

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE.
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique.
Université Mouloud Mammeri de Tizi-Ouzou.
Faculté de Génie Electrique et Informatique.
Département Informatique.

Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme de Master en Informatique

Thème

Etude comparative des ERP, Cas
d'étude : (Conception et réalisation
d'un
Module de gestion de fichiers avec
l'erp Odoo).

Dirigé par :
Mr: DIB A.

Réalisé par :
Melle: LADJ Amina.
Melle: SIAGH Fatma.

Année universitaire : 2014/2015

Dédicace

Je dédie ce travail à:

- ✓ Mes chères parents qui m'ont soutenu et encouragé durant mes études.
- ✓ AbdelHakim mon cher frère et Djouher ma chère sœur ainsi qu'à toute ma famille.
- ✓ Arezki Badsı Mon futur époux ainsi qu'à ses parents.
- ✓ Siagh Fatma mon binôme.
- ✓ Et enfin sans oublié tous mes ami(e)s de Saida, Sidi Bel Abbes, Bechar et Tizi-Ouzou qui m'ont accompagné durant tous les sicles de mes études.

Amina

Dédicace

Je dédie ce Modest travail à :

- ✓ Ma mère, ma raison d'être, ma raison de vivre, la lanterne qui éclaire mon chemin.
- ✓ Mon père, en signe d'amour, de reconnaissance et de gratitude pour tous les soutiens et les sacrifices dont il a fait preuve à mon égard.
- ✓ Mes chères frères et sœurs.

Aucun mot ne pourra décrire vos dévouements et vos sacrifices.

- ✓ Tous mes amis Lila, Linda et Sabrina , en témoignage de l'amitié sincère qui nous a liées et des bons moments passés ensemble.

- ✓ Mon binôme Ladj amira

Je vous dédie ce travail en vous souhaitant un avenir radieux et plein de bonnes promesses.

Tous les gens qui ont cru en moi et qui me donnent l'envie d'aller en avant, Je vous remercie tous, votre soutien et vos encouragements me donnent la force de continuer.

Fatma

Remerciements

Nous exprimons toute notre gratitude à Monsieur Dib Ahmed pour l'effort fourni, les conseils prodigués, sa patience et sa persévérance dans le suivi.

Nous adressons également nos remerciements, à tous nos enseignants, qui ont contribué à notre formation.

Nous remercions le personnel de l'O.P.G.I. en général, et plus particulièrement le responsable du Service Informatique pour avoir accepté de nous prendre en charge et d'avoir mis à notre disposition la documentation nécessaire.

Nous remercions sincèrement, les membres de jury d'avoir bien voulu accepter de faire partie de la commission d'examineurs.

A toute personne qui a participé de près ou de loin pour l'accomplissement de ce travail.

PDF Creator
www.nuance.com

Table des matières

Introduction générale	1
-----------------------------	---

Chapitre I : Les ERP

I.	Introduction	2
II.	Historiques des ERP	2
III.	Définition	3
IV.	Les ERP	3
IV.1	Caractéristiques d'un ERP	3
IV.2	Le besoin d'utiliser un ERP	4
IV.3	A qui sont destinés les ERP	4
IV.4	Principe de fonctionnement d'un ERP	5
IV.5	Classement des ERP	5
IV.5.1.	Le type	5
IV.5.2	La catégorie	6
IV.5.3	Le groupe	6
IV.6	Les avantages et les inconvénients des ERP	7
a)	Les avantages des ERP	7
b)	les inconvénients des ERP	7
IV.7	Quelques ERP gratuits	8
a)	Open Concerto	8
b)	Neogia	8
c)	OFBiz	8
V.	Le logiciel ERP	9
V.1	Logiciel ERP	9
V.1.1	L'Architecture d'un ERP	9
a)	l'architecture fonctionnelle	9
b)	architecture technique	10
c)	architecture modulaire des ERP	11
V.1.2.	Un module ERP	12

V.1.2.1.	Les différentes fonctionnalités des modules ERP -----	13
V.1.3.	Serveur ERP -----	14
V.1.3.1	Utilisation d'un serveur ERP à distance dans les Cloud-----	14
V.2	différents types de logiciels ERP-----	15
IV.	Conclusion -----	16

Chapitre II : L'ERP ODOO

I.	Introduction -----	17
II.	Historiques des ERP -----	17
III.	Définition -----	17
IV.	Pourquoi choisir l'ERP Odoo ?-----	18
V.	Principales Caractéristiques-----	19
VI.	Fonctionnalités d'Odoo (OpenERP)-----	20
VII.	Les avantages et les inconvénients -----	21
A.	Les avantages -----	21
B.	Les inconvénients-----	21
VIII.	Architecture d'OpenERP -----	22
a)	Architecture Client-serveur d'OpenERP -----	22
b)	Architecture modulaire d'OpenERP -----	25
IX.	Composants principaux d'Odoo -----	28
X.	Les applications Odoo -----	28
XI.	Open Source, quels domaines d'application ? -----	29
a)	Sur le poste client -----	29
b)	Applications d'entreprise -----	29
c)	Infrastructure -----	30
XII.	Odoo 8-----	30
XIII.	Conclusion-----	31

Chapitre III : Présentation de l'Organisme d'Accueil

I.	Introduction	32
II.	Présentation de l'organisation d'accueil.....	32
	II.1. Qu'est ce que l'O.P.G.I.	32
	II.2. Historique de l'O.P.G.I.	32
	II.3. Objectifs de l'O.P.G.I.	33
	II.4. Missions de l'O.P.G.I.	33
	II.5. Organisation générale de L'O.P.G.I. de Tizi-Ouzou.....	34
	II.5.1. Organisation structurelle de L'O.P.G.I.	35
	II.5.1.1. Direction Générale.....	35
	1. Directeur Général Adjoint.....	35
	2. Cellule Sécurité Interne.	35
	3. Cellule Affaires Juridiques et Contentieuses ----	36
	4. Cellule Audit Interne	36
	5. Cellule Organisation et Système d'Information et de Communication.....	36
	II.5.1.2. Agences	36
	II.5.1.3. Structuration des départements.....	37
	1. Département des Ressources Humaines et Moyens Généraux (D.R.H.M.G)	37
	1.1. Service Gestion des Ressources Humaines	37
	1.2. Service des Moyens Généraux	38
	2. Département Finances et Comptabilité (D.F.C.)	38
	2.1. Service Finance	39
	2.2. Service Comptabilité	39
	2.3. Service Investissement	39
	2.4. Service Recouvrement	40
	3. Département Gestion et Maintenance du Patrimoine (D.G.M.P.)	40
	3.1. Service Exploitation du Patrimoine.....	40
	3.2. Service Maintenance et Réparation.....	41
	4. Département Maitrise d'ouvrage (D.M.O.) ----	41
	4.1. Service Etudes et Programmation ----	41
	4.2. Service des Marchés.....	42
	4.3. Service de la Conduite des Opérations--	42
	5. Département de Développement de la Promotion Immobilière et Foncière (D.P.I.F.)	43
	5.1. Service de la Promotion Immobilière et Foncière.....	43
	5.2. Service Commercial.....	43
III.	Conclusion	43

Chapitre IV: Analyse et conception.

I.	Introduction	44
II.	Les étapes d'analyse et de la conception	45
III.	Phase d'analyse	46
III.1	Introduction	46
III.2	Définition des besoins	46
III.3	Identification des acteurs	47
III.4	Diagramme de contexte	47
III.5	Identification des cas d'utilisation	48
a)	Cas d'utilisation	48
b)	Spécification des cas d'utilisation	48
III.6	Spécification des tâches et des scénarios	49
III.7	Diagramme de cas d'utilisation	52
IV.	Phase de conception	55
IV.1	Introduction	55
IV.2	Diagrammes de séquence	55
IV.2.1	Définition	55
IV.2.2	Diagramme de séquence	55
IV.3	Diagramme d'activité	59
IV.4	Diagramme des classes	63
IV.4.1	Définition	63
IV.6	Diagramme de déploiement	65
IV.7	Diagramme d'architecture d'application	66
V.	Conclusion	67

Chapitre V: Réalisation.

I.	Introduction	68
II.	Partie I	68
	II.1. Présentation du matériel et les langages utilisés	68
	II.2. Le SGPD Postgre SQL8.3	68
	II.3. L'éditeur Notepad++	69
	II.4. Le langage de programmation python	70
	II.4.1 Définition	70
	II.4.2 caractéristiques du python	70
	II.5. L'XML	71
	II.5.1 Définition	71
	II.6. ODOO	72
	II.6.1 La démarche d'installation d'Odoo Sous windows	72
	II.6.2 la base de données dans PostgreSql	76
III.	Partie II: Présentation de l'application	77
	III.1. principe de bases	77
	III.2. les étapes de réalisation	77
	III.3. Fonctionnement d'ODOO	79
	III.3.1. Page d'authentification d'ODOO	79
	III.3.2. Espace administrateur	80
	III.3.3. Espace Utilisateur	84
IV.	Conclusion	87
	Conclusion générale	88
	Bibliographie	
	Annexe	

Liste des figures

Figure .I.1. Architecture fonctionnelle d'un ERP	9
Figure .I.2. Architecture technique d'un ERP	10
Figure .I.3. L'architecture modulaire qui intègre plusieurs modules retouchant aux grandes fonctions d'une entreprise.....	11
Figure .II.1. Architecture Client-serveur d'OpenERP.....	23
Figure .II.2. Architecture modulaire d'open ERP	25
Figure .II.3. Architecture en couche du module Odoo.....	26
Figure .II.4. Structure modulaire d'un ERP	26
Figure III.1 : Organigramme de l'O.P.G.I de Tizi-Ouzou.....	34
Figure IV.1: Cycle de modélisation de Module de la gestion de document	45
Figure .IV.2. Le diagramme de contexte.....	47
Figure IV.3 : Diagramme de cas d'utilisation de l'administrateur.....	53
Figure IV.4: Diagramme de cas d'utilisation « Utilisateur ».....	54
Figure IV.5: Diagramme de séquence de cas d'utilisation « s'authentifier »	56
Figure IV.6: Diagramme de séquence de cas d'utilisation « désactiver un compte Utilisateur	57
Figure IV.7: Diagramme de séquence de cas d'utilisation « Ajouter un document»	58
Figure IV.8: Diagramme d'activité de cas d'utilisation « Modifier un Utilisateur»	60
Figure IV.9: Diagramme d'activité de cas d'utilisation « Ajouter un Document»	61
Figure IV.10: Diagramme d'activité de cas d'utilisation « Supprimer un Document»	62
Figure IV.11: Diagramme de classe globale	64
Figure IV.12: Diagramme de déploiement	65
Figure IV.12: Diagramme d'architecture d'application	66

Figure. V.1.L'éditeur Notepad++	70
Figure. V.2. Les étapes d'installation d'Odoo sous Windows	72
Figure. V.3. Les étapes d'installation d'Odoo suite.....	73
Figure. V.4. La création d'une nouvelle base de données Odoo	74
Figure. V.5. Page Principale d'Odoo	75
Figure. V.6 .Présentation de la base de données	76
Figure. V.7. Le module Gesdoc installé.....	78
Figure. V.8. page d'authentification	79
Figure. V.9.Espace Administrateur « Menu Settings.....	80
Figure. V. 10. Espace Administrateur « Menu Settings/Spécification des droits d'accès ».....	81
Figure. V. 11. Espace Administrateur « Menu Settings/Création d'un groupe»	82
Figure. V.12.Espace Administrateur « Menu Settings/Création d'un utilisateur»	82
Figure. V.13. Espace Administrateur « Menu Gesdoc»	83
Figure. V.14.Espace Utilisateur « Menu Département des Ressources Humaines et Moyens Généraux (D.R.H.M.G.) Service des Moyens Généraux/Ajouter un document».....	84
Figure. V.15. Espace Utilisateur « formulaire d'ajout d'un Document ».....	85
Figure. V.16.Espace Utilisateur « formulaire de modification des informations d'un document ».....	85
Figure. V. 17. « La vue Kanban »	86
Figure. V. 18. « La vue Calendar »	86
Figure. V.19.Espace Utilisateur « absence des droit de création/d'édition et de suppression ».....	87

PDF Create! 5 Trial
www.nuance.com

Liste des tableaux

Tableau. I.1 Tableau des différentes fonctionnalités des modules ERP -----	13
Tableau. I.2 Tableau des différents types de logiciels ERP -----	15
Tableau II.1: représentant la Structure d'un module Odoo -----	27
Tableau IV.1 : Les différents cas d'utilisation -----	48
Tableau IV.2 : Résumé des scénarios par tâches d'un administrateur -----	49
Tableau IV.3 : Résumé des scénarios par tâches d'un utilisateur -----	51

PDF Create! 5 Trial
www.nuance.com

L'entreprise d'aujourd'hui est plus consciente de l'impact de la gestion efficace des ressources internes sur l'amélioration de sa performance et par la suite de sa compétitivité dans le marché. Or, la tâche de gérer s'avère de plus en plus difficile et complexe. En effet, la croissance des activités engendre un énorme flux de données, alors que la diversité des processus fonctionnels nécessite une gestion adéquate des compétences humaines. Pour surpasser ces difficultés, toute entreprise est prête à investir des sommes considérables dans l'implantation de technologies logicielles afin d'améliorer ses services et d'accroître son agilité vis-à-vis ses clients tout en optimisant la communication au sein de ses équipes. Pour la société OPGI, la gestion de l'ensemble de ses documents se fait manuellement. La société souhaite gérer la totalité de ses documents autour d'un même système d'information, à l'aide d'un progiciel de gestion intégrée connu sous l'acronyme ERP.

Un ERP est un système d'information composé de plusieurs applications informatiques relatives aux diverses fonctions de l'entreprise de manière modulaires sachant que ces modules sont indépendants entre eux et partagent une seule et la même base de données par le biais d'un système automatisé prédéfini et éventuellement paramétrable, un moteur de recherche workflow.

Dans notre cas on s'intéresse à OpenERP, c'est un progiciel de gestion intégré Open source et libre qui est téléchargé plus de 600 fois chaque jour, disponible en 20 langues, possède un réseau mondial de partenaires et de contributeurs (plus de 800 programmeurs) C'est dans cette optique que l'entreprise nationale OPGI cherche continuellement à améliorer son système d'informations, en introduisant un progiciel de gestion intégré ERP dans la gestion des documents.

Notre travail, s'inscrivant dans cette démarche, consiste justement à développer un module de gestion des documents sous l'ERP Open source OpenERP,

Le plan de ce mémoire s'articule autour des points de développement suivants :

- Le premier chapitre est dédié à la description des généralités sur les ERP qui ont des fonctions standards s'adaptant sur la principale règle de gestion de l'entreprise.
- Dans Le second chapitre, nous avons présenté le progiciel OpenERP (Odo) , avec le quel nous avons intégré notre module de gestion de document . notre choix s'est porté sur ce progiciel qui est un progiciel Open source.
- Le troisième chapitre est consacré à une étude détaillée de l'organisme d'accueil ou nous présentons les principales structures de l'entreprise OPGI ainsi que leurs activités.
- Le quatrième chapitre est réservé à l'analyse et la conception du module de la gestion de documents.
- Dans le dernier chapitre, nous présentons les outils de développement et quelques interfaces de fonctionnalités du module de la gestion de document
- Enfin, nous terminons ce travail par une conclusion générale qui illustre l'application qui traite la gestion de documents du personnel OPGI, ainsi que les perspectives projetées dans le futur.

I. Introduction

Dans une entreprise qui a commencé à s'informatiser, des outils informatiques hétérogènes coexistent à différents endroits souvent sans communiquer, et répondent à une seule fonction précise : la comptabilité, des tableaux du commercial avec des fichiers clients éparpillés, les informations techniques, les suivis des délais...

L'intérêt d'un outil de type Progiciel de Gestion Intégré est de centraliser toutes les informations nécessaires au bon fonctionnement de l'entreprise en informatisant les processus principaux

Dans ce chapitre, nous allons présenter l'ERP comme une technologie de l'intégration fonctionnelle qui offre à l'entreprise la possibilité d'une gestion intégrée et même intégrale de toutes ses fonctions ; Ce qui permet de comprendre l'utilité d'un ERP au sein de l'entreprise.

II. Historiques des ERP

Il est important de signaler à priori que les auteurs ne sont pas mis d'accord sur une même traduction en langue française du terme ERP (Entreprise Resource Planning). Certains le traduisent en tant qu'un système informatique de planification des ressources de l'entreprise. D'autres adoptent la traduction de gestion intégrée ou bien PGI : progiciel de gestion intégrée et c'est le terme le plus rencontré dans la littérature francophone. [1]

En effet Les premiers progiciels étaient apparus vers les soixante-dix. Ils recouvraient spécialement le domaine comptable. Ils ont pu satisfaire de nombreux besoins dans ce domaine alors que les développements spécifiques réalisés par les informaticiens de l'entreprise ont échoué de le faire. ces progiciels étaient le plus souvent la reprise d'une application développée par une entreprise qu'on adapte à une autre appartenant au même secteur. en outre leur dépendance par rapport à un environnement technique était énorme.

Pendant les années quatre-vingt les progiciels se développèrent pour couvrir les domaines de la paie, de la production et enfin celui des ventes tout en restant prisonniers de leur domaine technique. Cependant, on achetait un progiciel pour aussitôt l'adapter car on essaye de se claquer sur les besoins exprimés par les utilisateurs, au point qu'en quelque années le progiciel était complètement modifié et il devenait le plus souvent obsolète cinq à dix ans plus tard. Par conséquent, les progiciel devinèrent des briques applicatives qui répondent au besoin d'un domaine fonctionnel spécifique (finance, vente, achat...) mais également très spécifique à un type d'entreprise ou un type de secteur donné.

A la fin des années quatre-vingt, début des années quatre-vingt-dix, apparurent les premiers progiciels de gestion intégrée « les PGI¹ », appelé en anglais (ERP²) : Entreprise Ressources Planning.

¹ PGI : progiciels de gestion intégrée

²ERP: Entreprise Ressources Planning.

Les ERP sont maintenant présents dans l'industrie et dans la grande distribution, essentiellement dans les très grandes entreprises. Le marché des plus petites entreprises, le domaine de la finance et le secteur public commencent à être touché.

La conception et le développement des ERP ont été rendu possible par les évolutions technologiques : vitesse de calcul, technologie de réseau, système de gestion de base de données, stockage de données ... ce sont considérablement développés et améliorés

III. Définition

L'acronyme ERP signifie « Entreprise Resource Planning » traduit en français par progiciel de gestion intégré ou PGI. ERP est le terme le plus couramment utilisé.

Emanant d'un concepteur unique, un ERP est un logiciel qui permet de gérer l'ensemble des processus d'une entreprise intégrant l'ensemble de ses fonctions comme la gestion des ressources humaines, la gestion financière et comptable, l'aide à la décision, la vente, la distribution, l'approvisionnement, la production ou encore du e-commerce. Le principe d'un ERP est de construire des applications informatiques correspondantes à diverses fonctions citées précédemment de manière modulaire sachant que ces modules sont indépendants entre eux, tout en partageant une base de données unique et commune au sens logique. L'autre principe qui caractérise un ERP est l'usage de ce qu'on appelle un moteur de Workflow et qui permet, lorsqu'une donnée est enregistrée dans le système d'information (SI), de la propager dans les modules qui en ont l'utilité, selon une programmation prédéfinie. [2]

Ainsi, on peut parler d'ERP lorsqu'on est en présence d'un SI composé de plusieurs applications partageant une seule et même base de données, par le biais d'un système automatisé prédéfini et éventuellement paramétrable, un moteur de workflow.

Un ERP permet à une entreprise de mutualiser l'ensemble de ses systèmes d'informations et de ses processus opérationnels. Son objectif premier est de mettre à la disposition de tout le personnel d'une entreprise la gestion optimale de l'ensemble des données. Toutes les informations disponibles sont actualisées en temps réel et chaque utilisateur peut en connaître l'origine.

IV. Les ERP

IV.1 Caractéristiques d'un ERP [3]

- ERP est issu d'un concepteur unique.
- Une modification sur un module provoque une mise à jour en temps réel des autres modules liés.
- Un ERP garantit l'unicité des informations, grâce à la centralisation des données dans une base unique, accessible à tous les modules concernés ; il est facile de retrouver et d'analyser l'origine de chaque information.

- 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).
- Des logiciels évolutifs : Les ERP sont des logiciels évolutifs. Ils s'adaptent aux changements de l'entreprise. De nombreuses extensions ou services complémentaires viennent compléter une solution ERP classique.

IV.2 Le besoin d'utiliser un ERP

Les besoins d'utiliser les ERP sont nombreux, on cite [4]

- Faire des économies :
 - automatisation des tâches → réduction des coûts de maintenance.
 - réduction des inventaires.
 - identification et quantification des bénéfices.
 - réduction des coûts des matières premières.
 - en comptabilités : réduction de la période de fermeture de 80% .
 - réduction des coûts administratifs de 3.5% à 1.8 % des ventes.
- Améliorer l'organisation de l'entreprise et la performance de l'employé :
 - 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)
 - organise l'activité et la performance individuelle
- Pour les besoin globaux :
 - stockage en un seul endroit des données utiles et qualifiées.
 - garantie de la qualité des informations au sein de communautés.
 - système d'informations en temps réel comme support aux décisions d'affaires.
 - efforts de l'entreprise focalisés sur la réalisation des opportunités (plus besoin de déterminer qui a l'information véridique).
 - garantie de la qualité d'exécution des processus supportés par l'ERP.

IV.3 A qui sont destinés les ERP

Les ERP sont originellement destinées aux grandes entreprises ou multinationales du fait d'un coût important, cependant, le marché des ERP propose des solutions adaptées PME /PMI³. Certains éditeurs conçoivent un ERP uniquement pour ce type de structure. [4]

Enfin, il existe des ERP open source ce qui revient moins cher, puisqu'il y'a pas de coût de licence. En revanche, il faut inclure dans le calcul du coût d'acquisition total, les frais de maintenance et l'assistance technique.

³ PME /PMI : Petite et Moyenne Entreprise / Petite et Moyenne Industrie

IV.4 Principe de fonctionnement d un ERP

Un ERP contient généralement trois environnements de travail : [5]

- a. **Un environnement de développement** : qui permet d'adapter le progiciel standard à des besoins spécifiques de l'entreprise.
- b. **Un environnement de test** : dit encore environnement de recette qui permet de réaliser des simulations .ces simulations permettent de tester de nouveaux paramétrage et de vérifier le fonctionnement correct du progiciel par rapport à un processus de gestion donné (une vente, un achat, une sortie de stock).
- c. **Un environnement de production** : qui correspond au progiciel utilisé par les gestionnaires de l'entreprise.

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

IV.5 Classement des ERP

Les ERP sont classés par types, catégorie et groupe. [6]

IV.5.1. Le type

Tous les ERP ne présentent pas les mêmes caractéristiques de taille, de destination, d'usage et de risques. Ils sont donc répartis en types selon la nature de leur exploitation, classés en catégories d'après l'effectif du public et du personnel. Ils sont soumis à des dispositions générales communes ainsi qu'à des dispositions particulières qui leur sont propres issues du *Règlement de sécurité contre l'incendie et relatif aux établissements recevant du public*.

La typologie de l'établissement, qui correspond à son activité, est désignée par une lettre (article GN 1⁴ du règlement de sécurité incendie dans les ERP). Voici quelques établissements :

a) Établissements installés dans un bâtiment

- **J** : Structures d'accueil pour personnes âgées ou personnes handicapées.
- **L** : Salles d'auditions, de conférences, de réunions, de spectacles ou à usage multiple.
- **M** : Magasins de vente, centres commerciaux.
- **N** : Restaurants et débits de boisson.
- **O** : Hôtels et pensions de famille.
- **P** : Salles de danse et salles de jeux.

⁴ **Article GN 1** : Disposition générales : art. Gn règlement de sécurité contre les risques d'incendie et de panique dans les établissements recevant du public (E.R.P.) approuvé par arrêté du 25 juin 1980

- **R** : Établissements d'éveil, d'enseignement, de formation, centres de vacances, centres de loisirs sans hébergement.
- **S** : Bibliothèques, centres de documentation.
- **T** : Salles d'exposition à vocation commerciale.
- **U** : Établissements de soins.
- **V** : Établissements de divers cultes.
- **W** : Administrations, banques, bureaux.
- **X** : Établissements sportifs couverts.
- **Y** : Musées.

b) Établissements spéciaux

- **PA** : Établissements de Plein Air
- **CTS** : Chapiteaux, Tentes et Structures toile
- **SG** : Structures Gonflables
- **PS** : Parcs de Stationnement couverts
- **OA** : Hôtels-restaurants d'Altitude
- **GA** : Gares Accessibles au public (chemins de fer, téléphériques, remonte-pentes...)
- **EF** : Établissements flottants (eaux intérieures)
- **REF** : REFuges de montagne

IV.5.2 La catégorie

La catégorie est déterminée en fonction de l'effectif théorique susceptible d'être reçu dans l'établissement majoré de celui du personnel pour les établissements du 1er groupe et hors personnel dans les établissements du 2ème groupe.

La capacité, ou « **catégorie** », est désignée par un chiffre défini par l'article R123-19 du Code de la construction et de l'habitation:

- **1re catégorie** : au-dessus de 1 500 personnes ;
- **2e catégorie** : de 701 à 1500 personnes ;
- **3e catégorie** : de 301 à 700 personnes ;
- **4e catégorie** : 300 personnes et au-dessous, à l'exception des établissements compris dans la 5e catégorie ;
- **5e catégorie** : établissements accueillant un nombre de personnes inférieur au seuil dépendant du type d'établissement.

IV.5.3 Le groupe

Les ERP sont classés en 2 groupes :

Le 1er groupe comprend les ERP de 1ère, 2ème, 3ème et 4ème catégorie.

Le 2ème groupe comprend les ERP 5ème catégorie.

IV.6 Les avantages et les inconvénients des ERP [7]

a. Les avantages des ERP sont les suivants :

- pas de répétitions ou de saisies multipliées.
- logiciel disponible dans plusieurs langues.
- possibilité de travailler à partir de n'importe quelle devise.
- base de données unique et mise à jour en temps réel.
- aucune erreur lors de la transmission des informations.
- optimisation de la coordination entre les services.
- meilleure gestion des stocks et des Ressources humaines.
- Un gain de temps et d'argent avec l'ERP.
- La mise en place d'un projet ERP est aussi bénéfique sur la compétitivité de l'entreprise.

Voici les avantages PGI qui conduisent à une meilleure compétitivité :

✚ un gain de temps grâce à :

- la diminution des saisies.
- la diminution des erreurs dans les communications d'information.
- l'interconnexion entre les services.
- une meilleure réactivité.

✚ une augmentation du chiffre d'affaires, liée à :

- une meilleure gestion du CRM⁵.
- l'augmentation des ventes.
- une meilleure gestion des stocks et des approvisionnements.
- la motivation des salariés.

b. les inconvénients des ERP sont les suivants :

- coût élevé (cependant, il existe des ERP open sources (licence gratuite), les seuls coûts étant alors formation des utilisateurs et le service éventuellement assuré par le fournisseur du logiciel).
- Périmètre fonctionnel souvent plus large que les besoins de l'organisation ou de l'entreprise (le progiciel est parfois sous-utilisé).
- Lourdeur et rigidité de mise en œuvre.
- Difficultés d'appropriation par le personnel de l'entreprise.
- Nécessité parfois d'adapter certains processus de l'organisation ou de l'entreprise au progiciel.
- Nécessité d'une maintenance continue.
- Système unique donc risque de blocage de toute l'entreprise en cas de panne.

⁵ **CRM**: Customer Relationship Management.

IV.7 Quelques ERP gratuits

a) **Open Concerto [8]**

Open Concerto est un logiciel professionnel de gestion, ERP ouvert et modulaire, est adaptable facilement afin de fournir un outil simple et efficace.

- 6 bonnes raisons de choisir ce logiciel :
 - un logiciel ouvert (open source) et gratuit (pas de coût de licence).
 - un seul logiciel pour gérer l'ensemble de votre activité.
 - une ergonomie intuitive.
 - un logiciel professionnel adaptable à vos besoins.
 - fonctionne sous Windows, Linux et MacOS.
 - multi-société, multiutilisateur, mode monoposte ou multiposte en réseau.

b) **Neogia [8]**

Neogia est un progiciel de gestion intégrée, un ERP ou PGI spécifiquement conçu pour répondre aux besoins propres des PME/PMI. Elle recouvre la gestion financière, la collaboration logistique (y compris la gestion de production et la distribution), le CRM (y compris le marketing et la gestion des ventes et du service après-vente) ainsi que le commerce électronique.

D'autres modules sont en cours de développement) comme le suivi qualité et les services. Neogia est entièrement construit sous forme de serveur web, ce qui lui permet d'être installé très facilement en intranet, et le module d'e-commerce permet de créer un site marchand relativement facilement.

c) **OFBiz [8]**

- ❖ Apache OFBiz est un produit open source pour l'automatisation des processus d'entreprise qui comprend des éléments-cadres et des applications d'entreprise pour ERP (Enterprise Resource Planning), CRM (Customer Relationship Management), E-business / E-Commerce, SCM⁶, MRP⁷, MMS / EAM⁸, POS⁹.
- ❖ Apache OFBiz fournit un point de solutions d'entreprise fiables, sécurisées et évolutives fondation et départ. Utilisez-out-of-the-box, le personnaliser ou l'utiliser comme un cadre pour mettre en œuvre vos besoins d'affaires les plus difficiles. Avec OFBiz, vous pouvez commencer tout de suite sans les énormes coûts de déploiement et de maintenance des systèmes traditionnels d'automatisation de l'entreprise. Comme

⁶ **SCM** : Supply Chain Management.

⁷ **MRP** : Manufacturing Resource Planning.

⁸ **MMS / EAM** : Maintenance du système de gestion / Enterprise Asset Management.

⁹ **POS** : Point Of Sale.

vostra entreprise grandit, vous pouvez étendre les fonctionnalités pour répondre à vos besoins plus sophistiqués.

V. Le logiciel ERP

V.1 Logiciel ERP

L'ERP est un logiciel composé de plusieurs modules. [9]

Le logiciel ERP repose sur :

- Une architecture (*Architecture fonctionnelle, architecture technique, architecture modulaire*).
- un ou plusieurs modules.
- un serveur.

V.1.1 L'Architecture d'un ERP

- a. **l'architecture fonctionnelle** : l'architecture globale d'un ERP peut être considérée comme une pyramide à trois niveaux comme le montre le schéma suivant :

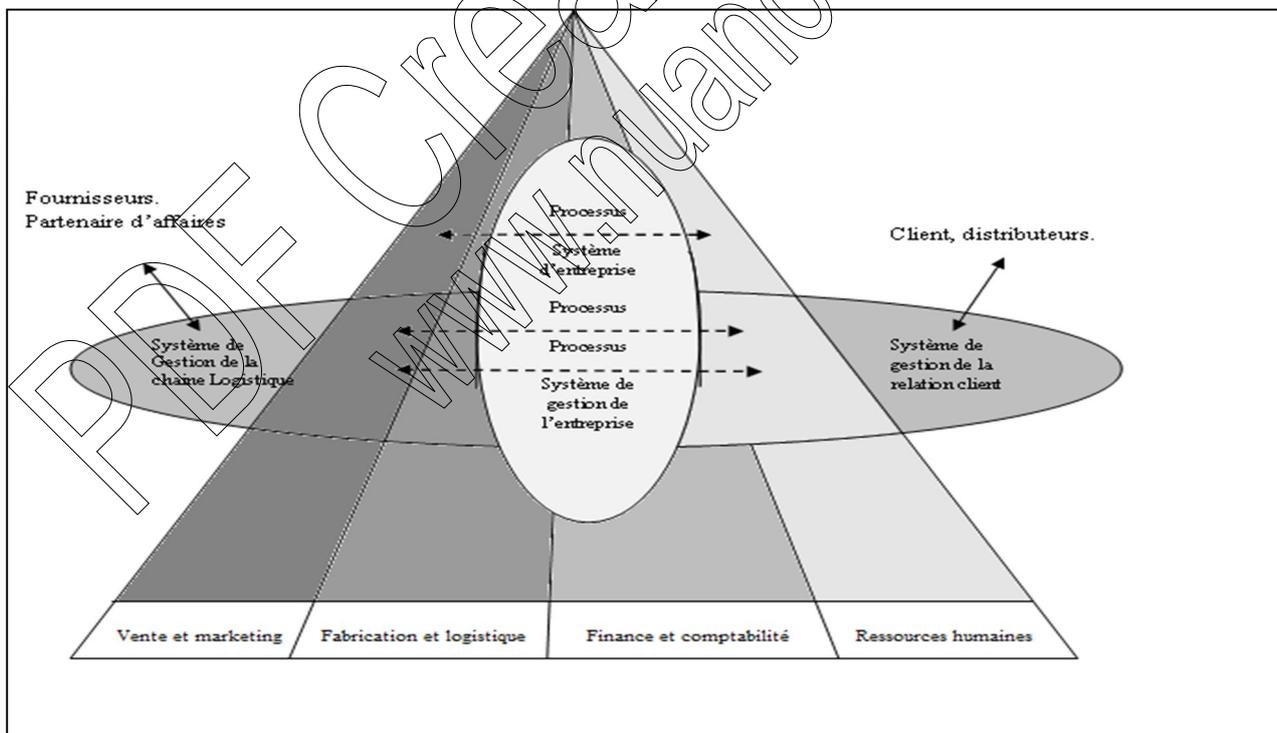


Figure .I.1. Architecture fonctionnelle d'un ERP [9]

Au plus haut de la pyramide résident les processus qui définissent le comportement des fonctionnalités de base que doit accomplir un ERP à savoir (la vente, la fabrication, les finances et les ressources humaines).

