

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

<u>Université 20 Août 1955 – Skikda</u>		<u>جامعة 20 أوت 1955 - سكيكدة</u>
<u>Faculté des Sciences</u>		<u>كلية العلوم</u>
<u>Departement d'informatique</u>		<u>قسم الإعلام الآلي</u>

**Mémoire De fin d'étude en vue de l'obtention du Diplôme de
Master en Informatique**

Option : Génie Logiciel Avancés et Applications (GLAA)

&

Systemes d'Information Avancés et Applications (SIAA)

Sujet :

**Conception et réalisation d'un module ERP pour la
gestion des escales au niveau de l'Entreprise Portuaire
de Skikda**

Réalisé par les étudiants :

- BOUREKKAB Chaima
- BELKAHLA Roumaissa

Encadré par :

Dr ZEGHIDA Djamel
Co-encadreur:
Dr LAROUM Toufik

Année Universitaire 2022-2023

Remerciements

Nous tenons à remercier DIEU le tout puissant de nous avoir donné la santé, le courage et la foi pour réaliser ce modeste travail avec volonté.

Nous voudrions au premier lieu remercier Dr. ZEGHIDA Djamel et Dr. LAROUM Toufik, en tant qu'encadrant, pour leur patience, disponibilité et surtout leurs judicieux conseils, qui ont contribué à alimenter notre réflexion.

Nous remercions les membres du jury pour avoir accepté d'examiner et évaluer notre travail.

Nous adressons nos sincères remerciements à tout le personnel de l'entreprise portuaire Skikda pour leur prise en charge, spécialement Mr. MGUELEB Ali directeur de la qualité et systèmes d'information, Mr. BENRABAH Omar directeur de la capitainerie, Mr. SID Ali chef département système d'information et Mme BRAHIMI Wafa Chef service maintenance et système d'information, pour avoir eu la patience de répondre à nos innombrables questions et de nous avoir donné l'occasion de réaliser notre travail sur terrain.

Un remerciement spécial pour les membres de l'équipe qui ont travaillé également sur ce projet, nos collègues BOUZNAD Housseem Eddine, FENAZI Idris et LAOUDJA Djihane, sous la supervision de Mr. SELLAMI Samir.

Enfin, nous tenons à remercier tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail.

Dédicaces

A mon père BOUREKKAB Walid grâce à toi j'ai pu aller à l'école. En guise de reconnaissance,
trouve ici mon amour filial.

Ma réussite est la tienne ! Qu'Allah t'accorde longue vie dans la santé!

A ma mère BOUTAGHENE Fatiha

Par les inestimables sacrifices que tu as consentis pour moi, tu as tant
souhaité que je parvienne à ce but.

Je te serai reconnaissant toute ma vie, qu'Allah t'accorde longue vie dans
la santé !

A mes chères sœurs, Sara et ses enfants Rawan et Nouh, Kawtar et sa fille Nadine, Khadija et ses
enfants Sallah Eddine et Zaid, Amel

A ma chère oncle, Mohammed Tahar Pour sa entente et sa sympathie.

Pour leurs indéfectibles soutiens et leur patience infinie.

A mes cousines Nour, Nada, safà, Rihab, Khalil EL Rahman et Anis

A mon amie intime, Katr Ennada

Qui m'a aidé et supporté dans les moments difficiles.

A mes chers amis, Meriem, Katr Ennada, Ferial, Roumaïssa, djihane, Housseem Eddine, Idris et
Karim

Pour leurs aides et supports dans les moments difficiles.

A toute ma famille, BOUREKKAB et BOUTAGHENE

A tous mes autres ami(e),

A tous ceux que j'aime et ceux qui m'aiment.

< Chaima >

Dédicaces

C'est avec profonde gratitude et sincères mots, que je dédie ce modeste travail de
fin d'étude à

mes chers parents qui ont sacrifié leur vie pour ma réussite et m'ont éclairé le
chemin par leurs conseils judicieux.

À mon papa Mouloud, ma boussole d'or, ma source de motivation et mon idole
dans la vie , sans lui je suis rien!

Maman, mon premier amour, Je serai reconnaissante toute ma vie, qu'Allah
t'accorde une longue vie .

Je dédie aussi ce travail à mes frères et sœurs, ma famille, mes amis, tous mes
professeurs qui m'ont enseigné et à tous ceux qui sont chers.

< Roumaïssa >

الملخص

الهدف من هذه الرسالة هو تصميم وإنشاء حزمة برامج إدارة متكاملة لإدارة ومراقبة رسو السفن ، وهي نتيجة دراسة متعمقة لنظام كان موجوداً خلال فترة تدريبنا في المؤسسة المينائية لولاية سكيكدة .
استخدمنا لغة نمذجة UML للتصميم ، وبالنسبة للبرمجة اخترنا Odoo كحزمة برامج مفتوحة المصدر مع Python التي تحترم نموذج تصميم MVC ، بالإضافة إلى PostgreSQL كنظام إدارة قواعد البيانات.
كلمات مفتاحية: شركة ميناء سكيكدة ، تخطيط موارد المؤسسات ، أودو ، رسو السفن ، UML

Résumé

L'objectif de ce mémoire est de concevoir et réaliser un ERP pour la gestion et le suivi des escales des navires, c'est le résultat d'une étude approfondie d'un système existant lors de notre stage à l'Entreprise Portuaire de Skikda. Nous avons utilisé le langage de modélisation UML pour la conception, et pour la programmation nous avons choisi Odoo comme un progiciel open source avec le langage python qui respecte le modèle de conception MVC, ainsi que PostgreSQL en tant que système de gestion de base de données.

Mots-clés : Entreprise Portuaire de Skikda, ERP ,Odoo, Suivi des escales, UML

Abstract

The objective of this thesis is to design and create an ERP for the management and monitoring of ship calls, it is the result of an in-depth study of an existing system during our stage at the Port Company of Skikda . We used development process, UML modeling language for design, and for programming we chose Odoo as an open source software package with python language which follows MVC design pattern, as well than PostgreSQL as a database management system.

Keywords : Skikda Port Company,ERP, Odoo, ship calls management, UML.

Table des matières

Remerciements	2
Dédicaces	3
Dédicaces	4
الملخص	5
Résumé	5
Table des matières	6
Table des figures	10
Liste des tableaux	12
Table des abréviation	13
Introduction générale	14
1. Introduction :	17
2. Entreprise Portuaire Skikda: [1]	17
2.1. Présentation générale de l'organisme d'accueil:	17
2.2. Aperçu historique:	18
2.3. Caractéristiques physiques des ports de l'entreprise:	20
3. Organisation et activité :	21
3.1. Activité :	21
3.2. Capacités portuaires de transit :	22
3.3. Surfaces commerciales :	22
3.4. Gamme des prestations de services :	22
4. L'organigramme de l'entreprise et ses missions :	23
4.1. Direction générale :	24
4.2. Direction d'audit et gestion des risques :	25
4.3. Direction financière et comptable:	25
4.4. Direction sûreté interne de l'entreprise (SIE) :	26
4.5. Direction de la qualité et des systèmes d'information :	26
4.6. Direction des affaires générales :	27
4.7. Direction des ressources humaines :	27
4.8. Direction du domaine et des travaux :	29

4.9. Direction du matériel :	29
4.10. Direction de la capitainerie :	30
4.11. Direction commerciale :	32
4.12. Direction du remorquage :	33
4.13. Direction de la Manutention et de l’acconage:	34
5. Présentation du Département informatique :	35
6. Etude de l’existant :	38
6.1. Présentation de système d’information SIGNE :	38
7. Problématique et Solution proposée:	40
7.1. Problématique:	40
7.2. Solution proposée:	40
8. Conclusion:	42
1. Introduction :	43
2. Présentation des ERP:	43
2.1. Définition:	43
2.2. Historique:	43
2.3. Principe de base d’un ERP:	44
2.4. Objectif des ERP:	45
2.5. Avantages et inconvénients des ERP:[8]	46
2.6. Les principaux éditeur d’ERP:	47
2.7. Les types ERP:	50
2.8. L’architecture d’un ERP:	51
3. Présentation ODOO	54
3.1. Définition:	54
3.2. Historique:	55
3.3. Source d’ODOO:	56
3.4. Composants principaux du module d’ODOO:	58
3.5. Les fonctionnalités d’ODOO:	59
3.6. Particularités de l’ERP ODOO:	60
4. CONCLUSION:	60
1. Introduction	61
2. Objectif de notre projet:	61
3. Unified Modeling Language (UML):	61
3.1. Définition :	61
3.2. A quoi sert UML?	62

4. Analyse:	63
4.1. Définition:	64
4.1.1. Spécification des besoins:	64
4.1.2. Acteurs:	64
4.1.3. Cas d'utilisation:	64
4.1.4. Relation:	65
4.1.5. Scénario :	68
4.2. Le modèle métier global d'EPS:	68
4.2.1. Décomposition du modèles global d'EPS en sous modèles:	69
4.2.2. Réalisation de la carte de contexte d'EPS:	71
4.3. Identification des acteurs :	73
4.3.1. Acteur principaux:	73
4.3.2. Acteur secondaire:	73
4.4. La spécification de besoins:	74
4.4.1. Besoins fonctionnels:	74
4.4.2. Besoin non fonctionnels:	74
5. La conception de microservice <ESCALE>:	75
5.1. diagramme de contexte:	75
5.2. Diagramme de cas d'utilisation:	75
5.2.1. Diagramme de cas d'utilisation de microservice <ESCALE>:	76
5.2.1.1. Cas d'utilisation associés à Communauté EPS:	76
5.2.1.2. Cas d'utilisation associé au directeur capitainerie:	77
5.2.1.3. Cas d'utilisation associés au Gestionnaire des escales:	78
5.2.1.4. Cas d'utilisation associés à l'officier de port :	79
5.2.1.5. Cas d'utilisation associés à l'officier radio :	80
5.2.2. Description textuelle des cas d'utilisation:	80
5.2.2.1. Cas d'utilisation « Authentifier »:	81
5.2.2.2. Cas d'utilisation « Chargement des nouvelles annonces »:	82
5.2.2.3. Cas d'utilisation « Ajouter à la liste attendu »:	83
5.2.2.4. Cas d'utilisation « Valider des navires en Rade » :	83
5.2.2.5. Cas d'utilisation « Ajouter - Prévision d'entrée »:	84
5.2.2.6. Cas d'utilisation « Modifier - Prévision d'entrée »:	85
5.3. Diagramme de séquence :	87
5.3.1. Diagramme de séquence du cas d'utilisation « Authentifier »:	88

5.3.2. Diagramme de séquence du cas « Charger les nouvelles annonces - Ajouter à la liste attendu »:	89
5.3.3. Diagramme de séquence du cas « Gérer les navires en rade - Ajouter à la liste enrade » :	90
5.3.4. Diagramme de séquence du cas « Prévision d'entrée »:	91
5.3.5. Diagramme de séquence du cas « Exécuter une décision »:	92
5.4. Diagramme de classe:	92
5.4.1. Présentation des classes et leurs attributs :	93
Dans le tableau 10, nous allons décrire en détails les différentes classes schématisées.	93
5.5. Présentation d'un modèle relationnel:	101
5.6. Règles de gestion:	102
6. Conclusion :	103
1. Introduction :	104
2. Les langages et environnements de développement :	104
3. les étapes de développement de module « gestion des escales » :	108
3.1. Installation Odoo16 sous Windows:	108
3.2. Configuration de notre module:	112
3.3. Développement de module :	116
● Lire et utiliser le paramétrer setting :	129
4. Conclusion :	129
Conclusion générale	130
Références	132

Table des figures

Figure 1 : Logo d'EPS.....	18
Figure 2 : Organigramme de l'entreprise portuaire Skikda	24
Figure 3 : Page d'accueil du système SIGNE.....	39
Figure 4 : Schéma d'évolution des ERP depuis les années 60[3].....	44
Figure 5 : Schéma d'un ERP [5].....	45
Figure 6 : Les composants d'un ERP [7].....	46
Figure 7 : Architecture technique d'un ERP [12].....	51
Figure 8 : Les modules principaux d'un ERP [13].....	52
Figure 9 : Architecture technique de Odoo [19].....	57
Figure 10 : Présentation de l'architecture du modèle MVC[19].....	58
Figure 11 : Les fonctionnalités d'odoo [20].....	59
Figure 12 : Les différents types de diagramme [24].....	63
Figure 13 : Formalisme d'un acteur en UML [25].....	64
Figure 14 : Représentation graphique d'un d'un cas d'utilisation [26].....	65
Figure 15 : acteur et cas d'utilisation [27].....	66
Figure 16 : Formalisme de la relation de généralisation[28].....	66
Figure 17 : Représentation graphique de l'inclusion (partager).....	67
Figure 18 : Représentation graphique de l'inclusion (decomposer).....	67
Figure 19 : Représentation graphique de l'extension.....	68
Figure 20 : Modèle métier global d'EPS.[1].....	69
Figure 21 : Le pattern bounded contexte[1].....	71
Figure 22 : La carte de contexte [1].....	72
Figure 23 : Diagramme de contexte	75
Figure 24 : Cas d'utilisation associés à Communauté EPS.....	76
Figure 25 : Cas d'utilisation associés au directeur capitainerie.....	77
Figure 26 : Cas d'utilisation associés au Gestionnaire des escales	78
Figure 27 : Cas d'utilisation associés à l'officier de port.....	79
Figure 28 : Cas d'utilisation associés à l'officier radio.....	80
Figure 29 : Diagramme de séquence du cas d'utilisation « Authentifier ».....	88
Figure 30 : Diagramme de séquence du cas « Charger les nouvelles annonces - Ajouter à la liste attendu ».....	89
Figure 31 : Diagramme de séquence du cas « Gérer les navires en rade - Ajouter à la liste en rade ».....	90
Figure 32 : Diagramme de séquence du cas « Prévision d'entrée ».....	91
Figure 33 : Diagramme de séquence du cas « Exécuter une décision ».....	92
Figure 34 : Diagramme de classe.....	93
Figure 35 : Structure d'un Module odoo.....	106
Figure 36 : Formulaire de création d'une base de données Odoo.....	113
Figure 37 : L'existence de la base de données Odoo16.....	113

Figure 38 : L'implémentation de fichier manifest .py de notre module EPS	114
Figure 39 : L'implémentation de fichier init .py de notre module EPS	114
Figure 40 : Le module EPS	115
Figure 41 : Login pour accéder à notre module.	115
Figure 42 : La création des groupes et compte.	116
Figure 43 : Menu principale de l'application.	116
Figure 44 : Liste view des escales	117
Figure 45 : archivage de l'enregistrement	117
Figure 46 : Ajouter des bouton et des actions.	118
Figure 47 : Manipulation de l'état d'un escale	118
Figure 48 : Afficher l'utilisateur qui a créé un enregistrement 01	119
Figure 49 : Afficher l'utilisateur qui a créé un enregistrement 02	120
Figure 50 : Afficher l'utilisateur qui a créé un enregistrement 03	120
Figure 51 : Afficher l'utilisateur qui a créé un enregistrement 04	120
Figure 52 : représente les attribut pour faire la recherche	122
Figure 53 : l'utilisation des filtres sur les attribut	123
Figure 54 : L'etat de l'escale	123
Figure 55 : Grouper par utilisateur qui a créé l'escale	124
Figure 56 : Les champs calculés	125
Figure 57 : Présentation des information d'un escale	126
Figure 58 : Le kanban view	127
Figure 59 : les escales sous forme de calendrier	127
Figure 60 : Sequences en odoo	128
Figure 61 : La création du paramètre de module	128
Figure 62 : L'utilisation des paramètre setting	129

