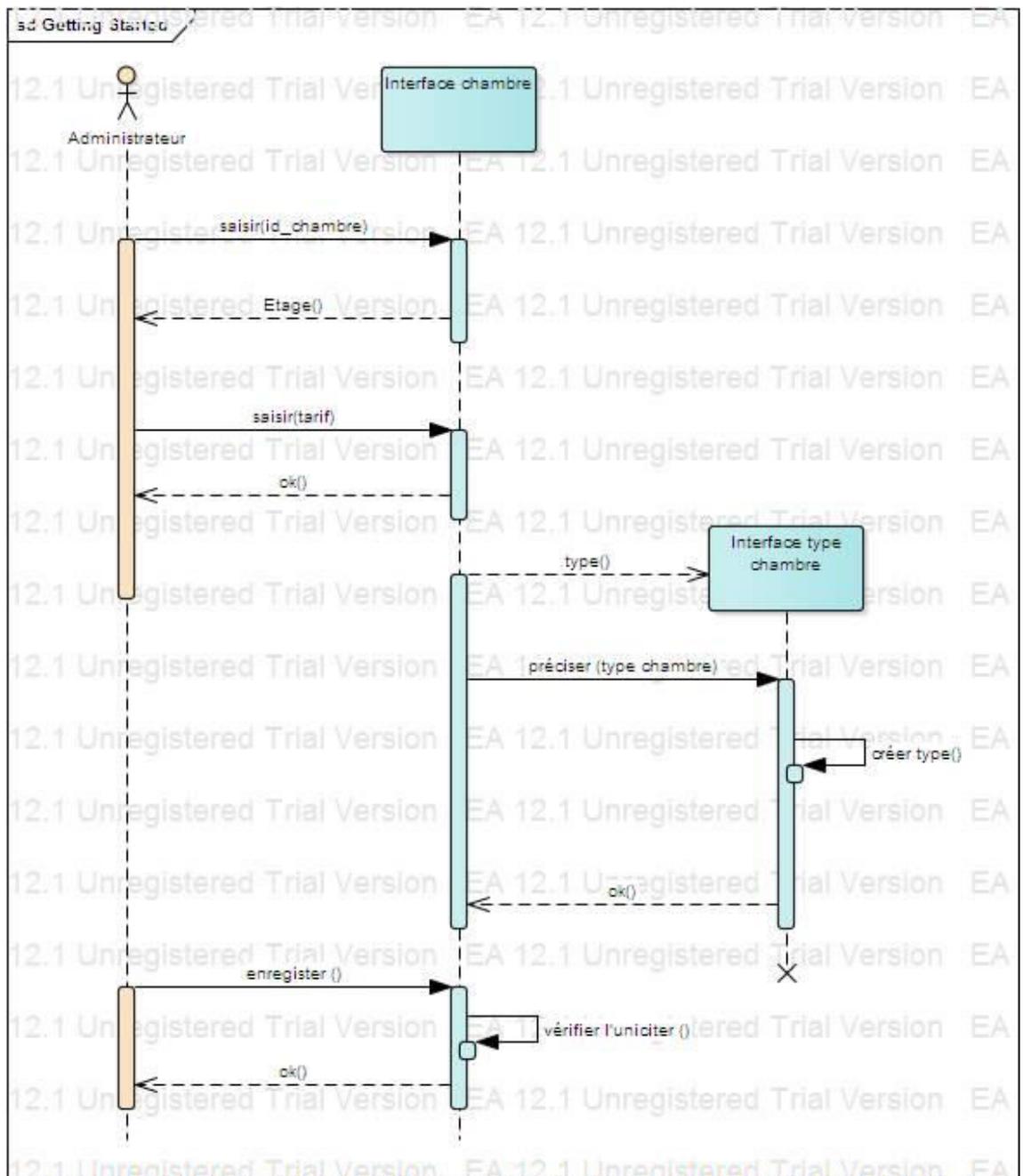


Figure II.4 : diagramme de séquence gestion chambre

Lors de la création d'une chambre l'administrateur gère :

- Les types des chambres : il peut créer des différents types pour les affecter à une chambre.
- Les prix : chaque chambre a son prix.
- La chambre est définie par un numéro unique.

II.4.2.3- Diagramme de séquence pour la gestion de réservation

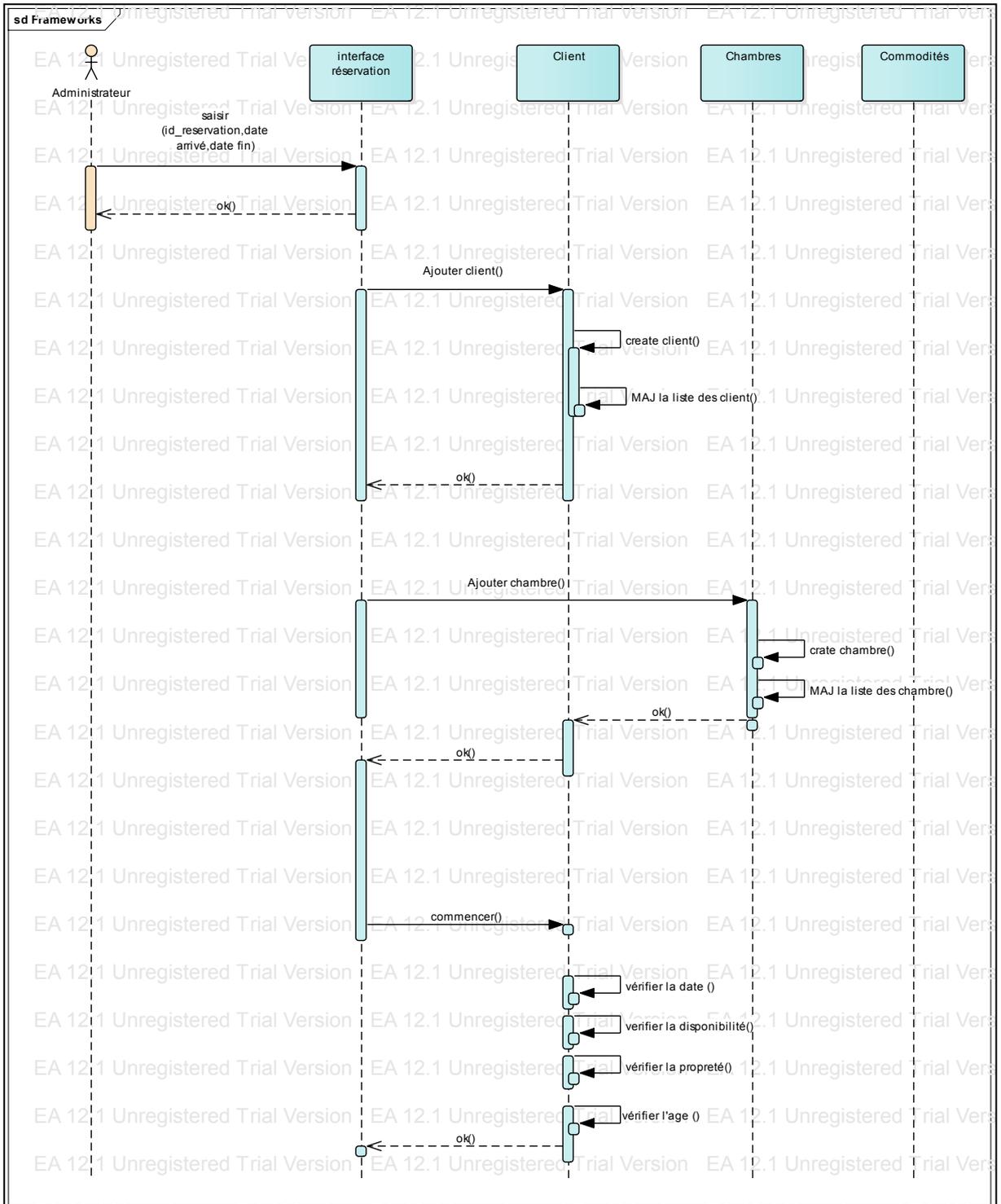


Figure II.5 : diagramme séquence de réservation

Le client demande une réservation, l'administrateur crée la réservation et remplit un formulaire en ajoutant :

- le client (créé le client s'il n'existe pas)
- une chambre
- les clients attachés (optionnel)
- Les commodités (optionnelles).