Au plus bas réside les systèmes de gestion des connaissances nécessaire pour le bon déroulement des processus du plus haut niveau.

Entre les deux se trouvent les systèmes de l'entreprise et les différents processus d'interaction de l'entreprise avec ses colorateurs externes (client, fournisseurs)

b. architecture technique : [10]

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

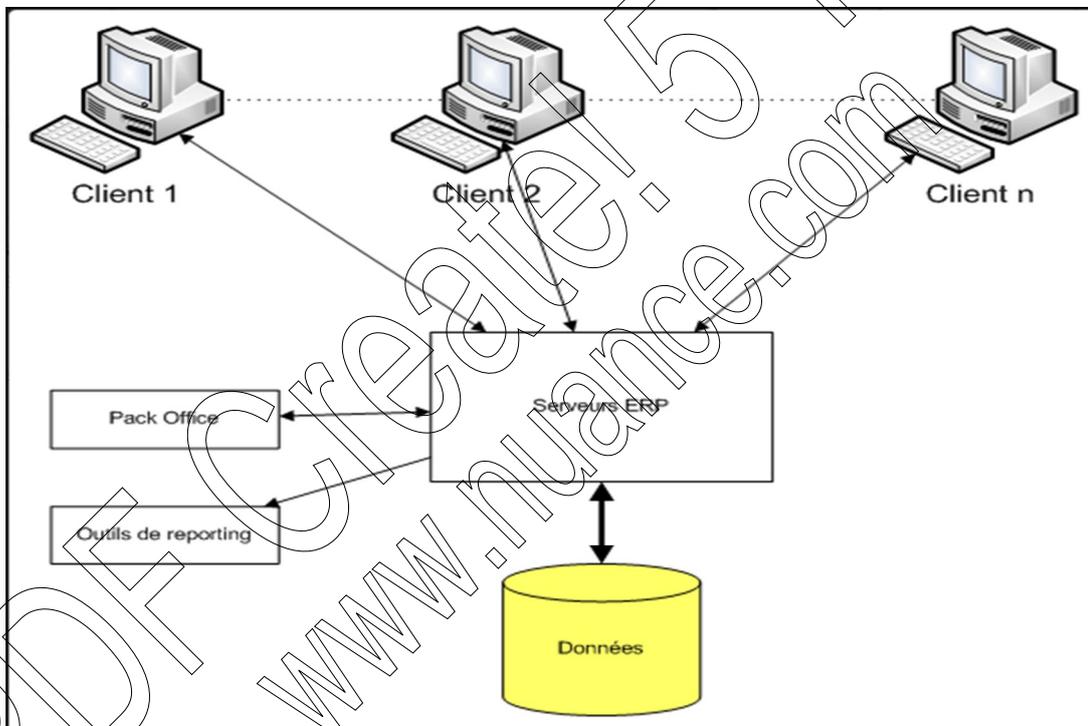


Figure .I.2. Architecture technique d'un ERP [10]

L'ERP est donc sur serveur. La majorité des ERP sont couplé à une base de données ORACLE [<http://www.oracle.com/fr/index.html>]. De plus, les ERP sont compatibles pack 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 compatible avec des outils de reporting.

c. architecture modulaire des ERP :

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. Les applications modulaires telles que les ERP permettent d'être sûr de la compatibilité des modules entre eux, ils s'imbriquent comme des blocs de Lego et fonctionnent ensemble (pas de vérification de compatibilité à effectuer).

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

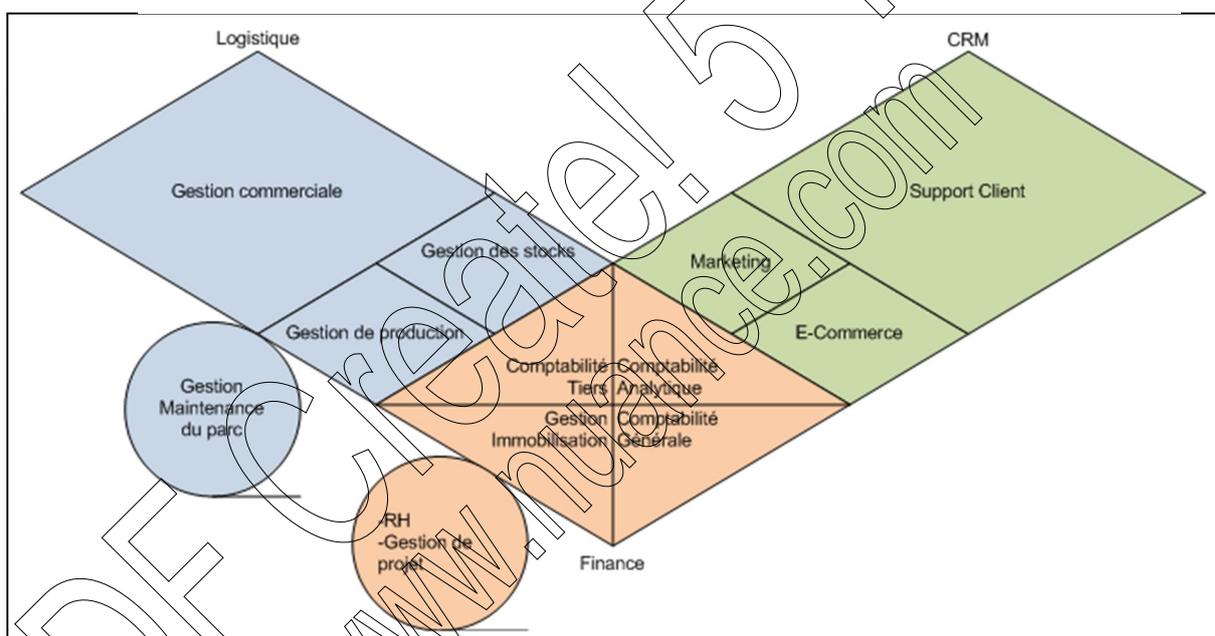


Figure .I.3. L'architecture modulaire qui intègre plusieurs modules retouchant aux grandes fonctions d'une entreprise [10]

V.1.2. Un module ERP [11]

Un module ERP est une fonctionnalité du logiciel ERP spécialisée dans la gestion d'une activité de l'entreprise. Ce module peut rassembler plusieurs fonctionnalités :

- compatibles avec les autres modules.
- connectées en permanence à la base de données.
- mutualisées et actualisées en temps réel.

Les ERP sont principalement destinés aux grandes entreprises ou multinationales du fait d'un coût important. Il existe des ERP open source qui revient moins cher, puisqu'il n'y a pas un coût de licence (ils sont gratuits) ; voila les modules ERP les plus courants dans le marché:

- ✓ gestion : des achats, des ventes.
- ✓ comptabilité : clients, fournisseurs.
- ✓ contrôle de gestion.
- ✓ Production.
- ✓ organisation du travail.
- ✓ stockage, archivage, inventaire.
- ✓ Logistique.
- ✓ gestion de projet.
- ✓ ressources humaines (paye, congés, pointage...).

V.1.2.1. Les différentes fonctionnalités des modules ERP

Voici un tableau récapitulatif contenant les différents modules ERP et leurs fonctionnalités.

MODULES ERP	FONCTIONNALITÉS
GESTION ACHATS	Transactions, écritures comptables, gestion des approvisionnements bons de commande et gestion de la production.
GESTION VENTES	Écritures comptables, devis, factures, CRM, e-commerce et e-procurement.
COMPTABILITÉ	Comptabilité multinationale, écritures comptables automatisées et gestion multi-devises.
STOCKAGE	Gestion approvisionnement, état des stocks en temps réel, SCM/GCL, mouvements des stocks et entreposage.
PRODUCTION	Gestion des besoins en fonction des commandes, régularisation des stocks et gestion des plannings de production.
GESTION DE PROJET	Gestion des plannings, affectation des tâches et suivi des projets en temps réel.
RESSOURCES HUMAINES	Gestion des plannings et gestion de paie.

Tableau. I.1 Tableau des différentes fonctionnalités des modules ERP [11]

V.1.3. Serveur ERP

Serveur ERP : Une base de données unique hébergée sur le serveur ERP ; elle est disponible pour tous

Un serveur ERP est un système similaire à un ordinateur. On y stocke l'ensemble des informations à mettre à la disposition des utilisateurs des différents modules ERP. Ce serveur est au cœur de l'architecture ERP puisqu'il est utilisé pour :

- administrer le réseau.
- gérer l'accès aux informations.
- gérer les connexions des différents utilisateurs.
- mutualiser les informations.
- assurer la traçabilité.

Son atout majeur : chaque utilisateur peut profiter de la même base de données actualisée en temps réel. **[11]**

V.1.3.1 Utilisation d'un serveur ERP à distance dans les Cloud

L'utilisation d'un serveur ERP est indispensable pour mutualiser les informations et les mettre en réseau.

Cependant, le marché des ERP est maintenant doté de solutions utilisant le Cloud computing donc des serveurs ERP à distance. Le Cloud¹⁰, communément appelé le nuage, est choisi par les entreprises dont les salariés ont besoin d'un accès au logiciel ERP en permanence depuis :

- tous les ordinateurs d'une société,
- un ordinateur portable,
- une tablette,
- un Smartphone.

¹⁰ **Le cloud** : Le cloud computing, ou l'informatique en nuage, est l'exploitation de la puissance de calcul ou de stockage des serveurs informatiques distants par l'intermédiaire d'un réseau, généralement Internet.

V.2 différents types de logiciels ERP

Voici les différents types de logiciels ERP que l'on trouve :

TYPES DE LOGICIELS ERP	
ERP GÉNÉRALISTES	<p>Un logiciel unique qui propose les fonctionnalités de base. Il répond à l'ensemble des besoins d'une entreprise et peut être implémenté sur une large majorité des secteurs d'activité.</p> <ul style="list-style-type: none"> • Non personnalisables.
ERP SPÉCIALISÉS	<p>Propose des fonctionnalités adaptées à chaque métier et surtout chaque secteur d'activité. Les secteurs d'activités couverts sont :</p> <ul style="list-style-type: none"> • la santé, la pharmacie, • l'agroalimentaire, • la chimie, la biologie, • le BTP¹¹, • le commerce, • la logistique.
ERP OPEN SOURCE	<p>Logiciels libres qui n'imposent pas l'acquisition d'une licence. Sont nettement moins chers, mais les services associés sont minimes.</p> <p>S'adaptent parfaitement aux besoins des PME, mais imposent de bonnes connaissances en informatique.</p>
ERP EN MODE SAAS	<p>ERP non commercialisés sous forme de logiciel, mais sous forme de service.</p> <p>La différence notable des ERP en mode SaaS¹² est la présence d'un serveur à distance qui permet une connexion à partir de n'importe quel ordinateur, mais aussi à partir des smartphones et des tablettes.</p>

Tableau. I.2 Tableau des différents types de logiciels ERP [12]

¹¹ **BTP** : Bâtiment et Travaux Public

¹² **SaaS** : Software as a Service : Le logiciel en tant que service ou Software as a Service (SaaS) est un modèle d'exploitation commerciale des logiciels dans lequel ceux-ci sont installés sur des serveurs distants plutôt que sur la machine de l'utilisateur.

IV. Conclusion

Les ERP ont des fonctions standards qui s'adaptent aux principales règles de gestion de entreprises (achats, ventes, comptabilité, paie,...). Cependant, chaque entreprise est unique et les ERP ne peuvent pas s'adapter à tous les cas particuliers d'entreprises. Toutes fois, la gestion d'une entreprise nécessite un ERP qui va répondre à ses besoins. En effet, nous avons fait appel à l'utilisation d'un ERP Open Source, ce dernier nous permet d'accéder aux programmes sources pour toute éventuelle modification. Dans notre cas, le choix du logiciel de gestion est porté sur Open ERP.

Après cet aperçu sur les ERP et leurs importances d'utilisation et afin d'assister les entreprises dans l'optimisation du travail en groupes et d'apporter la performance attendu, nous ferons appel aux Open Source dans le chapitre suivant.

PDF Create! 5.7.10
www.nuance.com

I. Introduction

Les différents ERP que nous avons sur le marché facilitent la gestion et le suivi des activités d'une entreprise. Pour notre cas ; nous intéressent à Odoo (Open Source), qui est un ERP gratuit et très utilisé par les entreprises. Odoo est une solution très riche fonctionnellement, ce qui permet de le déployer en tant qu'ERP d'une entreprise pour couvrir tous ses besoins, ou en tant qu'application métier sur un domaine précis comme la gestion des achats ou la gestion de la logistique d'un site e-commerce.

Odoo (ex-OpenERP) couvre tous les besoins, tels que : les ventes, achats, projets, logistique, stock, fabrication, facturation, ...

Odoo répond en effet aux besoins de tous types d'entreprises, quelles que soient leur tailles, pour toutes les industries et tous les secteurs d'activité.

II. Définition

Odoo, anciennement OpenERP et Tiny ERP³, est à la base un progiciel libre de gestion intégré comprenant de très nombreux modules permettant de simplifier la gestion d'entreprise dans son ensemble. De ce fait, il dispose d'un réseau de développeurs mondial qui est coordonné par la société belge Tinysprl¹. Ce mode de développement lui permet de disposer d'un grand nombre de modules (qui dépasse aujourd'hui 1800 modules) ; par la même occasion de plus de 2 millions d'utilisateurs à travers le monde, et d'une grande flexibilité.

C'est aussi un ERP Open Source offrant une multitude de possibilités d'intégration ; écrit en Python et repose sur un Framework orienté objet puissant ainsi que sur la base de données PostgreSQL²; Le logiciel est sous licence AGPL 3.07. [13]

III. Historique [14]

Odoo (ex-OpenERP), fondé en 2005 en Belgique par Fabien Pinckaers, combine à la fois la force d'un éditeur et celle d'une large communauté, comprenant des intégrateurs dans le monde entier, qui balisent l'ensemble des cas d'usages et fournit de précieux retours, notamment sous forme de modules réutilisables.

Ainsi, en 2005, il a commencé à développer le produit TinyERP¹, le logiciel qui allait changer le monde de l'entreprise.

en 2006, il a acheté le SorrySAP.com nom du domaine qui a été mis en attente pendant six ans, en attendant le bon moment pour l'utiliser. Trois ans plus tard il a estimé qu'il est préférable d'être un BigERP plutôt qu'un TinyERP³ et du coup il a renommé TinyERP³ à

¹ *Tinysprl* : une société d'édition basée en Belgique.

² *PostgreSQL* : PostgreSQL est un système de gestion de base de données relationnelle et objet (SGBDRO). C'est un outil libre

³ *TinyERP*: le logiciel libre de gestion d'entreprise le plus avancé sur le marché.

OpenERP; les choses ont commencé à évoluer. Il a élaboré des dizaines de modules pour OpenERP.

En 2010 Le groupe de travail est passé d'une société de services à une société éditrice de logiciels. Cela permettrait d'accroître les efforts dans les activités de recherche et développement.

Le 20 janvier 2011, OpenERP annonçait le lancement de la version 6.0 du logiciel, qui comprend une version à la demande (SaaS)⁴. Son approche modulaire permet aux utilisateurs de commencer avec une application, puis d'ajouter d'autres modules selon leurs besoins.

En décembre 2012, la version 7.0 d'OpenERP est lancée et peut être testée en ligne, téléchargée ou vue en version de démonstration. Mai 2014: OpenERP change de nom et devient Odoo ; et depuis, la suite d'applications de gestion d'entreprise est renommée Odoo. Cette suite d'applications permet aux entreprises de disposer d'une solution intégrée pour gérer les principales fonctions de l'entreprise. Son interface 100% Web, ses fonctions adaptées à l'entreprise en réseau en font l'un des ERP les plus moderne et performant du marché.

Été 2014, Odoo lance la version 8. Cette version enrichit principalement le logiciel de nouvelles applications qui font d'Odoo un logiciel allant au delà d'un ERP. Ces applications sont: Marketing (gestion d'événements, d'enquêtes de satisfactions, campagnes de mails auprès de la CRM⁵,...), CMS⁶ (construction d'un site internet - front-end lié au back-end - grâce au déplacement rapide et simple de "blocs" d'éditeurs), e-commerce (application pour vente en ligne),...

IV. Pourquoi choisir l'ERP Odoo ? [15]

L'ERP Odoo gère toutes les grandes fonctions de l'entreprise. Il propose aussi plus de 500 applications spécialisées qui facilitent la gestion et améliorent la productivité.

- **Gestion et création de site web** : Odoo V8 est à ce jour le seul ERP qui propose un CMS⁶ 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

⁴ **SaaS** : Software as a Service : Le logiciel en tant que service ou Software as a Service (SaaS) est un modèle d'exploitation commerciale des logiciels dans lequel ceux-ci sont installés sur des serveurs distants plutôt que sur la machine de l'utilisateur.

⁵ **CRM** : Customer Relationship Management : une méthode de planification commerciale, permettant d'encourager ou d'éviter la conception.

⁶ **CMS** : Content Management System : Un système de gestion de contenu ou SGC est une famille de logiciels destinés à la conception et à la mise à jour dynamique de sites Web ou d'applications multimédia.

dématérialisation des devis et factures. Fini les envois de PDF, avec Odoo vous envoyez un lien et votre 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 Adwords⁷ 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.
- **Améliorer la productivité avec Odoo** : Plus de 500 modules spécialisés permettent aux utilisateurs d'Odoo d'améliorer la productivité de l'entreprise. L'activation d'un module ne demande aucune intervention technique.

V. Principales Caractéristiques [16]

- **Complet** : OpenERP est une suite complète d'applications métiers. Englobe tous les besoins métiers de l'entreprise. Près de plus de 1500 modules disponibles.
- **Modulaire** : L'approche modulaire d'OpenERP, permet de démarrer avec une application, puis d'en ajouter d'autres pour les besoins de l'entreprise.
- **Flexible** : OpenERP est très flexible et offre la possibilité de modifier facilement de nombreux éléments (rapports d'extractions, écrans de saisie, workflows⁸ opérationnels, menus utilisateurs) afin de profiter d'un système parfaitement adapté aux besoins.
- **Installation en un clic** : OpenERP est construit comme un portail avec applications intégrées. De ce fait, quand un client lance OpenERP, il commencera avec une instance vide. Il pourra alors sélectionner les applications qu'il voudra utiliser à tout moment et en un seul clic.
- **Open Source** : OpenERP est en effet un logiciel Open Source sous licence AGPL⁹. Grâce à la communauté active, de nombreuses APIs¹⁰ ou programmes sont développés en permanence et permettent ainsi de faciliter l'exploitation et l'interaction avec certains outils métiers.

⁷ **Google Adwords** : Google AdWords ou AdWords est la régie publicitaire de Google créée le 23 octobre 2000 aux États-Unis. Celui-ci affiche des annonces ou bannières publicitaires, qui sont ciblées en fonction des mots-clés que tape l'internaute ou en fonction de son comportement de navigation.

⁸ **Workflows** : Un workflow est la représentation d'une suite de tâches ou opérations effectuées par une personne, un groupe de personnes, un organisme, etc. Le terme flow (flux) renvoie au passage du produit, du document, de l'information, etc., d'une étape à l'autre.

⁹ **AGPL 3.0** : Affero General Public Licence : une licence publique générale Affero. c'est une licence libre dérivée de la Licence publique générale GNU avec une partie supplémentaire couvrant les logiciels utilisés sur le réseau.

¹⁰ **APIs** : Application Programming Interface. En informatique, une interface de programmation est un ensemble normalisé de classes, de méthodes ou de fonctions qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. Elle est offerte par une bibliothèque logicielle ou un service web.

VI. Fonctionnalités d'Odoo (OpenERP) [17]

Odoo (OpenERP) est une solution ERP Open Source complètement personnalisable et modulaire. Chaque module permet de traiter une fonction d'une entreprise et peut être installé indépendamment.

Optimisation des ressources avec Odoo RH¹¹, amélioration des services client avec Odoo Ventes, pilotage de l'activité avec Odoo projets... Odoo couvre tous les domaines d'activités d'entreprise.

- **Gestion des achats** : création et suivi des commandes, gestion des informations sur les fournisseurs, contrôle du processus de réception des produits et vérification des factures fournisseurs.
- **Gestion de projets** : Suivi et gestion des projets en définissant des tâches pour l'exécution de projets à court terme ou la planification de différentes phases pour les projets sur le long terme.
- **Production** : Planification et contrôle d'une chaîne d'approvisionnement grâce aux fonctions du module de production. Personnalisation des données, configuration de la planification, gestion des ordres de fabrication.
- **Gestion d'entrepôt** : Odoo (OpenERP) a inventé le système de gestion des stocks à double entrée qui permet de gérer des besoins complexes : suivi des stocks des fournisseurs et des clients, traçabilité complète, liens avec la comptabilité, etc.
- **Point de vente** : Le point de vente tactile Odoo permet de gérer très facilement les ventes en boutique. Entièrement basé sur le Web, pas besoin d'installer ou déployer des logiciels. Il fonctionne en mode connecté et déconnecté pour pouvoir continuer à vendre en cas de perte de connexion internet.
- **Ressources Humaines** : Gestion du capital le plus important d'une entreprise : les Hommes. Avec ce module de gestion des ressources humaines, traçage des informations sur le personnel, les évaluations, la présence, les formations, les congés, les dépenses, le recrutement ou encore les départs.
- **Campagnes Marketing** : Les campagnes marketing peuvent aider à automatiser la création et la diffusion d'emails et à inciter les clients à contacter le bon département. Gain du temps et amélioration de la communication avec les clients sur le long terme.
- **Comptabilité & Finance** : Enregistrement des opérations comptables en quelques clics, gestion de l'ensemble des activités financières dans une plateforme unique.

¹¹ **RH** : Ressource Humaine : La gestion des ressources humaines (GRH) ex- gestion du personnel :est l'ensemble des pratiques mises en œuvre pour administrer, mobiliser et développer les ressources humaines impliquées dans l'activité d'une organisation.

- **CRM** : Le module CRM permet de suivre les leads¹² et opportunités. Personnalisation du cycle de vente, contrôle des prévisions commerciales, édition des tableaux de bord et pilotage des campagnes marketing pour amélioration de la performance commerciale.
- **Facturation** : Emission et paiement des factures en création et supervision de toutes factures des fournisseurs et clients et notes de crédits en quelques étapes simples.
- **Générateur d'Application** : Le générateur d'application d'Odoo permet de personnaliser chaque module pour l'ajuster aux besoins, directement depuis l'interface web et sans aucun développement.

VII. Les avantages et les inconvénients

A. Les avantages [18]

- Peu ou pas de coûts de licences ou de maintenance.
- Facile à déployer : vous pouvez l'installer dans autant de localisations souhaitées sans faire de suivi particulier
- Amélioration continue et en temps réel de la solution : tout le monde accède au code donc peut corriger les bugs et les erreurs sans attendre le prochain release
- Indépendance vis-à-vis de l'éditeur : si l'éditeur n'existe plus, la source et le code sont accessibles via la communauté et continuent à vivre.
- Pas de boîte noire : les équipes peuvent explorer le code pour réellement comprendre le fonctionnement de l'application

B. Les inconvénients [19]

- **Transparence** : Induite par l'accès au code source, la transparence est considérée comme une faiblesse en matière de sécurité.
- **Documentation rare** : Les communautés se concentrent davantage sur les développements que sur la rédaction de textes et de tutoriels.
- **Déficit de compétences** : Les informaticiens d'une entreprise en charge du support et de la maintenance doivent être formés lorsqu'une migration est envisagée.

¹² **Leads** : est un terme marketing désignant la détection de signaux d'intérêt émis par des clients potentiels (professionnels ou particuliers) pour les produits ou services d'une entreprise.

- **Risque de soft "orphelin"** : parfois, des projets open source s'arrêtent dans des organisations car des personnes clés partent, perdent du coup l'intérêt de développer la solution et passent à un autre projet
- **Ce n'est pas complètement gratuit** : il peut y avoir des coûts inattendus de mise en œuvre, de maintenance ou de support

VIII. Architecture d'OpenERP

OpenERP est basé sur une architecture client/serveur. Le serveur et le client communiquent via le protocole XML-RPC¹³. 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 http¹⁴, eux-mêmes sont encodés par le langage XML. [20] OpenERP est couplé à une base de données PostgreSQL². De plus, il est compatible au pack Open Office, et aussi avec des outils de reporting (ReportLab)¹⁵ pour produire des rapports en PDF¹⁶ ou en HTML.¹⁷

VIII.1 Architecture Client-serveur d'OpenERP

La logique d'open ERP est entièrement du côté serveur. La tâche du client se résume à demander les données 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 ou imprimer) il envoie cette action au serveur. Le serveur envoie alors la nouvelle action pour s'exécuter côté client. Un système d'OpenERP est formé de trois éléments principaux:

- Le serveur de base de données PostgreSQL, qui contient toutes les bases de données, dont chacune contient toutes les données et la plupart des éléments de la configuration du système d'OpenERP,
- Le serveur d'application d'OpenERP, qui contient toute la logique d'entreprise et assure que OpenERP fonctionne de manière optimale,
- Le serveur web, une application séparée appelée Open Object client Web, ce qui permet de connecter à OpenERP à partir du navigateur Web standard.

¹³**XML-RPC** : Extensible Markup Language (langage à balise extensible) **RPC** : un sigle qui peut signifier en informatique remote procedure call, un Protocol d'appel de procédure à distance.

¹⁴**Http** : HyperText Transfer Protocol : Protocol de transfert HyperText.

¹⁵**ReportLab** : est une librairie python pour générer des PDF. Cette version confirme le support de python.

¹⁶**PDF** : Portable Document Format : un langage de description de page créé par la société adobe système.

¹⁷**HTML** : HyperText Markup Language : est le format de données conçu pour représenter les pages web.

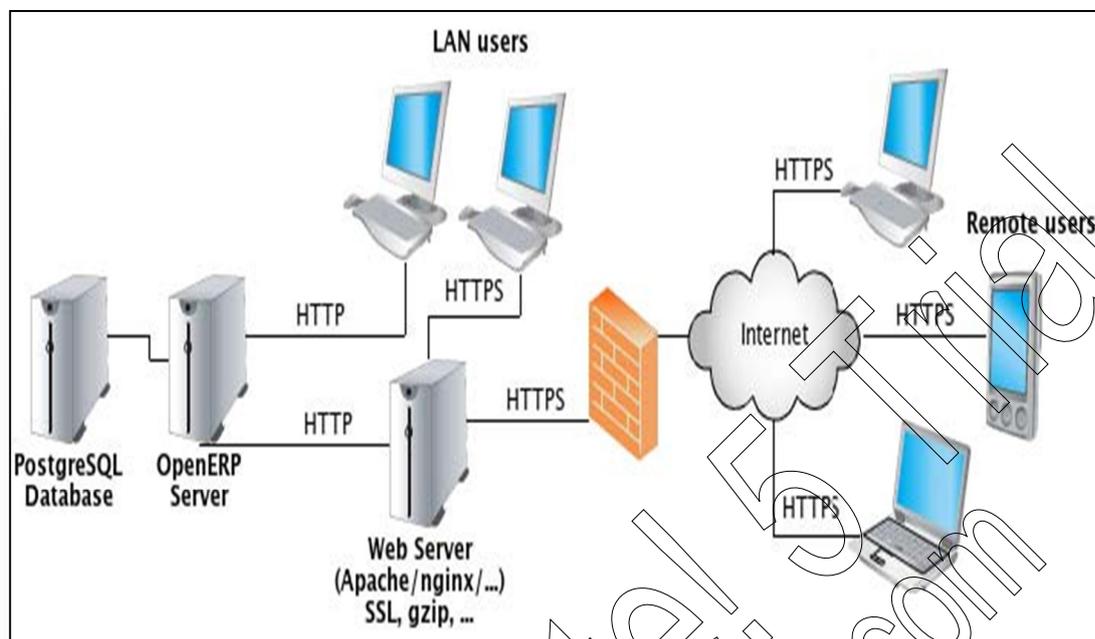


Figure .II.1. Architecture Client-serveur d'OpenERP [21]

La partie serveur est écrite en langage Python. Les différentes briques sont organisées en «modules». Un module est un dossier avec une structure prédéfinie contenant du code Python et des fichiers XML. Un module définit la structure de données, formulaires, rapports, menus, procédures, flux de travail, etc.

Le client est léger car il ne contient pas de logique d'entreprise (l'ensemble est embarqué dans le serveur d'application). Ainsi, l'ajout de nouveaux objets, comme les menus ou formulaires, le rend accessible à tout type de client graphique.

Framework OpenERP [21]

Le Framework de OpenERP est considéré comme le point le plus fort de ce progiciel de gestion intégré, il est indubitablement un avantage incontournable. Nous citons ci-dessous quelques points de force :

- **ORM** : Dès ses débuts, OpenERP s'est doté d'un ORM¹⁸, alors que cette technologie était encore très peu répandue. L'ORM permet d'avoir une couche d'abstraction par rapport à la base de donnée ; il gère les droits d'accès, les traductions et évite d'avoir à

¹⁸ **ORM** : Un mapping objet-relationnel (en anglais object-relational mapping ou ORM) est une technique de programmation informatique qui crée l'illusion d'une base de données orientée objet à partir d'une base de données relationnelle en définissant des correspondances entre cette base de données et les objets du langage utilisé. On pourrait le désigner par « correspondance entre monde objet et monde relationnel ».

écrire le code SQL dans lequel il faut refaire toutes les relations entre les tables avec des JOIN. OpenERP a fait évoluer son ORM au fur et à mesure.

Cependant, il reste possible d'utiliser des requêtes SQL dans le code d'OpenERP, par exemple pour certaines parties du code où les performances sont très importantes. Mais attention, le fait d'utiliser une requête SQL dans le code d'OpenERP contourne la gestion des droits d'accès d'OpenERP. L'ORM d'OpenERP ne fonctionne qu'avec PostgreSQL.

➤ **L'accès généralisé via les Webservices en XML-RPC et JSON-RPC.**

Tous les objets d'OpenERP (exemple d'objets : les commandes, les factures, les écritures comptables, ...) sont accessibles via les webservices en lecture, écriture, création et suppression. Mieux : toutes les fonctions d'OpenERP sont accessibles en web services. Cela signifie par exemple que n'importe quel clic sur un bouton de l'interface d'OpenERP (comme par exemple le bouton *Créer la facture* ou le bouton *Valider le mouvement de stock*) peut être fait depuis un web service.

➤ **Le moteur de Workflow** : OpenERP a un moteur de workflow simple mais généralement suffisant pour les besoins ERP des PME. Ce moteur de workflow est spécifique à OpenERP.

➤ **Serveur OpenERP** : Toute l'intelligence d'OpenERP est côté serveur ; il n'y a aucune intelligence côté client. Même à l'époque où le client Gtk¹⁹ était disponible (avant son abandon avec la sortie d'OpenERP 7.0), ce logiciel qu'on désignait sous le terme de "client lourd" (par opposition à l'interface Web qui est désignée sous le terme de "client léger") n'avait en réalité rien de lourd, car il se contentait de faire le rendu graphique d'un écran qui a été généré par le serveur OpenERP en XML.

➤ **L'interface Web** : L'interface Web d'OpenERP a été lancée avec OpenERP v4.2. Elle a été améliorée dans OpenERP version 5.0 et 6.0, puis elle a été entièrement réécrite dans OpenERP v6.1 sous l'impulsion du nouveau directeur R&D d'OpenERP, *Antony Lesuisse*, et son design et son ergonomie ont encore été grandement améliorés dans OpenERP version 7.0. Avec la nouvelle interface Web introduite dans OpenERP 6.1, l'interface Web est même devenue aussi rapide que le client Gtk¹⁹. Elle fonctionne bien sous Firefox, Chrome et Safari, ce qui en fait une solution d'accès à OpenERP multi-plateforme (Windows, Mac, Linux... mais aussi iPad ou tablette Android).