Liste des tableaux

Table 1 :Les caractéristiques physiques des ports d'entreprise.....	20
Table 2 : Tableau des différents types de logiciels ERP [10]	51
Table 3 :Description textuelle du cas d'utilisation "Authentifier"	81
Table 4 :Description textuelle du cas d'utilisation "Chargement des nouvelles annonces"	82
Table 5 :Description textuelle du cas d'utilisation "Ajouter à la liste attendu"	83
Table 6 :Description textuelle du cas d'utilisation "Valider des navires en Rade"	84
Table 7 :Description textuelle du cas d'utilisation "Ajouter - Prévision d'entrée"	85
Table 8 :Description textuelle du cas d'utilisation "Modifier - Prévision	86
Table 9 :Description textuelle du cas d'utilisation "Exécuter une décision"	87
Table 10 : Présentation des classes et leurs attributs.	101

Table des abréviations

EPS: Entreprise Portuaire de Skikda

EPE: Entreprise Publique Économique

SIE: Sûreté Interne d'Établissement

SIGNE: Système Intégré de Gestion des Navires et Escales

ERP: Entreprise Ressource Planning

PGI: Progiciel de gestion intégré

MVC: Modèle vue contrôleur

UML: Unified Modeling Language

CPN: Conférence de Placement des Navires

APCS: Algerian Port Community System

EDI: Environnement de Développement Intégré

XML: eXtensible Markup Language

Introduction générale

De nos jours, les entreprises évoluent dans un environnement en constante mutation, caractérisé par des demandes croissantes en termes d'efficacité, de productivité et de gestion des informations. Dans cette optique, l'implémentation d'un système d'information performant devient essentielle pour assurer le bon fonctionnement et la compétitivité de toute entreprise. L'Entreprise Portuaire de Skikda (EPS), consciente de ces enjeux, a décidé de mettre à jour son système d'information en créant un ERP (Enterprise Resource Planning). Cette décision découle de la nécessité de répondre aux besoins actuels de l'entreprise, qui ne sont plus satisfaits par l'ancien système en place.

Un ERP, ou Progiciel de Gestion Intégré, est une solution logicielle centralisée qui permet de gérer de manière automatisée et intégrée l'ensemble des processus et des données d'une entreprise. Il offre une vision globale des différentes activités de l'organisation, allant de la gestion des ressources humaines à la logistique, en passant par la finance, les ventes et bien d'autres domaines. L'ERP agit comme un véritable pilier pour l'entreprise en facilitant la prise de décisions stratégiques, en optimisant les processus métier et en favorisant la collaboration entre les différents services.

Pour l'Entreprise Portuaire de Skikda, l'adoption d'un ERP s'avère être un choix judicieux afin de moderniser son système d'information et de relever les défis auxquels elle est confrontée. L'ancien système, qui n'a pas évolué au rythme des besoins de l'entreprise, a montré ses limites en termes d'efficacité, de traçabilité des opérations et de gestion des flux d'informations. Face à ces contraintes, il devient primordial pour l'EPS de migrer vers un ERP qui permettra une meilleure coordination entre les différents services, une automatisation accrue des tâches et

une gestion plus efficace des ressources. Dans cette optique, ce projet de création d'un ERP au sein de l'Entreprise Portuaire X représente un tournant majeur dans sa transformation digitale.

L'outil que nous avons choisi pour créer ce système est Odoo, car il présente plusieurs avantages majeurs qui répondent aux besoins spécifiques de l'entreprise. Odoo est un outil open source mais il offre une large gamme de fonctionnalités modulaires qui couvrent tous les aspects essentiels de la gestion d'entreprise, tels que la comptabilité, les ressources humaines, la logistique, et bien d'autres. De plus, son intégration complète permet une circulation fluide des données et une vision globale des activités de l'entreprise. Odoo se distingue par son interface conviviale et sa facilité d'utilisation, ainsi, sa flexibilité et son évolutivité permettront d'adapter le système aux besoins changeants au fil du temps.

À la demande des dirigeants de l'entreprise portuaire, nous avons débuté l'implémentation de cet ERP en donnant la priorité à la gestion des escales. Cette tâche jugée essentielle par les dirigeants concerne la gestion complète des navires, depuis leur arrivée jusqu'à leur départ, et constitue le point de départ du Workflow du système d'information. Cette étape est interdépendante de toutes les autres étapes du processus, ce qui en fait un élément fondamental à mettre en place en premier. En se concentrant sur cette gestion des escales en amont, nous nous assurons de créer une base solide pour la suite de l'implémentation de l'ERP, facilitant ainsi la coordination des opérations portuaires, la gestion des ressources, la planification des activités et l'amélioration générale de l'efficacité opérationnelle de l'entreprise portuaire.

Outre l'introduction et la conclusion générale, ce mémoire est composé de 4 chapitres :

Chapitre 01: Présentation d'organisme d'accueil (rapport de stage): une brève présentation de l'entreprise portuaire de Skikda (E.P.Skikda) ainsi que son système informatique actuel.

Dans le chapitre 02 nous explorons le concept des ERP (Enterprise Resource Planning) et nous nous concentrons spécifiquement sur Odoo.

Le chapitre 3 de ce mémoire se concentre sur la conception de notre système basé sur l'ERP Odoo en utilisant le langage de modélisation UML.

Dans le chapitre 4, nous détaillons l'implémentation de notre application. Nous expliquons les outils et langages que nous avons utilisés pour développer l'application. Nous présentons également quelques interfaces graphiques du logiciel.

1. Introduction :

Skikda, appelée « Philippeville » jusqu'à la fin de la guerre d'indépendance en 1962, est une ville portuaire située dans le golfe de Stora , au nord-est de l'Algérie . Elle regroupe une population de 1115380 habitants (en 2020) et s'étend sur une superficie de 4137,68km², avec 130 Km² de côtes. Elle est limitée au Nord par la mer Méditerranée , et avoisine la wilaya de Annaba , Constantine , Guelma ,Jijel.

Au bord de la mer Méditerranée , elle dispose d'un important port commercial autonome , classé troisième d'Algérie avec une surface totale d'environ 600 hectares. Ce dernier assure entre autres des liaisons maritimes régulières de transport de voyageurs avec deux villes européennes Marseille en France et Gènes en Italie .

Dans ce chapitre , Nous allons mettre l'accent sur une brève présentation de L'Entreprise Portuaire de Skikda (EPS) et ses besoins , et étudier le système existant SIGNE.

2. Entreprise Portuaire Skikda: [1]

2.1. Présentation générale de l'organisme d'accueil:

- **Dénomination** : Entreprise Portuaire de Skikda, par abréviation E.P. Skikda.
- **Forme juridique** : Entreprise Publique Économique 'EPE', Société par actions régie par les lois et règlements relatifs à l'autonomie des entreprises.
- **Date de création** : Décret n° 82-284 du 14 Août 1982 et modification des statuts en société par actions en date du 21 mars 1989.
- **Capital Social** : 9.000.000.000DA détenu par un actionnaire unique, le groupe des Services Portuaires-SERPORT Spa.
- **Organisation de gestion** : Entreprise organisée en direction générale mono-unité.
- **Siège social** : Petite zone industrielle près embouchure Oued SAFSAF (ex. Domaine des lions) , BP 65-21 000Skikda- Algérie.
- **Compétences territoriales** :
 - Port mixte de Skikda (Marchandises générales et Hydrocarbures).
 - Port de Skikda El Djeddid spécialisé dans le transit des hydrocarbures.
 - Trois ports de pêche : STORA , COLLO et LA MARSA.
 - Trois sites extra-muros : 07ha, 12ha et 18ha.

• **Filiales et participations :**

L'entreprise Portuaire de Skikda contrôle trois (03) Filiales :

- Entreprise de Gestion des ports et abris de pêche de Skikda (EGPP Skikda) 100%.
- Skikda Conteneurs Service (SCS) 100%.
- Société intermodale de logistique (SIL) 66%.

Elle détient par ailleurs des participations dans 04 Sociétés :

- STORA (activités de transport – transit et magasinage) 20%
- STH (Société de gestion des terminaux à hydrocarbures) 15%
- CNAN-GROUP (transport maritime) 8,7%
- SIH (Société d'investissement hôtelier) 6,2%



Figure 1 : Logo d'EPS.

2.2. Aperçu historique:

Le cours débarcadère édifié pendant la période coloniale, est devenu aujourd'hui l'un des fleurons de l'économie nationale et un pion essentiel d'un pôle hydrocarbures intégré, qui véhicule l'essor et le développement économique de l'Algérie indépendante.

1854: les navires qui apportaient les marchandises de la Métropole déchargeaient en rade , à une demi-lieue du rivage, et par mauvais temps s'abritaient tant bien que mal dans l'anse de Stora où le mouillage n'offrait qu'une protection insuffisante.

1860 : Après la fondation de la ville, le programme des travaux de construction du port fut approuvé (décret du 28 juillet 1860).

Le 04 août est enfin décidée la construction d'un port mieux adapté.

1861 : La première tranche des travaux commence en février : un terre-plein est remblayé au pied du Skikda : c'est de là que s'élancera la grande jetée. Le port comprenait alors une

darse et un avant-port couverts des vents du Nord-est par une digue du large de 1.390m enracinée à la pointe du Skikda de direction Sud-est-Nord-Ouest et, de la côte du Nord-ouest, par une traverse dite jetée du Château-Vert.

1867 : En février : deuxième tranche : Le terre-plein de la traverse Sud sépare le port d'une plus petite darse qui ne sera comblée que plus tard.

1881 : Début de la construction des quais qui sont achevés (traverses Nord et Sud) en 1882.

1884 : La construction de la jetée du large de 1.625m de longueur.

1890 : On peut dire que le Port est terminé, il ne reste plus qu'à combler la petite darse, ce qui sera fait en 1893.

1895 : Installation de la 1 ère grue portique.

1938 : L'outillage public comprend quatre grues à vapeur d'un modèle désuet, installées sur les quais Sud et Sud-Est de la darse, un ponton-mât de 50 tonnes, La Société d'Entreprise Maritime et Commerciale possède deux grues flottantes de 12 et 8 tonnes. Les voies ferrées du port sont reliées à la gare tête de ligne de chemin de fer Philippeville-Constantine installée sur les terre-pleins de la darse. Skikda (Philippeville) était le cinquième port algérien en 1938.

1941 : Agrandissement de la surface des terre-pleins et Installation de quatre grues électriques de 1,5 t à 5 tonnes et, sur le nouveau môle de l'avant-port destiné aux charbons et aux matières pondéreuses et un portique à charbon de 1 tonne.

1962 : Indépendance de l'Algérie.

1965 : Réalisation du môle « Château Vert ».

1969 : début de la construction des 03 appontements pétroliers au niveau de l'ancien port.

1972 : Réception de la première tranche du nouveau port de Skikda (Appontements gazier). Le premier chargement du GNL est intervenu au mois de décembre. Instruction des 03 appontements pétroliers au niveau de l'ancien port .

1982 : Achèvement des infrastructures du nouveau port avec un début d'exploitation des postes à hydrocarbures (P1,P2,P3) spécialisés dans le trafic du pétrole brut et des produits raffinés .

1985: Construction du quai de la « Marinelle ».

2001: le Président de la République –Abdelaziz Bouteflika- inaugure la tour de contrôle de l'ancien port. Bâtiment de conception unique en Algérie.

2003: Aménagement du poste P13 pour la réception des navires de gros tonnage (35.000 Tonnes).

2005: Installation des Bouées OFFSHORE pour le chargement des super tankers de 320.000T.

2007: La mise en service du port sec de 07 Ha.

2006 – 2012: Confortement des quais de l'ancien port.

2.3. Caractéristiques physiques des ports de l'entreprise:

Caractéristiques	Port Mixte de Skikda	Port de Skikda EL- Djedid
Position géographique	-Longitude Est : 6° 54' 30'' -Latitude Nord : 36° 53' 20''	-Longitude Est : 6° 57' 00'' -Latitude Nord : 36° 53' 35''
Accès maritime	Passe d'entrée au port -15 mètres de profondeur -120 mètres de largeur	Passe d'entrée au port -18 mètres de profondeur -250 mètres de largeur
Ouvrages de protection	-Jetée principale : 1.625 m -Jetée secondaire : 300 m	-Jetée principale : 1.875 m -Jetée secondaire : 650 m
Surface plan d'eau	45 hectares	56 hectares
Cercle d'évitage	350 mètres	550 mètres
Linéaire de quai m/ses	2.180 ML	240 ML
Ouvrages d'accostage	-14 postes à quai général cargos -03 appontements pétroliers	-01 quai général cargos -07 appontements pétroliers
Surface commerciales dont :	198.631 M ²	
• Terre-pleins	180.975 M ²	40.000 M ²
• Hangars- magasins	17.650 M ²	
Surface commerciales Extra-portuaires	410.000 M ²	-

Table 1: Les caractéristiques physiques des ports d'entreprise.

- **Atouts du port de Skikda :**

- Position géographique centrale dans la façade maritime de l'Est – Algérien qui lui confère un rôle de premier plan dans les échanges commerciaux maritimes.
- Port attractif aussi bien pour sa position géographique que pour sa souplesse et sa fiabilité (organisation du travail des équipes de manutention en 03 shift et 7jous /7jours).
- Port particulièrement dynamique pour le traitement des produits industriels, les produits métallurgiques et les conteneurs.
- Son hinterland s'étend sur 15 wilayas de l'Est et du Sud - Est Algérien.
- Moyens matériels modernes et adaptés à tous types de trafics.
- Port relié par un vaste réseau ferré et routier tant vers les wilayas de l'Est que celles du centre et du Sud du pays et est, par excellence, considéré comme le port du Sud Est Algérien.
- Le port dispose d'un Système de management intégré suivant les normes :
 - ISO 9001 – 2005 / ISO 14001-2015 / ISO 45001-2018

Entreprise Certifié selon le code ISPS.

- Proximité immédiate de connexion du réseau ferrée (gare ferroviaire terminale à 50 mètres du port mixte).
- Important patrimoine foncier en dehors de l'enceinte portuaire (41Ha).

3. Organisation et activité :

L'entreprise Portuaire de Skikda est administrée par un conseil d'administration composé de sept (07) membres ; cinq (05) membres représentant l'unique actionnaire groupe des services portuaires « SERPORT » dont le directeur général de l'EPS en sa qualité de président et deux (02) membres de droit représentant les travailleurs. C'est une société organisée en direction générale mono-unité.

3.1. Activité :

L'entreprise portuaire de Skikda a pour objet sociale la gestion du domaine public portuaire et son corollaire : l'exercice des opérations de police et de sécurité relatives au mouvement de la navigation et à la préservation et sauvegarde du dit domaine public portuaire.

Le domaine public portuaire est constitué de :

- Plants d'eau.
- Terre-pleins, hangars, magasins.

- Divers locaux à usage administratif et commercial.

La gestion du domaine public portuaire s'effectue de manière directe pour les plants d'eau l'affectation des postes à quai aux navires et l'utilisation des terre-pleins et hangars.