II.4.2.4- Diagramme de séquence de réservation en ligne

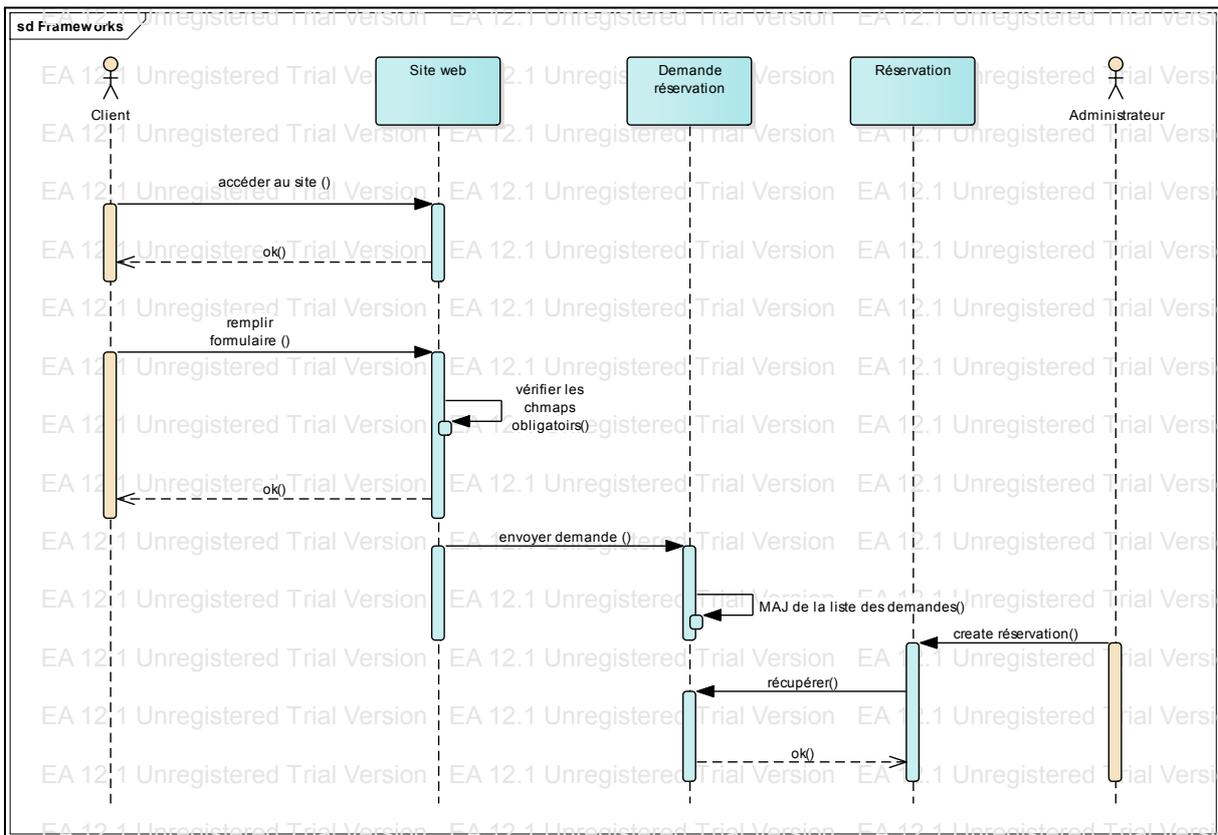


Figure II.6 : diagramme de séquence réservation en ligne

Une interface web est disponible pour le client. Il peut envoyer une demande de réservation en remplissant un formulaire qui contient toutes les informations nécessaires tel que : nom, prénom, date de naissance, date d'arrivée, date fin et le type du chambre.

II.4.3- Diagramme d'activité

Le diagramme d'activité [11] est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallélisables (multi-threads ou multi-processus).

Le diagramme d'activité est également utilisé pour décrire un flux de travail (workflow)

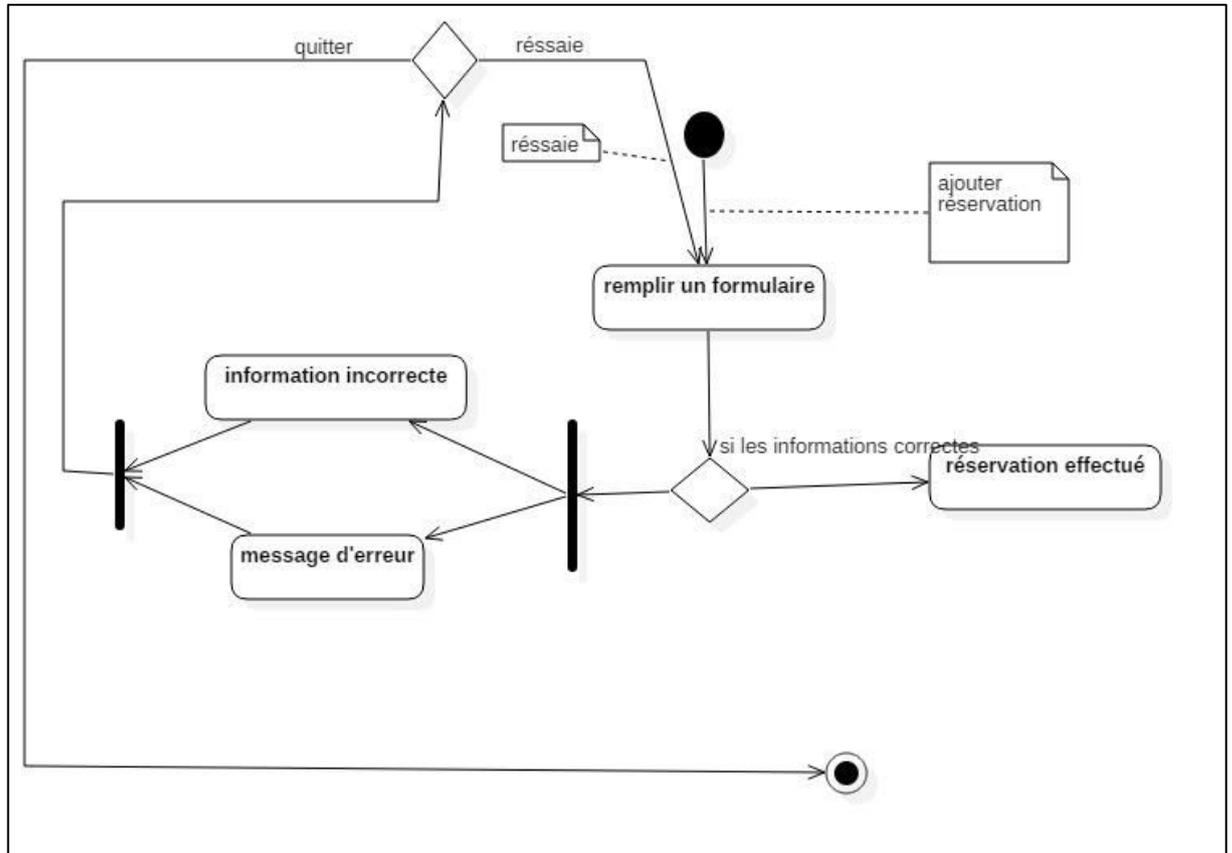


Figure II.7 : diagramme d'activité

Ce diagramme explique le cas d'utilisation le plus important dans notre système qui est **la réservation**.

II.4.4- Diagramme de classe

Le diagramme de classes est l'étape la plus importante dans la modélisation orientée objet d'un projet.

Il représente les classes intervenantes dans le système. Le diagramme de classe est une représentation statique des éléments qui composent un système et de leurs relations. Chaque application qui va mettre en œuvre le système sera une instance des différentes classes qui le compose [12].