¹⁹ **GTK** : (The GIMP Toolkit) est un ensemble de bibliothèques logicielles, c'est-à-dire un ensemble de fonctions permettant de réaliser des interfaces graphiques. Cette bibliothèque a été développée originellement pour les besoins du logiciel de traitement d'images

VIII.2 Architecture modulaire d'OpenERP

Cette architecture n'est pas propre à open ERP. Elle est en fait partagée par tous les ERP. Il s'agit de la faculté de construire des applications informatiques de manière modulaire (modules indépendants entre eux) tout en partageant une base de données unique. Ceci apporte une importance significative puisque les données sont maintenant standardisées et partagées. Ce qui élimine les saisies multiples et évite l'ambiguïté des données de même nature. L'architecture modulaire d'openERP lui permet de couvrir plusieurs domaines, dont on cite : Finance et comptabilité, Les ressources humaines, la logistique et CRM⁵. Comme illustrés sur la figure suivante :

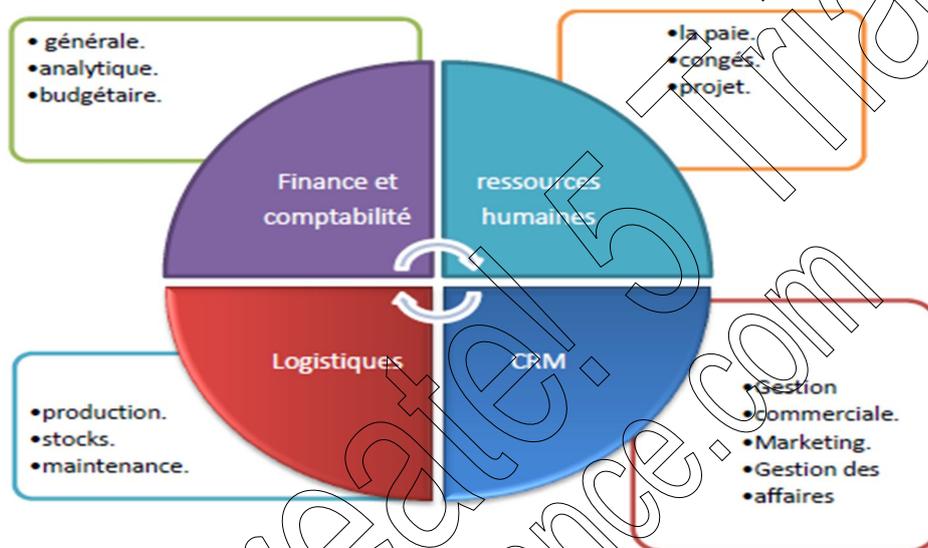


Figure .II.2. Architecture modulaire d'open ERP [20]

La compatibilité entre les modules qui composent Odoo est garantie par leur concepteur : par exemple la compatibilité du module achat avec le module stock est garantie, qui est lui-même compatible avec le module gestion de commandes. Les données sont désormais supposées standardisées et partagées, ce qui élimine les saisies multiples et évite l'ambiguïté des données multiples de même nature.

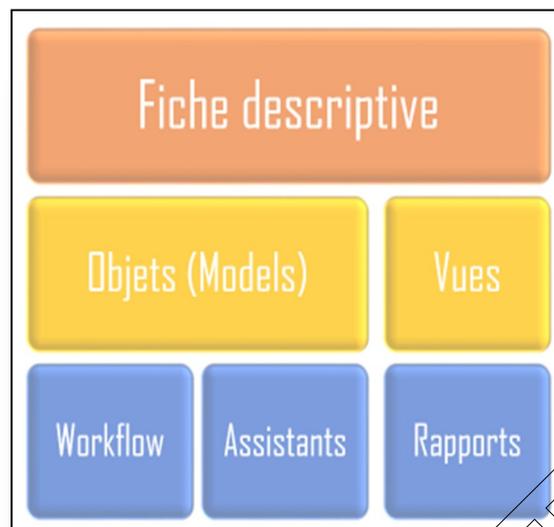


Figure II.3. Architecture en couche du module Odoo

Structure typique d'un module

Pour schématiser, on peut dire que la réalisation d'un module se fait en trois étapes :

- La création des fichiers d'initialisation du module.
- La création de l'objet en Python.
- La création des vues en XML.

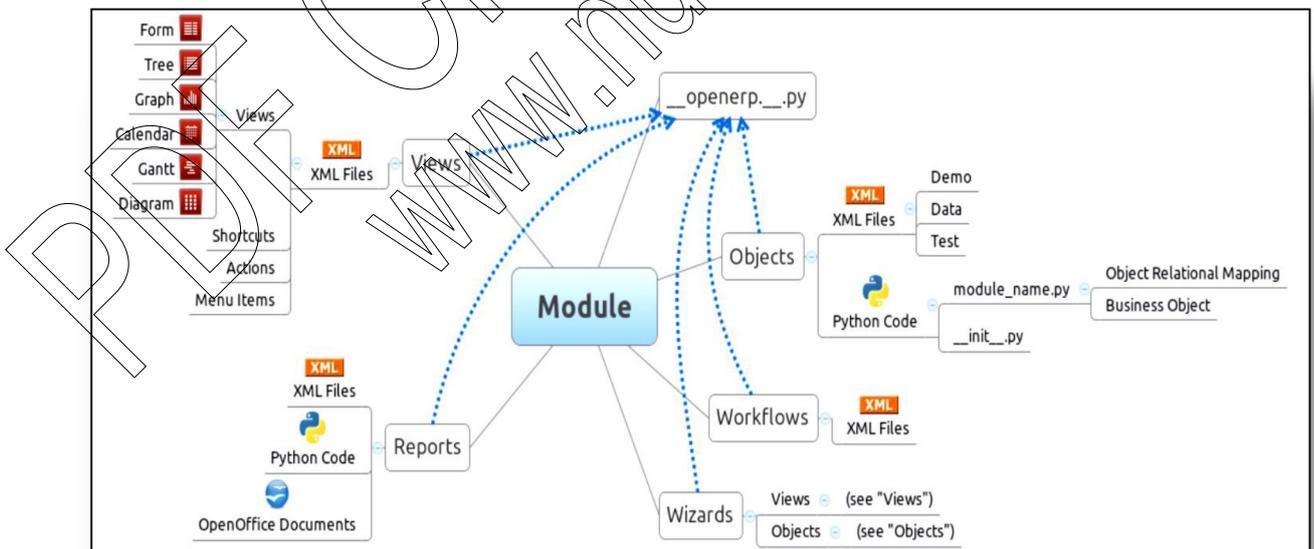


Figure .II.4. La structure d'un module sous Odoo[22]

Chaque module est contenu dans son propre répertoire, dans le répertoire d'installation du serveur `server\openerp\addons`. Voici les fichiers qu'un module peut contenir.

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

Tableau. II.1 Tableau représentant la Structure d'un module Odoe[22]

IX. Composants principaux d'Odoo

Les modules de l'ERP Odoo interagissent avec plusieurs autres composants et technologies, du fait, on distingue les composants suivants :

- **objets métier** : déclarés comme des classes Python qui étendent la classe *osv.Model*. La persistance de ces ressources est entièrement gérée par OpenERP ;
- **données** : les fichiers XML/CSV avec des métadonnées (vues et déclarations des flux de travail), les données de configuration (paramétrage des modules) et des données de démonstration (facultatives, mais recommandées pour tester, par exemple, des échantillons d'idées) ;
- **assistants** : formulaires interactifs utilisés pour aider les utilisateurs, souvent disponibles en actions contextuelles sur les ressources ;
- **rapports** : RML (format XML), MAKO ou OpenOffice modèles de rapports, qui seront fusionnés avec n'importe quel type de données de l'entreprise et généreront du HTML, ODT ou des rapports PDF.
- **Les workflows** : Odoo utilise systématiquement le moteur *workflow* qui permet, lorsqu'une donnée est entrée dans le système d'information, de la propager et d'offrir des vues logiques pertinentes dans tous les modules du système qui en ont besoin, selon une programmation prédéfinie. Ainsi, on peut parler d'ERP lorsqu'on est en présent dans un système d'information composé de plusieurs applications partageant une seule et même base de données.

X. Les applications Odoo [23]

Les applications d'Odoo officielles sont organisées en six groupes :

- Avant applications finaux : constructeur de site web, e-commerce.
- Application de gestion de vente : CRM¹⁸, point de vente, la citation constructive.
- Business operations applications : gestion de projet, de l'inventaire, de la fabrication, de la comptabilité et l'achat.
- Applications de marketing : événements, sondage, forum, chat en directe.
- Ressources humaines : applications annuaire des employés, réseau social d'entreprise, la gestion des feuilles, feuilles de temps.
- Application de productivité : business intelligence, la messagerie instantanée, les notes.

XI. Open Source : quels domaines d'application ? [24]

L'Open Source et le logiciel libre ont gagné toutes les couches informatiques du monde de l'entreprise. Des bases de données à la bureautique, ils s'installent sur les applications d'entreprises comme l'ERP (logiciel de gestion) et le CRM¹⁸ (logiciel de relation client).

a) Sur le poste client

➤ Système d'exploitation

Linux reste le système d'exploitation phare de l'Open Source. Il constitue le cœur de nombreuses distributions Open Source. Il reste toutefois assez peu déployé sur le poste de travail.

➤ Bureautique

La bureautique Open Source livre des alternatives de plus en plus performantes aux applications propriétaires :

- navigateur,
- suite bureautique,
- outils de messagerie instantanée...

b) Applications d'entreprise

➤ Gestion

Les progiciels de gestion intégrés libres (PGI et ERP) commencent à s'intervenir sur le segment des TPE et PME²⁰. Cependant, ce type d'application reste encore cantonné au monde propriétaire symbolisé par Oracle ou SAP²¹. Le faible coût des logiciels libres est souvent la première raison de leur adoption.

➤ Relation client

Comme les PGI, le logiciel relation client (CRM) Open Source perce les systèmes de petites entreprises. Les éditeurs sont de plus en plus nombreux sur ce segment, poussés par les applications hébergées (Saas) dont l'Open Source constitue le socle premier. Une application en mode Saas est une application installée sur des serveurs distants et accessible par les utilisateurs via un simple navigateur.

➤ Messagerie collaborative/Groupware²²

La gestion des e-mails, des agendas et des calendriers, ainsi que les espaces de travail collaboratif s'insèrent dans les systèmes existants grâce à leurs protocoles standards.

²⁰ **TPE et PME** : Très Petite Entreprise/Petite et Moyenne Entreprise

²¹ **SAP** : est une entreprise européenne qui conçoit et vend des logiciels, notamment des systèmes de gestion et de maintenance. Premier éditeur de logiciels en Europe et quatrième dans le monde. Son siège se trouve à Walldorf, en Allemagne. Son produit le plus connu est le progiciel de gestion intégrée SAP ERP.

²² **Groupware** : est un logiciel qui permet le travail collaboratif d'un groupe de personnes quelle que soit la position géographique de chacun des membres. Il permet en particulier l'échange de documents et la communication entre les membres.

➤ **E-learning**

Reposant sur des technologies web standards et libres, les solutions Open Source d'apprentissage à distance reposent sur des technologies web standards et libres. Elles ont trouvé naturellement leur essor à travers la démocratisation des usages d'Internet.

➤ **Décisionnel**

La Business Intelligence est le secteur de l'Open Source qui enregistre la progression la plus importante ces dernières années. Outils de génération de rapport, d'analyse prédictive, moteur décisionnel... les technologies libres du décisionnel profitent d'un monde où le reporting se banalise.

➤ **Sécurité**

VPN, firewall, anti-spam... L'Open Source parvient également à toucher le monde très critique de la sécurité informatique.

➤ **Administration de sites internet**

Les systèmes de gestion documentaire et de gestion de contenu web et entreprise Open Source proposent des fonctions sur mesure d'administration de sites Internet. Les entreprises apprécient leurs performances qui rivalisent avec les solutions dites « propriétaires ».

c) Infrastructure

Les couches basses des systèmes (celles qui supportent techniquement les applications métiers de l'entreprise) ont été les premières à démontrer l'efficacité et la robustesse de l'Open Source :

- les bases de données Open Source soutiennent la plupart des sites Internet,
- les distributions Linux pour serveurs rivalisent avec Microsoft,
- le n° 1 des serveurs web (Apache HTTP) est 100 % Open Source,
- les serveurs d'applications s'intègrent aux systèmes d'entreprises grâce au respect des standards.

XII. Odoo 8 [25]

Odoo 8 conquiert aujourd'hui de nouveaux territoires, au delà de l'ERP. Cette nouvelle version inclut un outil disruptif de construction de site web, un e-commerce intégré, un point de vente pour iPad/Android, un moteur de Business Intelligence et plusieurs applications de marketing entrant (inbound marketing). Aucun autre produit n'offre aujourd'hui un tel niveau d'intégration de manière native. C'est la raison pour laquelle OpenERP a été renommé en Odoo pour la version 8. Il n'y a plus de comparaison possible avec les PGI traditionnels.

Les améliorations principales de la version 8 sont, par ordre de priorité :

- **Amélioration ergonomiques et de productivité** : Odoo 8 est plus rapide, plus facile à utiliser et à configurer
- **Améliorations des applications existantes** : 732 points d'amélioration couvrant la plupart des applications
- **Nouveau CMS et outil de construction de site web**
- **Nouvelles applications** : Point de vente avec support matériel complet, applications de marketing, un nouveau système de gestion de stocks (WMS²³), et un CMS⁶,
- **La couche frontale de toutes les applications** : offres de job en ligne, réservation d'événements, outil de devis en ligne avec signature en ligne, etc.
- **Améliorations techniques** : refactoring du framework (nouvelle API²⁴)
- **Des modules et fonctionnalités supprimés** : cette nouvelle version est le résultat de 1432 tâches et 12.500 *jours*hommes* de développement pendant 18 mois. Un effort gigantesque a été fait sur le nettoyage du code existant, résultant en une augmentation de seulement 17.000 lignes de code ajoutée pour toutes ces fonctionnalités.

XIII. Conclusion

Dans ce chapitre nous avons défini odoo comme un progiciel adapté à toutes les entreprises. Très modulable et sans coût de licence, il répond aux besoins d'un déploiement rapide à un coût maîtrisé. Nous avons vu aussi les Caractéristiques, l'architecture et le domaine d'application.

²³ **WMS** : Web Map Service ou WMS est un protocole de communication standard qui permet d'obtenir des cartes de données géo référencées à partir de différents serveurs de données.

²⁴ **APIs** : Application Programming Interface. En informatique, une interface de programmation est un ensemble normalisé de classes, de méthodes ou de fonctions qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. Elle est offerte par une bibliothèque logicielle ou un service web.

I. Introduction

Après avoir défini certaines notions de base concernant les ERP et Odoos en général, ce présent chapitre entamera une présentation de l'organisme d'accueil qui est l'Office de Promotion et de la Gestion Immobilière O.P.G.I., comme étant un établissement public national à caractère social.

II. Présentation de l'organisation d'accueil

II.1. Qu'est ce que l'O.P.G.I. ?

L'Office de Promotion et de la Gestion Immobilière est un établissement public national à caractère social, doté de la personnalité morale et de l'autonomie financière. L'O.P.G.I. peut exercer ses activités sur l'ensemble du territoire national, chargé dans le cadre de la mise en œuvre de la politique sociale de l'Etat, de promouvoir le service public en matière de logements, notamment pour les catégories sociales les plus démunies. L'office assure aussi l'entretien, la maintenance, la réhabilitation et la restauration des biens immobiliers.

II.2. Historique de l'O.P.G.I.

Suite à la décentralisation de l'ex O.P.H.L.M. (Office Public d'Habitation à Loyer Modéré) de la wilaya d'ALGER entre 1969 et 1985, l'O.P.G.I. de Tizi-Ouzou a été créé par l'ordonnance N° 76/143 du 23 Octobre sous forme d'un Etablissement Public à caractère Administratif (E.P.A.), sa mission essentielle consistait en la réalisation dans le cadre du plan national de développement des programmes publics d'habitat.

Son statut juridique a connu par la suite des changements qui lui sont consacrés notamment par le décret 83/200 qui est venu lui conférer le statut d'E.P.L. (Etablissement Public Local) sur lequel s'exerçait une double tutelle : tutelle locale et tutelle centrale. Cette situation a réduit le champ d'action de l'office, limitant ainsi son évolution positive.

L'office est érigé en E.P.I.C. (Etablissement Public Industriel et Commercial) qui est doté de la personnalité morale et de l'autonomie financière suite au changement de statut à travers les décrets exécutifs N° 91/147 du 12 Mai 1991 et 93/08 du 28 Janvier 1993. L'O.P.G.I. est réputé commercial dans ses relations avec les tiers.

Le décret 93/08 du 02 Janvier 1993 a permis à l'O.P.G.I. de prendre des participations dans des sociétés civiles immobilières, des groupements et entreprises publiques, économiques et éventuellement créer des filiales.

L'arrêté ministériel N° 43/ SPM /MHAB du 19 Octobre 1998 a porté sur l'organigramme des O.P.G.I. érigés en 04 départements : Département Administratif et Financier (D.A.F.), Département des Gestions Patrimoines (D.G.P.), Département Maitrise d'Ouvrage (D.M.O.) et Unité de Réalisation et de Réhabilitation (U.R.R.) avec toujours la possibilité de créer des filiales.

II.3. Objectifs de l'O.P.G.I.

L'O.P.G.I a été créée par ordonnance N°76/143 du 23 octobre 1976 sous forme d'établissement public à caractère administratif (E.P.A.), son objectif essentiel constituait en la réalisation dans le cadre du plan national de développement des programmes publics d'habitats.

La tutelle des Offices de Promotion et de Gestion Immobilière peut prendre des participations dans les sociétés civiles immobilières, des groupements et entreprises publiques économiques et éventuellement créer des filiales.

Ils peuvent en outre, en vertu d'une concession ou d'une convention passée avec l'Etat, les collectivités locales ou tout autre organisme public, réaliser ou entreprendre toute opération d'aménagement, de rénovation et d'amélioration du patrimoine immobilier en vue de son maintien en état permanent d'habitabilité.

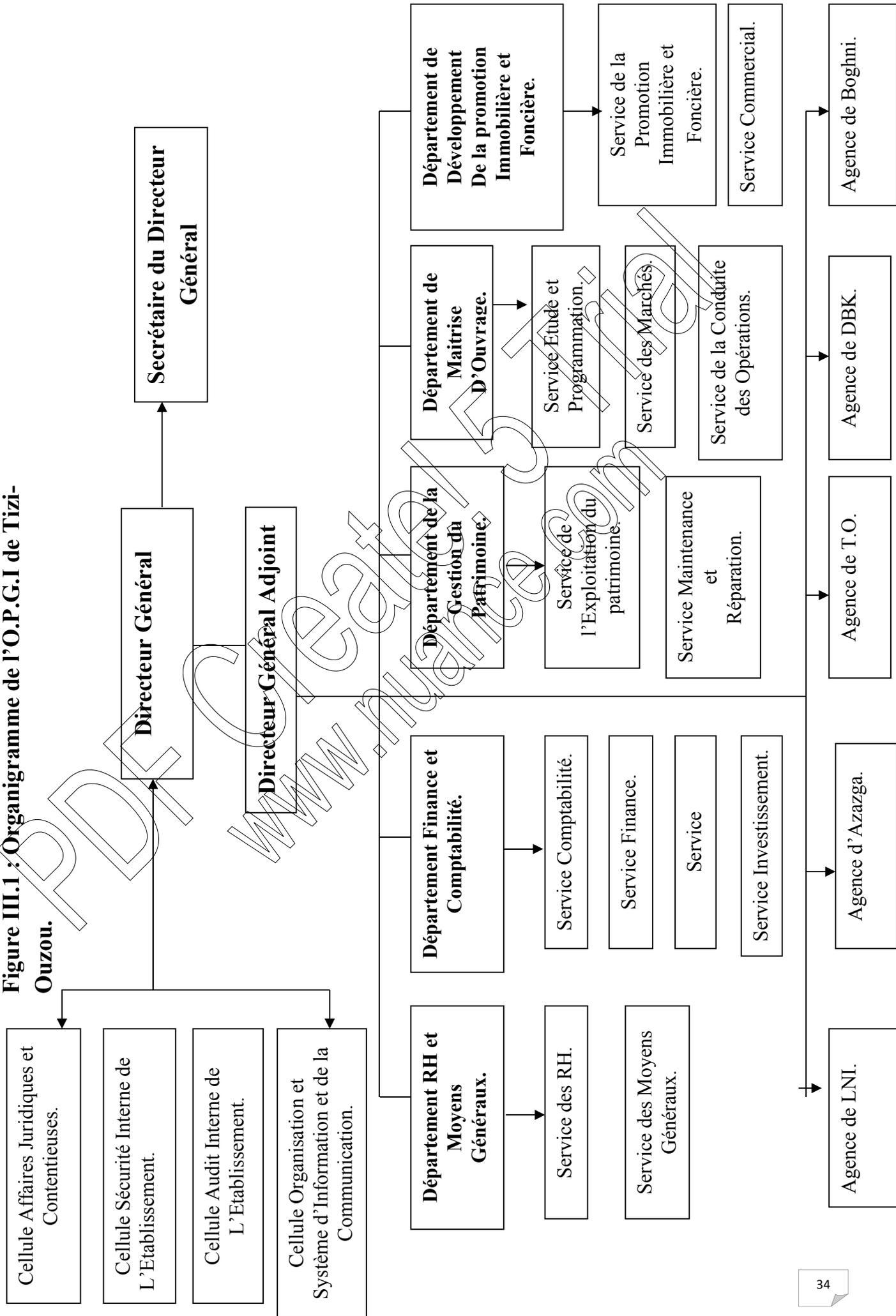
II.4. Missions de l'O.P.G.I.

L'O.P.G.I. est chargé de promouvoir le service public en matière d'immobilier : logements et locaux. Ses activités se résument comme suit :

- La location et/ou la cession des logements et locaux à usage professionnel, commercial et artisanal ;
- Le recouvrement des loyers et des charges locatives ainsi que des produits de la cession des biens immobiliers qu'il gère ;
- La préservation des immeubles et leurs dépendances en vue de leur maintien en état permanent d'habitabilité ;
- L'établissement et la tenue à jour de l'inventaire des immeubles constituant le parc immobilier dont il assure la gestion ainsi que le contrôle du statut juridique des occupants des logements et locaux ;
- L'organisation et la coordination de toutes les actions permettant une utilisation optimale des ensembles immobiliers qu'il gère ;
- Assurer selon des conditions particulières la gestion de tous les biens qui leur ont été ou qui leur seront rattachés dans le cadre d'unicité des règles de la gestion du patrimoine immobilier ;
- Toutes autres actions entrant dans le cadre de la gestion immobilière.

L'organigramme de l'O.P.G.I. de Tizi-Ouzou est représenté dans la figure III.1 suivante :

Figure III.1 : Organigramme de l'O.P.G.I. de Tizi-Ouzou.



II.5. Organisation générale de L'O.P.G.I. de Tizi-Ouzou

II.5.1. Organisation structurelle de L'O.P.G.I.

II.5.1.1. Direction Générale

- La Direction Générale est composée de :
 - Directeur Général ;
 - Directeur Général Adjoint.

Le Directeur Général est assisté :

- D'une cellule de Sécurité Interne de l'Office ;
- D'une cellule Affaires Juridiques et Contentieuses ;
- D'une cellule Audit Interne de l'établissement ;
- D'une cellule Organisation et Système d'Information et de Communication.

Les grandes lignes des missions confiées au Directeur Général Adjoint et les assistants chargés des différentes cellules sont résumées dans les points suivants :

1. Directeur Général Adjoint

Ses principaux rôles sont :

- Assister le Directeur Général et le seconder dans toutes les missions ;
- Analyser et évaluer les activités des structures de l'office ;
- Suivre et coordonner l'exécution des programmes de travail ;
- Contrôler et parapher tout courrier soumis à la signature du directeur ;
- Développer les activités liées au bon fonctionnement de l'office ;
- Evaluer les besoins en ressources humaines et en logistique de l'office.

2. Cellule Sécurité Interne

A pour missions principales :

- Veiller à la sécurité des lieux ;
- Veiller aux relations avec les services extérieurs en matière de sécurité ;
- Veiller à l'application des consignes et des mesures de sécurité entreprises au niveau de l'office ;
- Elaborer et concevoir les plans d'interventions et d'évacuations en cas d'incendie ou sinistre.

3. Cellule Affaires Juridiques et Contentieuses

A pour missions :

- De représenter l'office dans toutes les affaires où il est parti prenante auprès des instances concernées ;
- D'analyser les textes officiels et de veiller à leur application ;
- D'assister l'ensemble des structures dans le traitement des affaires juridiques ;
- D'étudier tout litige opposant à des personnes morales et physiques et de contrôler les actions et opérations visant à la passation de tout contrat.

4. Cellule Audit Interne

A pour missions :

- D'effectuer les missions d'audits couvrant l'ensemble des services dans le cadre d'un planning semestriel ;
- D'élaborer les rapports d'audits qu'elle transmet au Directeur Général ;
- D'élaborer les documents permettant au Directeur Général de disposer d'un instrument de contrôle de gestion efficient.

5. Cellule Organisation et Système d'Information et de Communication

A pour missions principales :

- Le développement et la généralisation de l'informatique au sein de l'office ;
- L'élaboration du schéma directeur informatique en assurant sa mise en œuvre ;
- Charger la distribution de la documentation au niveau des départements ;
- Collecter l'information et les données pour la préparation des synthèses et bilans et représenter le Directeur Général vis-à-vis des médias (radio, journaux etc.).

II.5.1.1. Agences

L'O.P.G.I. dispose de cinq agences qui dépendent de la Direction Générale à travers les daïra de la Wilaya de Tizi-Ouzou comme suit :

- ✓ Agence de Tizi-Ouzou, Nouvelle Ville ;
- ✓ Agence d'Azzazga ;
- ✓ Agence de l'Arabaa Nath Irathen ;
- ✓ Agence de Boghni ;
- ✓ Agence de Draa Ben Khedda.

Elles sont chargées essentiellement de :

- ✓ L'administration, la coordination, le développement et le contrôle de l'activité de l'agence ;
- ✓ La prise des mesures actives requises pour atteindre des objectifs fixes.

A ce titre, elles veillent :

- ✓ A l'exécution des missions et des actes de gestion qui lui sont confiés et notamment ceux relatifs au recouvrement des loyers et des charges locatives ;
- ✓ A la surveillance et au gardiennage du patrimoine ;

- ✓ A l'entretien et à la maintenance des locaux des immeubles et aussi le suivi de chantier à travers les projets implantés dans la zone géographique couverte par l'agence.

II.5.1.2. Structuration des départements

La structuration des départements est répartie comme suit :

1. Département des Ressources Humaines et Moyens Généraux (D.R.H.M.G.)

Ce département comporte deux (02) services :

1. Service Gestion des Ressources Humaines.
2. Service des Moyens Généraux.

Le chef de département a pour missions principales :

- ✓ De gérer le personnel mis sous son autorité ;
- ✓ De planifier, coordonner, organiser, diriger et contrôler le travail de son département en collaboration avec ses chefs de services, ainsi que de gérer les relations avec les autres départements en matière d'évaluation des emplois par rapport aux besoins de l'office ;
- ✓ D'élaborer des bilans d'activités du département ;
- ✓ D'assurer l'interface avec les autres départements et structures ;
- ✓ De mettre à jour les registres légaux en assurant une bonne communication en matière de discipline ;
- ✓ De veiller à la gestion des moyens logistiques (humain et matériel).

1.1. Service Gestion des Ressources Humaines

Ce service a pour missions principales :

- ✓ L'évaluation des emplois en adéquation avec les besoins de l'office ;
- ✓ L'élaboration et l'exécution des plans de formation ;
- ✓ La gestion des carrières des travailleurs ;
- ✓ Le suivi du mouvement du personnel ;
- ✓ La prévention et le règlement des conflits collectifs de travail ;
- ✓ Le traitement des affaires contentieuses en concertation avec le service juridique et contentieux de l'office ;
- ✓ L'entretien des relations avec le partenaire social et l'inspection du travail ;
- ✓ L'instauration de la gestion prévisionnelle des ressources ;
- ✓ Le contrôle de la paie ;
- ✓ La collecte et le classement des textes réglementaires et autres documents liés à la gestion des ressources humaines et l'élaboration des notes explicatives en vue de leur application ;
- ✓ L'élaboration des rapports et des bilans d'activités.

Le service de la Gestion des Ressources Humaines est composé de cinq (05) sections :

- 1) Section Gestion Administration du Personnel ;
- 2) Section Paie ;
- 3) Section Formation ;
- 4) Section de l'Action Sociale ;
- 5) Section Œuvres Sociales.

1.2. Service des Moyens Généraux

Les missions du chef du service Moyens Généraux se résument comme suit :

- ✓ Tenir à jour le registre du carburant et lubrifiants et assurer leur contrôle et leur distribution ;
- ✓ Contrôler les entrées et les sorties du matériel et fournitures de l'office ;
- ✓ Mettre à jour toutes les fiches de stocks pour arrêter l'inventaire ;
- ✓ Etablir les services faits de toutes les factures d'achats et contrôler leur conformité ;
- ✓ Classer les textes réglementaires et autres documents liés à la gestion des Moyens Généraux ;
- ✓ Elaborer les rapports et les bilans d'activités de la fonction.

Le service des Moyens Généraux est composé de six (06) sections :

- 1) Section Approvisionnement ;
- 2) Section Investissement ;
- 3) Section Gestion des Stocks ;
- 4) Section Parc Roulant ;
- 5) Section Hygiène ;
- 6) Section Archives et Documentations.

2. Département Finances et Comptabilité (D.F.C.)

Le D.F.C. est chargé de superviser les travaux réalisés par ses services au nombre de quatre (04) :

- 1) Service Finances ;
- 2) Service Comptabilité ;
- 3) Service Investissement ;
- 4) Service Recouvrement.

A ce titre, il doit veiller et s'assurer :

- ✓ Du suivi et de la tenue journalière des opérations de trésorerie de l'office ;
- ✓ De la tenue de la comptabilité par la passation des opérations et des écritures comptables selon les normes réglementaires ;
- ✓ Du suivi des recouvrements des créances que l'office détient sur ses clients et du contrôle financier des régies.

Le principal rôle du D.F.C. est d'assurer avec le Directeur Général toutes les dépenses liées à l'exploitation et à la gestion financière des opérations dont l'office est chargé de la maîtrise d'ouvrage.

2.1. Service Finance

Le service Finance a pour missions principales :

- ✓ D'assurer l'interface avec l'administration fiscale ;
- ✓ De veiller à la bonne gestion de caisse ;
- ✓ De suivre en collaboration avec les caissiers le fond de roulement ;
- ✓ D'effectuer des contrôles inopinés de caisse et régies ;
- ✓ De tenir à jour les mouvements de fond opérés sur le compte de l'office.

Il comporte trois (03) sections :

- 1) Section Trésorerie ;
- 2) Section Budget ;
- 3) Section Fiscalité.

2.2. Service Comptabilité

Ce service est chargé de veiller :

- ✓ A l'engagement des opérations liées à l'activité et au patrimoine de l'office et de contrôler la conformité et la légalité des documents émanant des structures de l'office et de l'extérieur ;
- ✓ A l'établissement du bilan annuel et des annexes réglementaires ;
- ✓ A analyser les comptes et procéder aux redressements appropriés, et à la mise à jour des livres légaux ;
- ✓ A l'établissement des états de rapprochement de banque ;
- ✓ A la sauvegarde des archives pour la durée légale.

Le service Comptabilité est composé de deux (02) sections :

- 1) Section Comptabilisation des Investissements et des Stocks ;
- 2) Section Opérations Diverses.

2.3. Service Investissement

Le service Investissement est chargé :

- ✓ De l'inventaire et du suivi des investissements de l'office ;
- ✓ Du suivi du parc immobilier ainsi que des remboursements auprès des organismes financiers ;
- ✓ Du suivi des biens de l'office, de l'amortissement du matériel de transport, du mobilier et du matériel.

Il est composé de deux (02) sections :

- 1) Section Suivi Financier des Investissements ;
- 2) Section Suivi comptable des Investissements.

2.4. Service Recouvrement

Ce service a pour missions :

- ✓ De veiller au recouvrement des recettes quelque soit l'origine ;
- ✓ De signaler toute anomalie relevée au niveau des antennes ;
- ✓ De fournir les éléments nécessaires à l'établissement des bilans.