Autre la gestion du domaine public l'entreprise exerce des activités de services se rapportant à :

- La gestion et l'exploitation de l'outillage et des installations portuaires.
- L'exercice des opérations de remorquage, de pilotage et de lamanage.
- L'exercice des opérations d'acconage et de manutention.
- L'exercice des missions de police et de sécurité portuaire.
- L'exécution des travaux d'entretien, d'aménagement et de renouvellement de la superstructure portuaire.
- L'élaboration, en relation avec les autres autorités concernées, de programmes de travaux d'entretien, d'aménagement et de création d'infrastructures portuaires.
- La gestion de site extra portuaire (bases logistiques, ports secs).

3.2. Capacités portuaires de transit :

- Transit marchandises générales : 4.000.000 tonnes/an ; 14 postes à quai avec un total de 2.180 MI de quai.
- Transit Hydrocarbures : 30.000.000 tonnes/an ; 10 appontements pour des navires de 25 à 130.000 tonnes DWT, deux bouées offshore de 3000.000 tonnes DWT.

3.3. Surfaces commerciales :

- Superficies commerciales couvertes (magasins, hangars modulaires, abris sous auvents) : 7815m².
- Superficies commerciales terre-pleins nouveau port : 21560m².
- Superficies commerciales terre-pleins ancien port : 58931m².
- Superficies commerciales parcs à conteneurs : 81239m².

3.4. Gamme des prestations de services :

- **Liées au domaine et apparente :**
 - Domaine maritime : séjour en rade, séjour à quai, occupation du plan d'eau, et le chargement en mer.

- Domaine Terrestre : concession à court, moyen et long terme du domaine portuaire, transit, dépôt et transfert des marchandises à quai en terre-pleins, magasins, et port sec.
- Installations spécialisées : usage de la cale de halage et services divers en rapport, usage voie ferrée , poids public , fourniture eau.
- **Liées au navire :**
 - Pilotage, lamanage, remorquage, course en pilotine, mise à disposition de défenses d'accostage.
 - Prestation de manutention et location d'engins divers (grues, chariots élévateurs).
 - Diverses prestations (fourniture d'énergie électrique, eau enlèvement déchets).
- **Liées à la marchandise :**
 - Manutention , (débarquement , embarquement).
 - Acconage (entreposage, bâchage , gardiennage, transfert et livraison).
 - Relevage.

4. L'organigramme de l'entreprise et ses missions :

L'entreprise portuaire de Skikda est organisée et structurée en une direction générale et douze (12) directions centrales :

- Direction audit et gestion de risques.
- Direction finances et comptabilité.
- Direction SIE.
- Direction de la qualité et des systèmes d'information .
- Direction des affaires générales.
- Direction des ressources humaines.
- Direction du domaine et des travaux.
- Direction du matériel.
- Direction de la capitainerie.
- Direction commerciale.
- Direction du remorquage.
- Direction de la manutention et de l'acconage.

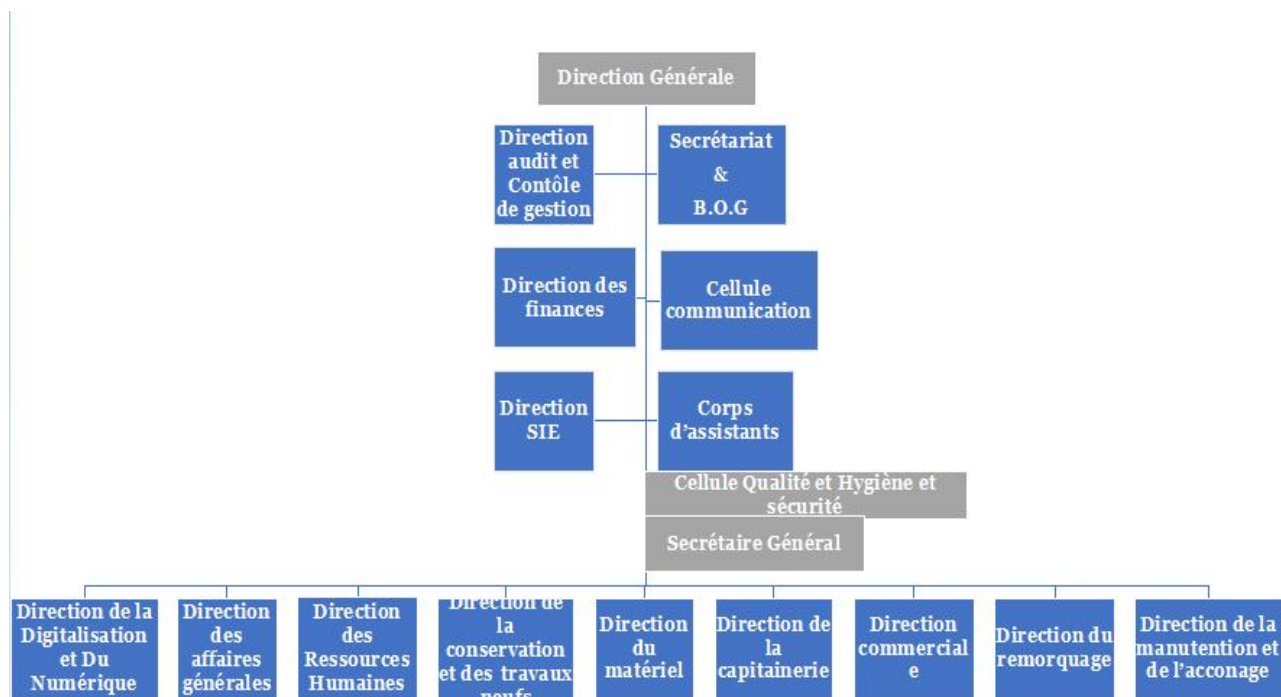


Figure 2: Organigramme de l'entreprise portuaire Skikda .

4.1. Direction générale :

Elle est chargée : de définir et d'élaborer les plans opérationnels de mise en œuvre de la stratégie ; de mettre en œuvre les décisions arrêtées par le conseil de l'administration ; et d'animer , coordonner et contrôler la gestion des directions fonctionnelles et opérationnelles qui concourent à la mise en œuvre des plans opérationnels.

Le directeur général est assisté, dans le cadre du management de l'entreprise par un secrétaire général et un staff constitué de :

- Un Directeur d'audit et gestion des risques.
- Un directeur finances et comptabilité.
- Un directeur de la sûreté interne d'établissement (SIE).
- Une cellule de communication.
- Un corps d'assistants.
- Un bureau d'ordre général.
- Un secrétariat.

4.2. Direction d'audit et gestion des risques :

Chargée de :

- Veiller à l'atteinte des objectifs ayant trait aux activités de l'entreprise et faire des recommandations d'amélioration nécessaires pour prendre en charge les dysfonctionnements éventuels.

- Élaborer, mettre en œuvre et évaluer le plan d'audit annuel validé par le conseil d'administration.

- Évaluer les systèmes d'organisation mis en place par l'entreprise, et notamment en terme de qualité et d'intégrité des informations opérationnelles et financières et en terme de gestion des risques et de protection du patrimoine.

- Veiller à l'application des procédures, instructions, règlements, lois, et des normes etc.

Les missions et tâches de la direction d'audit et gestion des risques sont assurées par un corps d'auditeurs pluridisciplinaires.

4.3. Direction financière et comptable:

La mise à disposition de l'entreprise des ressources et des techniques financières nécessaires à son exploitation , son développement et sa protection en procédant au contrôle de l'utilisation des actifs et la rentabilité des opérations auxquels ils sont affectés à l'effet d'assurer la pérennité financière de l'entreprise en maîtrisant sa rentabilité globale et en sauvegardant son autonomie financière ;

La gestion, la production et la conservation des documents comptables conformément à la réglementation en vigueur (livre journal, livre d'inventaire, grand livre , états financiers).

Département finances : il est chargé de veiller à la gestion et à l'optimisation de la trésorerie de l'entreprise, élaborer le budget de trésorerie et le plan de financement et respecter les engagements financiers de l'entreprise.

Département Comptabilité : il est chargé de veiller au traitement de la comptabilité générale et analytique dans les délais élaborer les états financiers périodiques et le bilan fiscal dans les délais impartis.

4.4. Direction sûreté interne de l'entreprise (SIE) :

- Mettre en œuvre l'ensemble des actions pour la protection du patrimoine public, des installations portuaires et extra portuaire, et les personnes.
- Assurer la coordination technique entre les agents relevant des institutions directement liés à la sûreté maritime et portuaire.
- Assurer le contrôle des marchandises, engins, et autres matériel à la sortie de l'enceinte portuaire.
- Assurer la maintenance des systèmes de vidéos et de télésurveillance, et des équipements sensibles (scanner, portique , etc.).
- Service Sûreté Interne d'entreprise : sa mission est de mettre e œuvre les mesures de nature à concourir à la prévention, la préservation, des risques de toute nature et notamment les actes d'agression, de sabotage, de destruction et de vol.

4.5. Direction de la qualité et des systèmes d'information :

Elle est chargée de vérifier la conformité des processus au regard de normes et des contraintes réglementaires en vigueur. Ainsi qu'entreprendre les travaux nécessaires qui s'inscrivent dans la démarche d'informatisation de l'ensemble du système d'information de l'entreprise (logistique n gestion des ressources humaines , gestion financière et comptable , gestion des stocks etc.)

Département Management Qualité : il est chargé d'impulser la mise en œuvre d'une politique qualité au sein des structures.

- Définir le périmètre, les objectifs et des indicateurs de suivi de la démarche qualité.
- Conduire des audits internes et réaliser des enquêtes sur la satisfaction client.
- Assurer le reporting des indicateurs et mesurer les écarts avec les prévisions.
- Analyser les risques et les opportunités.
- Assurer les mesures de réduction ou de traitement des déchets , vérifier les procédures de contrôle et de prévention par rapport aux risques environnementaux .

Département systèmes d'information : il est chargé de :

- Élaborer le plan directeur informatique et le cahier des charges superviser et évaluer les travaux nécessaires qui s'inscrivent , dans la démarche d'informatisation de l'ensemble du système d'information de l'entreprise (logistique, gestion des ressources humaines, gestion financière et comptable , gestion des stocks ... etc.).

- Développer et mettre en œuvre les systèmes d'échange d'informations à travers la réalisation , la gestion et la maintenance du réseau INTRANET de l'entreprise .
- Assurer le pilotage du processus « Gérer le système d'information ».

4.6. Direction des affaires générales :

Département approvisionnement et logistique : il est chargé de :

- Le suivi et la gestion du patrimoine mobilier et immobilier de l'entreprise.
- La documentation et la conservation des archives.
- Assurer les approvisionnements de l'entreprise, en pièces de rechange, équipements et fournitures consommables, de fournitures de bureaux et d'articles d'habillement ainsi que de l'exécution et du suivi des contrats d'achat.

Département affaires juridiques :

- Assurer une veille juridique.
- Assurer la défense des intérêts de l'entreprise devant les juridictions.
- Solliciter les expertises nécessaires dans le cadre de ses missions.
- Agir auprès des auxiliaires de justice (Avocats, Notaires, Huissiers, etc.).
- Agir auprès des administrations (CNRC, Direction du domaine de wilaya, etc.) pour solliciter une prestation ou disposer d'actes ou documents légaux ou juridiques au profit de l'entreprise.

Bureau des marchés : il est chargé de récupérer et conserver les plis des soumissions remis dans le cadre des consultations d'appels d'offres lancés par les différentes structures de l'entreprise portuaire de Skikda jusqu'à la date d'ouverture des plis par les commissions instituées au niveau de l'entreprise.

4.7. Direction des ressources humaines :

- Assurer la gestion des ressources humaines.
- Former et sensibiliser le personnel.
- Mettre en œuvre et superviser le plan d'actions des œuvres sociales.
- Assurer la médecine du travail.
- Assurer l'interface avec les organismes externes.
- Établir et suivre la réalisation du planning des visites médicales.

Service Médecine du travail : il relève hiérarchiquement du directeur de la structure , il est chargé de mettre en application les dispositions réglementaires en vigueur en matière de la médecine du travail.

- Département des ressources humaines : il est chargé de :
- Assurer la gestion administrative du personnel de l'entreprise.
- Appliquer la législation de travail, la convention collective, et le règlement intérieur.
- Établir la paie du personnel.
- Établir et transmettre les bilans périodiques ayant trait à la gestion du personnel.
- Établir les tableaux de bord.
- Établir le plan de relève et le plan prévisionnel de recrutement.

Département formation : il est chargé de :

- Décliner la politique de l'entreprise en matière de formation.
- Élaborer en concertation avec les autres directions le plan de formation.
- Veiller au respect des obligations légales en matière de formation réglementaire des gens de mer (STCW).
- Élaborer le bilan annuel des formations et évaluer leur efficacité.
- Suivre le personnel en formation sur le plan assiduité, discipline etc.
- Préparer les contrats d'apprentissage et suivre le personnel en apprentissage.

Collecter et analyser les fiches d'évaluation à chaud établies par les organismes de formation et les fiches d'évaluation à froid établies en concertation avec la structure concernée.

Département des œuvres sociales : Conformément à la réglementation en vigueur les œuvres sociales sont gérées par le comité de participation et dans ce cadre l'entreprise met à sa disposition l'organisation et les moyens humains et matériels nécessaires. Il est chargé également de :

- Élaborer, mettre en œuvre et évaluer le plan d'actions des œuvres sociales.
- Gérer, préserver et maintenir les infrastructures et les équipements du département.
- Établir les bilans comptables.
- Veiller à la tenue à jour du registre comptable.
- Veiller à la tenue à jour des registres d'inventaires des immobilisations et des éléments de faible valeur.

4.8. Direction du domaine et des travaux :

- Maintenir et développer les plans d'eau , ouvrages et installations gérés par l'entreprise,
- Gérer les installations spécialisées (avitaillement en eau, Cale de halage) ;
- Gérer tout ce qui se rapporte aux concessions et autres occupations temporaires accordées sur l'ensemble du domaine portuaire.

Département Réalisations : il est chargé par

- Exécuter le budget d'investissement de l'entreprise en termes de réalisations.
- Élaborer ou faire élaborer les descriptifs techniques et les devis quantitatifs et les estimatifs des cahiers des charges en rapport avec les acquisitions des équipements et des travaux.
- Assurer la maîtrise des sous-traitants et prestations engagés pour la réalisation des travaux.
- Veiller à la conformité de la réalisation des travaux avec les contrats.

Département Conservation : il est chargé de :

- Assurer la maintenance courante du patrimoine foncier bâti et non bâti, et des équipements associés.
- Élaborer les programmes des opérations d'entretien préventif et curatif au titre de la conservation des infrastructures et superstructures du domaine portuaire relevant de l'entreprise.
- Élaborer et faire exécuter les programmes d'entretien, de préparation et/ou de rénovation des divers réseaux ou équipements et dresser le rapport mensuel.
- Assurer le transport et l'évacuation des déchets.
- Veiller à prendre toutes les dispositions nécessaires au respect des consignes et règles d'hygiène, de sécurité et de préservation de l'environnement.
- Veiller à la concrétisation du programme SSTE et l'atteinte des objectifs y afférant.

4.9. Direction du matériel :

Elle est chargée de la maintenance , de l'entretien des équipements et du parc matériel gérés par l'entreprise portuaire de Skikda (engins terrestres roulants , grues terrestres , matériel de transport et équipements auxiliaires).