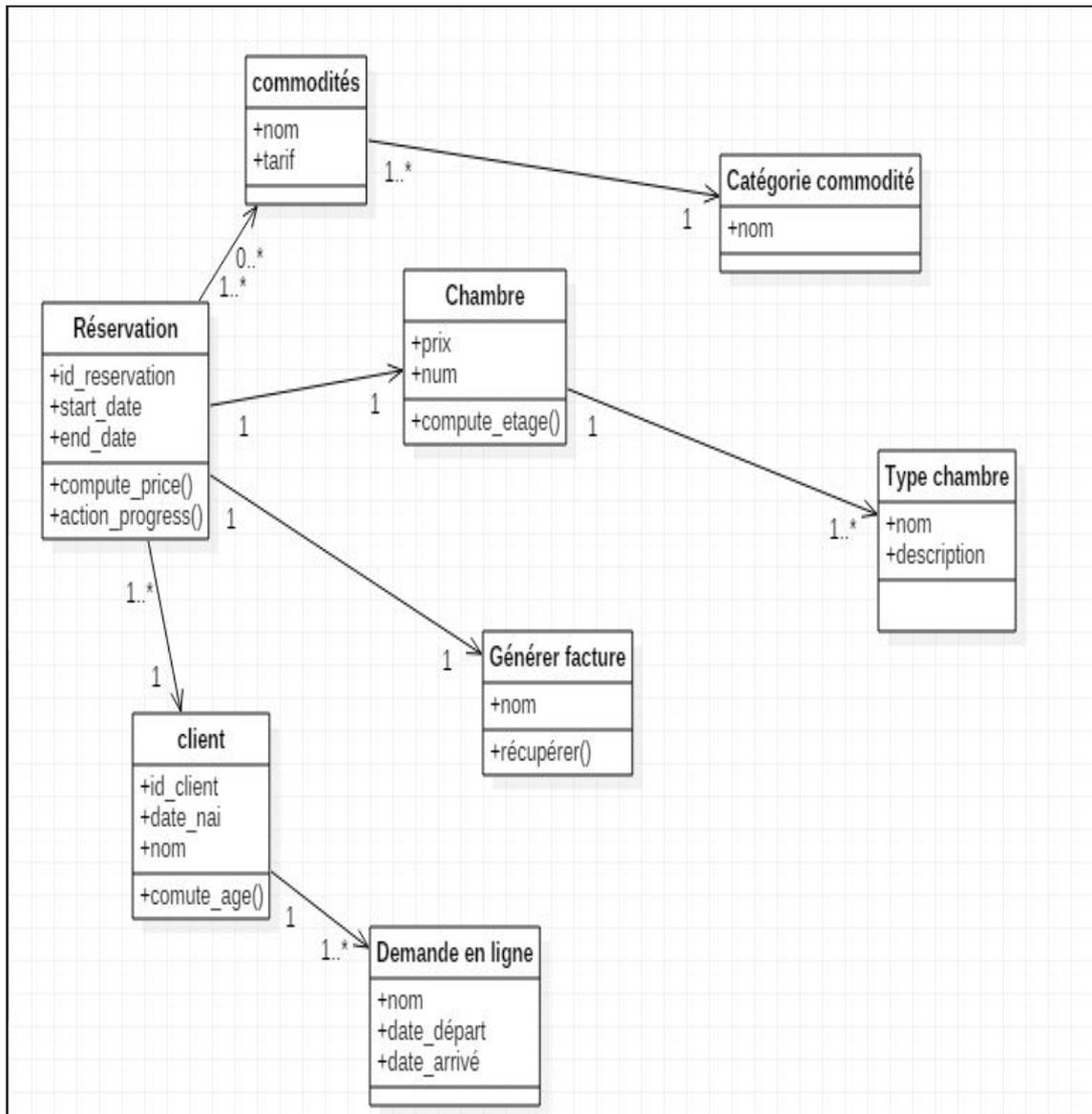


Figure II.8 : diagramme de classe

Notre application est composée de plusieurs classes : réservation, chambre, client, type chambre, demande de réservation avec les contraintes suivantes :

- Une chambre peut avoir un seul type.
- La réservation est affectée à un seul client.
- Le client peut effectuer plusieurs demandes de réservations.

II.5- Conclusion

Cette phase nous a permis d'identifier le processus de fonctionnement de l'application, les principaux cas d'utilisation, les diagrammes de séquence, diagramme d'activité et le diagramme de classe. Le chapitre suivant sera consacré à la présentation de la partie développement de notre application.

Chapitre III

III.1- Introduction

Dans ce chapitre nous présentons les différents outils et technologies adoptés et utilisés pour la réalisation de notre projet .Ainsi que l'ensemble des fonctionnalités offertes par notre application avec une description des différentes interfaces de celle-ci.

III.2- Environnement de travail

Pour le choix technique d'implémentation de notre application nous avons opté pour le PGI ODOO comme déjà cité au paravent. Cet ERP se base autour de deux langages :

- **Python**

Python [13] est un langage de programmation au source ouvert, généraliste et orienté objet. Python est conçu pour optimiser la productivité du développeur, la qualité du logiciel, la portabilité des programmes et l'intégration des composants.les programmes python fonctionnent sur la pluparts des plates-formes couramment utilisé.

```
1  #-*- coding: utf-8 -*-
2  from openerp import models, fields, api, exceptions
3  from datetime import datetime
4  from dateutil.relativedelta import relativedelta
5
6
7
8  class dzhotel_roomtype(models.Model):
9      _name = "dzhotel.roomtype"
10     name=fields.Char(u"Type de la chambre",size=64,required=True)
11 dzhotel_roomtype()
12
```

Figure III.1 : classe type chambre

La figure ci-dessus représente un extrait du programme python qui est la classe type chambre.

- **XML eXtensible Markup language (langage a balise extensibles)**

XML [14] est le standard soutenu par le W3C pour le balisage de documents, c'est une spécification du stockage des informations et de la structure des

informations stockées bien qu'il s'agisse d'un langage à balises (tout comme HTML), XML ne définit aucune balise qui lui soit propre et les utilisateurs peuvent créer celles dont ils ont besoin.

```
1
2 <?xml version="1.0" encoding="UTF-8"?>
3 <openerp>
4   <data>
5     <record model="ir.ui.view" id="roomtype_form_view">
6       <field name="name">roomtypes</field>
7       <field name="model">dzhotel.roomtype</field>
8       <field name="arch" type="xml">
9         <form string="Type de chambre">
10          <sheet>
11            <group col="2">
12              <field name="name" ></field>
13            </group>
14          </sheet>
15        </form>
16      </field>
17    </record>
18
```

Figure III.2 : vue forme.

La figure ci-dessus représente un extrait XML de la vue forme type de chambre.

III.3- Etapes de développement de notre application Dzhotel

III.3.1- installation d'Odoo 8.0 sous windows



Figure III.3 : installation étape 1.

La première étape de l'installation fait apparaître la figure ci-dessus.

Il faut suivre l'installation jusqu'à cette fenêtre, cocher « Installation du serveur de base de données PostgreSQL ».

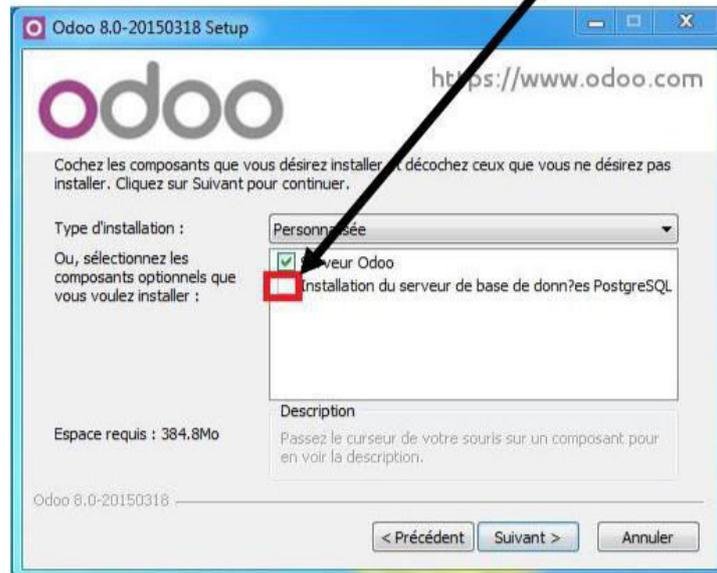


Figure III.4 : installation étape 2.

Cette étape permet d'installer le serveur odoo et la base de données postgresql.

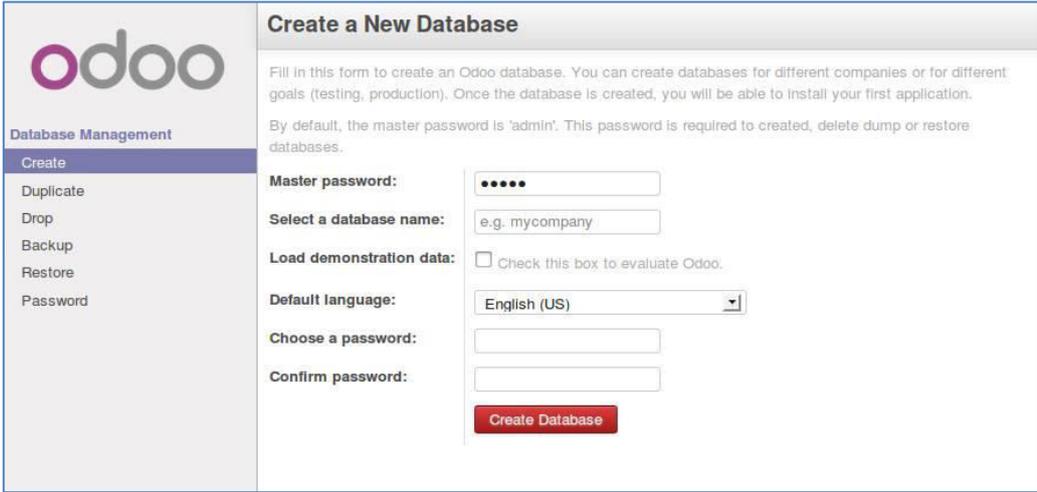


Figure III.5 : installation étape 3.