Il est composé de deux (02) sections :

- 1) Section Recouvrement ;
- 2) Section Contrôle des Régies.

3. Département Gestion et Maintenance du Patrimoine (D.G.M.P.)

Le chef de département de la Gestion du Patrimoine étant le responsable des opérations effectuées au sein de ses services a pour missions principales :

- ✓ De coordonner les actions de ses différents collaborateurs ;
- ✓ D'assurer l'interface avec les autres structures de l'office (D.M.O., D.A.G., D.F.C. et Régie) et éventuellement les structures externes ;
- ✓ De motiver et d'animer le personnel en place ;
- ✓ De trouver des solutions aux problèmes posés ;
- ✓ De faire toutes les recommandations jugées appropriées ;
- ✓ De contrôler la cohérence globale des travaux effectués.

Ce département comporte deux (02) services :

- 1) Service Exploitation du Patrimoine ;
- 2) Service Maintenance et Réparation.

3.1. Service Exploitation du Patrimoine

Le service Exploitation du Patrimoine a pour missions :

- ✓ De superviser les travaux réalisés par ses sections ;
- ✓ De veiller à la gestion locative des logements ;
- ✓ D'établir les fiches locatives et les contrats de location ;
- ✓ De mettre à jour les dossiers et de suivre les décisions d'attribution ;
- ✓ De repartir, coordonner et suivre les travaux relevant de sa structure ;
- ✓ D'assurer la transmission au service du fichier et quittancement, les éléments nécessaires pour l'établissement des quittances de paiement, des mensualités et des charges.

Il comporte quatre (04) sections :

- 1) Section Exploitation du Patrimoine ;
- 2) Section Suivi du dossier de la propriété et l'amélioration du cadre de vie ;
- 3) Section Cession ;
- 4) Section Commerciale.

3.2. Service Maintenance et Réparation

Ce service a pour missions :

- ✓ D'assurer la réception des doléances des occupants des cités ;
- ✓ D'assurer la collecte des données techniques à la confection des programmes d'intervention en matière de maintenance ;
- ✓ De veiller à la prise en charge des préjudices causés au patrimoine ;
- ✓ De tenir l'inventaire physique des immeubles et des réparations ;
- ✓ D'assurer l'interface avec les structures techniques pour les interventions en matière de maintenance ;
- ✓ De coordonner et suivre les opérations relatives aux travaux de sous traitance ;
- ✓ De travaillé en étroite collaboration avec le service recouvrement des loyers.

Il comporte trois (03) sections :

- 1) Section Entretien et Inspection ;
- 2) Section Travaux et Réparation ;
- 3) Section Maintenance (grosse réparation).

4. Le Département Maitrise d'ouvrage (D.M.O.)

Le D.M.O. a pour principales missions de superviser le travail réalisé par ses services, qui seront au nombre de trois (03), dans le cadre de l'activité technique.

- 1) Service Etude et Programmation ;
- 2) Service Marchés ;
- 3) Service de la Conduite des Opérations.

A ce titre, il doit veiller :

- ✓ A l'élaboration des programmes de travail ;
- ✓ Au suivi du déroulement des objectifs tracés ;
- ✓ A la proposition et à la mise en œuvre des actions susceptibles d'améliorer l'efficacité des différents services ;
- ✓ A assurer le suivi et la coordination de toute opération de maitrise d'ouvrage ;
- ✓ A assurer l'interface avec les départements.

4.1. Service Etudes et Programmation

Le service Etudes et Programmation est chargé de :

- ✓ L'établissement des fiches techniques relatives aux projets ;
- ✓ La prospection des terrains ;
- ✓ La préparation des dossiers techniques ;
- ✓ Le lancement des avis d'appel d'offres ou des concours à la sélection des bureaux d'études ;
- ✓ L'établissement des contrats et marchés en collaboration avec les bureaux d'études ;
- ✓ De suivre la préparation des études techniques ;

Il comporte deux (02) sections :

- 1) Section Programmation ;
- 2) Section Etudes.

4.2. Service des Marchés

Ce service est chargé de :

- ✓ La vérification des clauses contractuelles ;
- ✓ La vérification des dossiers administratifs des contractants ;
- ✓ La vérification des avenants ayant des incidences financières sur le projet en s'appuyant éventuellement sur la nomenclature actualisée des coûts des articles;
- ✓ Le contrôle des ordres de service ;
- ✓ La préparation des documents liés aux marchés pour les commissions des marchés ;
- ✓ Fournir les éléments nécessaires à l'établissement des bilans.

Il comporte deux (02) sections :

- 1) Section Vérification ;
- 2) Section Ordonnancement.

4.2. Service de la Conduite des Opérations

Le service de la Conduite des Opérations est chargé :

- ✓ Du suivi sur site des opérations engagées (conjointement avec le bureau d'études ou indépendamment du suivi effectué par ce dernier) ;
- ✓ De l'établissement des comptes rendus de visite de chantiers ;
- ✓ De la proposition de toute action de gestion des contrats (résiliation ; reprise etc.) ;
- ✓ D'approuver et légaliser les plannings de travaux des entreprises et de toute proposition de changement dans la réalisation des chantiers ;
- ✓ D'évaluer les moyens mis en place par l'entreprise ;
- ✓ De certifier les situations de travaux ;
- ✓ De proposer et approuver les variantes complémentaires éventuelles ;
- ✓ De l'évaluation éventuelle de sous-traitances ;
- ✓ De participer à la réception provisoire et définitive de travaux.

Ce service comporte :

- 1) Des chefs de projets ;
- 2) Des chargés d'études.

5. Département de Développement de la Promotion Immobilière et Foncière (D.P.I.F.)

Ce département est chargé :

- ✓ De la prospection foncière (promotion Immobilière, L.S.P. ; Social) ;
- ✓ Du montage financier des projets ;
- ✓ Du traitement des dossiers techniques des programmes projetés
- ✓ Du suivi des projets de promotion immobilière ;
- ✓ D'assurer l'interface avec les partenaires (C.N.L., Banque etc.) ;
- ✓ De la communication et vulgarisation du dispositif des formules L.S.P. et L.P.A.

Il est composé de deux (02) services :

- 1) Service de la Promotion Immobilière et Foncière ;
- 2) Service Commercial.

5.1. Service de la Promotion Immobilière et Foncière

Ce service est chargé des démarches relatives aux prospections des terrains :

- ✓ Assister aux différents choix de terrain ;
- ✓ Engagement de géomètre expert pour prise en charge des plans ;
- ✓ Recherche relative à la nature juridique des terrains ;
- ✓ Constitution des dossiers de transfert entre les différents organismes.

5.2. Service Commercial

Le service Commercial est chargé de :

- ✓ La communication et de la vulgarisation du dispositif régissant les formules LSP et promotionnelles ;
- ✓ La prise en charge des dossiers postulants ;
- ✓ La prospection du marché du logement ;
- ✓ La préparation des portes ouvertes et autres.

III. Conclusion

Dans ce chapitre, nous avons présenté l'organisme d'accueil, ses départements, ses services et ses différents objectifs et missions.

Les étapes et détails de la conception et la réalisation de notre application au sein de l'O.P.G.I. et plus précisément dans ses services sera l'objet du prochain chapitre.

I. Introduction

Tout processus de développement d'application ou système informatique doit suivre une méthode ou une démarche bien déterminée. Et ceci pour une meilleure organisation et une bonne maîtrise de travail.

Pour se faire, nous allons suivre dans ce chapitre une démarche qui s'appuie sur le langage UML, conçu pour la visualisation, la spécification et construction des systèmes logiciels. UML commence par une phase d'analyse pour spécifier les différents acteurs du système ainsi que leurs besoins, leurs tâches et leurs scénarios. Ensuite, une phase de conception qui va s'appuyer sur les résultats de la phase d'analyse, et sa modélisation s'appuie sur l'élaboration des diagrammes de séquences et de classes.

PDF Create! 5 Trial
www.nuance.com

II. Les étapes d'analyse et de la conception :

Pour mener à bien notre application, nous allons procéder comme suit :

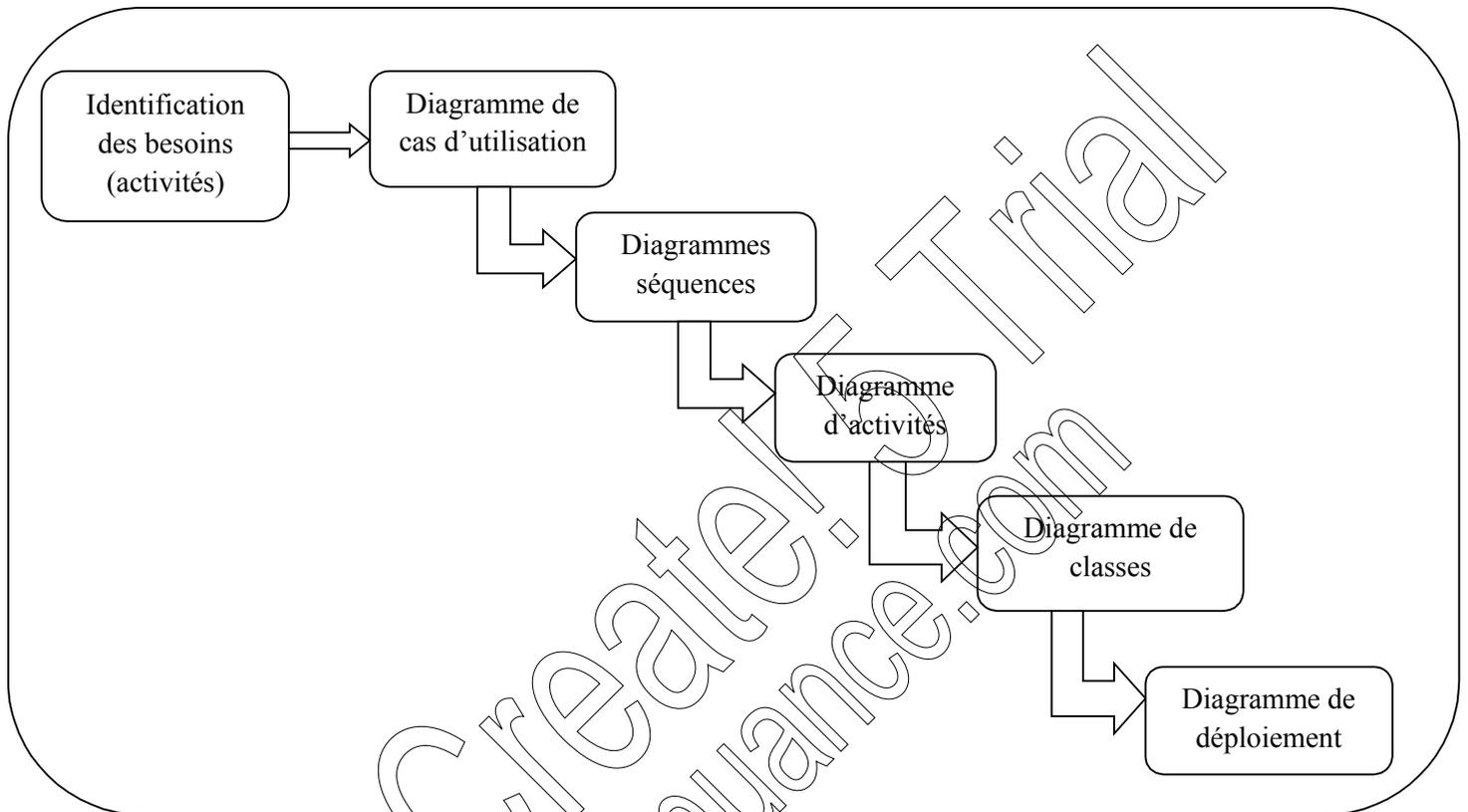


Figure .IV.1. Cycle de modélisation de Module de la gestion de document

III. Phase d'analyse

III.1 Introduction

Dans cette partie nous allons spécifier d'une manière bien détaillée et claire le module de la gestion de document, pour cela nous allons procéder en premier lieu à la détermination d'une manière globale ce qui se trouve dans le champ de l'application.

III.2 Définition des besoins :

Notre projet porte sur le développement d'un module qui gère la fonction de gestion de document sous Odoo

Etant un organisme chargé de promouvoir le service public en matière d'immobilier : Logements et locaux (O.P.G.I), est à la tête de plusieurs projets de construction répartis sur tout le territoire de la wilaya de Tizi-Ouzou et gère un volume important d'informations.

L'objectif principal de ce travail est de développer un module qui permettra de gérer des documents autour de chaque projet. Nous avons opté pour une solution open source implémentée avec le langage python, XML, PostgreSQL.

Sachant que la gestion des projets de construction fait participer différents collaborateurs, une application multiutilisateurs et multi rôles s'impose. Les événements et les étapes de chaque projet sont accompagnés par des documents officiels.

Notre application devra également fournir la possibilité de numérisation et de référencement des documents officiels ayant trait à la gestion des différents événements et étapes d'un projet

De ce fait notre application va permettre :

- ✓ Le traitement rapide de l'information ;
- ✓ Le diminution du travail manuel ;
- ✓ La minimisation des erreurs de travail ;
- ✓ A tout moment on peut consulter un document ;
- ✓ Garder trace de toute procédure ou traitement ;
- ✓ Une sécurité sûre pour l'information (droits d'accès).

III.3 Identification des acteurs

Pendant le stage que nous avons effectué au sein de l'organisme : l'Office de Promotion et de la Gestion Immobilière O.P.G.I., nous avons procédé, en premier lieu à la localisation des centres d'activités potentiels de la gestion de documents, et en second lieu à l'identification des acteurs qui seront le futur utilisateur de l'application, ces acteurs sont :

Administrateur(ADMIN) : C'est l'administrateur de notre application, son rôle principal se focalise sur l'ajout des comptes utilisateur. Il peut à tout moment consulter les documents concernant les services des différents départements

L'ensemble des employés des différents services (utilisateurs) :

C'est l'un des futurs utilisateurs de notre application, son rôle est de suivre son travail actuel (manuel) tel que l'ajout des documents, la recherche des documents tout en utilisant les différents moyens que notre système peut lui offrir.

III.4 Diagramme de contexte :

Le diagramme de contexte est un modèle conceptuel de flux qui permet d'avoir une vue globale des interactions entre le système et l'environnement extérieur. Il permet aussi bien délimiter le champ d'étude. Pour notre cas le diagramme de contexte est donné par la figure suivante :

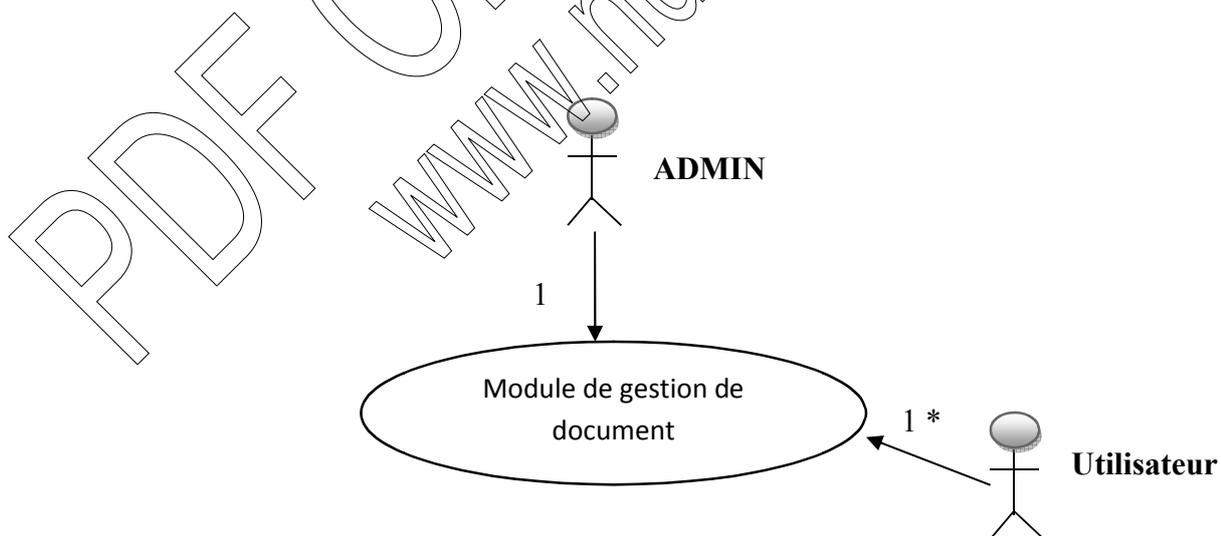


Figure .IV.2. Le diagramme de contexte

III.5 Identification des cas d'utilisation

a- Cas d'utilisation :

Les cas d'utilisations sont des outils formels qui permettent de consigner et d'exprimer les interactions et les dialogues des utilisateurs (acteurs) avec le système. Ils spécifient ce que le système devrait effectuer sans préjuger de la manière.

Dans ce qui suit nous allons identifier les différents cas d'utilisations de module gestion des documents

b- Spécification des cas d'utilisation :

Le cas d'utilisation (use case) est un ensemble de séquences d'actions réalisées par le système et qui produit un résultat visible pour un acteur particulier. Il permet de structurer les besoins des utilisateurs et les objectifs correspondants. Il identifie les utilisateurs (acteur) et leurs interactions avec celui-ci. Il permet aussi de classer les acteurs et structurer les objectifs de l'application.

Les uses cases permettent de décrire ce que le système devra faire sans spécifier comment le faire

Dans ce point, nous allons définir les cas d'utilisation de notre application et les acteurs, ainsi les diagrammes de cas d'utilisations relatifs à chacun de ces derniers

Pour notre cas, nous avons pu dégager les cas d'utilisation suivants :

Acteur	Cas d'utilisation
ADMIN (administrateur)	<ul style="list-style-type: none"> - S'authentifier. - Gestion des comptes utilisateur <ul style="list-style-type: none"> o Ajouter un compte utilisateur o Dupliquer un utilisateur o Modifier un compte utilisateur o Désactiver un compte utilisateur o Rechercher un utilisateur - Rechercher/Consulter un document
User (employés)	<ul style="list-style-type: none"> - S'authentifier. - Gestion des documents <ul style="list-style-type: none"> o Ajouter un document o Dupliquer un document

	<ul style="list-style-type: none"> ○ Rechercher un document ○ Modifier un document ○ Supprimer un document - Rechercher/Consulter un document
--	---

Tableau IV.1 : Les différents cas d'utilisation

III.6 Spécification des tâches et des scénarios :

Un scénario représente une succession particulière d'enchaînement étant l'unité de description de séquence d'actions. Un cas d'utilisation contient en général un scénario nominal et plusieurs scénarios alternatifs (qui se terminent d'une façon normale) ou d'erreurs (qui se terminent en échec).

Dans ce qui suit, on présente dans des tableaux, les différents scénarios associés à chaque acteur :

➤ **Administrateur**

Tâches	Scénarios
T01 : S'authentifier	<p>S01: Le système affiche un formulaire d'authentification.</p> <p>S02 :l'administrateur saisit son nom utilisateur et son mot de passe, puis il clique sur « connexion ».</p> <p>S03: Le système vérifie la validité des informations reçues et affiche la page d'accueil de l'espace demandé, sinon il retourne un message d'erreur.</p>
T02:Ajouter un nouveau compte utilisateur	<p>S04 :l'administrateur sélectionne le menu utilisateur et clique sur « create » pour ajouter un nouveau compte utilisateur</p> <p>S05 : Le système affiche un formulaire d'ajout.</p>

	<p>S06 : l'administrateur remplit le formulaire et clique sur « <i>save</i> »</p>
<p>T03: désactiver un compte utilisateur</p>	<p>S07:l'administrateur sélectionne le menu utilisateur</p> <p>S08: Le système lui retourne la liste des « <i>utilisateur</i> » ainsi leurs informations.</p> <p>S09: l'administrateur sélectionne la ligne qui contient l'utilisateur à désactiver et clique sur « <i>more</i> » → « <i>délete</i> »</p>
<p>T04: modifier un compte utilisateur</p>	<p>S10:l'administrateur sélectionne le menu utilisateur</p> <p>S11: Le système lui retourne la liste des « <i>utilisateur</i> » ainsi leurs informations.</p> <p>S12: l'administrateur sélectionne la ligne qui contient l'utilisateur à modifier et clique sur « <i>edit</i> »</p>
<p>T05: rechercher un compte utilisateur</p>	<p>S13:l'administrateur sélectionne le menu utilisateur</p> <p>S14:l'administrateur choisit le type de la recherche en sélectionnant le type de la vue</p> <p>S15: → Vue « <i>List view</i> » : Le système retourne une barre de recherche</p> <p>S16: l'administrateur saisit le nom de l'utilisateur</p> <p>S17: → Vue « <i>form view</i> » : Le système retourne un numéro séquentiel</p> <p>S18: l'administrateur parcourt les utilisateurs un par un</p> <p>S19: Le système retourne l'utilisateur ainsi leurs informations.</p>

Tableau IV.2 : Résumé des scénarios par tâches d'un administrateur

➤ *Utilisateur*

Tâches	Scénarios
T06 : S'authentifier	<p>S20: Le système affiche un formulaire d'authentification.</p> <p>S21 :l'utilisateur saisit son nom utilisateur et son mot de passe, puis il clique sur « connexion ».</p> <p>S22: Le système vérifie la validité des informations reçues et affiche la page d'accueil de l'espace demandé, sinon il retourne un message d'erreur.</p>
T07:Ajouter un nouveau document	<p>S23: l'utilisateur sélectionne le menu document et clique sur « create » pour ajouter un nouveau document</p> <p>S24: Le système affiche un formulaire d'ajout.</p> <p>S25 : L'administrateur remplit le formulaire et clique sur « save »</p>
T08: supprimer un document	<p>S26: l'utilisateur sélectionne le menu Document</p> <p>S27: Le système lui retourne la liste des documents ainsi leurs informations.</p> <p>S28: l'utilisateur sélectionne la ligne qui contient le document à désactiver et clique sur « more » → « delete »</p>
T09: dupliquer un document	<p>S29: l'utilisateur sélectionne le menu Document</p> <p>S30: Le système lui retourne la liste des documents ainsi leurs informations.</p> <p>S31: l'utilisateur sélectionne la ligne qui contient le document à désactiver et clique sur « more » → « duplicate »</p>

<p>T10: modifier un document</p>	<p>S32: l'utilisateur sélectionne le menu Document</p> <p>S33: Le système lui retourne la liste des « <i>documents</i> » ainsi leurs informations.</p> <p>S34: l'utilisateur sélectionne la ligne qui contient le document à modifier et clique sur « <i>edit</i> »</p>
<p>T11: rechercher/consulter un document</p>	<p>S35:l'utilisateur sélectionne le menu « <i>document</i> »</p> <p>S36: l'utilisateur choisit le type de la recherche en sélectionnant le type de la vue</p> <p>S37: → Vue « <i>List view</i> » : Le système retourne une barre de recherche</p> <p>S38 : l'utilisateur saisit le nom du document</p> <p>S39: → Vue « <i>form view</i> » : Le système retourne un numéro séquentiel</p> <p>S40: l'utilisateur parcourt les documents un par un</p> <p>S41: Le système retourne le document ainsi ces informations.</p>

Tableau IV.3 : Résumé des scénarios par tâches d'un utilisateur

III.7 Diagramme de cas d'utilisation

Bien souvent, les utilisateurs ne sont pas des informaticiens. Il leur faut donc un moyen simple d'exprimer leurs besoins. C'est précisément le rôle des diagrammes de cas d'utilisation qui permettent de recueillir, d'analyser et d'organiser les besoins, et de recenser les grandes fonctionnalités d'un système. Il s'agit donc de la première étape UML d'analyse d'un système.

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en unités cohérentes, les cas d'utilisation, ayant un sens pour les acteurs.

Il ne faut pas négliger cette première étape pour produire un logiciel conforme aux attentes des utilisateurs.

Les diagrammes de cas d'utilisation de notre application sont représentés par les figures ci-dessous :

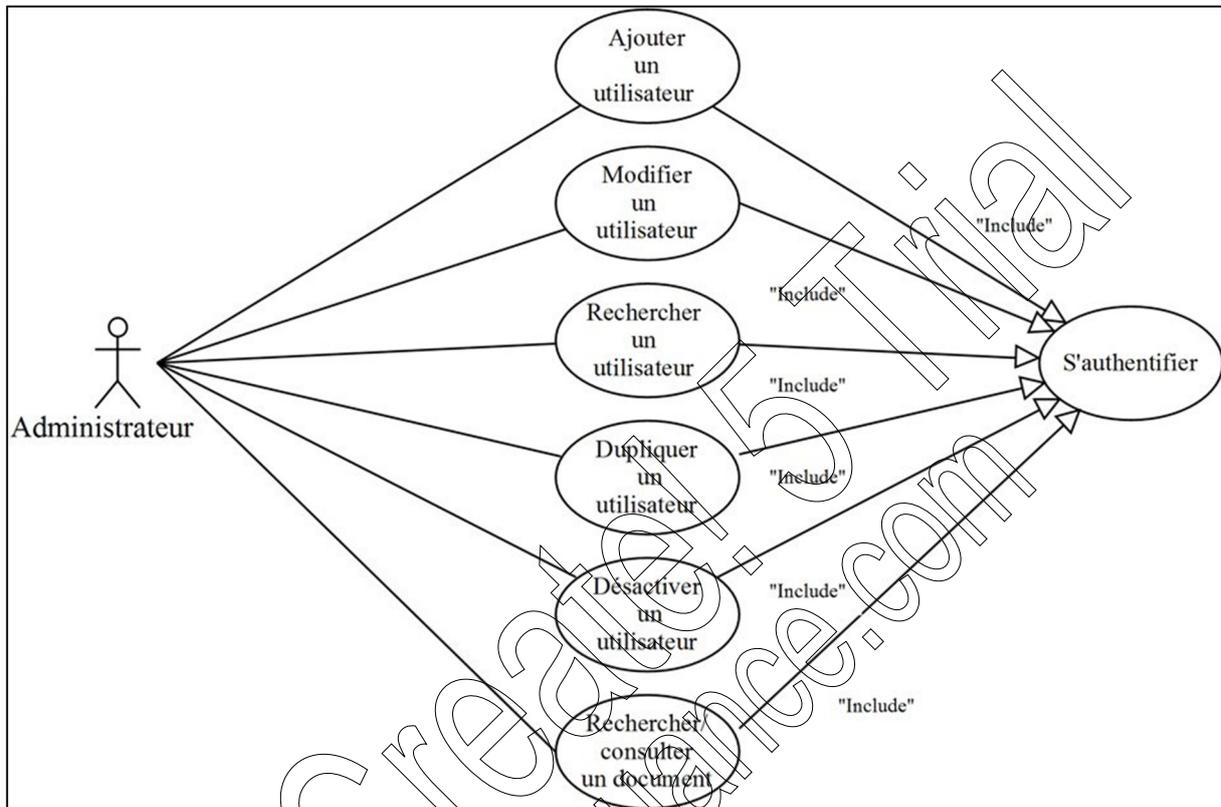


Figure IV.3 : Diagramme de cas d'utilisation de l'administrateur

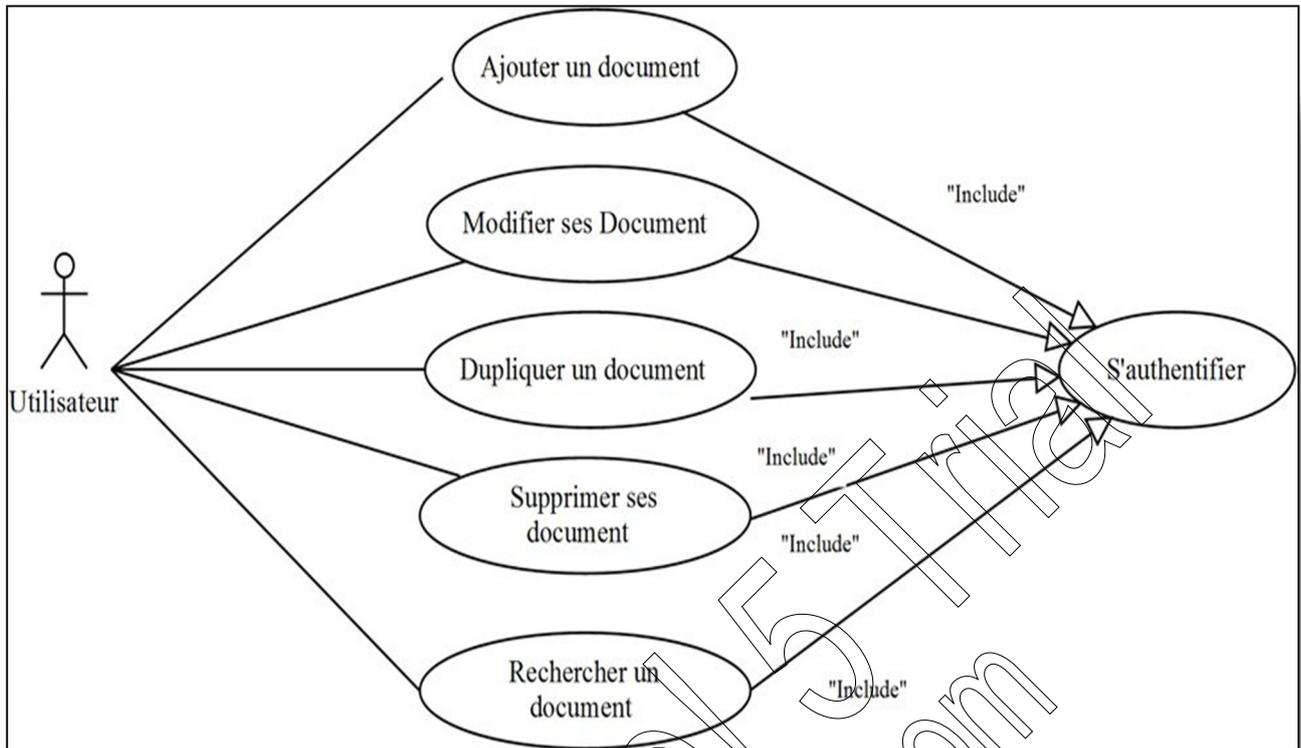


Figure IV.4: Diagramme de cas d'utilisation « Utilisateur »

PDF Create PDF
www.nuance.com

IV. Phase de conception :

IV.1 Introduction

Le but principal de la conception est de rendre le modèle d'analyse réalisable sous forme logicielle, c'est ici que l'abstraction métiers viennent pour la première fois, au contact de la réalité logicielle

IV.2 Diagrammes de séquence

IV.2.1 Définition :

Les diagrammes de séquences permettent de représenter des collaborations entre objets selon un point de vu temporel. Ce type de diagramme sert à modéliser les aspects dynamiques des systèmes temps réels et des scénarios complexes mettant en œuvre peu d'objets.

Les principales informations contenues dans un diagramme de séquence sont les messages échangés entre les lignes de vie, présentés dans un ordre chronologique.

Ils permettent de mieux visualiser la séquence des messages par une lecture de bas en haut. l'axe vertical représente le temps et horizontal représente les objets qui collaborent, une vertical en pointillé est attaché à chaque objet qui représente sa ligne de vie.