Département maintenance : il est chargé de :

- Maintenir et entretenir les engins terrestres roulants (Reachstackers et spreaders ; chariots élévateurs et fourches (gros et petit tonnage) ; tracteurs portuaires routiers et remorques ; matériels de transport ; équipements auxiliaires (groupes électrogènes , scanners, pompes à grains, etc.)
- Maintenir et entretenir les grues terrestres (grues portuaires et grues automobiles).
- Gérer les stocks de pièces de rechange, carburant et fournitures consommables.

Département relevage : le département relevage a pour mission :

- Veiller à la mise à disposition des moyens matériels adéquats pour les opérations navires, les opérations de transit et toute autre demande de moyens matériels de relevage en interne.
- Établir le rapport mensuel d'activité du département relevage en y consolidant toutes les informations concernant l'utilisation des moyens de relevage pour les opérations navires, location pour les clients externes et demandes internes.

Bureau des méthodes : il est chargé de :

- Réceptionner, traiter les demandes de réparation, émanant des diverses structures de l'entreprise et ordonnancement des opérations de maintenance (préventive et curative).
- Tenir à jour les dossiers technico-administratifs des équipements et le suivi de mise œuvre des programmes de levée des réserves et observations du bureau de contrôle réglementaire des équipements de production de l'entreprise .
- Assurer l'interface avec le département relevage , département conservation relevant de la direction des travaux et domaine , le parc auto , et les organismes externes.

4.10. Direction de la capitainerie :

La direction capitainerie a pour mission d'exercer dans les limites du domaine public maritime et portuaire , la police et la sécurité des plans d'eau , des terres pleines et des installations , et ce dans le but d'exercer , en permanence, la conservation des ouvrages, la circulation et la sécurité des personnes et des biens , et de la contribution de l'élaboration de la politique générale en matière d'hygiène , sécurité et environnement ainsi que de la mise en œuvre des mesures de prévention.

Elle est également responsable de l'ensemble des actions relatives à la navigation , du pilotage , du lamanage et du placement des navires . Elle incarne l'autorité portuaire .

La direction de la capitainerie est composée de deux départements , et une station de pilotage , le directeur de la capitainerie assume les missions de PFSO.

- **PFSO** : Port Facilities Office est chargé de coordonner les activités de sûreté , sécurité au sens de la préservation du domaine portuaire et maritime (infrastructures, personnels, matériels , navires et marchandises) contre les risques d'intrusion, sabotage, vol, terrorisme et ce conformément aux dispositions du décret exécutif 04-418 du 20 décembre 2004, et du code ISPS.

Le bureau de sûreté portuaire est placé sous l'autorité du PFSO, et comprend :

- Un adjoint PFSO (chef de bureau sûreté portuaire).
- Officiers de port chargés de la sûreté (assistants PFSO).

Département police et sécurité :

Il est tenu de représenter l'autorité portuaire et d'appliquer le règlement de sécurité et d'exploitation du port et toute réglementation découlant du code maritime.

A ce titre il est chargé de :

- **Préparer la conférence de placement des navires:**

- Exécuter les décisions de la conférence de placement des navires.
 - Assurer les relations de la capitainerie avec les commandants de navires et les consignataires.
 - Assurer la gestion des équipements de radio télécommunication.
 - Tenir à jour les dossiers navires.
 - Gérer les statistiques en matière de mouvements particuliers de trafic.
 - Tenir à jour les informations relatives aux infrastructures portuaires (tirant d'eau, longueur, des postes à quai, distance entre les bollards).
 - Traiter les réclamations des amateurs ou leurs représentants.
 - Veillez à l'application des règlements en matière de police et sécurité dans l'ensemble du domaine public maritime et portuaire : plan d'eau, terre-plein, docks-installations.
 - Veiller au respect de la réglementation régissent le transport et la manutention des marchandises dangereuses.
 - Dresser des PV de constatation et verbaliser tout contrevenant.
- Le personnel « officiers et surveillants de port » est assermenté, conformément aux dispositions du code maritime. il comprend :
 - Chefs de quart.
 - Officiers de port.
 - Officiers Radio.
 - Section police et sécurité.

- Surveillants de port.
- Ambulanciers et accompagnateurs.

Département aides à la navigation:

Le département aide à la navigation est chargé des activités d'assistance à la navigation à l'entrée et sortie du port dans les bassins portuaires .

Le département aide à la navigation est composé d'un service et d'une section environnement et lutte antipollution, et une station de pilotage.

Service lamanage : sa mission est d'assurer la coordination et la supervision des opérations de lamanage et de largage des navires dans l'enceinte portuaire.

Il comprend une section de lutte antipollution. Il assure aussi :

- L'exploitation et l'armement des engins de servitude.
- Suivi des engins de servitude en matière d'hygiène et de sécurité , ainsi que leurs entretiens internes et externes.
- Elaboration du tableau de position mensuel des engins de servitude.
- Veiller à la validité et le renouvellement des titres de navigation et de sécurité.
- Assurer les services d'urgences telles que le déversement d'hydrocarbures ou d'autres incidents graves.

Station de pilotage : la station de pilotage placée sous l'autorité du Directeur de la Capitainerie, elle comprend :

- Un chef pilote, et son adjoint, le corps des pilotes maritimes, et fonctionne selon le régime de travail permanent.
- Le pilotage maritime est obligatoire dans les limites administratives du port pour l'ensemble des navires.
- Le pilotage est effectué par les pilotes dotés de qualifications et d'expériences professionnelles prévues par la réglementation.

4.11. Direction commerciale :

La direction commerciale a pour mission la gestion du volet facturation de l'ensemble des prestations fournies par l'entreprise, le couvremet des créances sur clients et le traitement des réclamations clients, en coordination avec les structures concernées, et l'écoute et la satisfaction client, dans le cadre de la promotion des prestations de services de l'entreprise .

Département relations clients : il est chargé de :

- Recouvrir les créances sur clients.
- Traiter et répondre aux réclamations des clients.
- Constituer et transmettre à la DAG les dossiers litigieux.
- Promouvoir l'écoute et la satisfaction client.
- Élaborer les états des chiffres d'affaires, impôts et taxes sur le chiffre d'affaires, droits et timbres etc.
- Transmettre à la DFC , les chèques et les copies des virements des clients.
- Effectuer les rapprochements avec la DFC.

Service Recouvrements : il est chargé de :

- Relancer les clients conformément à la procédure en vigueur.
- Constituer et transmettre les dossiers litigieux au service contentieux (phase du contentieux)

Service Marketing : il est chargé de :

- Traiter les réclamations des clients en coordination avec les structures concernées de l'entreprise .
- Réaliser les études économiques et statistiques nécessaires à l'adaptation des tarifs portuaires pour la satisfaction des clients.
- Calculer et analyser la rentabilité financière des opérations de traitement des différents produits manutentionnés.

Département facturation : il est chargé de :

- Élaborer, suivre et superviser la facturation de l'ensemble des prestations.
- Consolider , analyser et transmettre le chiffre d'affaires réalisé.
- Dresser les rapports d'activité périodique.

4.12. Direction du remorquage :

Mettre à la disposition des navires des remorqueurs durant leur appareillage, accostage , dé-halage, et veille de sécurité.

Assurer la fourniture des services annexes et des prestations exceptionnelles telles que l'assistance et le sauvetage, l'affrètement, et le remorquage hauturier.

Département Armement : il est chargé de :

- Organiser , planifier et suivre l'état des remorqueurs pour une bonne disponibilité ,fiabilité et performance.
- Mettre les remorqueurs en exploitation en toute sécurité.

- Veiller à la mise à jour des documents nécessaires à l'exploitation des remorqueurs.
- Veiller à ce que tous les remorqueurs soient bien armés et bien équipés.
- Travailler conjointement avec le département de la maintenance navale afin de résoudre les problèmes rapidement.
- Veiller à la formation des équipages et améliorer leur cadre de vie à bord.
- Assurer la veille technologique et la gestion documentaire technique.

Département Maintenance : il est chargé de :

- Élaborer le planning de maintenance préventive et curative des remorqueurs , et des engins de servitude, en gérant les priorités , la sécurité et les contraintes de disponibilité des remorqueurs et des engins de servitude, avec l'aide d'un système de GMAO (gestion de maintenance assistée par ordinateur).
- Réagir en cas de panne grave et résoudre les problèmes rapidement .
- Gérer les relations avec les structures internes, les constructeurs et les fournisseurs de pièces de rechange et les outils pour la maintenance .
- Contrôler les interventions réalisées, et proposer l'amélioration des performances des machines en termes de taux de disponibilité et en terme de coûts.
- Assurer la veille technologique et la gestion de la documentation technique.

4.13. Direction de la Manutention et de l'acconage:

Sa mission principale est de contrôler , coordonner et diriger l'ensemble des activités liées aux opérations , de reconnaissances, pointage , débarquement, embarquement , de réception, du magasinage , du gardiennage et de la livraison (enlèvement) des marchandises.

C'est la direction qui contient le plus d'effectifs en vue de son activité

Département Manutention : il est chargé de :

- Assurer la coordination des structures en charge de la manutention dans les opérations de chargement et de déchargement des navires.
- Fournir les accessoires afférents aux opérations de chargement et déchargement des navires .
- Assurer l'affectation et le suivi des moyens humains.

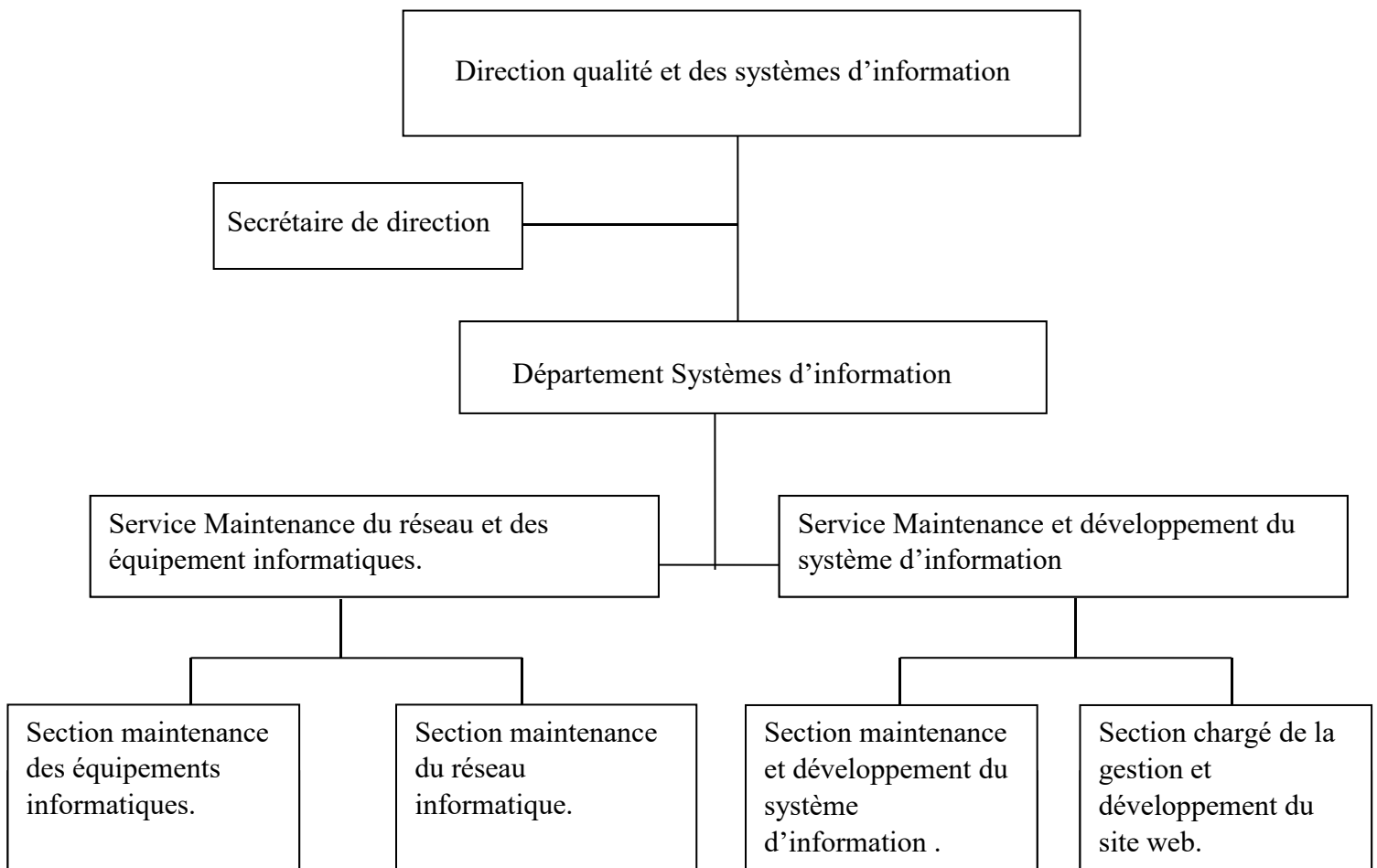
Le personnel de manutention est constitué de deux équipes A et B et qui compte en totale 446 employés de différentes fonctions.

Avant d'accoster un bateau doit communiquer au département un « CARGO PLAN » qui compte toutes ses caractéristiques pour lui préparer le quai et toute l'équipe qui se charge de la manutention.

Département Acconage :il est chargé de

- Assurer la coordination des opération destinées à assurer la réception, le pointage et la reconnaissance à terre des marchandises embarquées ou débarquées .
- Procéder au magasinage, gardiennage jusqu'à leur embarquement ou livraison.
- Livrer la marchandise à son destinataire ou mandataire habilité.
- Conserver les marchandises débarquées et/ou embarquées dans son état initial.
- Déclarer et constituer les dossiers relatifs aux marchandises avariés.
- Constituer les dossiers de la commission de traitement des avaries.

5. Présentation du Département informatique :



Le département a pour charges :

- Élaborer et mettre en œuvre le plan directeur informatique .
- Entreprendre les travaux nécessaires qui s'inscrivent dans la démarche d'informatisation de l'ensemble du système d'information de l'entreprise (logistique, gestion des ressources humaines, gestion financière et comptable , gestion des stocks etc.)
- Développer et mettre en œuvre les systèmes d'échange d'informations à travers notamment la réalisation , la gestion et la maintenance du réseau INTRANET de l'entreprise .
- Établir le plan informatique annuel ainsi que les budgets correspondants .
- Superviser le travail du personnel du département en coordination avec les deux chefs de service .
- Effectuer un suivi régulier du plan d'action de la structure .
- Orienter le développement du site web de l'entreprise en collaboration avec les parties concernées.
- Gérer les dossiers d'acquisitions de matériel et de logiciels ainsi que ceux concernant la fourniture et installation du réseau en collaboration avec ses subordonnés directs.
- Communiquer en permanence avec le personnel sur les objectifs et résultats à atteindre .
- Assurer le pilotage du processus « Gérer le système d'information ».
- Service maintenance du réseau et des équipements informatiques :il est chargé de :
 - L'entretien et la réparation du matériel informatique et l'installation de nouveaux équipements .
 - Formation des utilisateurs sur le bon fonctionnement des équipements dont ils ont la charge .
 - Mise en place de systèmes de sécurité permettant la protection des équipements, installations réseau etc.