Laisser les informations de connexion pour le serveur PostgreSQL par défaut.

PostgreSQL [18] est un système de gestion de base de données relationnelle et objet. PostgreSQL peut stocker plus de types de données que les types traditionnels entiers, caractères, etc. L'utilisateur peut créer des types, des fonctions, utiliser l'héritage de type, etc.

Une fois l'installation est terminée nous devons créer notre base de données :



The screenshot shows the 'Create a New Database' form in Odoo. On the left is a sidebar with 'Database Management' and options like 'Create', 'Duplicate', 'Drop', 'Backup', 'Restore', and 'Password'. The main form area has the title 'Create a New Database' and instructions: 'Fill in this form to create an Odoo database. You can create databases for different companies or for different goals (testing, production). Once the database is created, you will be able to install your first application. By default, the master password is 'admin'. This password is required to create, delete dump or restore databases.' The form fields include: 'Master password:' (masked with dots), 'Select a database name:' (with 'e.g. mycompany' as an example), 'Load demonstration data:' (checkbox, 'Check this box to evaluate Odoo.'), 'Default language:' (dropdown menu set to 'English (US)'), 'Choose a password:' (empty text box), and 'Confirm password:' (empty text box). A red 'Create Database' button is at the bottom.

Figure III.6: création de la base de données

L'interface fournit par Odoo est la suivante:

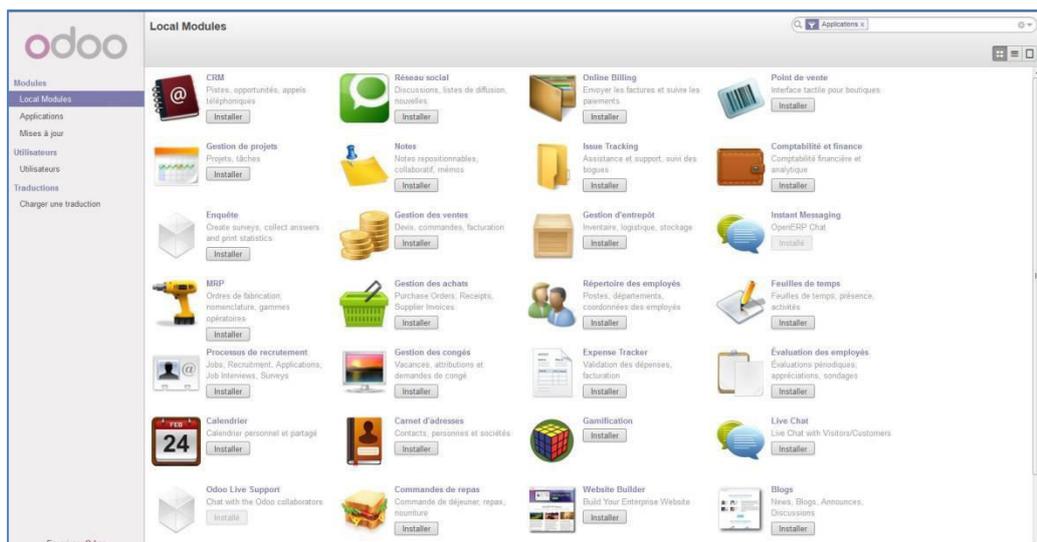


Figure III.7 : interface odoo.

Pour pouvoir créer et installer notre module il nous a fallu activer les menus de la configuration avancée :

Configuration / utilisateur / modifier l'utilisateur / onglet droit d'accès

- Cocher caractéristiques techniques
- Actualiser la page

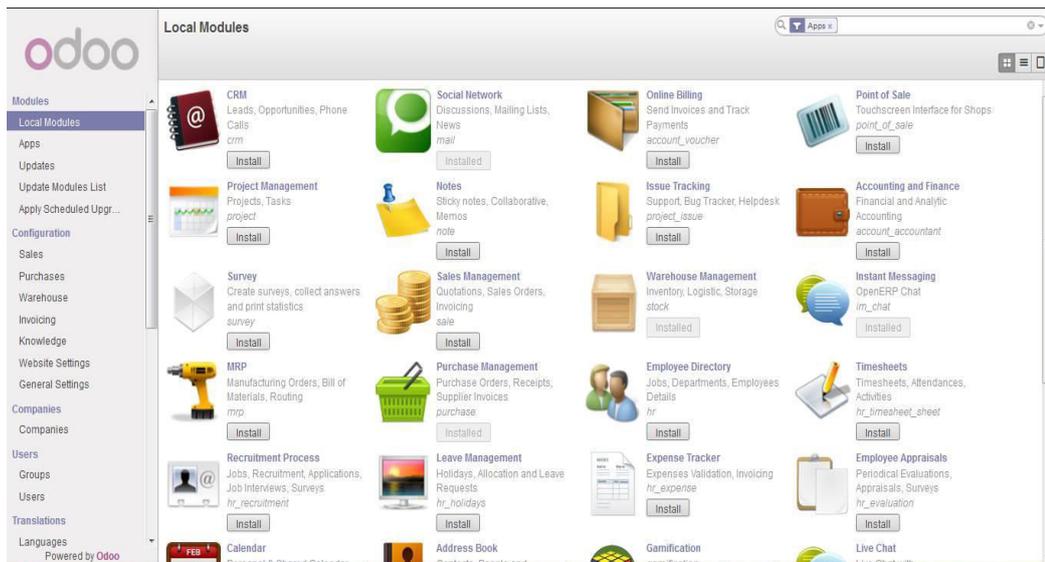


Figure III.8: interface odoo 2.

III.3.2- Mise en place de l'application DzHotel :

Quatre fichiers sont obligatoires lors de la création de notre module sont : `__init__.py` , `__openerp__.py` , `fichier.xml`, `fichier.py`

- `__init__.py` : représente les fichiers python importé

```

__init__.py
1 import client
2 import chambre
3 import chambretype
4 import reservation
5 import commodité
6

```

Figure III.9: fichier `__init__.py`

La figure ci-dessus représente le fichier python `__init__.py`. Nous avons importé dedans les classe `client`, `chambre`, `chambre type`, `réservation`, et `commodités`. Les classes citez peuvent être créé dans un seul fichier python. Tout fichier python doit être importé dans le fichier `__init__.py`

- `__openerp__.py` : ce fichier contient la description de notre module

```
__openerp__.py
1  # -*- coding: utf-8 -*-
2  #####
3  #
4  #####
5  {
6      'name': 'Dzhotel',
7      'version': '1.0.0',
8      'category': 'sale',
9      'description': ""
10
11     Module de gestion d'hotels
12     =====
13
14
15     """
16     'author': 'sihem & sara',
17     'website': 'https://dzhotel.com',
18     'data' : [
19
20         'views/client_view.xml' ,
21         'views/room_view.xml' ,
22         'views/roomtype_view.xml' ,
23         'views/booking_view.xml' ,
24         'views/menuitems.xml' ,
25         'views/amenities_view.xml' ,
26     ],
27
28     'auto_install': False,
29 }
30
31
```

Figure III.10 : fichier `__openerp__.py`.

Le fichier `openerp` contient la description du module tel que le nom, la version, l'auteur, ...etc.

Tous les fichiers XML doivent être déclarés dans le fichier `__openerp__.py`

- **Fichier.py** : parmi les fichiers python dans notre module nous avons **chambretype.py**

```

1  -*- coding: utf-8 -*-
2  from openerp import models, fields, api, exceptions
3  from datetime import datetime
4  from dateutil.relativedelta import relativedelta
5
6
7
8  class dzhotel_roomtype(models.Model):
9      _name = "dzhotel.roomtype"
10     name=fields.Char(u"Type de la chambre",size=64,required=True)
11     dzhotel_roomtype()
12
13

```

Figure III.11 : classe type chambre.

La figure représente la classe python type chambre.

- **Fichier.xml** : parmi les fichiers python dans notre module nous avons **room.xml**

```

room_view.xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <openerp>
3      <data>
4          <record model="ir.ui.view" id="dzhotel_room_form_view">
5              <field name="name">dzhotel.room.form.view</field>
6              <field name="model">dzhotel.room</field>
7              <field name="arch" type="xml">
8                  <form string="Gestion de chambres">
9                      <sheet>
10                         <header>
11                             <button string="Nettoyé" name="action_done" type="object" class="oe_highlight"/>
12                         </header>
13                         <group col="2" >
14                             <group col="2">
15                                 <field name="num" ></field>
16                                 <field name="etage" />
17                                 <field name="price" />
18                                 <field name="is_cleaned" />
19                                 <field name="end_d" invisible="True"/>
20                                 <field name="stat" />
21                             </group>
22                             <group col="2">
23                                 <field name="type" ></field>
24                                 <field name="phone" />
25                             </group>
26                         </group>
27                     </sheet>
28                 </form>
29             </field>
30         </record>
31

```

Figure III.12: vue forme de la chambre.