IV.2.2 Diagramme de séquence :

Dans la description des cas d'utilisation nous avons pu identifier les scénarios. Dans ce qui suit nous allons traduire quelques-uns en diagramme de séquence :

- S'authentifier ;
- Désactiver un compte utilisateur;
- Ajouter un document;

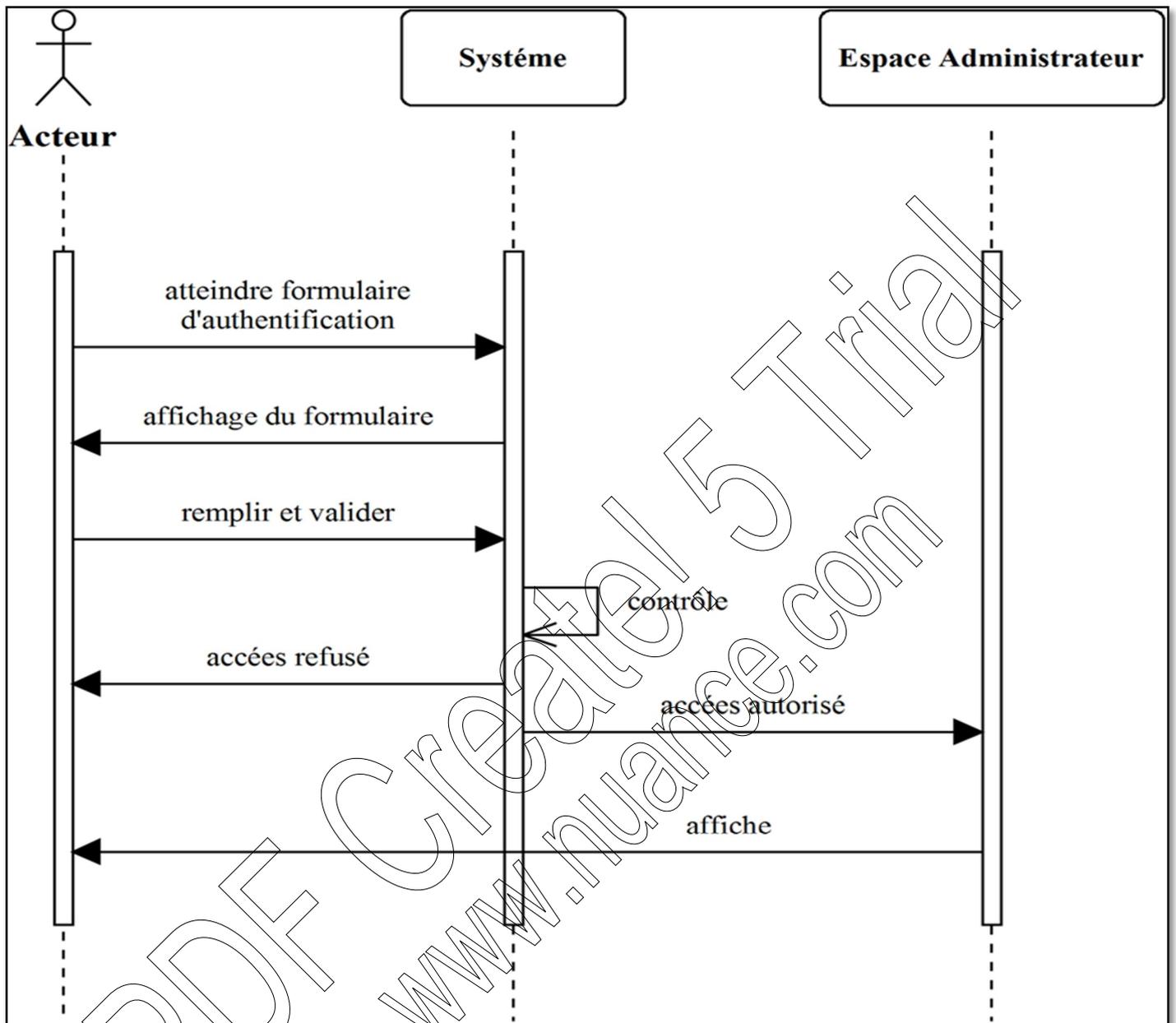


Figure IV.5: Diagramme de séquence de cas d'utilisation « s'authentifier »

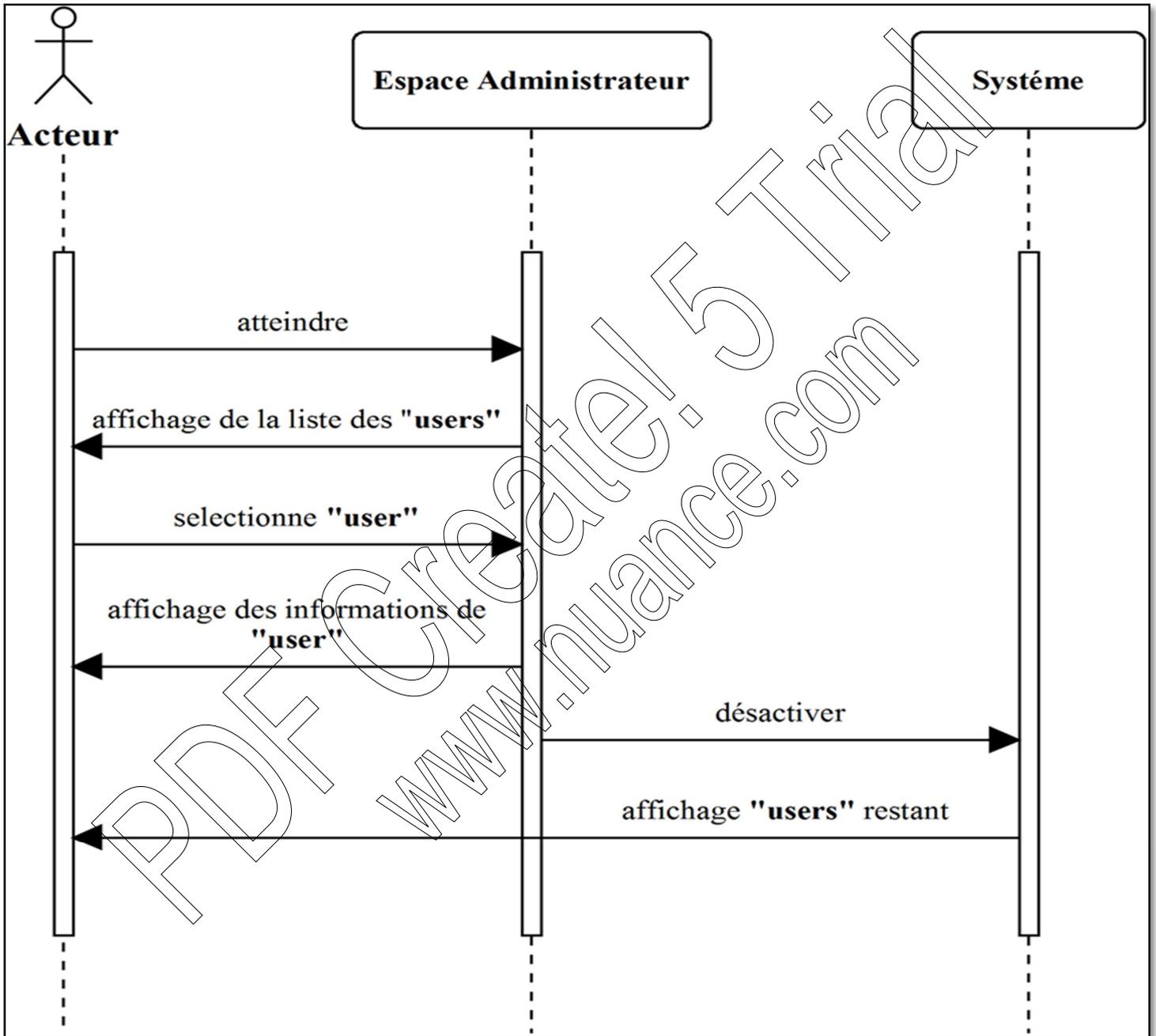


Figure IV.6: Diagramme de séquence de cas d'utilisation « désactiver un compte utilisateur »

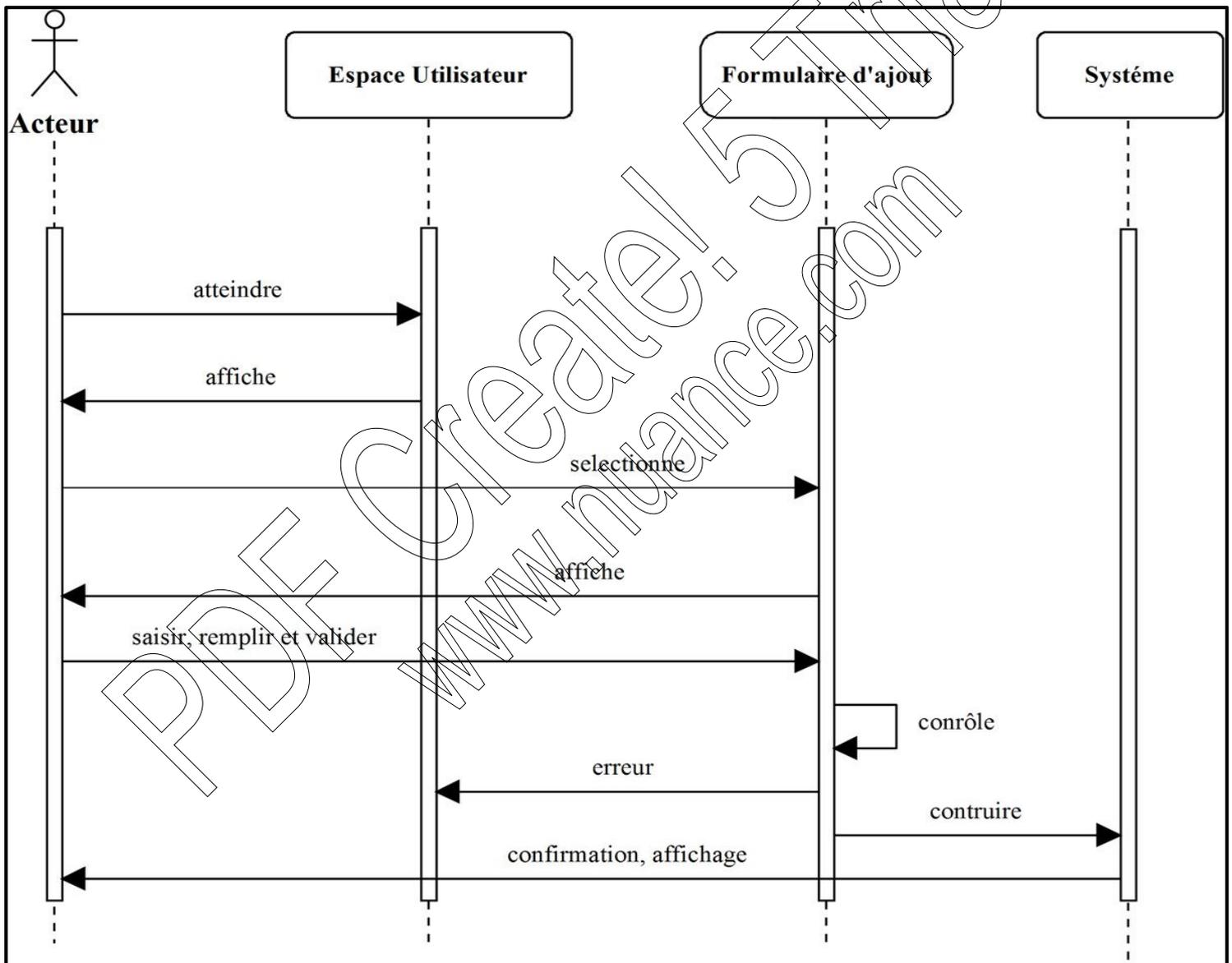


Figure IV.7: Diagramme de séquence de cas d'utilisation « ajouter un document »

IV.3 Diagramme d'activité :

Un diagramme d'activité est une variante des diagrammes d'état/transition organisé par rapport aux actions et principalement destiné à représenter le comportement interne d'une méthode (la réalisation d'une méthode ou d'un cas d'utilisation).

Un diagramme d'activité représente l'état de l'exécution d'un mécanisme, sous la forme d'un déroulement d'étapes regroupées séquentiellement dans des branches parallèles de flots de contrôles.

Nous allons représenter le diagramme d'activité pour les cas d'utilisations suivants :

- Modifier un Utilisateur ;
- Ajouter un Document ;
- Supprimer un Service ;

PDF Create! 5 Trial
www.nuance.com

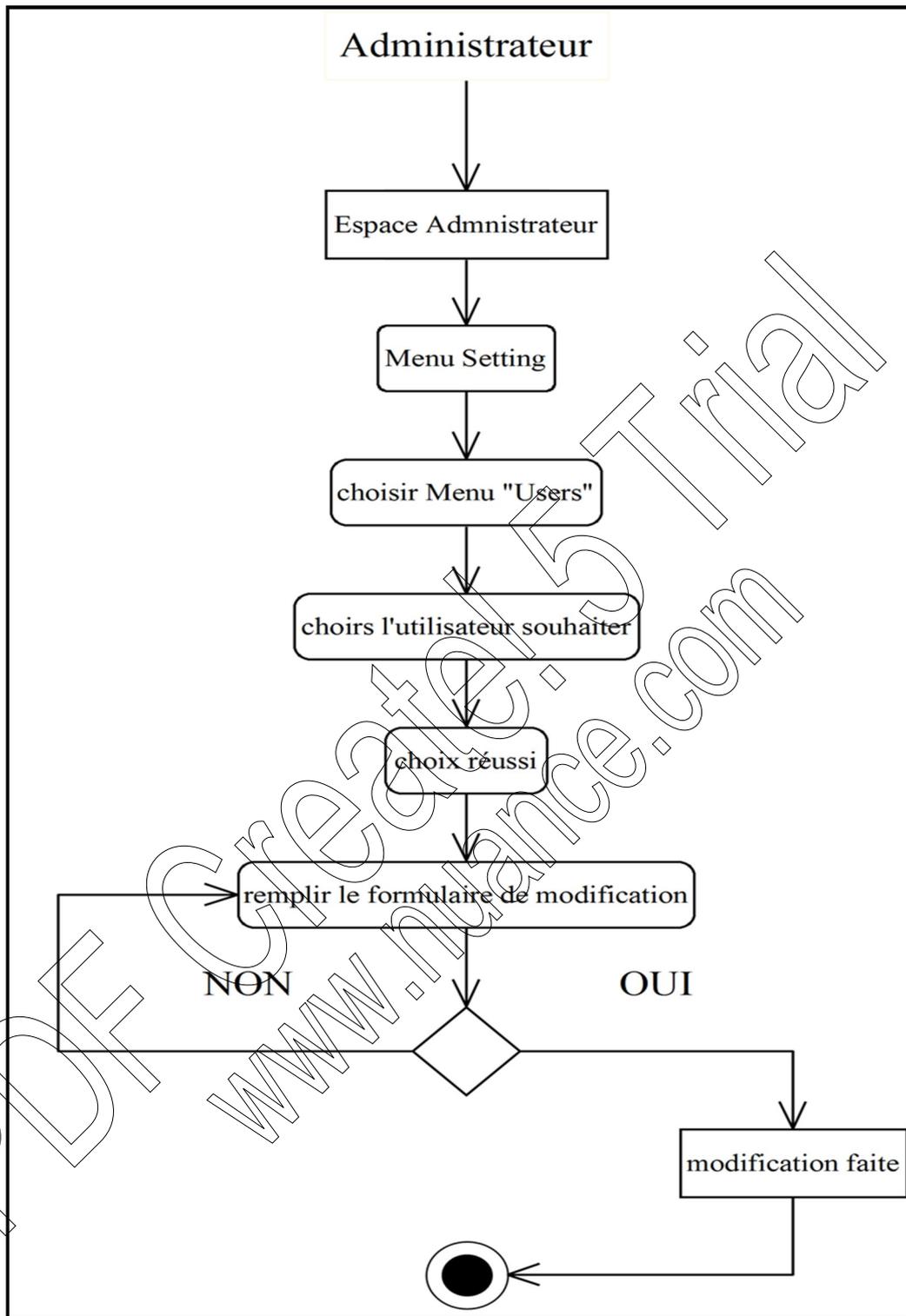


Figure IV.8: Diagramme d'activité de cas d'utilisation « Modifier un Utilisateur»

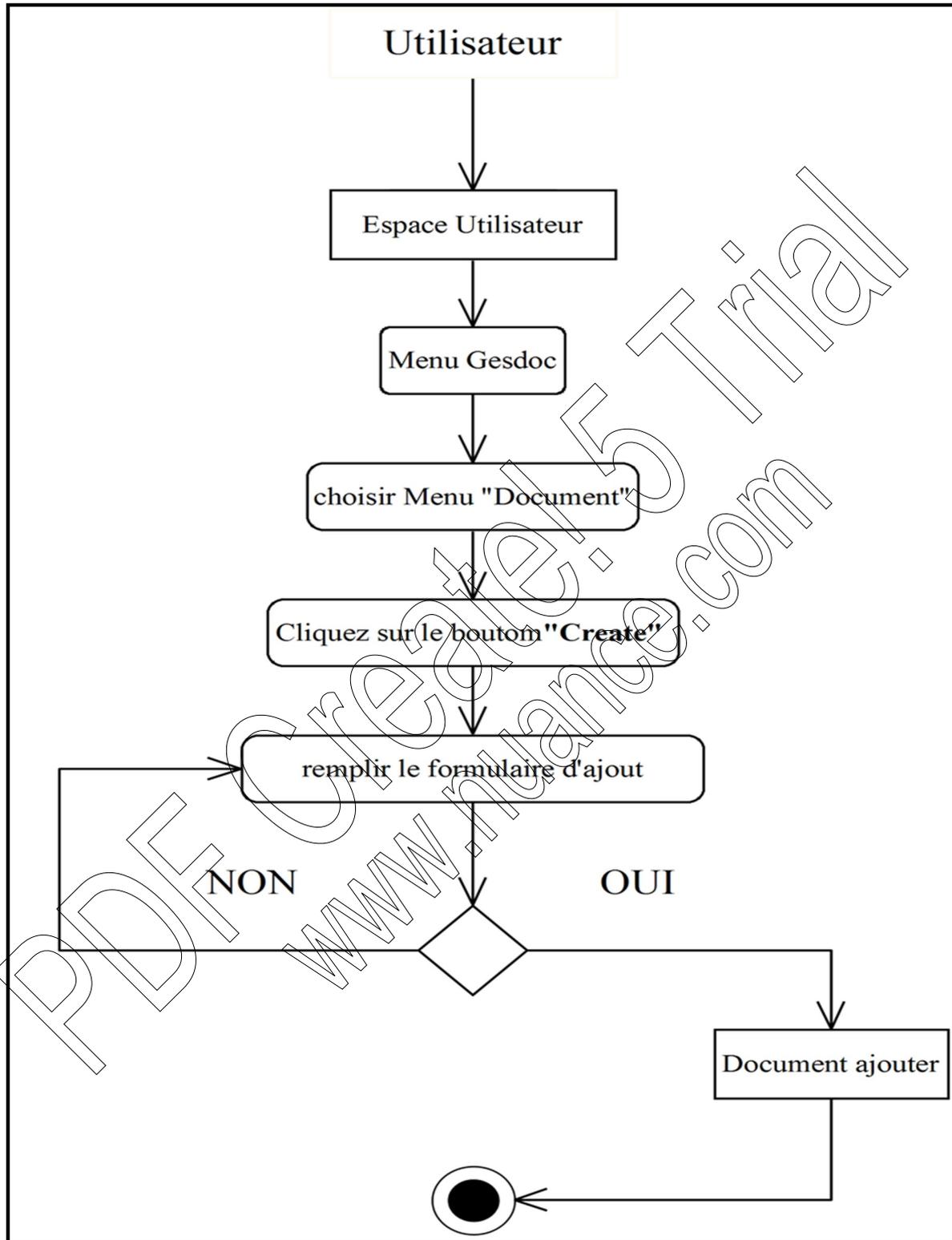


Figure IV.9: Diagramme d'activité de cas d'utilisation « Ajouter un Document »

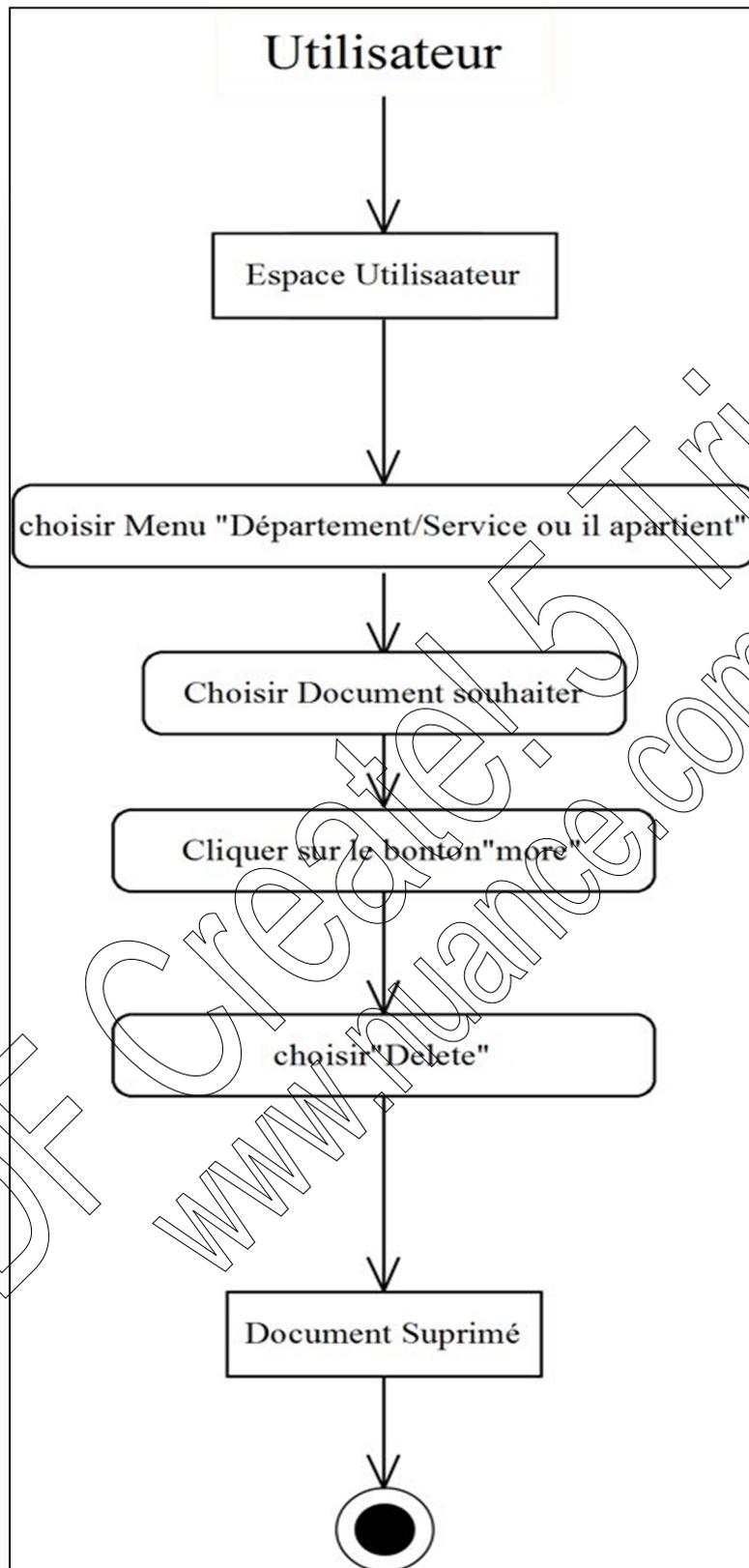


Figure IV.10: Diagramme d’activité de cas d’utilisation « Supprimer un Document »

IV.4 Diagramme des classes:

IV.4.1 Définition

Le diagramme de classe montre la structure interne du système. Il permet de fournir une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisation

Du coup après l'élaboration des diagrammes des cas d'utilisation et les diagrammes de séquences, nous allons élaborer le diagramme de classe global qui sera la référence pour l'implémentation de la base de données, et cela parce qu'il met en évidence toutes les classes et leurs attributs ainsi leurs méthodes.

PDF Create! 5 Trial
www.nuance.com

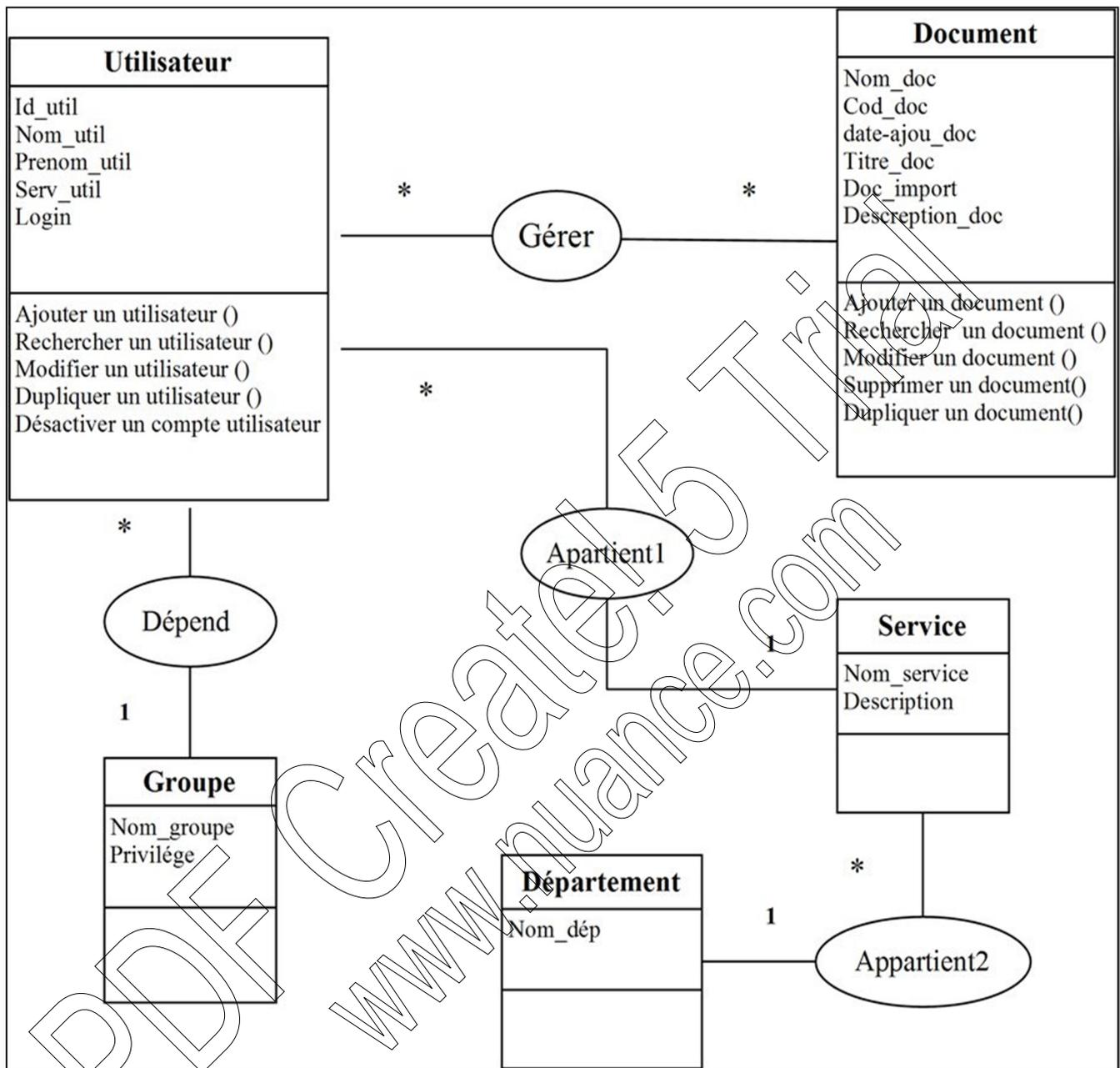


Figure IV.11: Diagramme de classe globale

IV.5 Diagramme de déploiement

Le diagramme de déploiement est une vue statique qui sert à représenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que la relation entre eux. Les éléments utilisés par un diagramme de déploiement sont principalement les nœuds, les composants, les associations. Les caractéristiques des ressources matérielles physiques et des supports de communication peuvent être précisées par stéréotype.

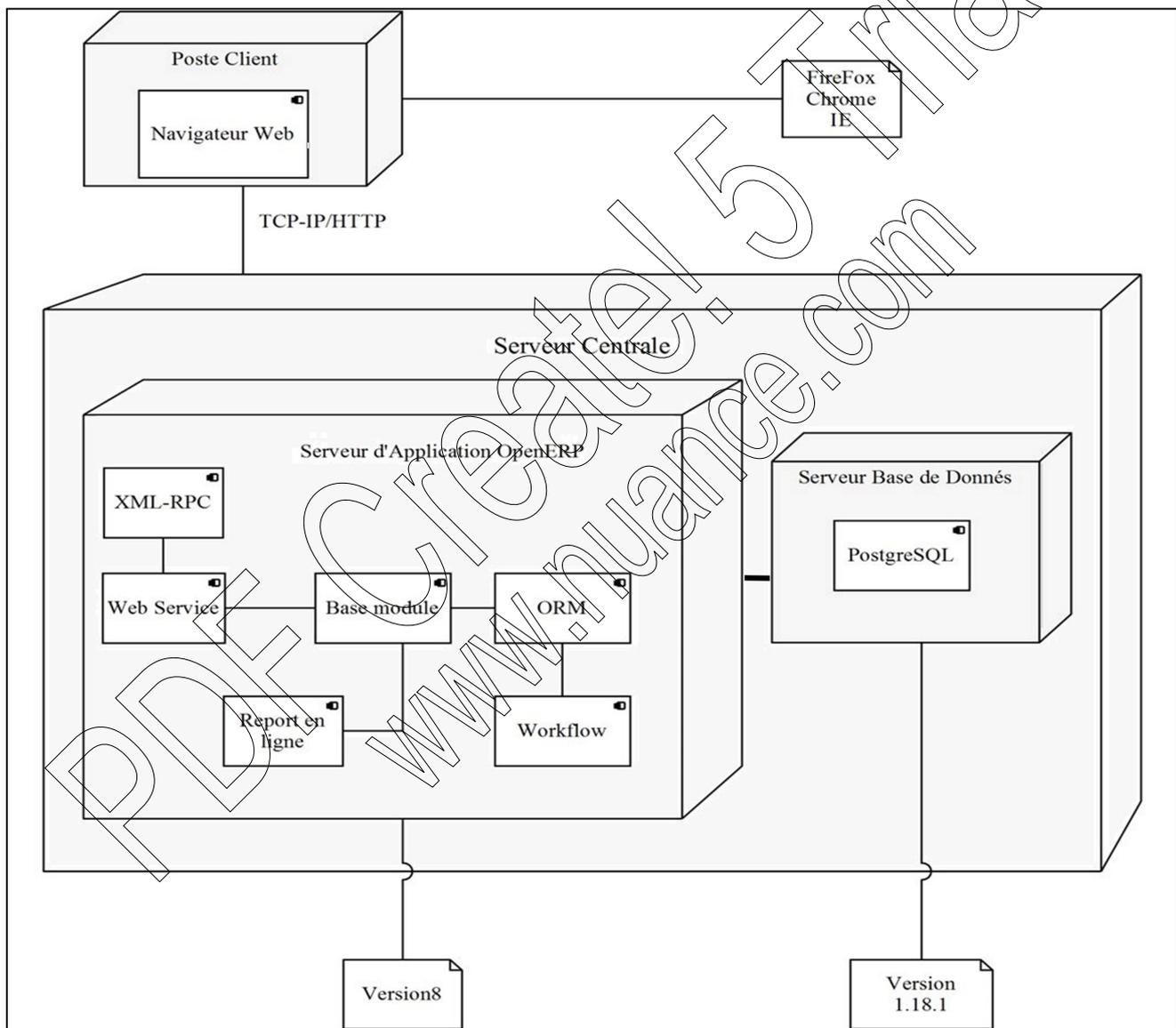


Figure IV.12: Diagramme de déploiement

IV.6 Diagramme d'architecture d'application

Ce diagramme représente l'architecture d'application de notre ERP dans lequel on possède un serveur SGBD qui représente par PostgreSQL, et un serveur d'application openerp qui contient tous les modules concernés par notre système en interaction avec un interpréteur des scripts python, puis des services web, des appels XML-RPC qui assurent l'interface entre l'utilisateur et le système.