- Animer et coordonner toutes les actions concourant au développement de l'outil informatique au sein de l'entreprise .
- Veiller à la bonne gestion du courrier électronique à travers le site internet .
- Le service Maintenance du réseau et des équipements informatiques est organisé en deux (02) sections :
- Section maintenance du réseau informatique :
- Ingénieur chargé du développement et de la maintenance du système d'information.
- Section maintenance des équipements informatiques :
- Technicien en maintenance informatique .
- Service Maintenance et Développement du système d'information : il est chargé de :
- Étudier et entreprendre les démarches opérationnelles appropriées visant l'exploitation en réseau de l'échange d'informations avec les opérateurs portuaires.
- Études et développement de nouvelles applications informatiques selon les besoins exprimés par les utilisateurs .
- La mise en service et la maintenance des applications développées.
- Assister les utilisateurs en vue d'aboutir à une exploitation efficace des applications développés en conformité avec les besoins de l'entreprise .
- Entreprendre des actions d'encadrement et de formation des utilisateurs à chaque fois que le besoin est exprimé par les structures .
- Mise en place d'un planning de sauvegarde (logiciels et bases de données) et sécurité du système informatique de l'entreprise .
- Veiller au bon fonctionnement du réseau informatique de l'entreprise et l'amélioration de ses performances.
- S'assure que le plan de développement garantit une cohérence globale du système d'information et ce notamment via des interfaces et passerelles entre les différents modules .
- Répartit le plan de charge sur les différents éléments du service et suit le plan d'action établi pour chacun en conséquence .
- Effectue un contrôle régulier sur les différentes bases de données de l'entreprise et notamment en ce qui concerne le volet sécurité .

- Développe elle-même des applicatifs de gestion et assure la maintenance des logiciels qu'elle a précédemment créés .
- Superviser le travail du chargé de la maintenance du site web
- Le service maintenance et développement du système d'information est organisé en :
 - Section maintenance et développement du système d'information :
 - Ingénieur chargé du développement et de la maintenance du système d'information
 - Section chargé de la gestion et développement du site web.

6. Etude de l'existant :

L'entreprise portuaire de Skikda adopte environ 15 systèmes d'information pour des différents taches et services, on cite parmi eux SIGNE, Planetecom, BIGFinances, MEDSTATS, SIDOM, GPRPO .

Dans notre projet, Nous intéressons par étudier le système d'information SIGNE, mettre le point sur ses insuffisances afin de proposer une solution qui répond aux nouveaux besoins des utilisateurs.

6.1. Présentation de système d'information SIGNE :

Le Système Intégré de Gestion des Navires et Escales est un système d'information permettant la gestion des navires accostés au niveau du port et de toutes les activités liées aux prestations de services, offertes aux clients (importateurs, transitaires, consignataire, armateur etc.).

Le système SIGNE est basé sur une architecture client/serveur . Il est conçue et implémenté d'une façon modulaire, permettant ainsi de l'adapter aux nouveaux besoins des utilisateurs.

Il constitue un système sécurisé grâce à un verrouillage des accès.



Figure 3: Page d'accueil du système SIGNE.

6.1.1. Composants de base de SIGNE:

Le système intégré de gestion des navires et escales pose sur des éléments de base :

Les navires , les escales, les droits de navigation , les états , le Suivi des conteneurs , la facturation de pilotage, la facturation de domaine, la facturation de manutention , facturation de remorquage .

6.1.2. Les traitements de SIGNE:

Le système est responsable des traitements suivants :

- Suivi des navires et escales .
- Planification de conférence de placement .
- Situation Rade/Quai
- Gestion des droits de navigation .
- Suivi de la marchandise : Manifeste, débarquement/embarquement, évacuation .
- Autorisation d'accès aux marchandises.
- Suivi des bons de relevage .
- Suivi des conteneurs .
- Facturation .

6.1.3. Les résultats de SIGNE:

Les résultats des traitements effectués par le système SIGNE seront présenter sous forme de :

- Factures.
- Journal de factures.

- États de souffrance des marchandises.
- Situation à quai/en rade.
- Avis d'arrivée marchandises.
- Relance des clients.
- Statistiques :
 - Entreposage conteneurs/divers.
 - Pilotage/remorquage.
 - Attentes hydrocarbures. etc.
- Chiffre d'affaires :
 - Pilotage, amarrage.
 - Remorquage.
 - Entreposage , etc.
- Bordereau d'envoi :
 - Manutention .
 - Entreposage conteneurs .
 - Entreposage divers.
 - Relevage

7. Problématique et Solution proposée:

7.1. Problématique:

Nous avons constaté un certain nombre de difficultés et d'incohérences sur le système existant tel que :

- Le partage et l'échange d'informations avec d'autres systèmes se font manuellement ou via des passerelles, ce qui fait perdre du temps et augmente le risque d'erreur.
- Absence d'interactions entre les systèmes ce qui provoque l'incohérence des données.
- La base de données (BDD) utilisée est incomplète en termes de relations entre les données.
- Manque des fonctionnalités dans les systèmes pour faciliter la manipulation.

7.2. Solution proposée:

Pour résoudre les problèmes posés par le système existant, nous avons proposé de concevoir et implémenter un ERP avec Odoo .

La solution projetée devra être en mesure de satisfaire certaines fonctionnalités de base :

- **Restriction des accès et personnalisation des menus** : L'accès aux différentes sessions sera contrôlé par l'identification de l'utilisateur (Login) assorti d'un mot de passe. Les menus et le mode d'accès aux sessions du progiciel seront personnalisables par utilisateur (lecture, écriture, suppression). Certaines données jugées confidentielles telles que les prix, les tarifs seront masqués à tous les utilisateurs non autorisés.
- **Import et export des données** : Le système permettra d'importer et d'exporter des données. L'import des données sera contrôlé pour interdire la mise en cause de l'intégrité des tables. Les import pourront être réalisés à partir de fichiers Excel mis en forme. Des exports sur Excel seront facilement réalisables.
- **Éditions et états personnalisés** : Des outils simples permettront d'effectuer des requêtes paramétrables dans les tables pour éditer des états (numériques ou graphiques), afficher des listes à l'écran ou exporter des données sur Excel. Des outils permettront aussi de personnaliser des éditions. Ces personnalisations pourront comporter des mises en page avec zones topologiques (première page, en tête, pied de page, saut de page, etc.) et permettront l'intégration de données issues de n'importe quel champ enregistré dans le système, sous réserve du respect de l'intégrité des requêtes effectuées au sein d'une même page. Des champs calculés (fonctions dépendantes d'autres champs) pourront également être mis en œuvre. L'utilisation de ces outils sera à la portée d'utilisateurs spécialement formés, mais qui ne seront pas nécessairement des informaticiens spécialistes en programmation.
- **Tables et champs libres** : Dans les tables existantes, des "champs libres" seront paramétrables pour gérer des informations spécifiques. Par ailleurs, des "tables libres" seront paramétrables pour gérer des informations indépendantes.
- **Gestion du Workflow** : Pour chaque processus mettant en œuvre un enchaînement d'opérations, un Workflow pourra être paramétré qui décrira :
 - La nature des opérations,
 - Leur enchaînement,
 - Les prérequis et les attendus de l'opération,
 - Les utilisateurs concernés
 - Les délégations de signature

- La validation (signature électronique) d'une opération générera un message d'invitation (e-mail) à l'utilisateur suivant du processus. La validation de l'opération suivante sera conditionnée par la validation de la précédente.
- **Historique** : Toutes les saisies et transactions enregistrées feront l'objet d'un enregistrement historique comportant le Login, la date, l'heure et la nature de la transaction. Certains champs (par ex. prix), à définir lors de la mise en œuvre, doivent garder l'historique des modifications/suppressions. Il sera possible d'interroger l'historique sur n'importe lequel des champs précédemment mentionnés.
- **Autres fonctionnalités** : Sans constituer une liste exhaustive, les fonctionnalités suivantes seront couvertes :
 - Multidevises
 - Multilingue : Le système doit supporter plusieurs langues applicables sur les éditions;
 - Synchronisation des utilisateurs : La solution doit permettre une intégration des annuaires d'utilisateurs (ex : AD).
 - Gestion automatique des sauvegardes et mise en mode sûr en cas de coupure d'énergie de longue durée ;

Gestion simple des procédures de sauvegarde et de restauration.

8. Conclusion:

Dans ce chapitre, nous avons présenté l'Entreprise Portuaire de Skikda. Ensuite, nous avons étudié le système utilisé actuellement par l'EPS pour la gestion des escales dans le but de faire ressortir ses limites et faiblesses afin de proposer une solution sous forme d'un ERP regroupant toutes les fonctionnalités intervenant lors de la gestion des escales et intégrant de nouvelles fonctionnalités indispensables lors de cette gestion.

1. Introduction :

Dans ce chapitre, nous allons présenter les principaux concepts utiles pour la réalisation de notre application, et qui permettent d'offrir une vision claire sur les différents éléments du application.

Nous allons décomposer ce chapitre en deux parties :

- La première partie donne une idée générale sur ERP : Entreprise Ressource Planning, leur définition, leur histoire ainsi que le principe de base de leur fonctionnement. Nous donnerons aussi leurs avantages et inconvénients, types et son architecture

- La deuxième partie nous allons définir ODOO, la raison pour laquelle nous l'avons choisi son historique, ses structure, ses composantes, ainsi que ses différentes fonctionnalités.

2. Présentation des ERP:

2.1. Définition:

L'ERP vient de l'anglais « Entreprise Ressource Planning ». On utilise parfois dans le monde francophone la dénomination PGI (Progiciel de gestion intégré) mais la terminologie anglo-saxonne prime.

Un système ERP (Enterprise resource planning) est un type de logiciel que les entreprises utilisent pour gérer leurs activités quotidiennes telles que la comptabilité, les achats, la gestion de projets, la gestion des risques et la conformité, ainsi que les opérations de supply chain. Une suite ERP complète comprend également un logiciel de gestion de la performance (EPM) qui aide à planifier, budgétiser, prévoir et générer un rapport sur les résultats financiers d'une entreprise.

Les systèmes ERP relient une multitude de processus métier et activent le flux de données entre eux. En collectant les données transactionnelles partagées d'une entreprise à partir de plusieurs sources, les systèmes ERP rendent la duplication des données inutile et assurent l'intégrité des données avec une "source unique d'informations fiables".[2]

2.2. Historique:

Le MRP (Materials) est l'ancêtre de l'ERP (Enterprise Resource Planning) a vu le jour dans les années 1960 et s'inspire du programme de production de Toyota. Cela visait

spécifiquement à résoudre le problème de la planification des besoins en composants. Pour de nombreuses organisations, l'ERP est avant tout une fonction financière, mais la logistique est un domaine historique.

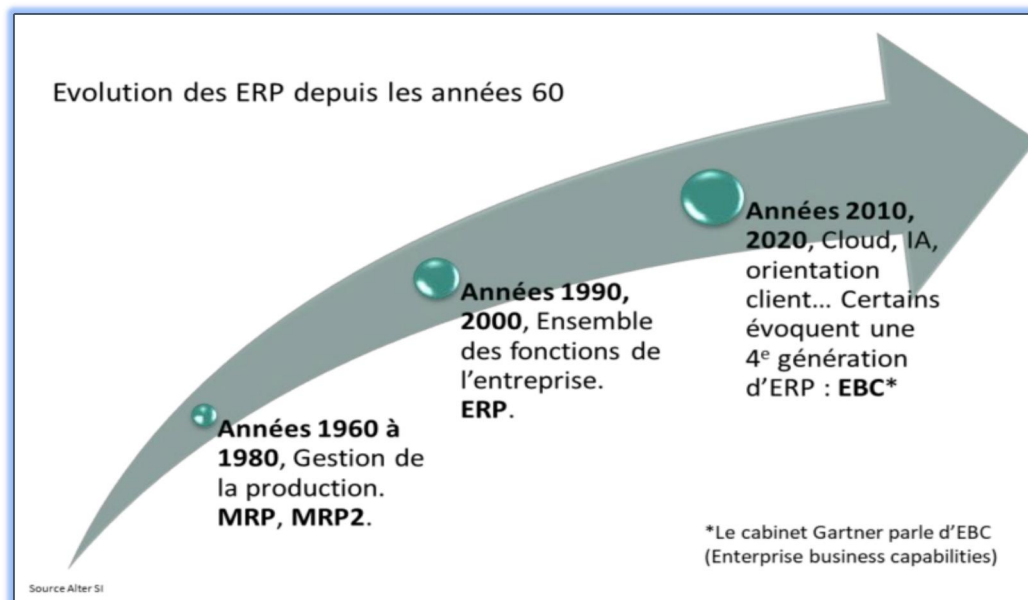


Figure 4: Schéma d'évolution des ERP depuis les années 60[3]

C'est à partir des années 1990 que le nom d'ERP est utilisé, son utilisation est à présent étendue à l'ensemble des domaines de l'entreprise. A commencer par la finance, la vente, les ressources humaines etc...

Avec l'essor d'internet, les logiciels ERP sont devenus presque indispensables aux entreprises. Ils permettent une meilleure organisation de ces dernières, en participant à instaurer des processus d'entreprise. Les erreurs de manipulation des données diminuent et la fiabilité de celles-ci augmente.

Nous vivons dans les années 2010, 2023, probablement l'émergence d'une nouvelle génération d'ERP. Il devient désormais possible aux éditeurs de logiciels ERP d'envisager l'ajout de fonctionnalités en faisant appel à l'Intelligence artificielle, à la manière du modèle Cloud, qui permet l'accès à de grandes puissances de calcul ou de stockage à la demande.[4]

2.3. Principe de base d'un ERP:

Le principe de base de cet outil (ERP) est de fournir aux entreprises des applications informatiques servant différentes fonctions (vente, approvisionnement, production, etc.) de façon modulaire tout en partageant une base de données commune unique .



Figure 5: Schéma d'un ERP [5]

Une autre partie importante de l'ERP qui a révolutionné l'utilisation de ces outils de manière impressionnante est l'utilisation de moteurs dits de Workflow. Cela vous permet de déclencher automatiquement des actions, des alertes et des informations selon des critères précisément définis.

Pour résumer, on peut parler d'ERP lorsqu'on est en présence d'un système informatique composé de plusieurs modules partageant une seule et même base de données et un système automatisé prédéfini et paramétrable (moteur de Workflow).

Les flux de travail ou le Workflow peuvent être associés à n'importe quel objet dans OpenERP, et sont entièrement personnalisables. Le Workflow est utilisé pour structurer et gérer les cycles de vie des objets et documents commerciaux (ex: bon de commande, facture, ordre d'approvisionnement ...), et pour définir des transitions, des déclencheurs, etc. avec des outils graphiques.

2.4. Objectif des ERP:

Les implémentations ERP sont les entreprises transforment leurs investissements en profits. Ainsi, l'objectif de la mise en place d'un progiciel de gestion pour l'entreprise l'intégration donne à l'organisation un levier de croissance clé. Dans les temps Dans le numérique, il est important d'avoir un ensemble d'outils comme celui-ci fournis par les ERP pour mieux gérer et améliorer la production, Réduisant ainsi les coûts et les pertes et augmentant les marges bénéficiaires.