La figure représente la vue form de la classe type chambre.

- ❖ Après la création de ces fichiers nous redémarrons le serviced'Odo, mettons à jours la liste des modules et installons DzHotel.La figure présente le résultat obtenu.



Figure III.13 : installation du module dzhotel.

III.4- Description de notre module

DzHotel offre principalement les fonctionnalités suivantes :

III.4.1- Gestion des clients

L'interface clients sur figure ci-dessous permet de créer et mettre à jours la liste des clients. Les champs en bleu sont obligatoires.

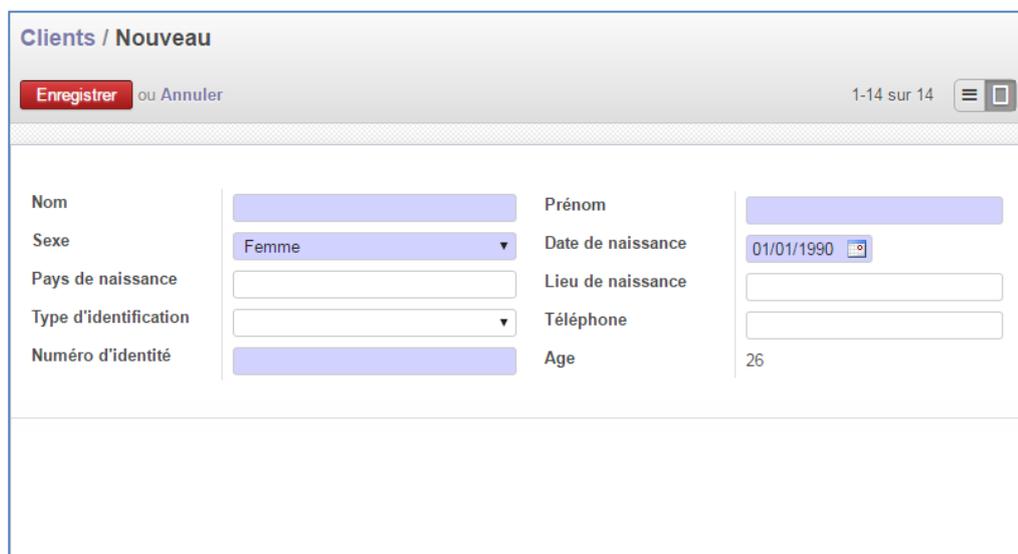
The image shows the 'Clients / Nouveau' form in Odoo. At the top, there are buttons for 'Enregistrer' (highlighted in red) and 'ou Annuler'. The page number '1-14 sur 14' and a menu icon are on the right. The form contains several fields: 'Nom' (text input, highlighted in blue), 'Prénom' (text input, highlighted in blue), 'Sexe' (dropdown menu with 'Femme' selected, highlighted in blue), 'Date de naissance' (date picker with '01/01/1990' selected, highlighted in blue), 'Pays de naissance' (text input), 'Lieu de naissance' (text input), 'Type d'identification' (dropdown menu, highlighted in blue), 'Téléphone' (text input), 'Numéro d'identité' (text input, highlighted in blue), and 'Age' (text input with '26' entered).

Figure III.14 : interface client.

III.4.2- Gestion des chambres

L'interface chambre ci-dessous permet de gérer la liste des chambres en indiquant (le tarif, le type, Téléphone.....)

The screenshot shows a web interface titled "Chambres / Nouveau". At the top left, there are two buttons: "Enregistrer" (highlighted in red) and "ou Annuler". At the top right, there are icons for a menu and a window. The main content area is divided into two columns of form fields:

Numéro de chambre	<input type="text" value="123"/>	Type	<input type="text" value="double"/>
Etage	<input type="text" value="1"/>	Téléphone	<input type="text"/>
Tarif de la chambre	<input type="text" value="7000"/>	Chambre nettoyée	<input checked="" type="checkbox"/>
Dernier nettoyage	<input type="text"/>		

Figure III.15 : interface chambre.

Il est aussi possible de mettre à jour les différents types des chambres à partir de l'interface sur la figure suivante :

The screenshot shows a web interface titled "Nouveau". At the top left, there are two buttons: "Enregistrer" (highlighted in red) and "ou Annuler". At the top right, there are icons for a window and a menu. The main content area is titled "Type chambre" and contains the following form fields:

Type de la chambre	<input type="text" value="Double"/>
déscription	<input type="text" value="Ces chambres doubles standard ont été décorées individuellement et sont idéales pour les touristes cherchant un hébergement économique."/>

Figure III.16 : interface type chambre.

III.4.3- Gestions des réservations

Lors de la création d'une réservation l'administrateur doit remplir un formulaire contenant les coordonnées du client. La réservation est affectée pour un seul client.

Réservations / Nouveau

Enregistrer ou Annuler

Commencer **Ouverte** En cours Terminee

Client Chambre

Début de réservation Fin de réservation

Tarif 0,00

Clients attachés Commodités

Figure III.17 : interface réservation

La réservation peut aussi être effectuée à travers un planning :

bookings

Aujourd'hui **16/05/2016 — 22/05/2016** Mois Semaine Jour

W20 lun. mar. mer. jeu. ven. sam. dim. 22/05/2016

16/05/2016 17/05/2016 18/05/2016 19/05/2016 20/05/2016 21/05/2016

Journée entière CH166

06

07

08

09

10

11

Créer : Reservation

Résumé de l'événement :

Créer un événement

Modifier l'événement

mai 20

lun. mar. mer. jeu.

2 3 4 5

9 10 11 12

16 17 18 19

23 24 25 26

30 31

CH166

Figure III.18 : vue calendrier.

Lors de la création d'une réservation l'utilisateur peut ajouter les clients attachés à la chambre :

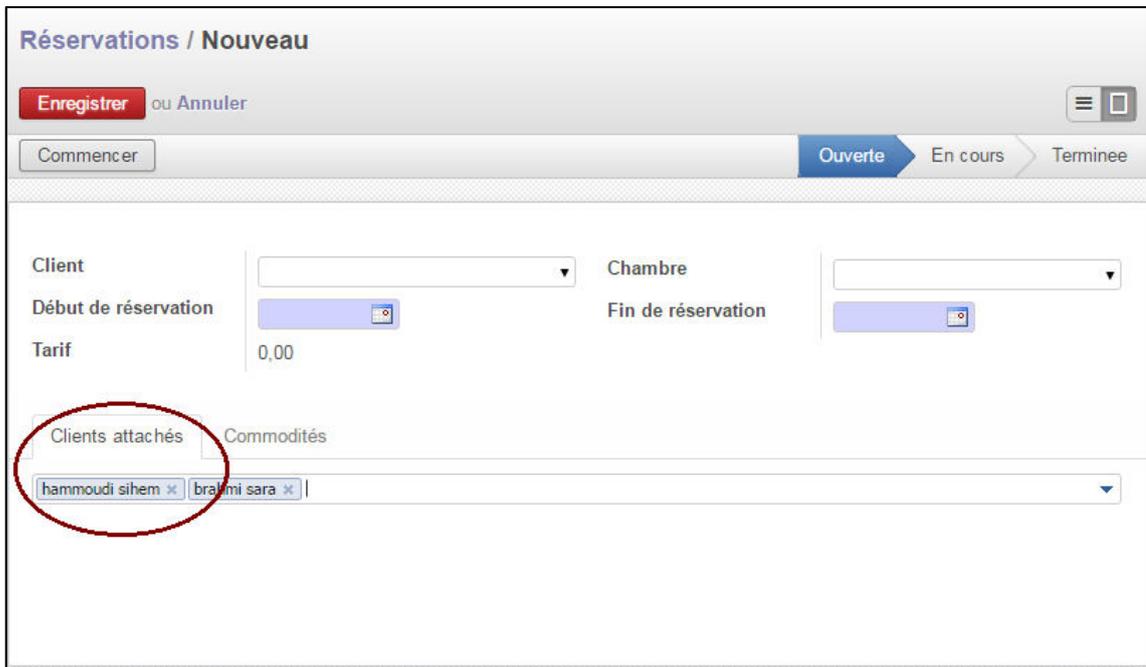


Figure III.19 : Ajouter clients attachés

Il peut aussi spécifier des commodités :