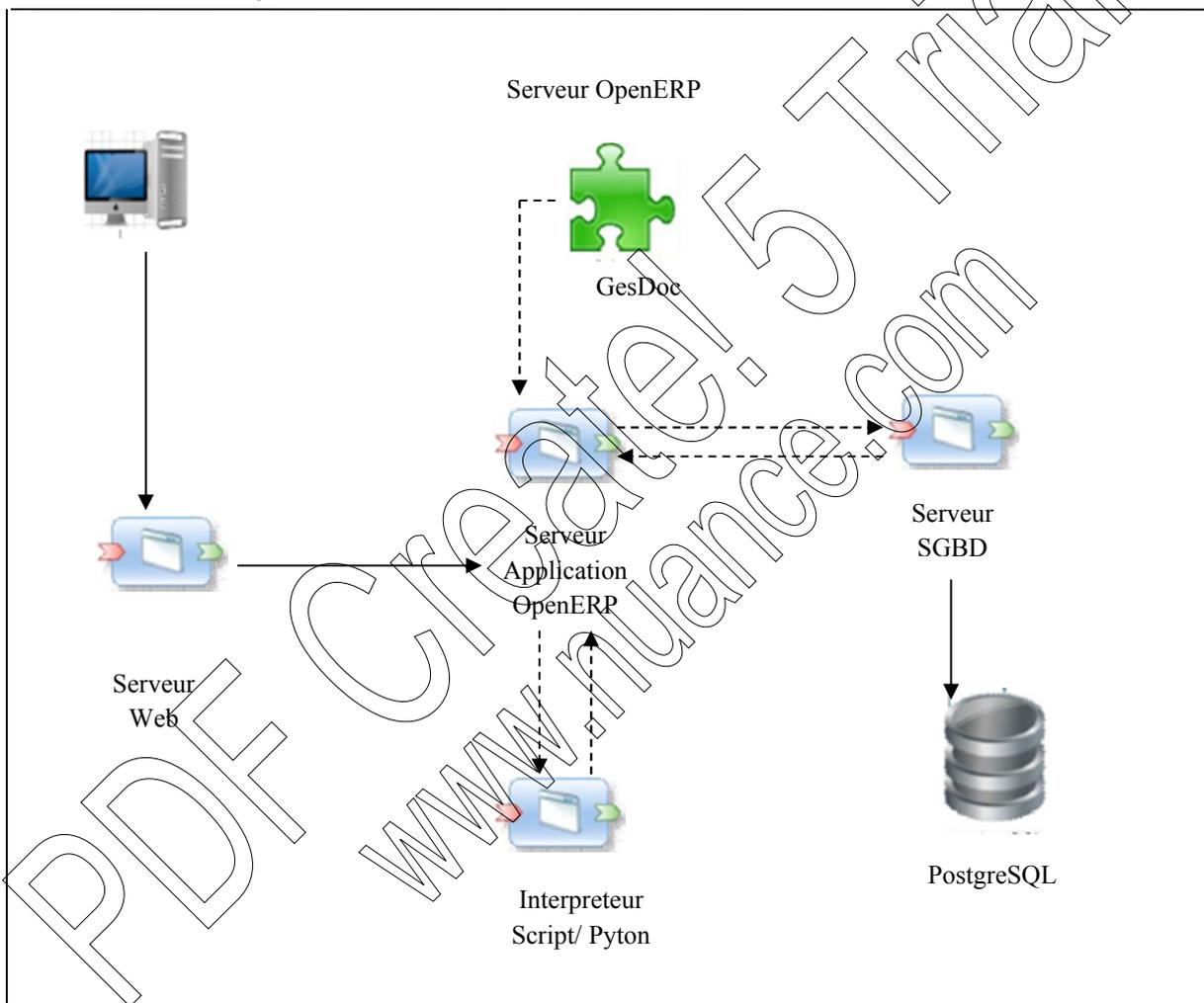


Figure IV.13: Diagramme d'architecture d'application

V. Conclusion :

Ce chapitre a présenté la modélisation du système à réaliser en utilisant les diagrammes d'UML, en commençant par la spécification des besoins et les divers cas d'utilisation, puis la conception des diagrammes des cas d'utilisation, diagramme des séquences ,diagrammes d'activités , diagrammes de classes

Dans le chapitre qui suit, nous allons décrire les outils ayant servis à développer notre application, ainsi que ses interfaces principales.

PDF Create! 5 Trial
www.nuance.com

I. Introduction

Dans ce chapitre, nous allons présenter le support matériel et logiciel qui nous a servi d'appuis pour le développement de notre module tout en focalisant notre attention sur la technologie Windows, Par la suite, nous allons présenter notre application qui couvre la création et l'implémentation de la base de données et les différents programmes qui vont permettre la mise en marche des fonctionnalités envisagées, ainsi que la présentation de quelques interfaces de notre module.

II. Partie I :

II.1. Présentation du l'environnement utilisés

Durant la réalisation de notre module, nous avons utilisé l'environnement suivant comprenant :

- Le SGPD Postgre SQL1.18.1
- Notepad++version 6.1.5
- Le langage de programmation python
- L'XML
- L'ERP open source «OPENERP » (ODOO version 8)

II.2. Le SGPD Postgre SQL8.3 [26]

PostgreSQL est un système de gestion de base de données relationnelle et objet(SGBDRO). C'est un outil libre disponible selon les termes d'une licence de type BSD. Il a été développé à l'université de Californie au département des sciences informatiques de Berkeley

Ce système est concurrent d'autres systèmes de gestion de base de données, qu'ils soient libres (comme MySQL et Firebird), ou propriétaires (comme Oracle, Sybase, DB2, Informix et Microsoft SQL Server). 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. PostgreSQL est pratiquement conforme (de plus en plus conforme) aux normes ANSISQL 89, SQL 92 (SQL 2), SQL 99 (SQL 3), SQL:2003 et SQL:2008. Il fonctionne sur diverses plates-formes matérielles et sous différents systèmes d'exploitation.

PostgreSQL fonctionne sur Solaris, SunOS, Mac OS X, HP-UX, AIX, Linux, IRIX, Digital Unix, BSD, NetBSD, FreeBSD, OpenBSD, SCO unix, NeXTSTEP, UnixWare et toutes sortes d'Unix. Depuis la version 8.0, PostgreSQL fonctionne également nativement sur Windows. Avant la version 8, il fallait un émulateur de type cygwin pour faire fonctionner PostgreSQL sur ce système d'exploitation.

PostgreSQL est largement reconnu pour son comportement stable, proche d'Oracle. Mais aussi pour ses possibilités de programmation étendues, directement dans le moteur de la base de données, via PL/pgSQL. Le traitement interne des données peut aussi être couplé à d'autres modules externes compilés dans d'autres langages. - [docs.postgresqlfr.org , Rapport PFE Hamli]

II.3. L'éditeur Notepad++ [27]

Notepad++ est un éditeur de texte très léger, très puissant et libre (licenceGPL). Il est parfait pour programmer avec des langages ne nécessitant pas d'environnement de développement (html, css, javascript, php...) ou en ayant un peu pratique (python, processing...), ou pour du traitement de données. Il prend en charge par défaut une cinquantaine de langages différents, et vous laisse libre d'en ajouter d'autres. Notepad ++ est éditeur de code source pour une utilisation avec Microsoft Windows . Contrairement

Bloc-notes , le haut-éditeur de texte Windows, il prend en charge onglets édition, qui permet de travailler avec plusieurs fichiers ouverts dans une seule fenêtre. Le nom du projet provient du C opérateur d'incrémentation .Il est distribué comme logiciel libre

Il a été codé par Don Ho, un informaticien basé à Paris diplômé de l'Université Paris VII - Diderot en 2000.

Ces avantages sont principalement :

- Les onglets pour éditer plusieurs fichiers dans la même fenêtre.
- Ou même les multi-vues pour voir tout en même temps.
- Sa rapidité (il est assez léger et se lance aussi vite que le Notepad Windows).
- La coloration syntaxique de nombreux langages (entre autres C, C++,HTML, XML, PHP, Java, Perl, Pascal, Python les fichiers Makefile, TeX, Flash ActionScript, Haskell et InnoSetup) ainsi que ceux que vous ajoutez vous même.
- Le choix du Codage de caractères (ANSI, UTF-8, UCS-2)
- Le formatage type Unix, Macintosh ou Windows.
- La recherche d'expression régulière, ainsi que la recherche dans les dossiers ("Find in Files").
- Il s'insère au menu local (clic-droit) pour ouvrir n'importe quel type de fichiers rapidement sans avoir à les attribuer.
- Sa configuration se fait entièrement via des menus et des boîtes des dialogues.
- La numérotation des lignes, l'auto-complétion, plein d'options d'édition en icônes et les raccourcis clavier qui vont avec.

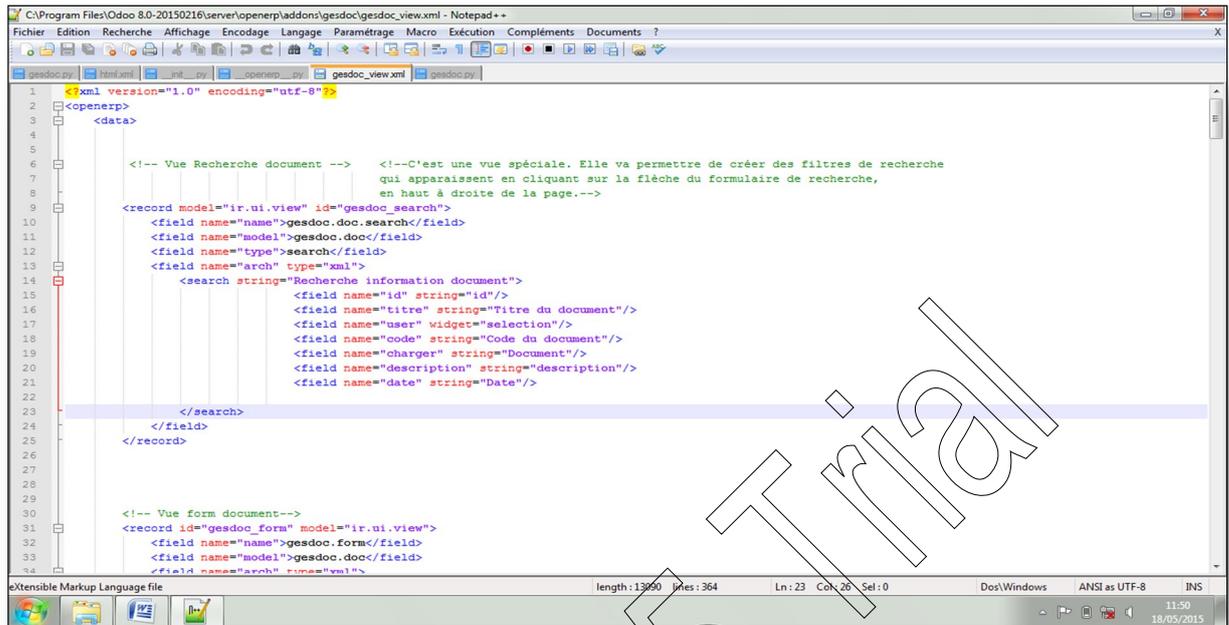


Figure. V.1.L'éditeur Notepad++

II.4. Le langage de programmation python

II.4.1 Définition

Python est un langage portable, dynamique, extensible, gratuit qui permet une approche modulaire et orienté objet de la programmation. Python est développé depuis 1989 par Guido van Rossum et de nombreux contributeurs bénévoles. Il est utilisé par un public très large : des développeurs web professionnels, des chercheurs en intelligence artificielle ou en bioinformatique, des administrateurs systèmes, ou même des programmeurs occasionnels.

II.4.2 caractéristiques du python

Les principales caractéristiques de python sont : [20]

- **De haut niveau, orienté objet et totalement libre** : même si elle n'est pas imposée, Python permet la programmation orientée objet. Tous les mécanismes objet essentiels sont implémentés et toutes les données manipulées sont des instances de classes, comme pour les langages SmallTalk ou Ruby. Enfin, le code peut être structuré en modules (fichiers) qui sont ensuite importables dans l'interpréteur. Ce découpage, permet d'organiser le code et son utilisation par des espaces de noms, et aussi de faciliter l'extension du langage par des bibliothèques tierces compilées dans d'autres langages.
- **Python est portable** : non seulement sur les différents variantes d'UNIX, mais aussi sur les OS propriétaires, NEXTSep, MS_DOS et les différents variantes de

Windows. Un nouveau compilateur, nommé Python, est écrit en java et génère du *bytecode* java.

- **Python est gratuit** : on peut l'utiliser sans restriction dans des projets commerciaux.
- **La syntaxe** de python est très simple et combiné à des **types de données évolués** (listes, dictionnaires,...), conduit à des programmes à la fois très importants et très lisibles. A fonctionnalités égales, un programme python est souvent de 3 à 5 fois plus court qu'un programme C ou C++ (ou même java) équivalent, ce qui représente en général un temps de développement de 5 à 10 fois plus court et une facilité de maintenance largement accrue.
- **Dynamique** : dans la plupart des implémentations, le code source n'est pas compilé contrairement à des langages comme C ou Pascal, mais exécuté à la volée. On parle alors de langage interprété. Ce mode de fonctionnement rend la programmation beaucoup plus souple puisqu'il est possible de changer un programme en cours d'exécution, ou de tester du code en mode interactif sans disposition particulière.
- Python intègre, comme java ou les versions récentes de C++, un système d'exception, qui permettent de simplifier considérablement la gestion des erreurs.

Voici une liste de projets représentatifs basés sur python :

- **Zope**, un serveur d'application innovant, et **CPS**, un Framework de gestion de contenu et de travail collaboratif basé sur Zope.
- Les programmes d'admissions système spécifiques à la distribution Red Hat Linux.
- Des moteurs de recherche comme **Google** ou **Yahoo!**
- **Chandler**, le projet de PIM (Personnel Information Manager) de l'Open Source Applications Fondation

II.5. L'XML: [20]

II.5.1 Définition

XML a été mis au point par le XML Working Group sous l'égide du World Wide Web Consortium (W3C) dès 1996. Depuis le 10 février 1998, les spécifications XML 1.0 ont été reconnues comme recommandations par le W3C, ce qui en fait un langage reconnu.

XML est un sous ensemble de SGML (Standard Generalized Markup Language), défini par le standard ISO8879 en 1986, utilisé dans le milieu de la Gestion Electronique Documentaire (GED).

Le langage XML (eXtended Markup Language) est un langage de format de document. Il dérive de SGML (Standard Generalized Markup Language) et HTML (HyperText Markup Language). Comme ces derniers, il s'agit d'un langage formé de *balises* qui permet de structurer les documents.

XML s'est imposé comme le format standard pour les communications entre applications. Il est utilisé dans la plupart des projets de publication sur le WEB ainsi que dans les bases de données. XML fait partie aussi du code des modules composants OpenERP, les vues par lesquelles sont représentés les différents objets sont écrites en XML, ainsi nous y

trouvons la description détaillée de l’affichage des arbres, formulaires, menus et autres actions. [Introduction à XML - Comment Ça Marche]

II.6. ODOO

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

II.6.1 La démarche d’installation d’Odoo Sous windows (toutes les versions):

Installation d’odoo et Postgresql

Odoo fonctionne en mode SaaS (Software as a service, en fait logiciel en ligne) ou en mode client/serveur. On a opté pour le mode « logiciel en ligne » pour deux raisons

- ✚ il suffit d’installer sur un PC (ou un serveur) le serveur et le serveur web, on aura pas besoin de l’installer sur les autres PC, l’utilisation du logiciel se fait avec un navigateur web.
- ✚ L’interface graphique est plus « actuelle ».

En cliquant sur le programme d’installation, tout est automatisé, l’installation du Server, du Web client ainsi que le gestionnaire de base de données.

Il suffit de cliquer sur *suivant*, à chaque étape de l’installation :

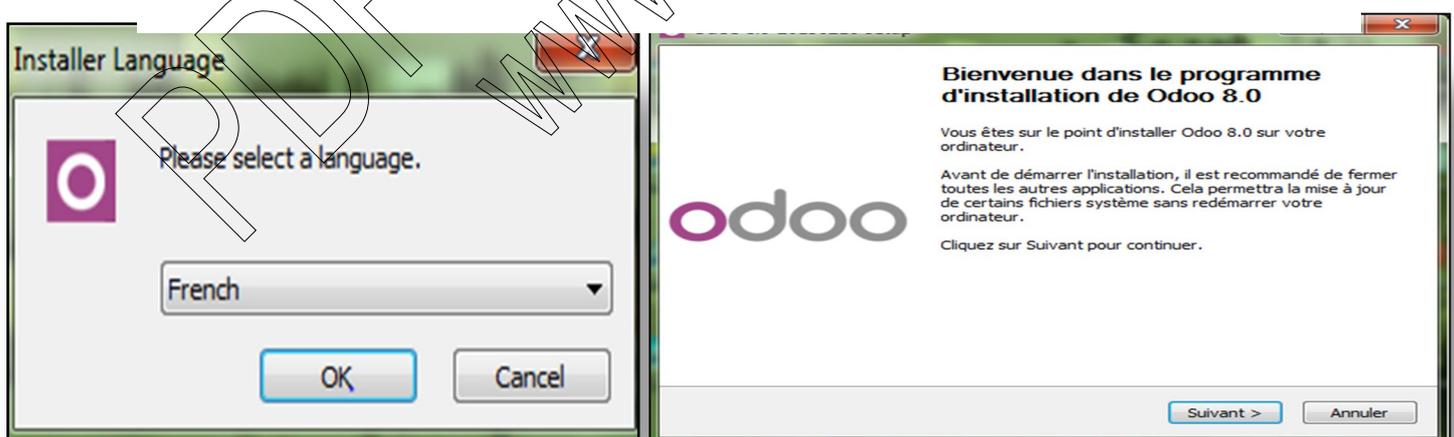


Figure. V.2. Les étapes d’installation d’Odoo sous Windows

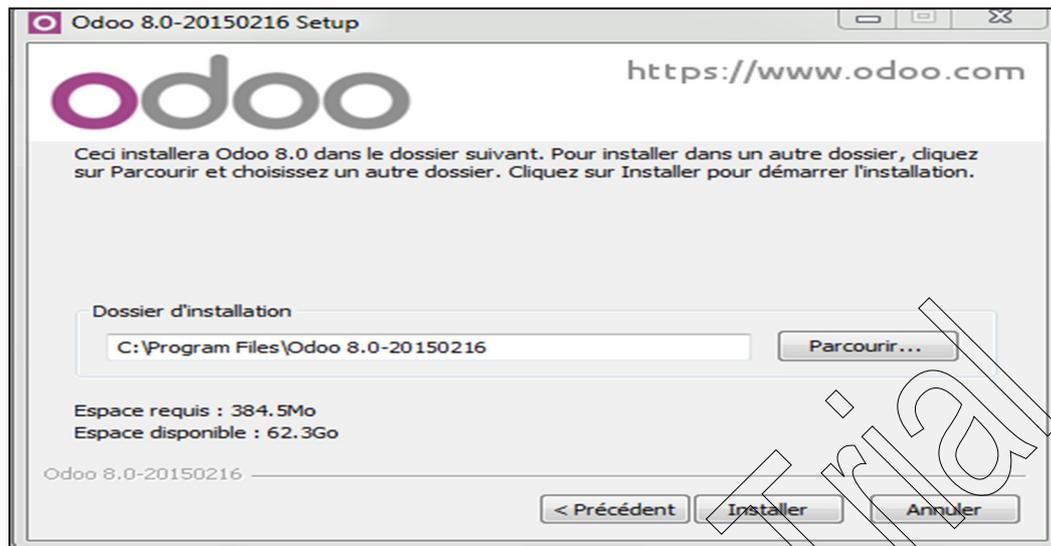


Figure. V.3. Les étapes d'installation d'Odoo suite

- Une fois odoo et postgres SQL sont installés sur notre ordinateur, on clique sur finish (après avoir coché la case « start odoo »). Odoo commencera ensuite automatiquement.
- notre navigateur par défaut sera ouvert avec le panneau d'Odoo de contrôle de la base de données.

Création d'une nouvelle base de données

Odoo utilise une base de données (PostgreSQL) pour stocker toutes les données. Après l'installation locale d'Odoo sur notre propre ordinateur cette base de données doit d'abord être créée.

- Maître mot de passe: admin
- Sélection du nom de base de données: gdoc
- Chargement des données de démonstration: décochez
- Langue par défaut: English (US)
- Choix d'un mot de passe: admin
- Confirmation du mot de passe: admin
- Confirmation de la création de la base de données en cliquant sur « crée la base de donnée »



Figure. V.4. La création d'une nouvelle base de données Odoo

Après quelques instants, la nouvelle base de données sera chargée dans Odoo et nous serons automatiquement connectés avec l'utilisateur par défaut « Administrator » et on aura la liste de tous les modules installer sur Odoo.

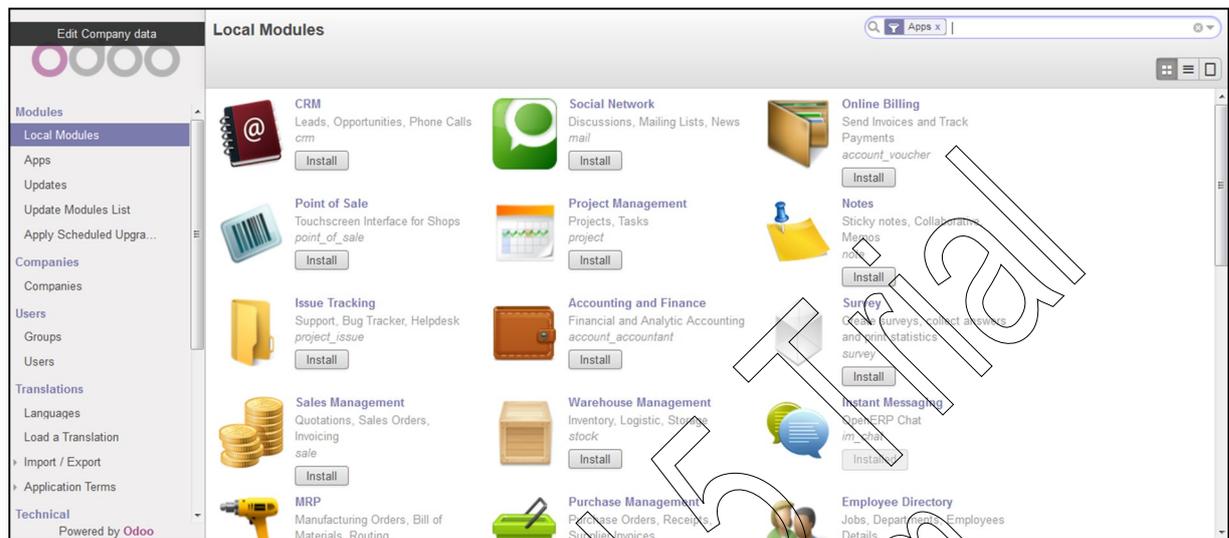


Figure. V.5. Page Principale d'Odoo

II.6.2 la base de données dans PostgreSQL:

La base de données «Gdoc» a été générée à partir du client ERP, les différentes tables qui composent la base sont implémentées sous le schéma « PUBLIC ».

La figure suivante montre l'interface graphique de pgAdmin dans la quelle sont montées les tables ainsi que leur appartenance.

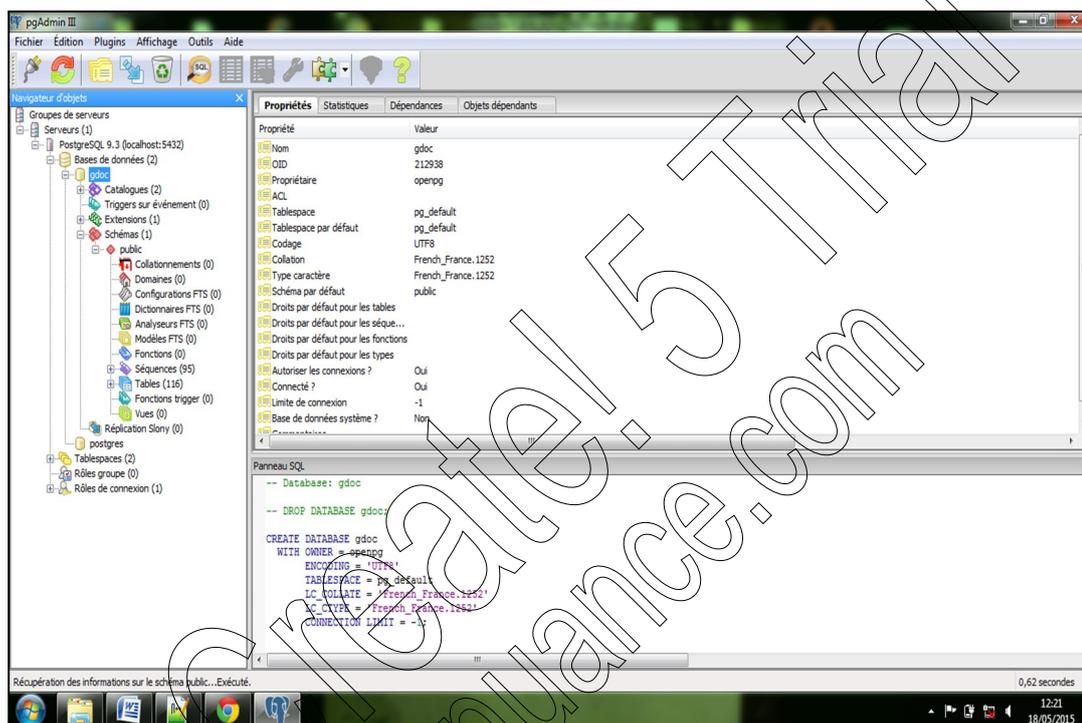


Figure. V.6. Présentation de la base de données

➔ Pour accéder au logiciel en mode Web il faut simplement ouvrir un navigateur web et saisir l'adresse suivante : <http://localhost:8069>

III. Partie II: Présentation de l'application:

III.1. principe de bases:

Pour schématiser, on peut dire que la réalisation de notre module s'est faite en trois étapes.

- La création des fichiers d'initialisation du module.
- La création de l'objet en Python.
- La création des vues en XML.

III.2. les étapes de réalisation :

- Créer un package python dans le répertoire /addons portant le nom de notre module
- Création des fichiers d'initialisation du module : **__init__.py**
- Création du fichier Python qui contiendra l'Objet : **__openerp__.py**
- Création des vues tableau/formulaire : **gesdoc_view.xml**
- Création du fichier python contenant le modèle (classe + méthodes) : **gesdoc.py**
- Création d'un menu pour le groupe Gestion de Document
- Création de l'icône pour le module.
- Création du fichier d'Index.html pour le design du module

En gros y'a trois fichiers obligatoires lorsque vous créez un module.

- **__init__.py**
- **__openerp__.py**
- **gesdoc.py**

a) **__init__.py** :

C'est le fichier qui permet à OpenERP de charger notre module.

b) **__openerp__.py**

C'est le fichier qui contient toutes les informations sur notre module : le nom, la version, la catégorie, la description, les fichiers à charger, etc.

- name: le nom de votre module;
- version: la version du module;
- category: la catégorie dans laquelle vous classez votre module;
- séquence: c'est un nombre qui permet de définir l'ordre de votre module dans la liste des modules. 1, il sera en haut, 100 il sera en bas
- author: l'auteur du module;
- description: la description complète du module;
- depends: les dépendances votre module;

- installable: si votre module est installable ou non;
 - application: laissez à False. Votre module ne sera pas reconnu comme une application. C'est OpenERP qui délivre les certificats qui qualifient votre module d'application;
- c) **gesdoc.py** : c'est le fichier qui contient les classes et les méthodes de notre module

Tous les modules d'Odoo sont créés au-dessous du répertoire :

<C:\Program Files\Odoo 8.0-20150216\server\openerp\addons>

Une fois terminé, notre module Gesdoc apparaîtra dans la liste des modules installés, comme dans l'image ci-dessous.



Figure. V.7. Le module Gesdoc installé

III.3. Fonctionnement d'ODOO:

III.3.1. Page d'authentification d'ODOO:

Elle représente la porte principale d'odoo, c'est la première page qui apparaîtra à tout utilisateur d'odoo. Elle permet également l'accès à la page d'accueil en cliquant sur le bouton « *login* », et de quitter l'application en utilisant le bouton « *logout* »

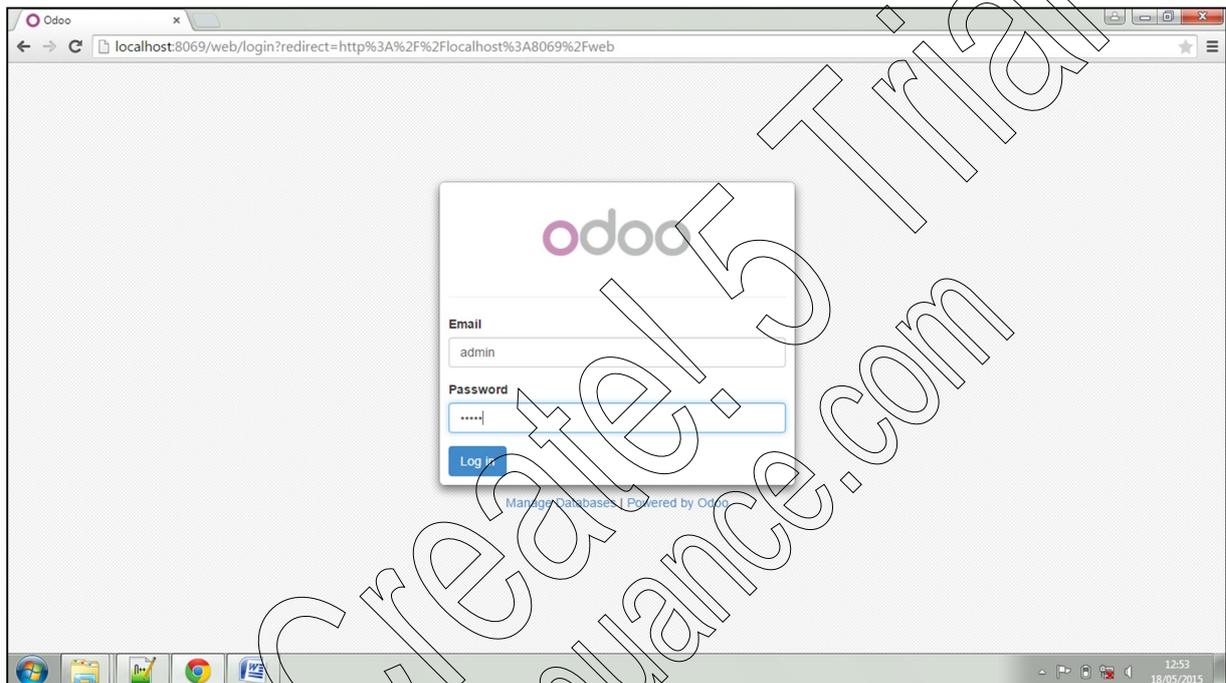


Figure. V.8. page d'authentification

III.3.2. Espace administrateur :

C'est un espace qui contient 3 menus principaux :

- Le premier menu « **gesdoc** » permet à l'administrateur de gérer notre module
- Le menu « setting » qui permet à l'administrateur la gestion de notre module (configuration, création des groupes et des utilisateurs).
- Le menu « administrator » : c'est le menu qui affiche le nom de l'utilisateur et qui possède les options générales concernant **Odoo** dont les quelles on trouve « log out » afin de quitter son espace.

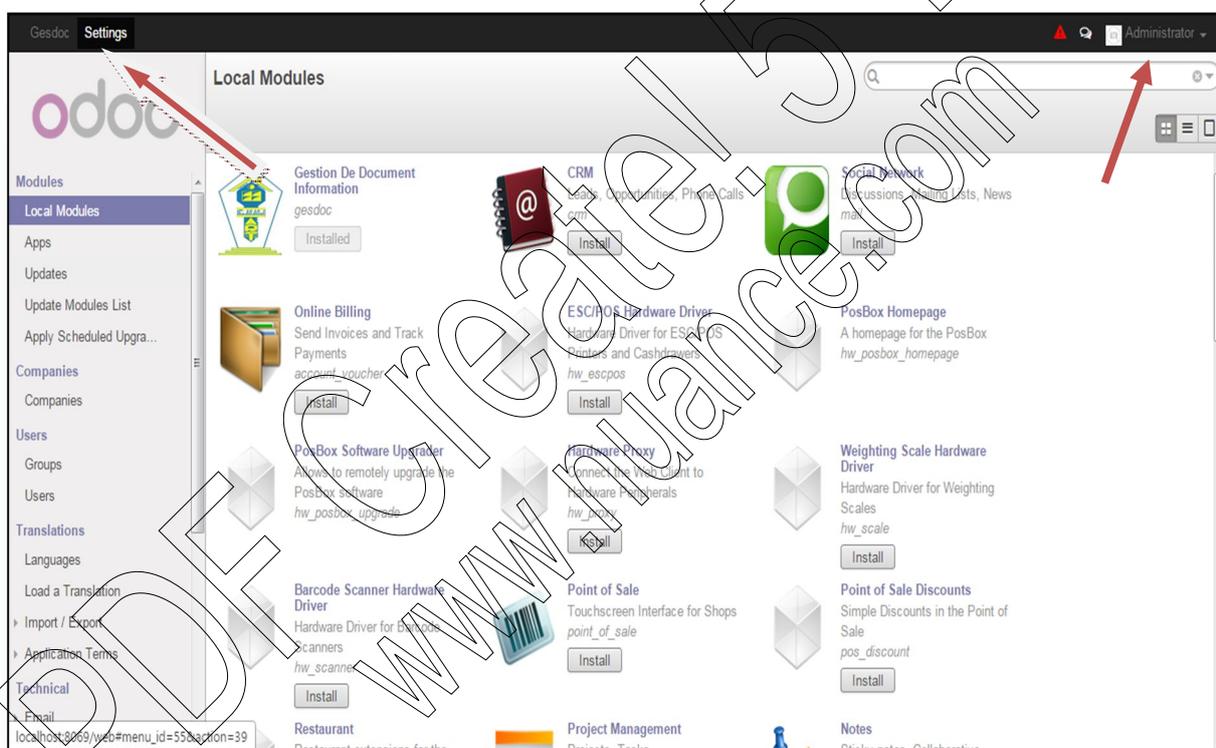


Figure. V.9.Espace Administrateur « Menu Settings »

- Le menu « Settings » permet à l'administrateur de crée des utilisateurs, des groupes ainsi de spécifier les droit d'accès concernant chaque utilisateur pour accéder au modules.