Les ERP grâce à leur architecture modulaire, permettent à l'entreprise d'adapter les fonctionnalités fournies à son mode de travail et de perfectionner ses processus et ses activités. De ce fait, il est indispensable de bien identifier au préalable ses besoins pour que les fonctionnalités de l'ERP soient en parfaite adéquation avec les objectifs visés par l'entreprise.

[6]

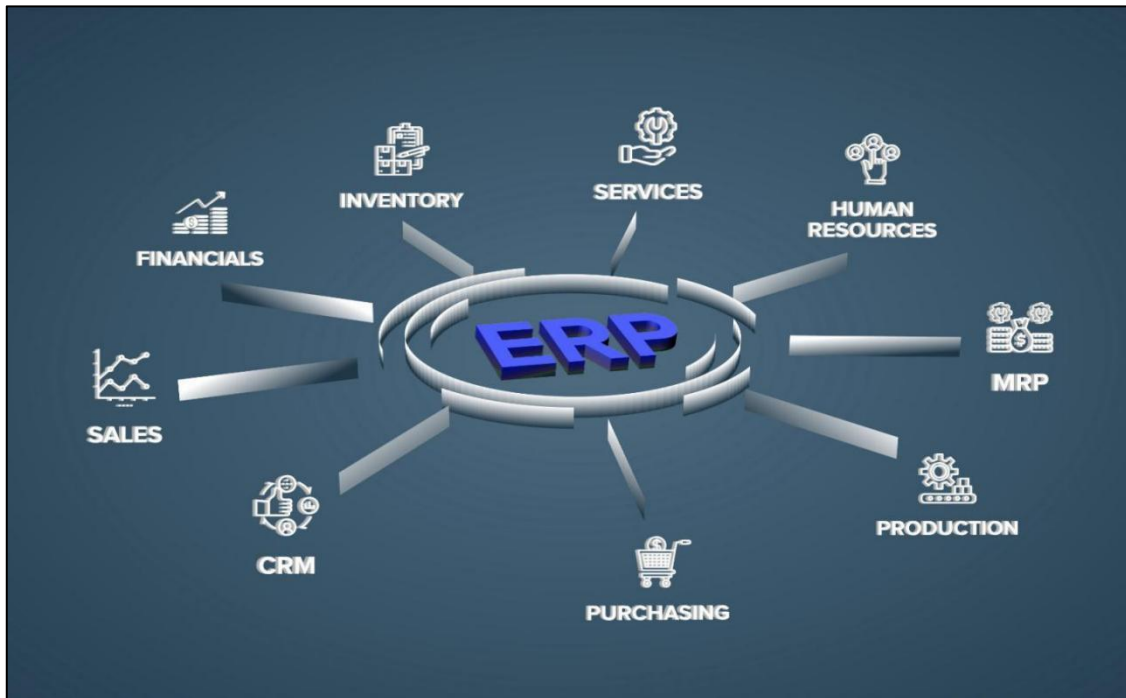


Figure 6: Les composants d'un ERP [7]

2.5. Avantages et inconvénients des ERP:[8]

2.5.1. Avantages:

- Centralisation de l'ensemble des données pour optimiser la recherche de l'information.
- Partage des données en temps réel facilitant les flux d'informations et d'échanges en interne.
- Aide à la prise la décision se basant sur les statistiques qui permettent d'être réactif en cas d'urgence.
- Accessible sur plusieurs appareils connectés à internet, s'adaptant ainsi aux conditions de télétravail.
- Optimisation de certains coûts, par exemple pour les ERP SaaS, la maintenance est effectuée par l'éditeur.

- L'accès à l'information est uniformisé et cohérent, il est vérifié par l'ensemble des membres.
- Les équipes projet sont formées à un seul outil
- Un outil permettant de standardiser les processus internes pour avoir une certaine cohérence aux niveaux des réalisations.
- C'est la solution évolutive qui s'adapte en fonction des innovations technologiques et des évolutions du marché.

2.5.2. Inconvénients:

- L'implémentation d'un logiciel ERP au sein de l'entreprise prend du temps. Est prise en compte l'analyse des besoins actuels pour personnaliser l'interface et la formation de l'équipe projet jusqu'à la prise en main complète du logiciel.
- Le coût de la licence du produit, 5 000 à 12 000 € par utilisateur (matériel + licence + intégration + formation + maintenance),
- Le mauvais choix du logiciel peut coûter très cher à votre entreprise. Assurez-vous qu'il corresponde bien aux attentes et besoins de vos équipes.
- La mise en place d'un ERP en entreprise n'est pas une étape simple. La rigueur est primordiale aussi bien dans le paramétrage que dans les mises à jour des données.

Plus qu'un simple logiciel, un ERP est un véritable projet stratégique demandant une intégration totale d'un outil logiciel au sein d'une organisation et d'une structure spécifique. Sa mise en place dans l'entreprise entraîne des changements importants des habitudes de travail d'une grande partie des employés et requiert l'implication de l'ensemble du management

2.6. Les principaux éditeur d'ERP:

Il existe deux types d'ERP. ERP propriétaire que les entreprises publient et doivent acheter une licence, et ERP open source "gratuit".

2.6.1. Les ERP propriétaire du marché :

Aujourd'hui, de nombreux ERP propriétaires existent sur le marché. Nous évoquerons dans ce qui suit, quelques grands éditeurs :

SAP (Business One): En tant que premier fournisseur ERP mondial, SAP votre serveur. Ses modules couvrent toutes les fonctions de l'entreprise et chaque module couvre tous les besoins de gestion. Ce progiciel a rapidement rencontré un énorme succès auprès des grandes entreprises en proposant un progiciel Multi-langues et Multi-devises.



Oracle (Jad Edwards): Oracle a lui fait le choix des acquisitions en se dotant des suites Peoplesoft et JD Edwards au début des années 2000. A cette époque, la réputation de Peoplesoft n'était plus à faire en matière de gestion des ressources humaines, avec des fonctionnalités de pointe et particulièrement novatrices par rapport à la concurrence. Au fur et à mesure du temps, la solution s'est vu adjoindre des modules de gestion financière, de la distribution et de la production. Le rapprochement avec JD Edwards a permis à la solution Peoplesoft de cibler des entreprises de plus petite taille (PME). [9]



NFOR: Infor propose une solution ERP extrêmement flexible et personnalisable, Infor® cloud ERP. Il s'agit d'un logiciel en SaaS qui offre une gestion rationalisée et une visibilité approfondie de tous les aspects de l'entreprise, intégrant, entre autres, la comptabilité, les ressources humaines mais aussi la gestion de la chaîne d'approvisionnement, des projets et de la relation client. Grâce à des outils de reportage et des plates-formes de collaboration intuitifs et simple d'utilisation, votre équipe peut prendre les meilleures décisions plus rapidement et surtout de n'importe où. De plus, grâce à l'hébergement sécurisé par Amazon Web Services, vous disposerez toujours de la dernière version de votre logiciel sans avoir à vous occuper des mises à jour et de la maintenance matérielle. [9]



MICROSOFT: Microsoft est devenu l'ERP idéal pour les organisations PME (petites et moyennes entreprises). Depuis 2000, Microsoft a acquis Great Plains (qui a déjà acquis RealWorld et Solomon) et Navision (qui a déjà acquis Axapta). Alors que Microsoft a échoué sur l'infâme intégration de Project Green qui consiste à fusionner ces solutions.

Dans une centrale ERP, la société fait un travail impressionnant pour continuer à faire

progresser ces solutions de manière indépendante. Les produits Axapta et Navision offrent une forte capacité de distribution et de fabrication et sont beaucoup plus populaires en Europe qu'en Amérique du Nord. Great Plains reste une chaîne favorite et offre les meilleurs résultats financiers parmi les suites de produits MS. Solomon dispose également d'une chaîne fidèle VAR, de Financial Finance solides et d'une comptabilité de projet et d'un coût de travail compétitifs.

Great Plains et Solomon ont des ressources de canal plus profondes et beaucoup plus de parts de marché aux États-Unis que Navision et Axapta. Malheureusement, la maintenance et l'évolution de Solomon sont largement externalisées (retour à certains des fondateurs et auteurs de logiciels originaux), de sorte que nous ne saurons pas combien de temps ce produit survivra.

2.6.2. Les ERPs Open Source:

Ils sont relayés par des partenaires (SSII, Cabinets de Conseil) pour le support. L'implémentation d'un progiciel Open Source revient moins chère, puisqu'il n'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 de l'assistance technique. Voici la liste des principaux progiciels Open Source :

- **Aria** : est le cœur de la gamme de produits Aria 4 XP, il couvre tous les domaines fonctionnels internes nécessaires pour gérer une entreprise. Il a été développé à base d'un ERP open source appelé Nola, et son environnement est PHPMYSQL.

- **Compiere** : Progiciel de Gestion Intégré (PGI), Gestion de la Relation Client (CRM) Open Source, Open Source pour les Petites et Moyennes Entreprises (PME) dans la Vente et le Service.

L'application est fournie sous double licence GPL et sa propre licence.

Les sources peuvent être personnalisées.

Le support technique et la documentation sont payants. Son origine est JörgJank.

- **Odoo** : Anciennement OpenERP ou encore TinyERP, est un progiciel libre de gestion intégrée comprenant des modules de gestion des ventes, des relations clients, des projets, des entrepôts, de production, de comptabilité et de ressources humaines. Son environnement est constitué principalement du PostgreSQL et XML.

2.7. Les types ERP:

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

TYPES DE LOGICIELS ERP	
ERP GENERALISTES	Un logiciel unique qui propose les fonctionnalités de bases. Il répond à l'ensemble des besoins d'une entreprise et peut être implémenté sur une large majorité des secteurs d'activité. Non personnalisables.
ERP SPECIALISES	Propose des fonctionnalités adaptées à chaque métier et surtout chaque secteur d'activité. Les secteurs d'activités couverts sont : <ul style="list-style-type: none"> La santé, la pharmacie, L'agroalimentaire, La chimie, la biologie, Le commerce La logistique.
ERP OPEN SOURCE	Logiciels libres qui n'imposent pas l'acquisition d'une licence. Sont nettement moins cher, mais les services associés sont minimes. S'adaptent parfaitement aux besoins des PME, mais imposent de bonnes connaissances en informatique.
	ERP non commercialisés sous forme de

<p>ERP EN MODE SAAS</p>	<p>logiciel, mais sous forme de service.</p> <p>La différence notable des ERP en mode SaaS (Software as a Service) 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>
------------------------------------	--

Table 2 : Tableau des différents types de logiciels ERP [10]

2.8. L'architecture d'un ERP:

2.8.1. L'architecture technique :[11]

L'ERP se caractérise par une architecture Client/serveur à trois niveaux, nous proposant la figure suivante pour décrire cette architecture :

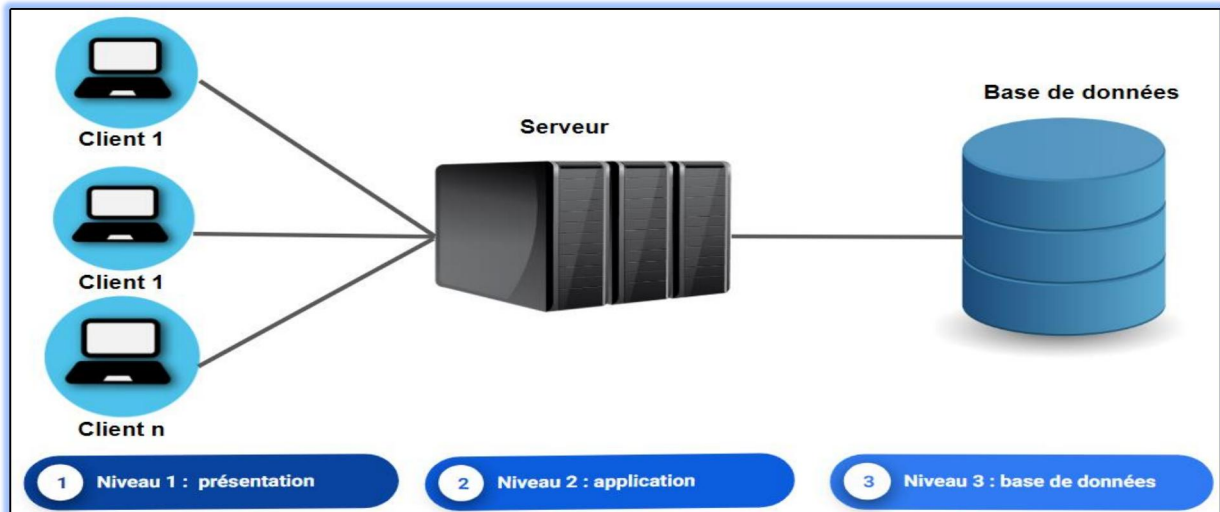


Figure 7 : Architecture technique d'un ERP [12]

- **Niveau 1**

Ce niveau correspond à la couche présentation, équipé d'un interface utilisateur (l'interface du logiciel) qui va être chargé de la présentation de l'information.

- **Niveau 2**

Ce niveau correspond à celle du serveur d'application qui fournit l'information mais en faisant appel à un autre serveur.

- **Niveau 3**

C'est le niveau d'accès à la base de données, cette partie conserve toutes les données de l'entreprise.

2.8.2. L'architecture modulaire :[13]

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

Un ERP est modulaire dans le sens où il est possible de n'avoir qu'une ou plusieurs applications en même temps, ou peu à peu. Les applications modulaires telles que les ERP permettent d'être sûr de la compatibilité des modules entre eux, ils s'imbriquent comme des blocs de Loge et fonctionnent ensemble (pas de vérification de compatibilité à effectuer).



Figure 8 : Les modules principaux d'un ERP [13]

- **CRM (Customer Resource Management)**

Ce module permet de créer et de gérer une relation entre l'entreprise et le client afin d'identifier les besoins et préférences de ce dernier et ainsi travailler sur des offres commerciales et publicitaires.

Le but de ce module est de fidéliser un grand nombre de clients. En effet, cela permet aux entreprises d'améliorer leurs compétitivités et augmenter leur chiffre d'affaires.

- **FRM (Finance Resource Management)**

Ce module nous permet de gérer en temps réel tous les aspects liés à la comptabilité financière de l'entreprise. En effet il inclut des fonctions liées à la comptabilité tiers, analytique et générale, il offre également des modules de gestions des investissements et de l'immobilisation de l'entreprise.

- **Vente et marketing**

Ce module est très convoité par les entreprises, en effet, il permet à ces dernières d'améliorer leurs stratégies de vente et de mettre en place des stratégies de marketing adopté au marché et aux clients afin de rester compétitives et d'augmenter leurs chiffres d'affaires.

Ce module inclut entre autres les fonctionnalités suivantes :

- o Gestion de la vente au comptoir.
- o Gestion de l'e-commerce/ e-service.
- o Génération des devis et factures.
- o Gestion des livraisons.
- o Gestion des stratégies de marketing.
- o Génération d'un rapport de vente détaillé.