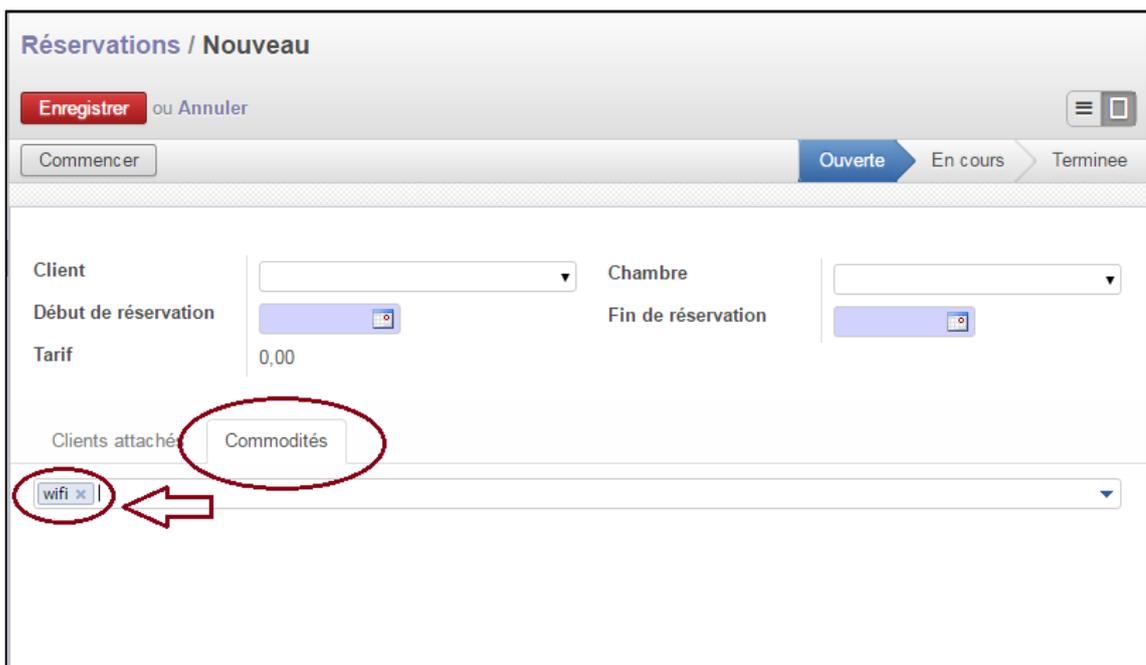


Figure III.20 : Ajouter commodités

Il est possible de mettre a jour les commodités a partir de cette interface :

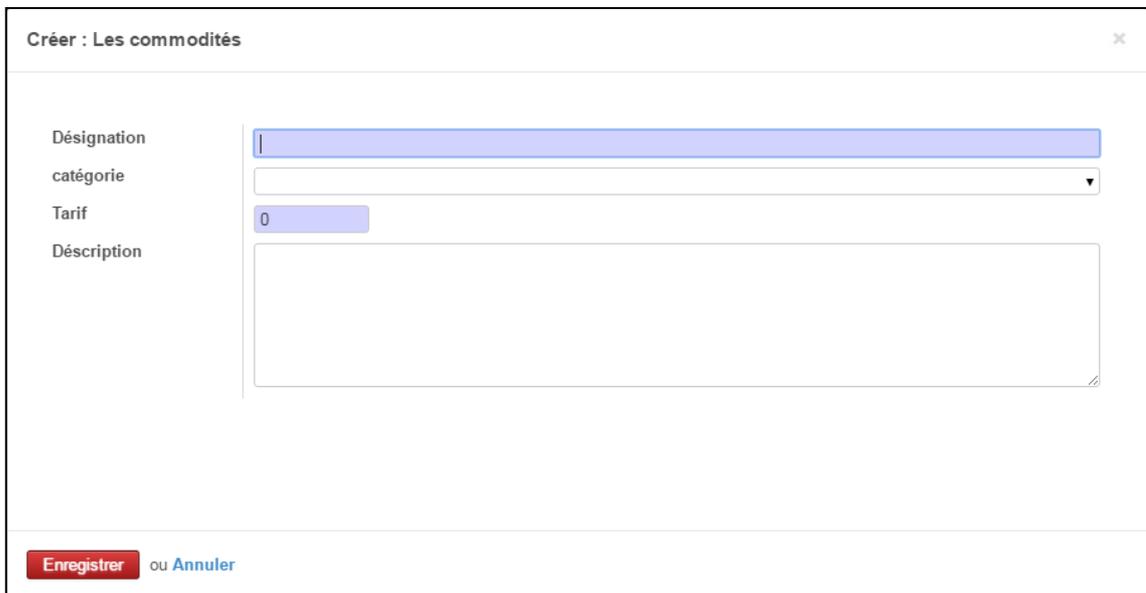
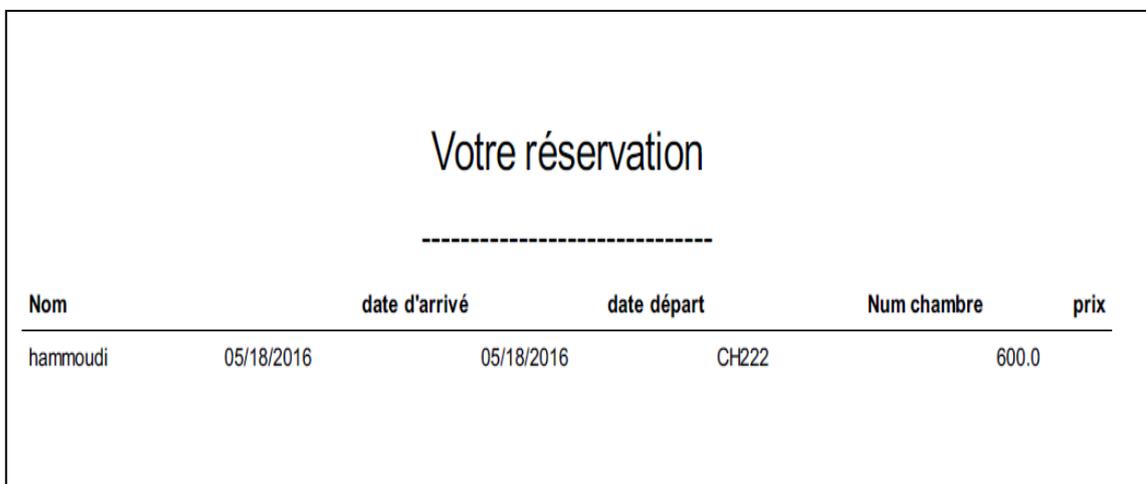


Figure III.21 : créer commodités.

III.4.4- Gestions des rapports

L'administrateur peut imprimer un rapport PDF de réservation qui contient le nom du client, date d'arrivé, date départ, ainsi le montant à payer.



Votre réservation				
Nom	date d'arrivé	date départ	Num chambre	prix
hammoudi	05/18/2016	05/18/2016	CH222	600.0

Figure III.22 : Rapport de réservation

III.5- Développement du site web de la réservation en ligne

III.5.1- les outils utilisés :

- Serveur d'application Python Flask :
Flask [15] est un framework open-source de développement web en Python. Son but principal est d'être léger, afin de garder la souplesse de la programmation Python, associé à un système de templates
- Bootstrap :
Nous avons choisis le *framework* web bootstrap [16] qui facilite la création de sites internet et d'applications web. Il contient des modèles HTML et CSS qui permettent de créer rapidement des formulaires, des boutons, des outils de navigation et d'autres éléments dynamiques.
- HTML :
HTML [17] est un langage dit de « marquage » (de « structuration » ou de « balisage ») dont le rôle est de formaliser l'écriture d'un document avec des balises de formatage. Les balises permettent d'indiquer la façon dont doit être présenté le document

III.5.2- L'installation du module website builder :

Pour créer le site sur ODOO, nous étions obligés d'installer le module website builder :

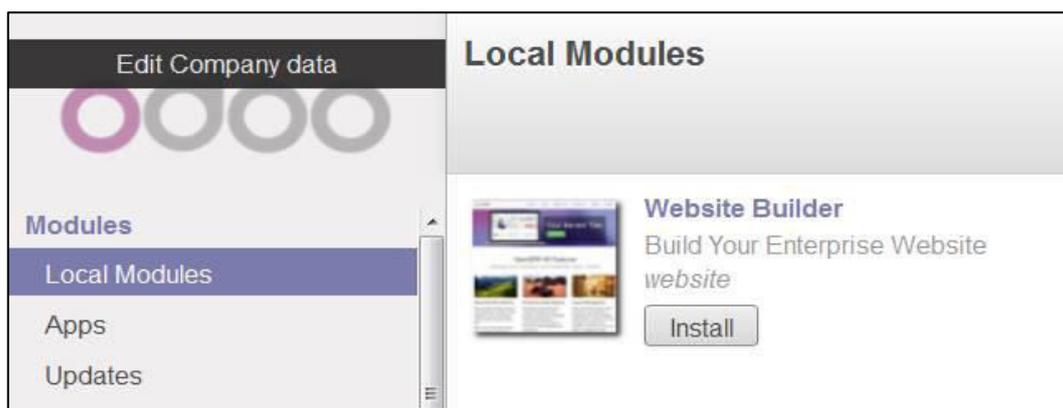


Figure III.23: Le module website builder.