Quelque définition :

- *Utilisateur*: une personne identifiée par son login et mot de passe. Notez que tous les employés d'une entreprise ne sont pas nécessairement les utilisateurs OpenERP, un utilisateur est une personne qui accède à l'application.
 - *Groupe* : un groupe d'utilisateurs qui a des droits d'accès. Un groupe donne ses droits d'accès à ses utilisateurs.
 - *Règles de sécurité*: une règle qui définit les droits d'accès d'un groupe donné. Les règles de sécurité sont fixées à une ressource donnée.
- Voilà la Présentation de quelques interfaces montrant comment l'administrateur spécifie les droits d'accès, crée des utilisateurs et des groupes.

La figure suivante représente la spécification des droits d'accès :

Object	Read Access	Write Access	Create Access	Delete Access	Name
gesdoc.dep5service2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	dep5ser2
gesdoc.dep5service1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	dep5ser1
gesdoc.dep4service3	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	dep4ser3
gesdoc.dep4service2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	dep4ser2
gesdoc.dep4service1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	dep4ser1
gesdoc.dep3service2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	dep3ser2
gesdoc.dep3service1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	dep3ser1
gesdoc.dep2service4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	dep2ser4

Figure. V. 10. Espace Administrateur « Menu Settings/Spécification des droits d'accès »

Cette figure représente la création d'un groupe :

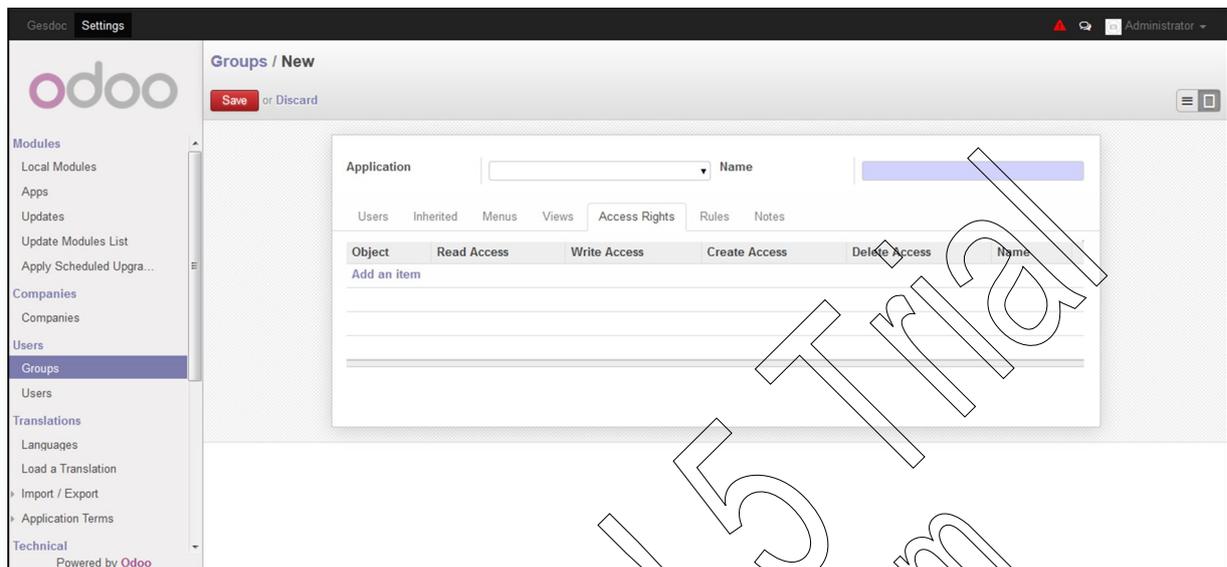


Figure. V. 11. Espace Administrateur « Menu Settings/Création d'un groupe »

Concernant les utilisateurs cette figure représente leur création :

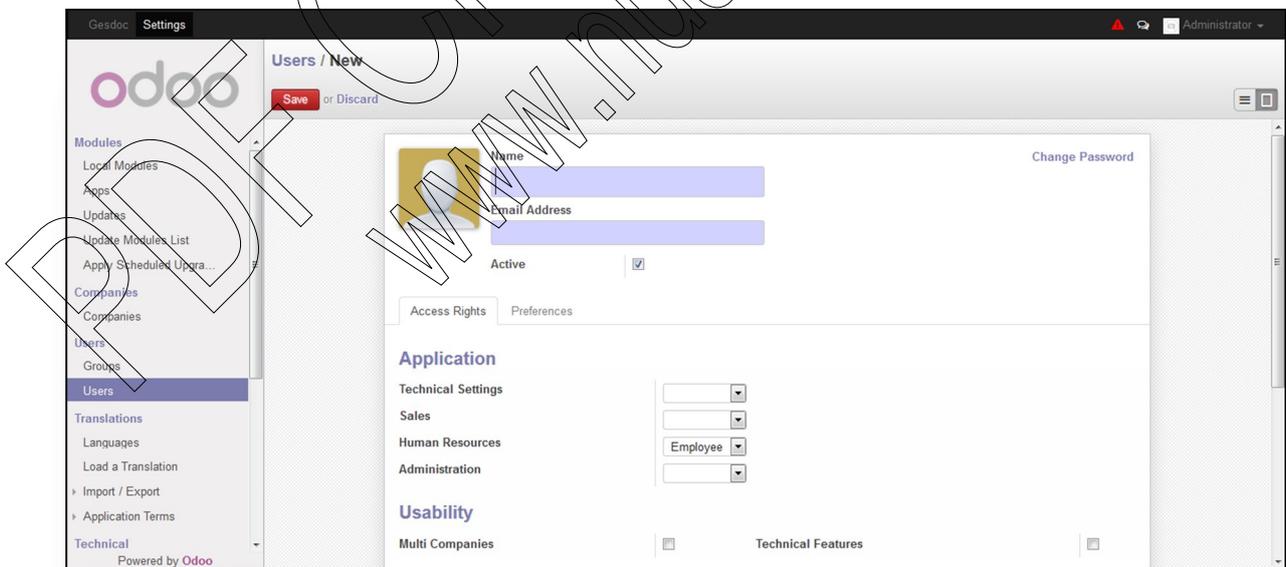


Figure. V.12.Espace Administrateur « Menu Settings/Création d'un utilisateur »

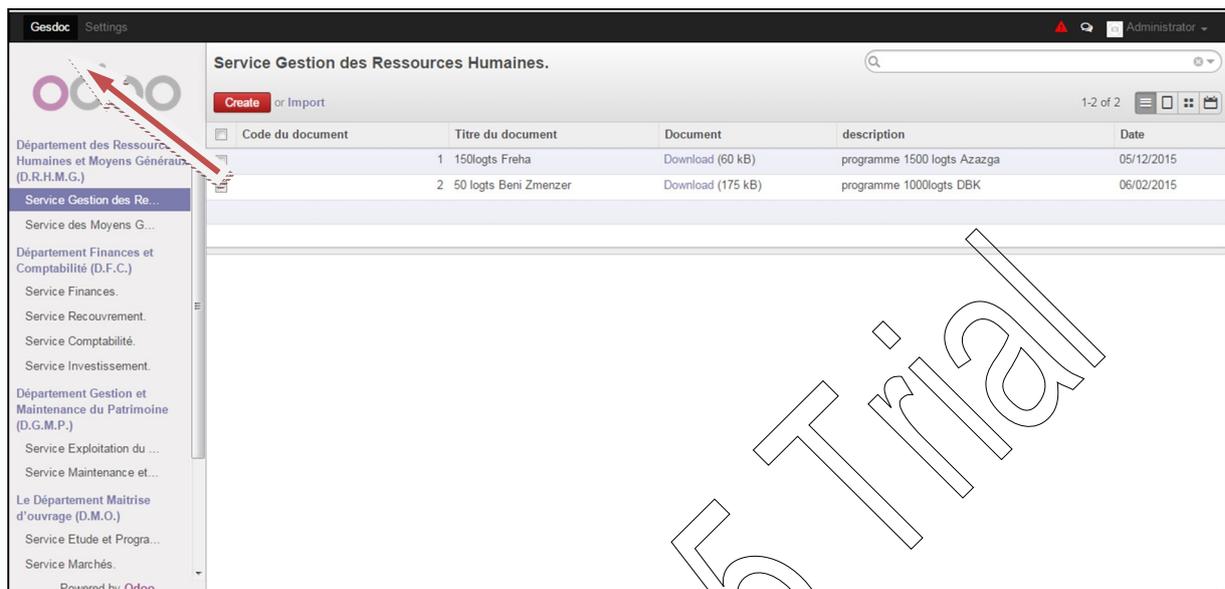


Figure. V.13. Espace Administrateur « Menu Gesdoc »

III.3.3. Espace Utilisateur :

C'est un espace qui contient :

- Un menu qui affiche le nom d'utilisateur et qui possède les options générales concernant Odoo dont les quelles on trouve « log out » afin de quitter son espace et de retourner à la page d'authentification.
- Plusieurs sous menus dans lesquelles l'utilisateur a le droit d'utiliser un seul menu, il utilise le département et le service qui le concerne ou il peut ajouter, modifier et supprimer des documents pour les autres menus il a le droit juste de consulter.

Cette figure montre par exemple un utilisateur avec le nom « *Ladj* » et il a l'accès juste à son service « *service des moyen généraux* » dont il peut créer, modifier et supprimer.

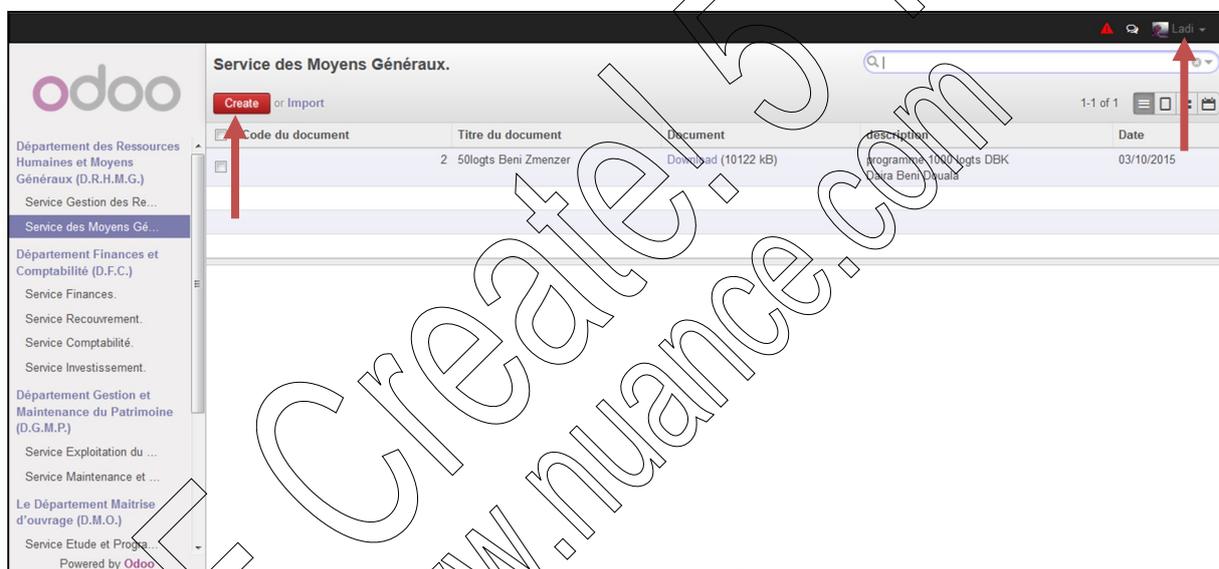


Figure. V.14.Espace Utilisateur « Menu Département des Ressources Humaines et Moyens Généraux (D.R.H.M.G.)/ Service des Moyens Généraux/Ajouter un document»

Cette figure montre le formulaire d'ajouter un document qui apparaît après avoir cliqué sur le bouton « *creat* »

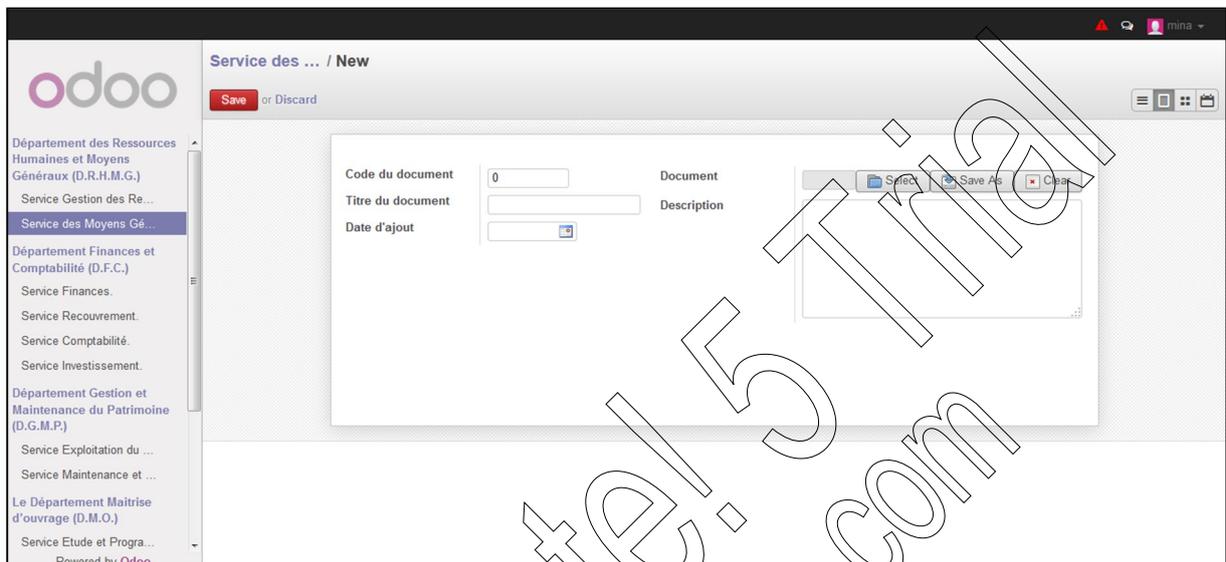


Figure. V.15. Espace Utilisateur « formulaire d'ajout d'un Document »

Cette figure montre la modification des informations d'un document

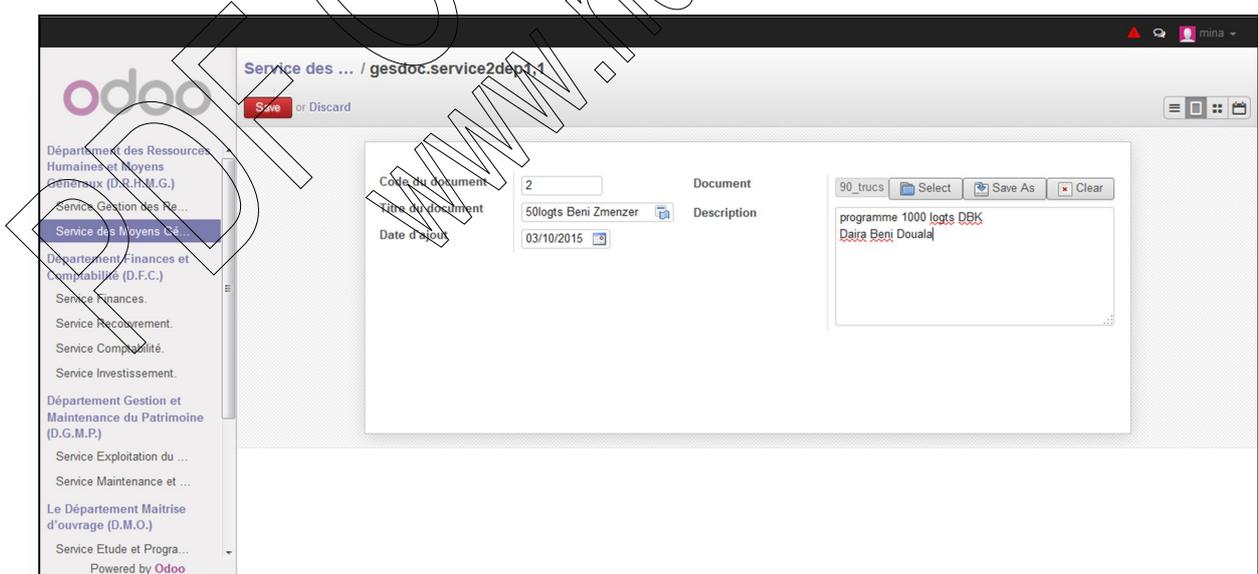


Figure. V.16. Espace Utilisateur « formulaire de modification des informations d'un document »

L'utilisateur a plusieurs vue : « *form* », « *list* », « *kanban* et *calendar* ».voila quelque figure qui montre quelque vue.

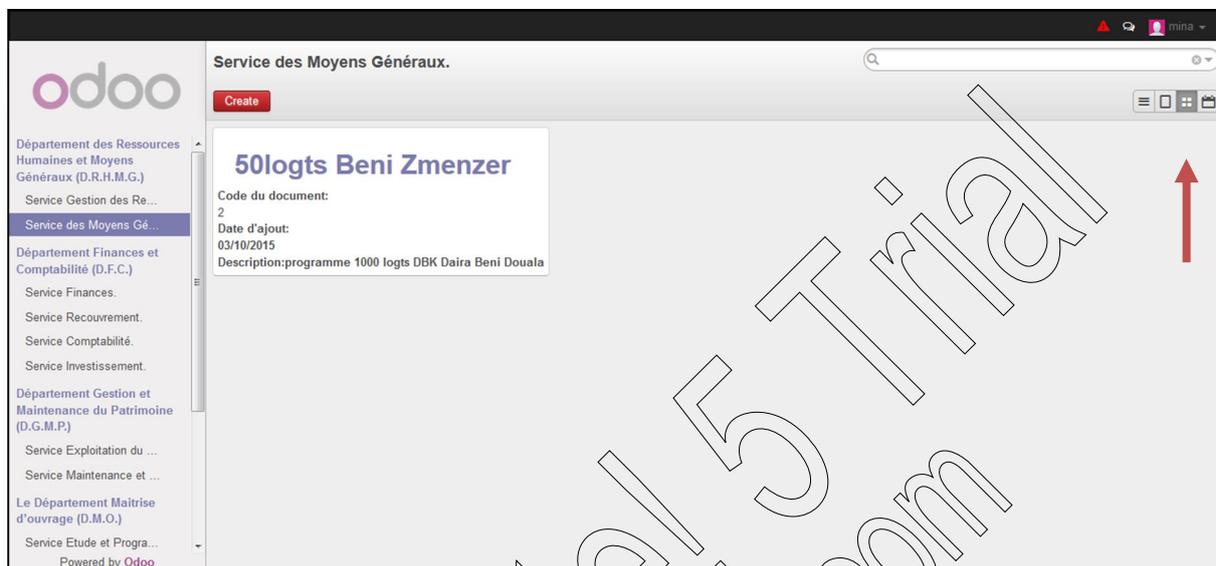


Figure. V. 17. « La vue Kanban »

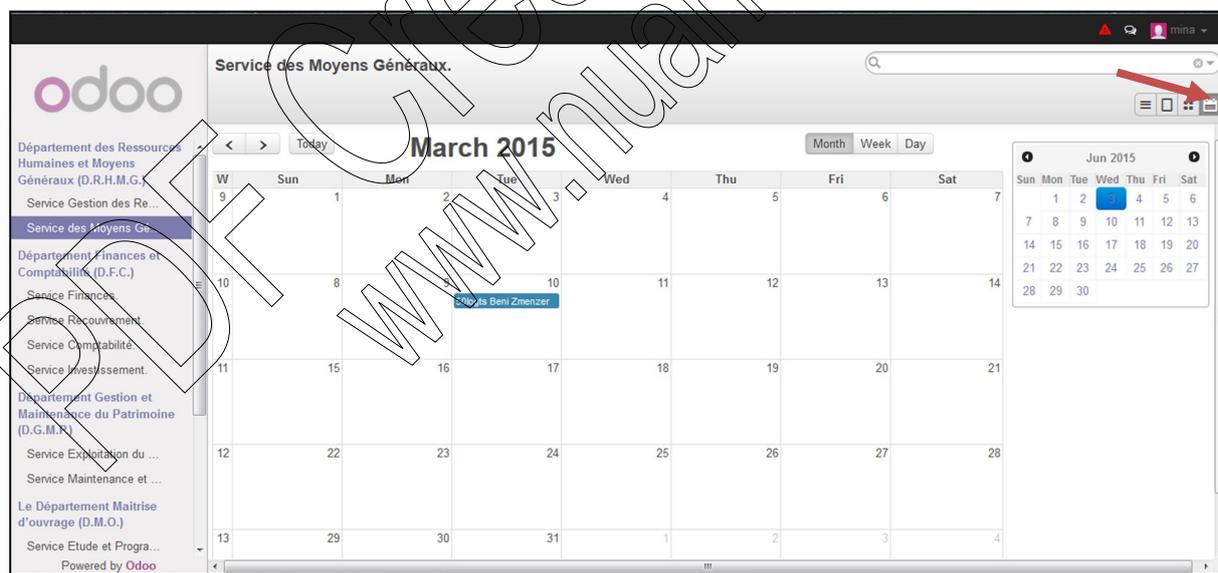


Figure. V. 18. « La vue Calendar »

Cet Utilisateur a le droit d'ajouter des documents juste pour son service, il ne peut pas ajouter/modifier et supprimer des documents des autres services cette figure montre l'absence des boutons « *creat* » et « *more* ».

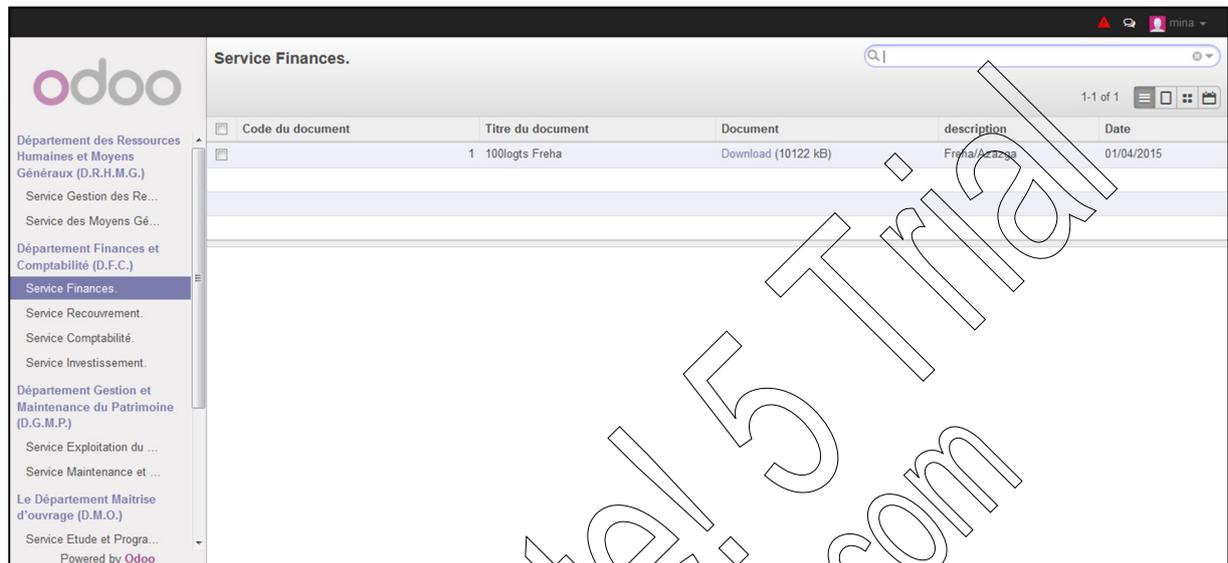


Figure. V.19.Espace Utilisateur « absence des droit de création/d'édition et de suppression »

IV. Conclusion

Après la phase d'analyse et de conception, ce dernier chapitre a été consacré à la présentation des différents outils et logiciels qui nous ont permis de réaliser notre application, à savoir : les langages de programmation et les logiciels de création. Pour conclure, nous avons présenté quelques interfaces d'utilisation de notre application.

L'entreprise d'aujourd'hui est plus consciente de l'impact de la gestion efficace des ressources internes sur l'amélioration de sa performance et par la suite de sa compétitivité dans le marché. Or, la tâche de gérer s'avère de plus en plus difficile et complexe. En effet, la croissance des activités engendre un énorme flux de données, alors que la diversité des processus fonctionnels nécessite une gestion adéquate des compétences humaines. Pour surpasser ces difficultés, toute entreprise est prête à investir des sommes considérables dans l'implantation de technologies logicielles afin d'améliorer ses services et d'accroître son agilité vis-à-vis ses clients tout en optimisant la communication au sein de ses équipes. Pour l'organisme OPGI, la gestion de l'ensemble de ses activités se fait manuellement ou par le biais des applications informatiques. La société souhaite gérer la totalité de ses activités autour d'un même système d'information, à l'aide d'un progiciel de gestion intégrée connu sous l'acronyme ERP.

C'est dans ce cadre que s'inscrit notre projet de fin d'études qui a pour objectif de mettre en place une solution de gestion intégrée au sein de l'organisme OPGI, pour cela la réalisation de ce travail nous a permis d'approfondir nos connaissances en systèmes d'informations des entreprises (comptabilité, gestion des ressources humaines, gestion commerciale, gestion de la production...etc.), de voir sur le terrain du travail les différentes tâches effectuées par le gestionnaire de différents services, et d'acquies un certain nombre de connaissances et règles de gestion. Comme elle nous a permis de nous familiariser avec les outils de développement Open ERP (PYTHON, XML, RML, OPEN OFFICE, POSTGRESQL) ainsi de maîtriser la conception orienté objet, les concepts et les outils de la méthode de conception UML.

L'application à laquelle nous avons abouti, est un module qui traite la gestion des documents du personnel OPGI. elle offre aux différents utilisateurs plus de souplesse et d'efficacité dans l'exécution de leurs tâches.

Notre module présente diverses fonctionnalités, cependant quelques améliorations et extensions peuvent être apportés à ce module.

Dans la perspective d'améliorer ce travail, il serait souhaitable de développer d'autres modules pour chaque service de cette entreprise et de les intégrer conjointement dans l'Open ERP afin d'avoir un ERP complet qui va gérer toutes les fonctionnalités de l'organisme OPGI aussi nous projetons d'utiliser cet ERP complet pour gérer toutes les filiales de l'organisme OPGI se trouvant sur tout le territoire national.

Bibliographie

- [1] : <http://www.open-source-guide.com/Solutions/Applications/Erp-pgi/Odoo>.
- [2] : [<http://www.memoireenligne.com/05/07/463m-impact-erp-performance>].
- [3] : [Philippe Norigeon,Cours PGI,ERP :Master CCA].
- [4] : [ABISSA informatique PDF (tec_erp.pdf)].
- [5] : [ERP : publié par Marc Bequigneau-DESS QUASSI-Promotion 2002/2003].
- [6] : [IMMO PARTNERS,La Technique de l'Immobilier,PARIS].
- [7] : CARLOT Guillaume , «choix d'une architecture technique,Rapport de syntese ERP_VF2.0 »,2008memoire2010/2011.
- [8] : [<http://www.openconcerto.org/fr/index.html>].
- [9] : Raphail Valyé, «ERP Open source,livre blanc,-Expert Technique – smile » ;version 1.1.2009.
- [10] : Fleur-Anne Blain, «présentation général des ERP et leur architecture modulaire »,2006.
- [11]: <http://erp.comprendrechoisir.com/comprendre/module-erp>.
- [12]: <http://erp.comprendrechoisir.com/comprendre/logiciel-erp>.
- [13]: <http://erp-logiciel.fr/erp-definitions-lexique>.
- [14]:<http://www.open-source-guide.com/Solutions/Applications/Erp-pgi/Odoo>.
- [15] :<http://www.audaxis.com/Odoo-OpenERP>.
- [16]: [SANAE BEKKAR, consultante BPM/ERP, le logiciel de gestion intégré open ERP vesrion7.0, 2014].
- [17]:<http://www.audaxis.com/Odoo-OpenERP/Fonctionnalites>.
- [18]: <http://www.wingoo-solutions.fr/blog/wingoo-monitoring/avantages-et-inconvenients-des-logiciels-open-source.html>.

[19]:<http://reseau-informatique.prestataires.com/conseils/open-source-avantages-inconvenients>.

[20] : [Ait-Mlouk Addi, Mémoire fin d'étude en vue de l'obtention du diplôme d'ingénieur d'état en télécommunication, implémentation d'une solution de géolocalisation des véhicules pour le module open TMS d'openERP à l'aide d'openGTS, CADI AYAD MARRAKECH Maroc, Promotion 2012 – 2013].

[21] :M. Mohamed Amine AZAHAF M. Mohammed OMARI, Mise en place d'un Système d'Information pour la gestion des collectivités locales (OpenERP7), pour l'obtention du diplôme d'ingénieur d'état en télécommunication, institue nationale des postes et des communications, Maroc2013/2014].

[22] : Ait-Mlouk Addi, Paramétrage et développement spécifique des modules odoo(OpenERP) Partie I, université CADI AYAD MARRAKECH].

[23]:<http://monindependancefinanciere.com/lenciclopedie/seccion-o/odoo.php>.

[24] :[http://reseau-informatique.prestataires.com/conseils/open source%C2%A0-quels domaines d'application](http://reseau-informatique.prestataires.com/conseils/open%20source%C2%A0-quels%20domaines%20d%27application).

[25] :<http://anybox.fr/blog/odoo-8-est-publie-quelles-nouveautes>.

[26] : [Guide de l'utilisateur de postgresSQL/PostGis. Université Californie.2008.]

[27] : [Un article de Wikipédia, l'encyclopédie libre.]

Introduction

Avant l'implémentation et la réalisation de toute application informatique, il convient de suivre une démarche méthodologique pour mettre en place cette dernière. Pour réaliser une application, il faut commencer par une analyse profonde et bien réfléchie, suivie d'une conception en se basant sur le formalisme UML (Unified Modeling Language) qui permet de bien représenter les aspects statiques et dynamiques d'un projet par la série des diagrammes qu'il offre. UML est une notation permettant de modéliser un problème de façon standard.

Présentation d'UML

UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. UML unifie à la fois les notations et les concepts orientés objet. Il ne s'agit pas d'une simple notation graphique, car les concepts transmis par un diagramme ont une sémantique précise et sont porteurs de sens au même titre que les mots d'un langage. C'est un langage très expressif, UML est simple à comprendre et à utiliser.

Pour apprendre à s'en servir efficacement, il faut d'abord s'appuyer sur une représentation conventionnelle de ce langage, ce qui nécessite l'assimilation de trois éléments fondamentaux : les briques de base d'UML, les règles qui déterminent la manière de les assembler et quelques mécanismes généraux qui s'appliquent à ce langage.

Pourquoi la modéliser :

Un modèle est une simplification de la réalité qui permet de mieux comprendre le système à développer.

Il permet :

- De visualiser le système comme il est ou comme il devrait l'être.
- De valider le modèle vis à vis des clients
- De spécifier les structures de données et le comportement du système.
- De fournir un guide pour la construction du système.
- De documenter le système et les décisions prises.

Les principes de la modélisation:

- Le modèle doit être connecté au monde réel.
- Un modèle peut être exprimé avec différents niveaux de précision.
- Un simple modèle n'est pas suffisant, il y a plusieurs façons de voir un système.
 - ✓ Plan de masse.
 - ✓ Vue de face, de côté, ...
 - ✓ Plan des niveaux.
 - ✓ Plan électrique.
 - ✓ Plan de plomberie.
 - ✓ Plan des calculs de construction.

Qu'est-ce que UML :

UML, c'est l'acronyme anglais pour « Unified Modeling Language » On le traduit par « Langage de modélisation unifié ». La notation UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc.

À quoi sert UML ?