- **HRM (Human Resource Management)**

Ce module est conçu pour automatiser les processus liés à la gestion des employés de l'entreprise. Il offre des fonctionnalités pour :

- o La gestion administrative du personnel
 - Gestion des absences (vacance, maladie, accident)
 - Gestion du paiement
- o La gestion opérationnelle
 - Rendement et performance
- o Autres
 - Gestion de l'apprentissage et formation
 - Gestion de la carrière

- Recrutement

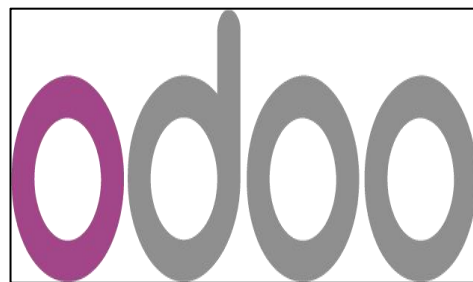
- **SCM (Supply Chain management)**

Certains ERP incluent un module de gestion de la chaîne d’approvisionnement, qui permet à l’entreprise d’avoir une meilleure visibilité du stock en temps réel. Ce module inclut la gestion des prestations des fournisseurs, la gestion des achats, gestions des prévisions et la planification de l’approvisionnement.

3. Présentation ODOO

3.1. Définition:

Odoo, anciennement OpenERP2 et Tiny ERP, est initialement un progiciel de gestion intégré open-source comprenant de très nombreux modules permettant de répondre à de nombreux besoins de gestion des entreprises ou de gestion de la relation client (CRM). Le logiciel est utilisé par



plus de cinq millions d’utilisateurs pour gérer leurs entreprises à travers le monde³. Odoo est le système ERP open-source le plus populaire.[14]

Odoo propose donc , une suite d’applications interconnectées , et partagées au sein d’une seule base unique, et qui vont être unifiées entre l’ensemble des fonctions de l’entreprise. Sa modularité permettant de s’adapter à des entreprises de toutes tailles, son interface 100% web et responsive, et la richesse des fonctionnalités d’Odoo en font l’un des ERP les plus téléchargés au monde [15]

Odoo est le plus évolutif et le plus installé au monde grâce à des applications qui couvrent tous les besoins d’une entreprise, depuis la gestion de la relation client, la création de sites web et d’e-mails, le commerce, la production, la gestion des stocks et la comptabilité. Et les ERP, etc. sont tous entièrement intégrés. C’est la première fois qu’un éditeur de logiciel atteint un tel niveau de fonctionnalité.

La mission d’Odoo est de fournir aux entreprises de toutes tailles et de tous secteurs des solutions professionnelles et faciles à utiliser pour tous les utilisateurs. Parmi ses clients figurent WWF, Danone, Hyundai, La Poste, Toyota et Jamba Juice.[16]

Odoo est une suite d’applications open source qui propose de multiples applications pour la gestion de votre entreprise : CRM, e-commerce, comptabilité, gestion des stocks, gestion de projet... Le logiciel est entièrement modulable, et permet aux entreprises de construire un

ERP sur mesure en choisissant les applications dont ils ont besoin.

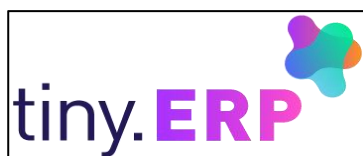
La plate-forme propose 8 modules, avec des applications variées :

- ✓ Finance : comptabilité, facturation, notes de frais, feuille de calculs, signature
- ✓ Ventes : CRM, gestion des points de vente, connecteur Amazon, gestion des abonnements
- ✓ Sites web : création de sites, gestion e-commerce, chat en ligne, e-learning
- ✓ Inventaire et MRP : gestion d'entrepôt, maintenance, gestion du cycle de vie produit, achat fournisseurs, bons de commande
- ✓ Ressources humaines : gestion des employés, congés, évaluations
- ✓ Marketing : e-mail et SMS marketing, automatisation, création de sondages
- ✓ Services : feuille de présence, gestion de projet, planification de rendez-vous
- ✓ Productivité : VoIP, discussions internes, validations

Les applications sont compatibles et permettent de centraliser les informations tout en automatisant les processus de gestion de l'entreprise et de la relation client. [17]

3.2. Historique:

L'idée de Odoo a germé il y a maintenant 18 ans, en 2005. Fabien Pinckaers, le fondateur, émet le rêve de devenir le leader sur le marché du management d'entreprise avec un logiciel de gestion complet Open Source.

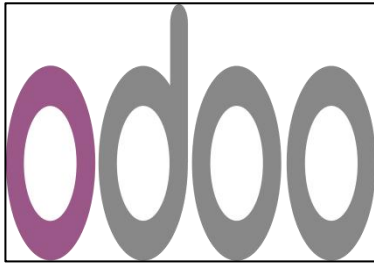


En 2005, il a développé Tiny ERP le progiciel libre de gestion intégré qui a pu à peine prendre sa place avec les géants comme SAP. Les versions lancées sous le nom de Tiny ERP sont 1.0, 2.0, 3.0 et 4.0.

En 2008, TinyERP devient OpenERP. Avec un tout petit ERP (« tiny » signifie petit en anglais), la question pour Fabien Pinckaers est de savoir comment développer une offre pour les entreprises qui gèrent des milliers d'employés et des flux importants de données et de capitaux. La communauté Odoo s'accroît, propose à la vente toujours plus de modules et la structure de



l'entreprise augmente. Fabien Pinckaers garde en tête l'idée de concurrencer les plus grands leader du marché.



En 2015, OpenERP devient Odoo : son activité se diversifie et l'entreprise ajoute plus de 3000 nouveaux modules à son logiciel. En Octobre 2017, la version 11 de Odoo devient disponible, avec toujours plus de fonctionnalités, toujours dans le but de booster la performance des entreprises.[18]

3.3. Source d'ODOO:

ODOO est basé sur une structure modulaire, permettant d'ajouter et de modifier très facilement de nouveaux modules et fonctionnalités. Vous pouvez supprimer d'autres utilisateurs sans toucher à l'ensemble du système.

Un module est une arborescence de fichier placé dans un dossier dédié, avec une structure prédéfinie contenant du code Python et des fichiers XML et dans lesquels existe la définition de structure de données, les formulaires, les rapports, ainsi que la définition du Workflow, etc ...

3.3.1. L'architecture du système [19]:

L'architecture du système Odoo est composée de 3 couches :

- ◆ **Un serveur de base de données PostgreSQL:** Odoo utilise une couche ORM « Object Relational Mapping » pour la persistance de ses objets métier et la gestion des la base de données.
- ◆ **Un serveur d'application:** contenant les objets de gestion, le moteur de Workflow, le générateur d'édition, etc...
- ◆ **Un serveur de présentation:** qui permet à l'utilisateur de se connecter à Odoo avec n'importe quel navigateur internet (Google chrome, Firefox,...).

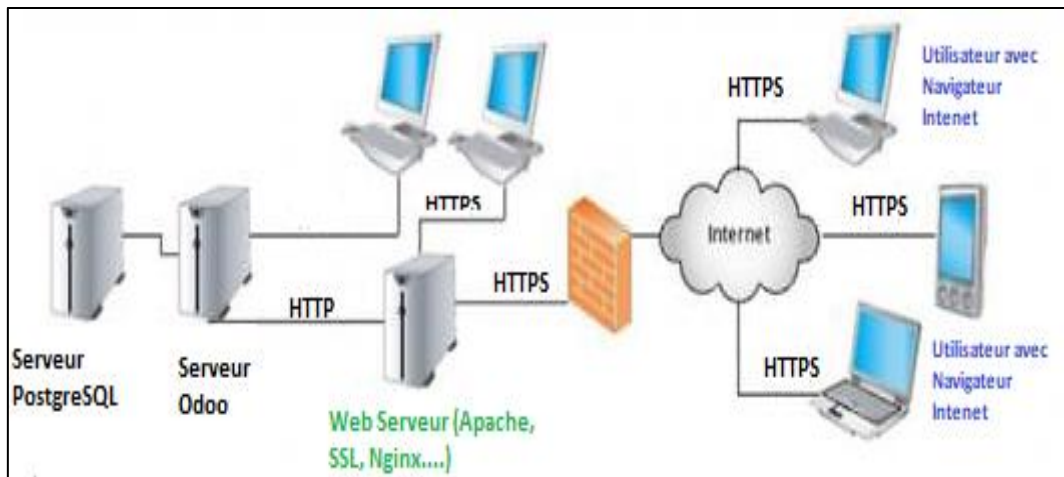


Figure 9 : Architecture technique de Odoo [19]

3.3.2. Modèle vue contrôleur [19]:

MVC est un patron de conception très répandu pour la réalisation de sites web.

Il nous permet de structurer notre code en séparant l’affichage, les contrôles ainsi que l'accès à la base de données.

MVC est l’acronyme de Model View Controller, elle permet de décomposer une application informatique en trois sous parties dans le but d'améliorer la maintenance de l'application :

- **Modèle** : s’occupe de gérer les données de l’application, elle contient également la logique en rapport avec les données : validation, lecture et enregistrement (bases de données, fonctions, procédures).

- **Vue** : s’occupe de la partie visible de l’interface graphique (présentations, saisie et validation des données).

- **Contrôleur** : s’occupe des actions effectuées par l’utilisateur, elle est en quelque sorte l'intermédiaire entre le modèle et la vue (fait le lien entre l'utilisateur et le reste de l'application.).

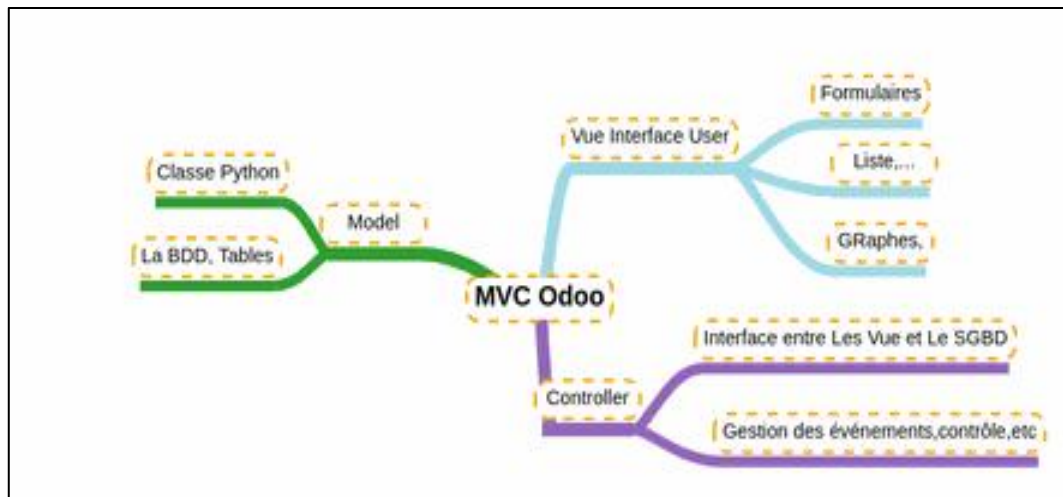


Figure 10: Présentation de l'architecture du modèle MVC[19]

3.4. Composants principaux du module d'ODOO:

Le module Odoo ERP interagit avec plusieurs autres composants et technologies. Par conséquent, nous distinguons les composants suivants:

- **Objet métier** : déclarés comme des classes Python (modèle) qui étendent la classe « OSV. Modelé » ; toute classe modèle doit en hériter pour le déploiement du module OpenERP. La persistance de ces ressources est entièrement gérée par OpenERP.
- **Données** : Fichiers XML/CSV contenant des métadonnées (vues et déclaration de Workflow), données de configuration (paramètres du module) et données de démonstration (Facultatif, mais recommandé pour les tests exemple d'idée).
- **Assistants** : formulaires interactifs utilisés pour aider les utilisateurs, souvent disponibles en actions contextuelles sur les ressources.
- **Rapports** : Modèles de rapport RML (au format XML), MAKO ou OpenOffice. Combinez avec n'importe quel type de données d'entreprise pour générer des rapports HTML, ODT ou PDF.
- **Les Workflow** : Odoo utilise systématiquement un moteur de Workflow afin que lorsque les données sont saisies dans un système d'information, elles puissent être diffusées et fournir une vue logique pertinente dans tous les modules du système qui en ont besoin selon un planning. Défini. Ainsi, on peut parler d'ERP lorsque l'on dispose d'un système d'information constitué de plusieurs applications partageant des applications de base de données unique.

3.5. Les fonctionnalités d'ODOO:

Odoo est une solution ERP open source entièrement personnalisable et modulaire. Chaque module nous permet de gérer les fonctionnalités de vous pouvez l'installer indépendamment de votre entreprise.

Odoo est une suite complète d'applications métier couvrant tous les besoins métiers d'une entreprise. Environ 1500 modules sont disponibles.

Il existe des modules pour les activités et domaines suivants :

- ◆ **Gestion de la comptabilité et des finances:** préparez vos déclarations de TVA, générez les écritures comptables des factures, gérez vos rapports de gestion...
- ◆ **Gestion de la relation client avec le CRM:** gérez vos contacts commerciaux, les campagnes de mailing, l'avancement des négociations commerciales...
- ◆ **Gestion des ressources humaines:** créez vos fiches employés, enregistrez les frais professionnels, les demandes de congés ...
- ◆ **Gestion des ventes:** créez des devis personnalisables, gérez les tarifs clients, facturez...
- ◆ **Gestion des stocks:** visualisez vos stocks, validez les réceptions et expéditions...
- ◆ **Gestions des achats:** gérez les fournisseurs et leurs tarifs, gérez les livraisons directes...
- ◆ **Gestion de la production:** suivez vos chaînes de production, les nomenclatures, les ordres de fabrication, MPS...
- ◆ **Gestion de projet:** planifiez des tâches pour vos projets, suivez les avancements des tâches réalisées...



Figure 11 : Les fonctionnalités d'odoo [20]

3.6. Particularités de l'ERP ODOO:

Odoo ERP: gère toutes les fonctions critiques de l'entreprise. Il propose également plus de 500 applications spécialement conçues pour faciliter la gestion et augmenter la productivité.

- **Gestion et création de site internet :** Odoo V8 est actuellement le seul ERP qui propose un CMS intégré 64 . Cette solution est particulièrement efficace pour les projets de boutiques en ligne, les portails clients et les sites publicitaires.

spécialisé.

- **Gestion commerciale:** Odoo V8 couvre toutes les fonctions de gestion commerciale (CRM - Devis - Commande - Facturation - Comptabilité) et digitalise davantage les devis et factures. Plus besoin d'envoyer des PDF. Utilisez Odoo pour envoyer le lien et l'interlocuteur s'occupe du reste

- **Marketing:** Odoo V8 peut être utilisé pour gérer toutes vos campagnes de suivi de portefeuille de recherche de prospects et clients. Utilisons des modules plus dédiés à la gestion des campagnes Google Adwords 7 pour les sites Web.

- **Ressources Humaines :** Odoo V8 gère du recrutement à la gestion des employés via le site internet. Cela permet à tous les employés de s'inscrire aux événements et d'être avertis.

- **Augmenter la productivité avec Odoo:** Avec plus de 500 modules dédiés, les utilisateurs d'Odoo peuvent augmenter la productivité de leur entreprise. L'activation du module ne nécessite aucune intervention technique. [20]

4. CONCLUSION:

Dans ce chapitre, nous avons fourni le contexte général du projet à travers la définition des ERPs et la définition de progiciel ODOO . Ce dernier est actuellement une alternative de marché libre aux systèmes de gestion des entreprises avec toutes leurs solutions conçues pour satisfaire le plus grand nombre de personnes, les entreprises en intervenant dans plusieurs domaines.