III.5.3- Présentation du site

Notre site web comprend :

III.5.3.1- Page d'accueil « Notre hôtel »

Notre site à une interface graphique dont la page d'accueil est comme suit :



Figure III. 24: Page d'accueil

III.5.3.2- Page Chambres

Cette figure contient tous les types des chambres et leurs descriptions ainsi que leur prix, le nécessaire pour effectuer un choix de réservation.

		
Chambre seule	Chambre double	Chambre triple
Déscription: Ces chambres individuelles sont confortables et ont été décorées individuellement. Elles sont économiques et se trouvent au cœur de ville x. Elles disposent d'une décoration classique avec des tapis, des papiers peints colorés et un mobilier ancien, qui vous feront vous sentir comme chez vous. Nos salles de bain privées disposent soit d'une baignoire, soit d'une douche.	Déscription: Ces chambres doubles standard ont été décorées individuellement et sont idéales pour les touristes cherchant un hébergement économique au cœur de la ville x. Elles disposent d'une décoration classique avec des tapis, des papiers peints colorés et un mobilier ancien, qui vous feront vous sentir comme chez vous.	Déscription: Ces chambres triples sont spacieuses et joliment décorées. Elles sont parfaites pour les couples avec enfants ou les amis voyageant ensemble et pour ceux qui cherchent un hôtel économique dans le centre de la ville. Nos salles de bain privées disposent soit d'une baignoire, soit d'une douche.
prix: 20 000 DA	prix: 50 000 DA	prix:60 000 DA

Figure III.25 : Chambres.

III.5.3.3- Page Demande de réservation

Le client doit remplir le formulaire sur figure ci-dessous pour pouvoir envoyer une demande réservation :

Informations personnelles

Nom	<input type="text" value="text input"/>
Prénom	<input type="text" value="text input"/>
CIN / passeport	<input type="text" value="text input"/>
Date de naissance	<input type="text" value="text input"/>
Date d'arrivé	<input type="text" value="text input"/>
Date départ	<input type="text" value="text input"/>
E-mail	<input type="text" value="text input"/>
Type chambre	<input type="text" value="text input"/>
	<input type="button" value="Envoyer"/>

Figure III.26 : formulaire de demande de réservation.

III.5.3.4- Page Contacter nous

Cette figure présente la page web du site qui contient toutes les informations sur l'hôtel. Aussi elle décrit la situation géographique de l'hôtel sur la carte Google maps.

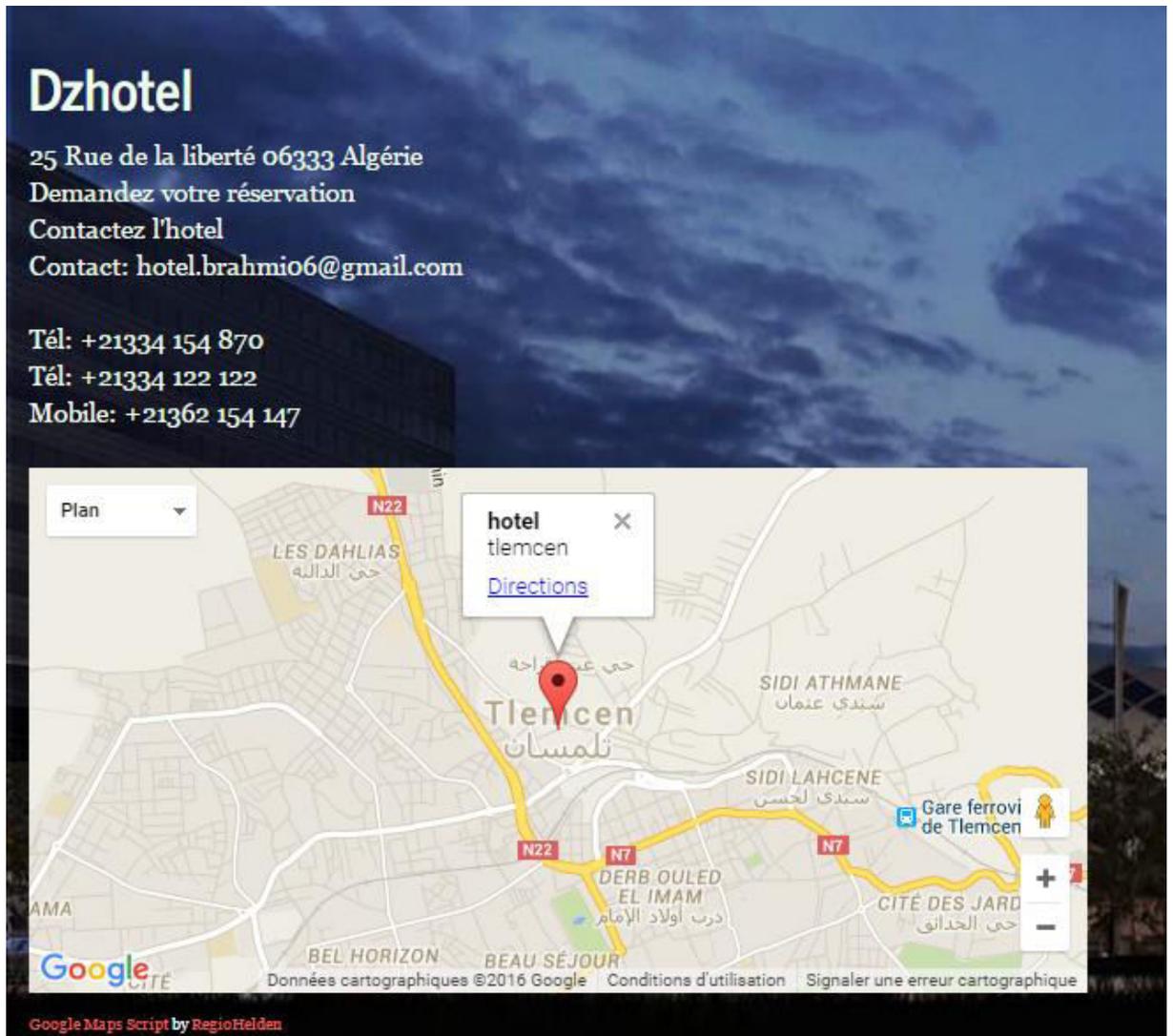


Figure III.27: page contacter nous.

III.6- Conclusion

Dans ce chapitre nous avons décrit le processus de développement de notre projet. Ceci en spécifiant l'environnement et la démarche suivie pour la réalisation du module DzHotel et le site de réservation en ligne. Nous avons aussi détaillé les différentes fonctionnalités offertes par notre application que cela soit sur la gestion d'hôtel ou bien la réservation en ligne.

Conclusion Générale

Conclusion générale

Dans notre projet de fin d'étude nous avons mis en place un PGI (progiciel de gestion intégré) de gestion d'hôtel ainsi qu'un site de réservation en ligne.

Notre module gère les principales fonctionnalités attaché à une gestion d'hôtel tel que la gestion des clients, des chambres, des commodités offerts par l'hôtel, et des rapports de réservation.

Notre site quand a lui, offre la possibilité d'une demande de réservation en ligne, a partir de ce site l'utilisateur peut aussi contacter l'hôtel pour des renseignements ou pour consulter la situation géographique de l'hôtel sur la carte Google Mapp.

Ce projet de fin d'étude été pour nous une occasion qui nous a permis de côtoyer le monde des ERPs. Travailler sur une open source tel que ODOO nous a permis d'acquérir des connaissances et des compétences de gestion intégrée.

Plusieurs technologies ont été nécessaires pour la réalisation de notre application, nous citerons donc le langage python, XML, HTML, l'environnement du travail Odoo, et une base de donnée postgreSql.

Ce travail étant une œuvre humaine, n'est pas un modèle unique et parfait.

En termes de perspectives nous souhaiterons :

- Lier le module dzHotel avec un module restaurant qui gère la restauration de l'hôtel.
- Gérer les travaux et les autres services de l'hôtel tel que le personnel, financement...etc.