UML permet de définir et de visualiser un modèle, à l'aide des diagrammes. Un diagramme UML est une représentation graphique, qui s'intéresse à un aspect précis du modèle ; c'est une perspective du modèle, pas "le modèle" lui-même. Chaque type de diagramme UML possède une structure (les types des éléments de Modélisation qui le composent sont prédéfinis). Un type de diagramme UML véhicule une sémantique précise. Combinés, les différents types de diagrammes UML offrent une vue complète des aspects statiques et dynamiques d'un système. Sa notation graphique et son aspect visuel en font aussi de lui un support de communication.

Il permet aussi de :

- Visualiser : chaque symbole graphique a une sémantique
- Spécifier : de manière précise et complète, sans ambiguïté,
- construire : les classes, les relations SQL peuvent être générées automatiquement
- documenter : les différents diagrammes, notes, contraintes, exigences seront présentées dans un document.

Historique :

UML est né en octobre 1994 chez Rational Software Corporation à l'initiative de G. Booch et de J. Rumbaugh et Ivar Jacobson. Issu en 1996 de la pratique industrielle et de la modélisation des systèmes logiciels.

UML 1.1 a été standardisé par l'OMG (Object Management Group) le 17 novembre 1997 suite à la demande émanant de la collaboration de plusieurs entreprises (Hewlett-Packard, IBM, i-Logix, ICON Computing, IntelliCorp, MCI Systemhouse, Microsoft, ObjecTime, Oracle, Platinum Technology, Ptech, Rational Software Corporation, Reich Technologies, Softeam, Sterling Software, Taskonet Unisys).

La version actuelle (depuis juin 1999) est UML 1.3 (la version 1.4 sera bientôt prête, afin de préparer la prochaine version 2.0).

Evolution d'UML

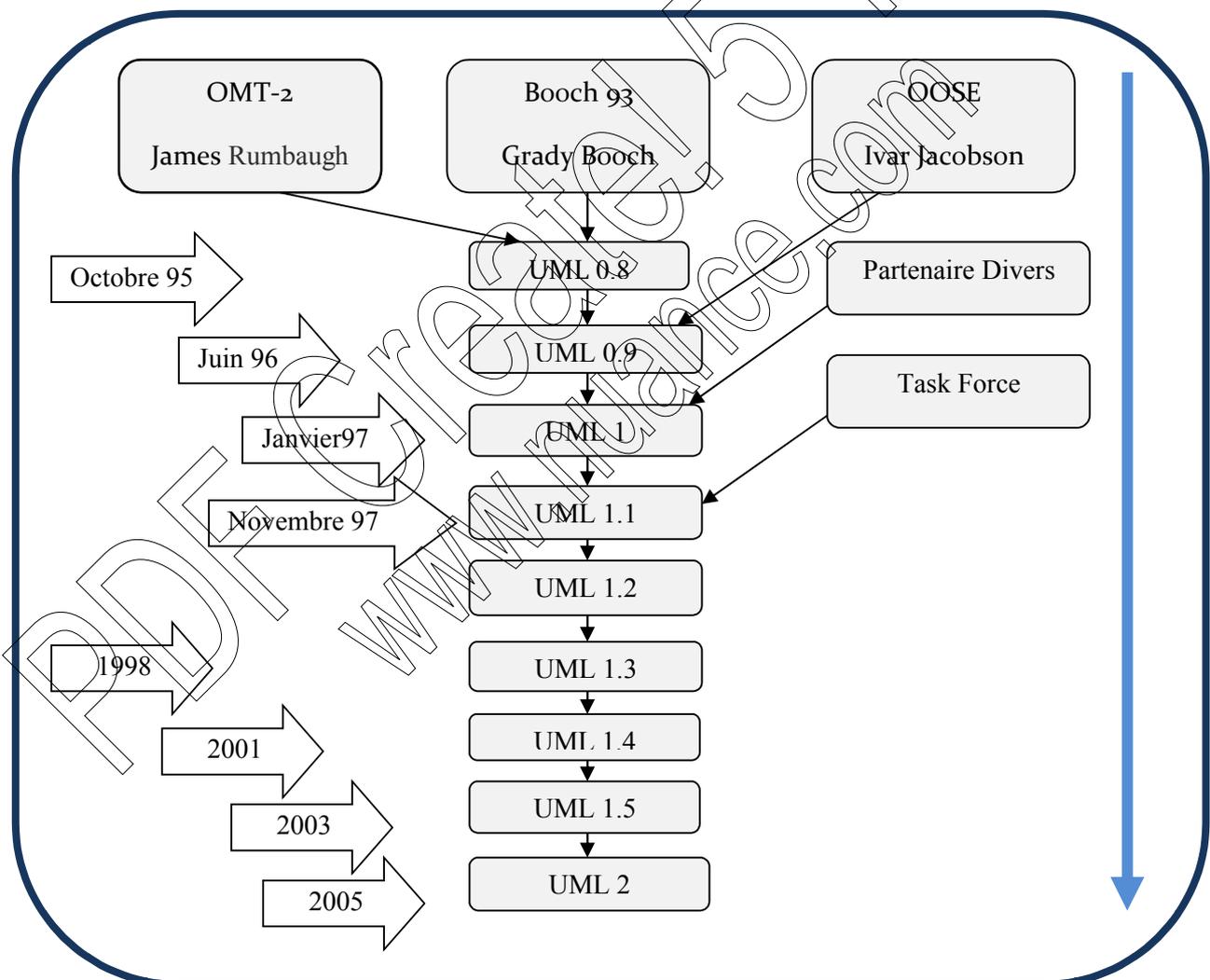


Figure 1 : représentation d'évolution d'UML.

Dans quels domaines UML peut-il être utilisé :

UML est destiné tout au système à forte composante logicielle ; et a été utilisé avec succès dans des domaines tels que :

- ✓ Les systèmes informatiques d'entreprise ;
- ✓ Les services bancaires et financiers ;
- ✓ Les télécommunications ;
- ✓ Les transports ;
- ✓ La défense / l'aérospatiale ;
- ✓ Le commerce de détail ;
- ✓ L'électronique médicale ;
- ✓ Les sciences ;
- ✓ Les services distribués basés sur le web.

L'utilisation d'UML, n'est pas limitée à la modélisation de logiciels. En effet, ce langage est suffisamment expressif pour permettre de modéliser des systèmes qui n'appartiennent pas à cette catégorie.

Les diagrammes en UML :

UML est un langage graphique et repose sur neuf types de diagrammes. Chacun de ces diagrammes utilise le même principe : les concepts sont représentés par des symboles, et les relations entre les concepts sont représentées par des lignes qui relient les symboles. Le vocabulaire et la grammaire d'UML sont très réduits.

Les trois éléments de base en UML

- 1) les blocs de base pour construire
 - les entités utilisées (Entités structurelles, Entités de comportement, Entités de regroupement, Entité d'annotation)
 - la notion de relation
 - les diagrammes
- 2) les règles à observer pour utiliser ces blocs de base
 - règles sémantiques
 - règles de présentation
- 3) les mécanismes communs
 - spécification
 - présentation
 - extension des modèles

La modélisation avec l'UML

1. Les concepts

L'UML utilise une notation simple, précise, et homogène permettant un bon rendu visuel. Elle décrit la réalité plutôt que le processus de réalisation.

1.1 Les acteurs

Un acteur représente un rôle que peut jouer l'utilisateur ou toute entité externe pouvant interagir avec le système. Autrement dit c'est un objet actif qui utilise les fonctions du système.

L'acteur dans l'UML est représenté comme suit :

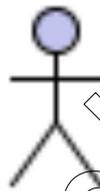


Figure 2 : représentation graphique d'un acteur dans l'UML.

1.2 Les composants

Il existe 4 grands types de composants en UML :

1.2.1 Les composants structuraux : ce sont les parties statiques d'un modèle. on en dénombre cinq voici les noms et la représentation graphique.

❖ Une classe

- **Définition** : une classe est une description abstraite (condensée) d'un ensemble d'objets du domaine de l'application : elle définit leur structure, leur comportement et leurs relations.
- **Représentation** : les classes sont représentées par des rectangles compartimentés:
 - le 1er compartiment représente le nom de la classe
 - le 2ème compartiment représente les attributs de la classe
 - le 3ème compartiment représente les opérations de la classe

- **Formalisme d'une classe :**

<nom classe>
Attribut_1: int Attribut_2: int Attribut_3: int : :
Operation_1 (): void Operation_2 (): void

Figure 3 : représentation graphique d'une classe.

- **Attribut :** Un attribut présente la modélisation d'une information élémentaire représentée par son nom et son format.
- **Opération :** l'opération représente un élément de comportement des objets, défini de manière globale dans la classe.

Une opération est une fonctionnalité assurée par une classe. La description des opérations peut préciser les paramètres d'entrée et de sortie ainsi que les actions élémentaires à exécuter.

❖ Une interface

Une interface est un ensemble d'opérations utilisées pour définir un service d'une classe ou d'un composant. Elle est représentée par un cercle, mais elle peut être représentée par une classe stéréotype en plaçant le mot clé « interface » avant le nom de la classe. Dans tous les cas, elle doit être accompagnée d'une classe qui implémente les opérations définies par l'interface.

❖ Cas d'utilisation

- **Définition :** Un cas d'utilisation décrit un ensemble de séquences d'actions, qu'un système exécute pour produire un résultat observable pour un acteur particulier. Il modélise un service rendu par le système.

- **Formalisme de cas d'utilisation :** un cas d'utilisation est représenté par une ellipse

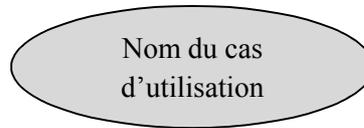


Figure 4: représentation graphique d'un cas d'utilisation.

- **La relation :** Elle exprime l'interaction existant entre un acteur et un cas d'utilisation.

Il existe 3 types de relations entre cas d'utilisation :

- la relation de généralisation
- la relation d'extension
- la relation d'inclusion

➤ **Relation de généralisation :**

Un cas A est une généralisation d'un cas B si B est un cas particulier de A. Le symbole utilisé pour la généralisation entre ces utilisations est une flèche avec un trait plein dont la pointe est un triangle fermé désignant le cas le plus général.

➤ **Relation d'inclusion :**

Elle indique que le cas d'utilisation source contient aussi le comportement décrit dans le cas d'utilisation destination. L'inclusion a un caractère obligatoire, la source spécifiant à quel endroit le cas d'utilisation cible doit être inclus. Cette relation permet ainsi de décomposer des comportements et de définir des comportements partageables entre plusieurs cas d'utilisation.

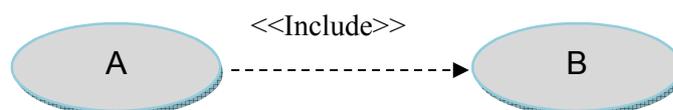


Figure 5: représentation graphique d'une relation d'inclusion.

➤ **Relation d'extension :**

Elle indique que le cas d'utilisation source ajoute son comportement au cas utilisation destination. L'extension peut être soumise à condition. Le comportement ajouté est inséré au niveau d'un point d'extension défini dans le cas d'utilisation destination. Cette relation permet de modéliser les variantes de comportement d'un cas d'utilisation (selon les interactions des acteurs et l'environnement du système).

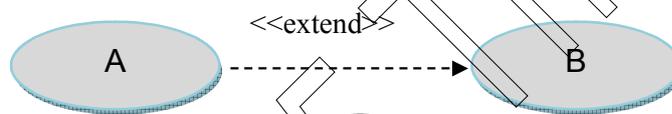


Figure 6: représentation graphique d'une relation d'extension.

❖ **Un composant**

- **Définition :** un composant est une partie physique remplaçable d'un système qui se conforme à un ensemble d'interface et en permet la réalisation dans un système. Il peut être du code, du script, un fichier de commande...etc.
- **Formalisme d'un composant :**



Figure 7: représentation graphique d'un composant.

❖ Un nœud

- **Définition :** Le nœud est l'élément physique qui existe au moment de l'exécution et représente une ressource de calcul. En règle générale, il a au moins de la mémoire et souvent des capacités de traitement.

- **Formalisme d'un nœud**

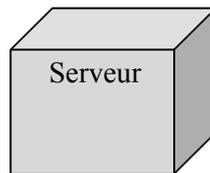


Figure 8: représentation graphique d'un nœud.

1.2.2 Les composants comportementaux : ils représentent la partie dynamique des modèles UML, il existe deux types d'élément fondamentaux.

❖ Une interaction

- **Définition :** une interaction est un comportement qui correspond à un envoi de message entre objet pour accomplir une tâche bien définie.
- **Formalisme d'une interaction :** un message est représenté par une flèche accompagnée du nom du message.

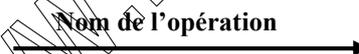


Figure 9: représentation graphique d'une interaction.

❖ Un état

- **Définition :** un automate à état finis est un comportement qui précise les séquences d'états d'un élément au cours de son existence.
- **Formalisme d'un état :** un état est représenté par un rectangle aux angles arrondis.

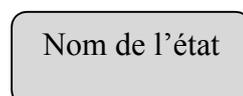


Figure 10: représentation graphique d'un état.

1.2.3 Les composants de regroupement : les éléments de regroupement représentent les parties organisationnelles des modèles UML. Ce sont des boîtes dans lesquelles un modèle peut être décomposé. Il existe un seul type fondamental d'élément de regroupement : « les paquetages ».

❖ **Un paquetage**

- **Définition :** un paquetage est un mécanisme général qui permet de regrouper des éléments structurels, des éléments comportementaux, ou même d'autres paquetages.
- **Formalisme d'un paquetage :** un paquetage est représenté par un dossier étiqueté, il contient son nom parfois son contenu.

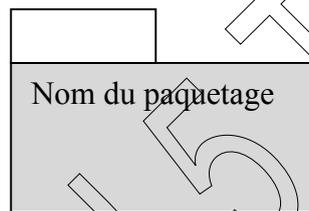


Figure 11: représentation graphique d'un paquetage.

1.2.4 Les composants d'annotation : ils représentent les parties explicatives des modèles UML. Ce sont les commentaires qui peuvent accompagner tout élément dans un modèle à des fins de description, d'explication et de remarque.

- **Formalisme d'une annotation :** une annotation est représentée par un rectangle écorné qui contient N commentaire textuel ou graphique.

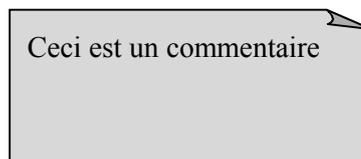


Figure 12: représentation graphique d'une annotation.

2. Les relations dans l'UML

Il existe quatre types de relations en UML. Chacune symbolisée différemment :

- ❖ **L'association** : l'association est une relation structurelle précisant que les objets d'une classe sont en relation avec les objets d'une autre classe.



Figure 13: représentation graphique d'une association.

- ❖ **L'agrégation** : dans l'UML l'agrégation n'est pas un type de relation mais une variante de l'association. Elle représente une association non symétrique dans laquelle une des extrémités (agrégée) joue un rôle prédominant par rapport à l'autre extrémité (agrégat).

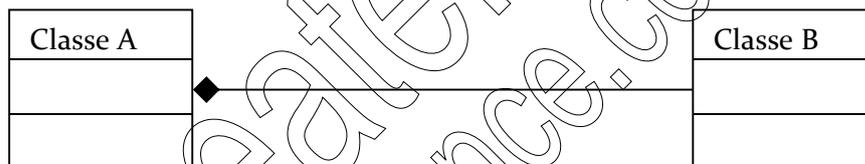


Figure 14: représentation graphique d'une agrégation.

- ❖ **Composition** : la composition est un cas particulier de l'agrégation dans laquelle la vie des composants est liée à celle des agrégats. La composition implique en plus de l'agrégation, une coïncidence des durées de vie des composants : la destruction de l'agrégat implique automatiquement la destruction de tous les composants liés.

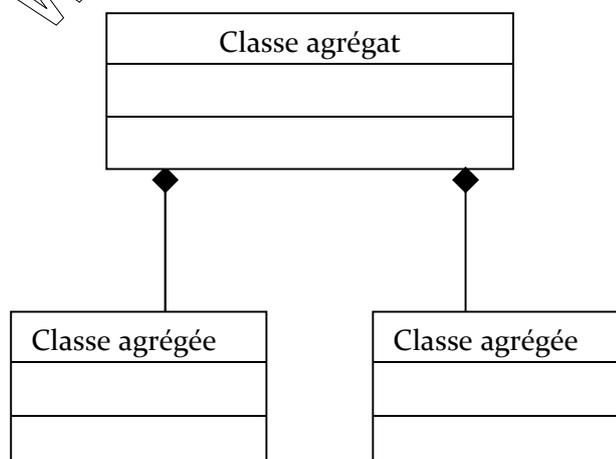


Figure 15: représentation graphique d'une composition.

- ❖ **La généralisation :** une généralisation est une relation transitive entre une classe nommée *surclasse* et une ou plusieurs autres classes nommées *sous-classes* ; chaque sous classe hérite des caractéristiques (attributs et opérations) de la surclasse

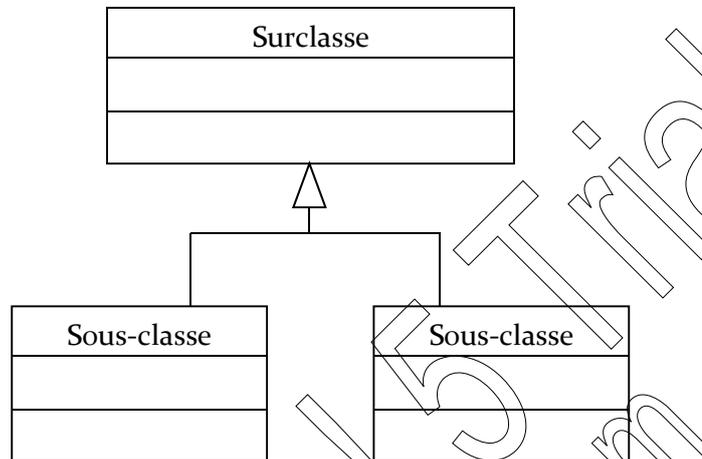


Figure 16: représentation graphique d'une généralisation.

3. Autres Composants :

- ❖ **Objet :** Les objets sont des instances des classes, et sont rangés horizontalement. La représentation graphique pour un objet est similaire à une classe (un rectangle) précédée du nom d'objet (facultatif).

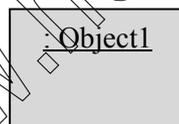


Figure 17: représentation graphique d'un objet.

- ❖ **Ligne de vie :** les lignes de vie, LifeLine, identifient l'existence de l'objet par rapport au temps. La notation utilisée pour une ligne de vie est une ligne pointillée verticale partant de l'objet.



Figure 18: représentation graphique d'une ligne de vie.

- ❖ **Activation** : les activations, sont modélisées par des boîtes rectangulaires sur la ligne de vie. Elles indiquent quand l'objet effectue une action.



Figure 19: représentation graphique d'une activation.

- ❖ **Message** les messages, modélisés par des flèches horizontales entre les activations, indiquent les communications entre les objets.



Figure 20: représentation graphique d'un message.

Les principaux 9 diagrammes en UML

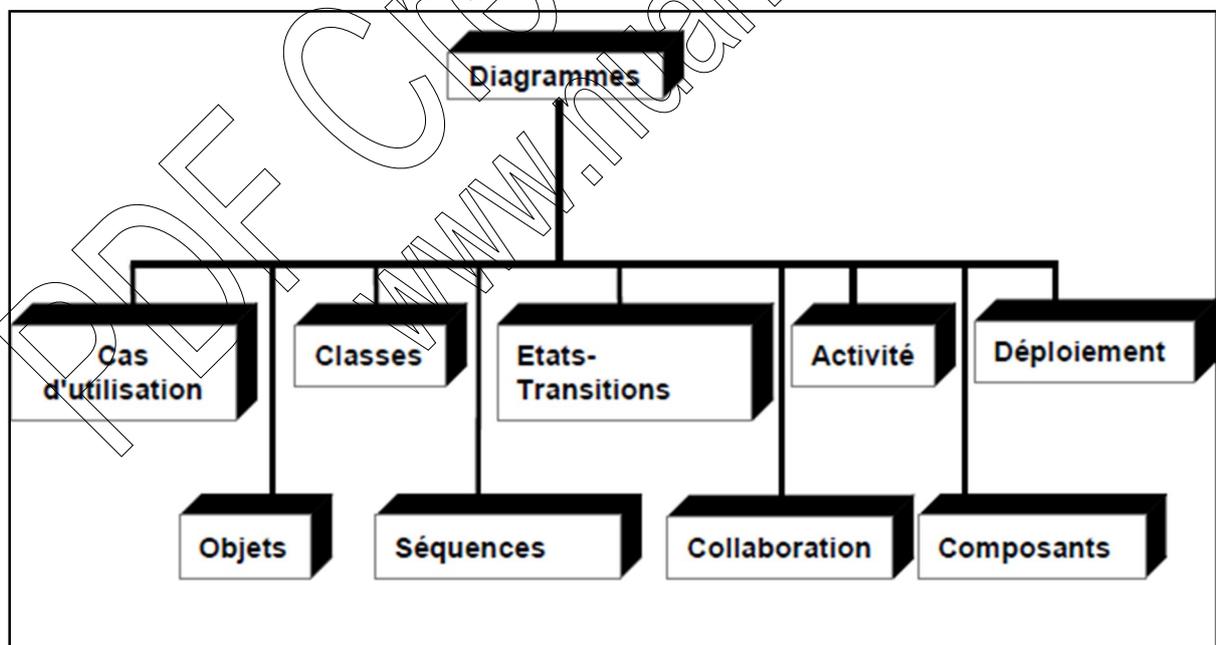


Figure 21 : Les principaux diagrammes d'UML.

Les vues d'UML :

Toutes les vues proposées par UML sont complémentaires les unes des autres, elles permettent de mettre en évidence différents aspects d'un logiciel à réaliser.

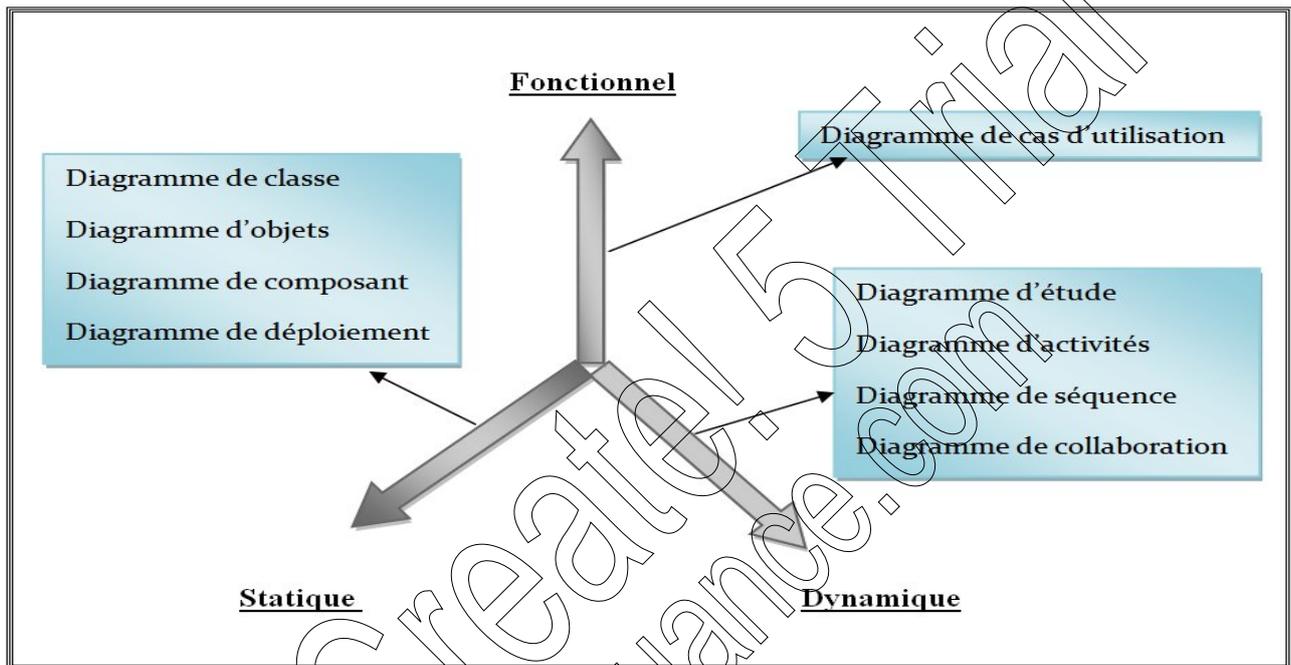


Figure 22 : Diagramme représentant les vues d'UML.

Présentation les différents vues :

1. Le mode statique : Il est utilisé dans l'activité de spécification des besoins. Le mode de représentation statique ou structurel s'appuie sur le diagramme Le diagramme de classe.

1.1 Diagramme de classe

Diagramme de classe est l'outil principale de l'aspect statique du système. Il montre l'ensemble des classes et leurs relations. Ce sont les diagrammes les plus utilisés dans la modélisation des systèmes orientés objet.

2. **Le mode dynamique** : le mode dynamique ou comportementale s'appuie sur les diagrammes suivants :

2.1 Les diagrammes séquence

Diagramme de séquence représente les échanges de message entre objet. Il permet de représenter un processus de façon simplifiée et permet de visualiser les messages par une lecture de haut en bas. L'axe vertical représente le temps, l'axe horizontal les objets qui collaborent. Une ligne verticale en pointillé est attachée à chaque objet et représente sa durée de vie.

Formalisme : Dans un diagramme des séquences, les classes et les acteurs sont énumérés en colonnes, avec leurs lignes de vie verticales indiquant la durée de vie de l'objet.

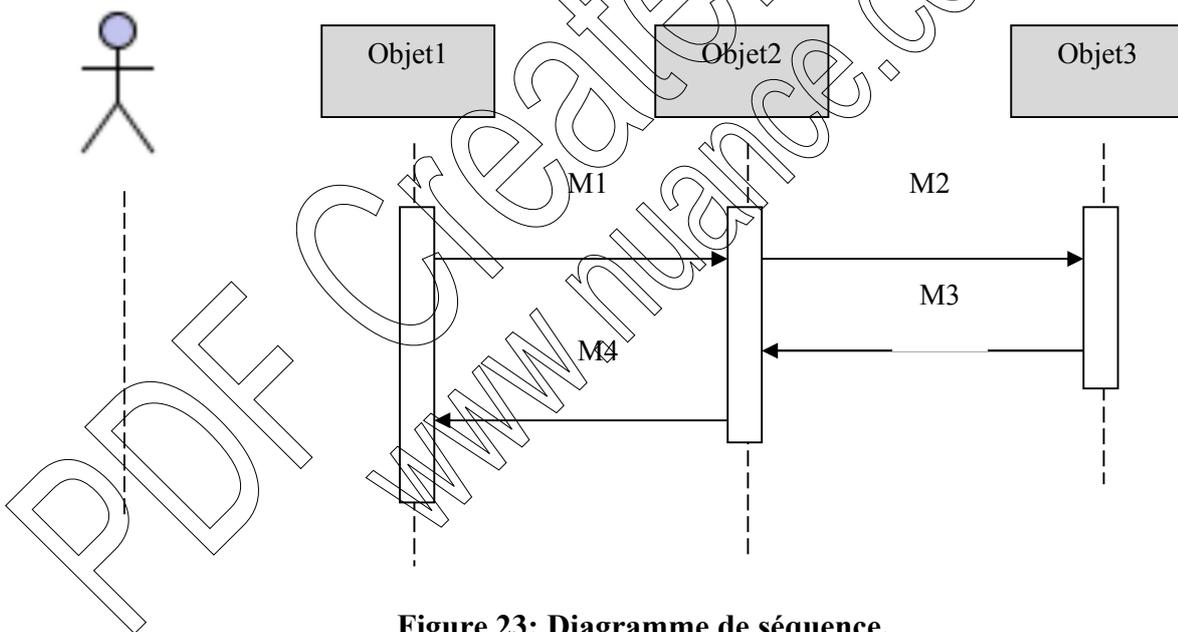


Figure 23: Diagramme de séquence.

2.2 Les diagrammes de collaboration

Le diagramme de collaboration permet de mettre en évidence les interactions entre les différents objets du système.

Dans le cadre de l'analyse, il sera utilisé

- Pour préciser le contexte dans lequel chaque objet évolue.

- Pour mettre en évidence les dépendances entre les différents objets impliqués dans l'exécution d'un processus ou d'un cas d'utilisation.

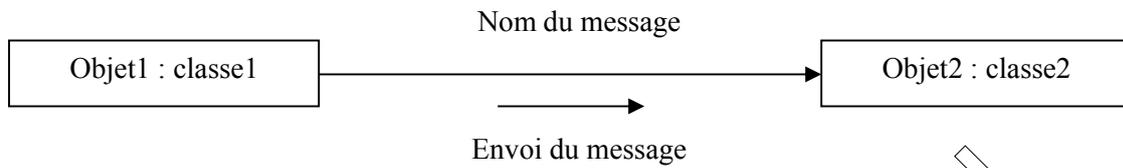


Figure 24: Diagramme de collaboration.

3. Le mode fonctionnel : le mode fonctionnel s'appuie sur le diagramme suivant :

3.1 Les diagrammes de cas d'utilisation :

- **Définition :** le diagramme de cas d'utilisation permet de recueillir, d'analyser et d'organiser les besoins des utilisateurs, et recenser les grandes fonctionnalités d'un système. Il s'agit de la première étape UML d'analyse d'un système.

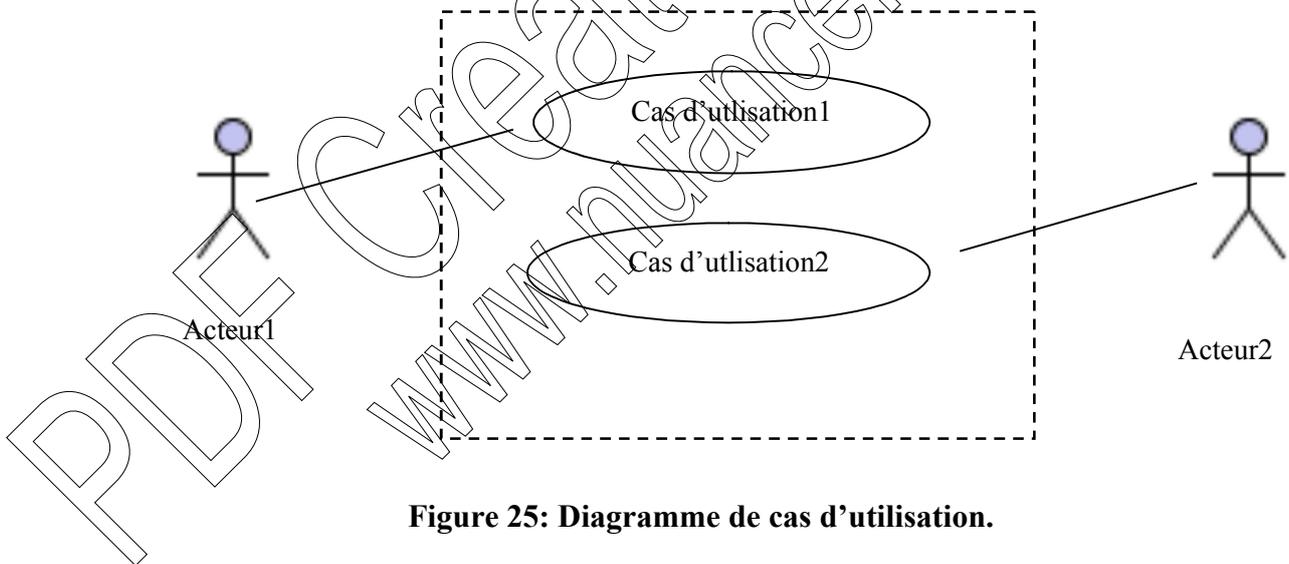


Figure 25: Diagramme de cas d'utilisation.

Conclusion

Cette annexe a proposé un survol et une présentation brève des principaux concepts de modélisation UML ainsi que leur notation.

Dans le **chapitre IV** nous avons mis en œuvre les différents processus nécessaires à la conception de notre application et cela, en montrant d'abord de manière globale l'architecture du système et l'interaction de ses différents composants. Nous avons ensuite détaillé le processus de conception objet, de manière progressive, à travers les diagrammes UML des cas d'utilisation, de séquence UML, et le diagramme de classes. Ce qui nous a permis d'aboutir de façon efficace à du code résultant proche de l'implémentation. La mise en œuvre de cette dernière étape est explicitée dans le chapitre suivant.

PDF Create! 5 Trial
www.nuance.com