Références bibliographiques

- [1] Exposés encadrés et dirigés par Dominique Revuz, directeur de l'UFR Ingénieurs 2000 à l'Université de Marne-la-vallée, et par Etienne Duris.
<http://www.wigm.univmlv.fr/~dr/XPOSE2008/xpose%20HIROOKA%20SHOOTCHI/erp.html>
- [2] Cours master CCA pgi erp AUTEUR PHILIPPE NORIGEON.
http://www.guillaumeriviere.name/estia/si/pub/cours_ERP_PGI_2010.pdf
- [3] Le site officiel d'entreprise-erp articles.
<http://www.entreprise-erp.com/articles/definition-erp.html>
- [4] MAOUCHE Cilia et MENDIL Radia, Intégration et personnalisation d'OpenERP au sein de l'entreprise ISATIS ,thèse de Master ,université A.MIRA- Bejaia promotion :2013-2014.
<http://fr.slideshare.net/celiamaouche/memoire-corrig>
- [5] M. DAMIR Ayoub, la mise en place d'une solution pour la gestion des projets, gestion de ressources humaines Sous la plateforme Odoo, mémoire de projet de fin d'étude pour l'obtention du titre Ingénieur d'Etat en Informatique promotion :2010-2015.
- [6]Le site officiel Aoik-Conseils, 2015.
<http://www.apik-conseils.com/odoo/>
- [7] Ait mlouk Addi ,Conception et réalisation d'une application de gestion intégrée au sein de la société Eone Group basée sur OpenERP ,la faculté des sciences Semlalia de Merrakech pour obtenir : Le diplôme du master spécialisé « ingénierie des systèmes d'information »promotion 2012-2013.
<http://fr.slideshare.net/mloukaddi/conception-et-ralisation-dune-application-de-gestion-intgre-au-sein-de-la-socit-eone-group-base-sur-openerp>
- [8] Jean François Pillou , uml_modeliser ,2002.
http://www.lomag-man.org/informatique/conception_logiciels/umlmodeliser.php
- [9] David Janiszek, cours de génie logiciel.
http://www.mi.parisdescartes.fr/~janiszek/_media/enseignement/06_diagramme_cas_d_utilisation.pdf
- [10] ABDELAALI AIT BOUGARMOUD Rapport de projet de fin de formation Dossier De Conception DEVELOPPEMENT D'UNE APPLICATION DE RESERVATION POUR LA PLATEFORME ANDROID 2012/2013 .

Références bibliographiques

toubkalit.com/chapitre/Rapport_pff_abdelaali_ait_bougarmoud_GA.pdf

[11] Projet De StijlPlate-forme pour robots mobiles Dossier de conception Version 1.0.2 (4 mars 2015) Institut National des Sciences Appliquées de Toulouse.

moodle.insa-toulouse.fr/mod/resource/view.php?id=10610

[12] Le site officiel uml-sysml.org

<http://www.uml-sysml.org/diagrammes-uml-et-sysml/diagramme-uml/diagramme-de-classe>

[13]Mark Lutz , Python : précis & concis

<https://books.google.fr/books?isbn=2841773604>

[14] Elliotte Rusty Harold,W. Scott Means, XML En concentré Manuel de référence rapide .

<https://books.google.com/books?isbn=2841773531>

[15]Flask(framework), dernière MAJ : 20 mai 2016.

http://fr.wikitally.com/wiki/Flask_%28framework%29.html

[16]Carmen Olazabal, le framwork le plus populaire au monde , 20 février 2015

<https://mastercaweb.u-strasbg.fr/blog/bootstrap-le-framework-le-plus-populaire-au-monde>

[17]Cours les langages HTML et CSS

<http://www.maths.lyceebellepierre.fr/IMG/pdf/index-5.pdf>

[18] Cours et tutoriels sur le langage SQL

<http://sql.sh/sghd/postgresql>

Figure I.1 : Organisation traditionnelle	6
Figure I.2 : Organisation avec un ERP	7
Figure I.3 : Architecture technique des ERP	10
Figure I.4 : Architecture modulaire des ERP	11
Figure I.5 : Architecture modulaire des ERP	14
Figure I.6 : Modèle vue contrôleur	17
Figure II.1 : processus de fonctionnement de notre application.....	21
Figure II.2 : Diagramme de cas d'utilisation globale	23
Figure II.3 : diagramme séquence gestion client.....	24
Figure II.4 : diagramme de séquence gestion chambre.....	25
Figure II.5 : diagramme séquence de réservation	26
Figure II.6 : diagramme de séquence réservation en ligne.....	27
Figure II.7 : diagramme d'activité.....	28
Figure II.8 : diagramme de classe	29
Figure III.1 : classe type chambre.....	32
Figure III.2 : vue forme.....	33
Figure III.3 : installation étape 1	33
Figure III.4 : installation étape 2.....	34
Figure III.5 : installation étape 3.....	34
Figure III.6 : création de la base de données.....	35
Figure III.7 : interface odoo.....	35
Figure III.8 : interface odoo 2.....	36
Figure III.9 : fichier __init__.py.....	36
Figure III.10 : fichier __openerp__.py.....	37
Figure III.11 : classe type chambre.....	38
Figure III.12 : vue forme de la chambre.....	38
Figure III.13 : installation du module dzhotel.....	39
Figure III.14 : interface client.....	39
Figure III.15 : interface chambre.....	40
Figure III.16 : interface type chambre.....	40
Figure III.17 : interface réservation	41

La liste des figures

Figure III.18 : vue calendrier.	41
Figure III.19 : Ajouter clients attachés.....	42
Figure III.20 : Ajouter commodités	42
Figure III.21 : créer commodités.....	43
Figure III.22 : Rapport de réservation.....	43
Figure III.23 : Le module website builder.	44
Figure III. 24 : Page d'accueil.....	45
Figure III.25 : Chambres	46
Figure III.26 : formulaire de demande de réservation.	47
Figure III.27 : page contacter nous.	48

Résumé

Pour améliorer leur performance, les organismes d'aujourd'hui visent à automatiser la gestion interne de leurs activités en faisant appel à des technologies informatiques plus sophistiquées comme les PGI (Progiciels de Gestion Intégrés), ou ERP (Progiciel de Gestion Intégré). La gestion de réservation de chambre est une vitalité indispensable dans le déroulement des activités d'un hôtel. Dans ce mémoire nous avons mis en place un ERP de gestion de réservation de chambre d'hôtel et un site de réservation en ligne sous ODOO V8. Ce travail consiste d'une part à gérer la réservation de chambre et les principales fonctionnalités de l'hôtel qui lui sont liées. D'autre part à offrir aux clients de l'hôtel une interface conviviale qui leur permet d'effectuer en ligne leur demande de réservation (en choisissant les dates, le type de chambre...). Ceci en prenant en considération toutes les contraintes qui peuvent surgir lors d'une réservation.

Les mots clé : ERP (Entreprise Ressource Planning), PGI (Progiciel de Gestion Intégré), ODOO, gestion de réservation de chambre d'hôtel, XML, MVC, Python

Abstract

To improve their performance, today's organizations seek to automate the internal management of their activities by using more sophisticated computer technologies such as integrated management software package or ERP (Enterprise Resource Planning). Room reservation management is an essential vitality in the running of a hotel business. In this paper, we have introduced a hotel room reservation management ERP and an online booking site under Odoos V8. This work aims at firstly managing room reservation and the main features of the hotel linked to it. Its other aim is to offer to hotel guests a friendly interface that allows them to perform their online reservation request (by choosing the dates, room type ...), taking into account all the constraints that may arise when making a reservation.

Key words: ERP (Enterprise Resource Planning), integrated management software package, Odoos, hotel room reservation management, XML, MVC, Python

ملخص

لتحسين الأداء؛ تسعى الهياكل لأتمتة الإدارة الداخلية لأنشطتها باستخدام تكنولوجيا الكمبيوتر الأكثر تطوراً كحزمة برامج متكاملة للإدارة أو تخطيط موارد المؤسسات. إدارة حجز الغرف هي قسم حيوي و أساسي في إدارة الأعمال في الفنادق . تضمن هذا العمل، استخدام مجموعة من التطبيقات المتعلقة بتخطيط موارد المؤسسات لإدارة حجز الغرف وموقع للحجز عبر الإنترنت عبر النسخة المعدلة 8 من برنامج "أودو". و ذلك بهدف إدارة حجز الغرف و ميزات الفندق الرئيسية المرتبطة بها . و من جهة أخرى، تقديم واجهة سهلة لنزلاء الفندق تسمح لهم بأداء طلب حجز عبر الإنترنت (اختيار مواعيد وصنف الغرفة ...). هذا مع الأخذ بعين الاعتبار جميع القيود التي قد تنشأ عند إجراء الحجز .

الكلمات الرئيسية: حزمة برامج متكاملة للإدارة، تخطيط موارد المؤسسات، أودو، إدارة حجز غرفة فندق، لغة التوصيف الموسعة، نموذج عرض تحكم ، بايثون.