1. Introduction

La conception de toute solution informatique est d'une grande importance et elle doit être traitée avec rigueur et précision, car elle constitue la base du système à développer.

Pour ce 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 leur besoins, leur tâches et scénario.

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.

2. Objectif de notre projet:

Le principal objectif de notre projet est de développer le cœur d'un web ERP, une base de données, où certaines tables seront créées, par l'utilisateur selon les besoins de son entreprise, à travers une interface .

3. Unified Modeling Language (UML):

3.1. Définition :

UML est un acronyme anglais de Unified Modeling Language ou Le Langage de Modélisation Unifié en français.

L'UML est un langage graphique de modélisation informatique. Ce langage est désormais la référence en modélisation objet, ou programmation orientée objet. Cette dernière consiste à modéliser des éléments du monde réel (immeuble, ingrédients, personne, logos, organes du corps...) ou virtuel (temps, prix, compétence...) en un ensemble d'entités informatiques appelées « objet ».

L'UML est constitué de diagrammes qui servent à visualiser et décrire la structure et le comportement des objets qui se trouvent dans un système. Il permet de présenter des systèmes logiciels complexes de manière plus simple et compréhensible qu'avec du code informatique. L'UML a des applications dans le développement logiciel, mais aussi dans l'industrie (pour modéliser les flux de processus par exemple), dans l'ingénierie ou le marketing.[22]

L'objectif principal d'UML est de définir une manière standard de visualiser la façon dont un système a été conçu et aide les équipes de projet à communiquer et à explorer les conceptions potentielles ainsi qu'à valider la conception architecturale du logiciel.

3.2. A quoi sert UML?

Dans la réalisation d'une application il est mieux de passer par la modélisation de système.

Le choix d'UML nous permet de :

- Faire collaborer des participants de tous horizons autour d'un même document de synthèse.
- Obtenir une modélisation de très haut niveau indépendante des langages et des environnements.
- Faire des simulations avant de construire un système.
- Exprimer dans un seul modèle tous les aspects statiques, dynamiques, juridiques, spécifications, etc.
- Documenter un projet.
- Encadrer l'analyse.
- Faciliter la compréhension de représentations abstraites complexes.

3.3. Type de diagrammes UML:

Les normes UML prévoient plusieurs diagrammes qui sont réparties en deux catégories :

- Diagrammes structurels qui montrent la structure statique du système

Cette catégorie regroupe les diagrammes suivants :

- Diagramme de classe.
- Diagramme des composants.
- Diagramme de déploiement.
- Diagramme d'objets.
- Diagramme de package.

- Diagramme de structure composite.
- Diagramme de profil.
- Diagrammes comportementaux qui montrent le comportement dynamique des objets dans un système UML propose les diagrammes comportementaux suivants :
 - Diagramme de cas d'utilisation.
 - Diagramme d'activité.
 - Diagramme de la machine d'état.
 - Les diagrammes d'interaction qui regroupent
 - o Diagramme de séquence.
 - o Diagramme de communication.
 - o Diagramme de timing.
 - o Diagramme de présentation des interactions.[23]

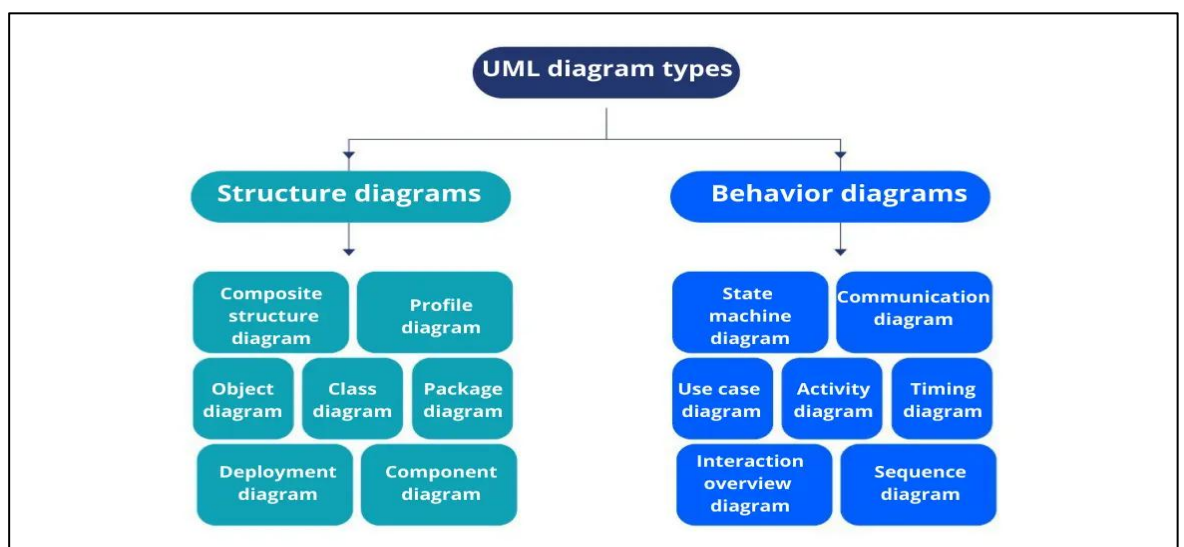


Figure 12: Les différents types de diagramme [24]

4. Analyse:

L'objectif de cette partie est de décrire notre application de façon précise et détaillée. Pour cela, nous allons commencer par spécifier les besoins fonctionnels auxquels notre application finale se doit de répondre, puis nous procédons à l'identification des futurs utilisateurs de notre système ainsi que les cas d'utilisation relatifs à chacun d'eux.

Enfin, le résultat de cette analyse sera illustré à travers des diagrammes de cas d'utilisation ainsi que quelques scénarios nominaux relatifs à certains cas d'utilisation et leurs utilisateurs.

4.1. Définition:

4.1.1. Spécification des besoins:

La spécification des besoins consiste en la détermination et la description des besoins que notre futur système devra couvrir afin de satisfaire les utilisateurs finals de notre application.

4.1.2. Acteurs:

Un acteur, au sens UML, représente le rôle d'une entité externe (utilisateur humain ou non) interagissant avec le système. Il est représenté par un bonhomme en fil de fer. On représente généralement à gauche l'acteur principal (idéalement il y en a un seul), et à droite les acteurs secondaires. Il est à noter qu'un utilisateur peut amené à jouer plusieurs rôles vis-à-vis du système et à ce titre être modélisé par plusieurs acteurs.[25]

Le formalisme d'un acteur en UML est le suivant :

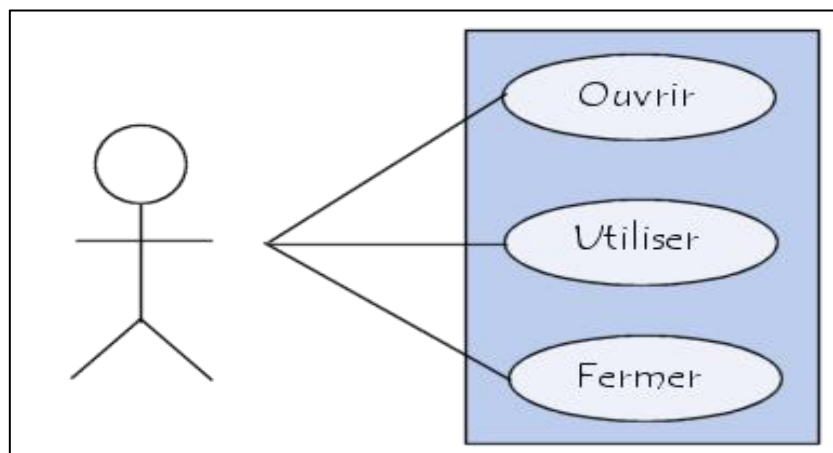


Figure 13 : Formalisme d'un acteur en UML [25]

4.1.3. Cas d'utilisation:

Un cas d'utilisation (en anglais use case) permet de mettre en évidence les relations fonctionnelles entre les acteurs et le système étudié.

Le format de représentation d'un cas d'utilisation est complètement libre mais UML propose un formalisme et des concepts issus de bonnes pratiques. [25]

- Les cas d'utilisation permettent de trouver :

- 1) les principales tâches de chaque acteur.
- 2) les fonctionnalités mal décrites dans les spécifications.
- 3) les modifications (lecture et écriture) des données du système.
- 4) les cas d'anomalie.

Soit un cas d'utilisation "recherche de produit" identifié pendant l'analyse des besoins d'un futur système logiciel de vente de produits cosmétiques en ligne, selon la notation UML celui-ci se représente comme ci-dessous :[26]



Figure 14: Représentation graphique d'un d'un cas d'utilisation [26]

4.1.4. Relation:

Une relation est une connexion entre des éléments. Dans la Modélisation, les trois relations majeures sont les dépendances, les généralisations et les associations. Une relation est représentée graphiquement par différents types de lignes utilisées pour distinguer les types de relations.

- Généralisation et implémentation
- Dépendance
- Association (association simple, association n-aire, agrégation et composition)
- Interfaces fournies et interfaces requises (UML 2.0) [27]

Une relation correspond à une interaction entre :

- **Un acteur et un cas d'utilisation :**

Cette relation décrit un échange entre l'application et un acteur, sa représentation graphique est la suivante:



Figure 15 : acteur et cas d'utilisation [27]

• **Relation entre cas d'utilisation:**

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

▪ **Relation de généralisation :**

- ⌊ Comme nous l'avons découvert lorsque nous avons traité la notion d'objet, il est également possible de spécialiser un cas d'utilisation en un autre cas d'utilisation. Nous obtenons alors un sous-cas d'utilisation.
- ⌊ Comme pour les classes, le sous-cas d'utilisation hérite du comportement du sur-cas d'utilisation. Le sous-cas d'utilisation hérite aussi de toutes les associations du sur-cas (relations d'association avec les acteurs, relations d'inclusions, et relations d'extensions).
- ⌊ Quelquefois, le sur-cas d'utilisation est abstrait (c'est-à-dire qu'il ne peut pas être instancié). Il correspond à un comportement partiel et sert uniquement de base pour les sous-cas d'utilisation qui en hériteront.
- ⌊ La relation de généralisation est représentée par une flèche avec une extrémité triangulaire.
- ⌊ Le nom d'un cas d'utilisation abstrait est écrit en italique (ou accompagné du stéréotype« abstract »).[28]

La figure suivante décrit le formalisme de cette relation :



Figure 16 : Formalisme de la relation de généralisation[28]

▪ **Relation d'inclusion :**

La relation d'inclusion sert à enrichir un cas d'utilisation par un autre cas d'utilisation (c'est une sous fonction).

La relation d'inclusion est impérative et donc systématique. Dans un diagramme des cas d'utilisation, cette relation est représentée par une flèche pointillée reliant les 2 cas d'utilisation et munie du stéréotype « include ». [28]

L'inclusion permet de :

- ⌋ **Partager** une fonctionnalité commune entre plusieurs cas d'utilisation (fig.17).
- ⌋ **Décomposer** un cas d'utilisation complexe en décrivant ses sous fonctions (fig.18).

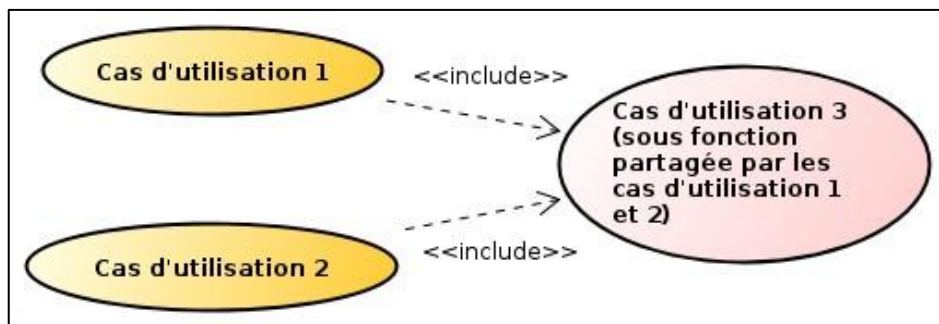


Figure 17: Représentation graphique de l'inclusion (partager)

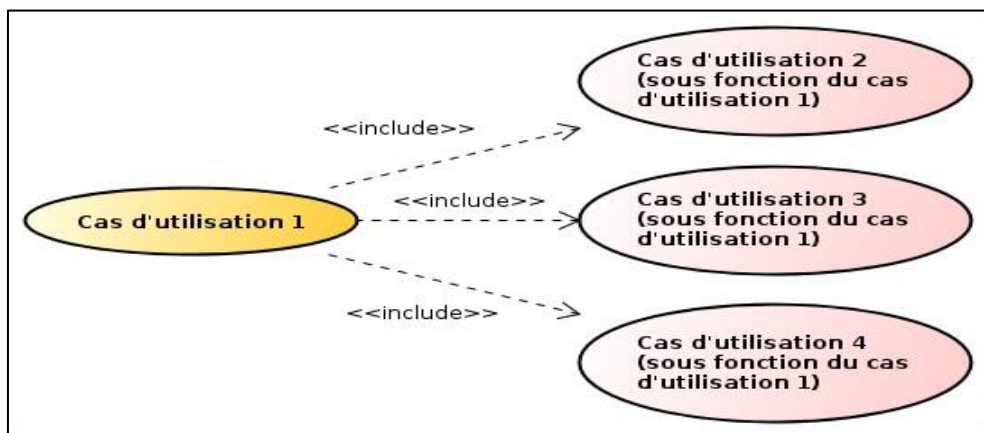


Figure 18 : Représentation graphique de l'inclusion (décomposer)

▪ **Relation d'extension :**

L'extension peut intervenir à un point précis du cas étendu. Ce point s'appelle le point d'extension. Il porte un nom, qui figure dans un compartiment du cas étendu sous la rubrique

point d'extension, et est éventuellement associé à une contrainte indiquant le moment où l'extension intervient. Une extension est souvent soumise à condition. [28]

Son formalisme est décrit par la figure suivante:

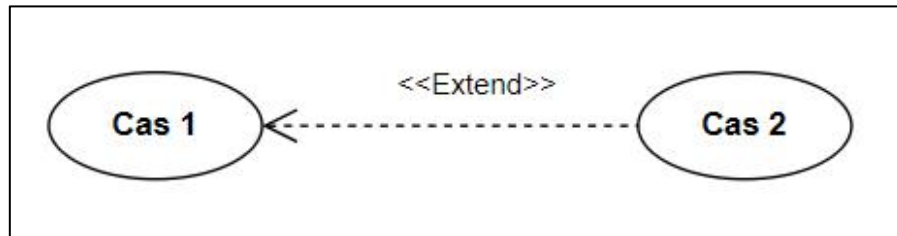


Figure 19: Représentation graphique de l'extension

4.1.5. Scénario :

Un scénario correspond à un ensemble de tâches successives qui s'exécutent par l'application lorsqu'un cas d'utilisation est sollicité par un acteur.

Un cas d'utilisation peut avoir plusieurs scénarios, qui décrivent toutes les possibilités d'exécution lors de sa mise en œuvre, il existe deux types de scénarios :

- **Scénario nominal** : Il correspond à une exécution sans erreur, et qui satisfait le besoin visé.
- **Scénario alternatif** : Il s'agit d'exécution exceptionnelle ou avec erreur qui n'atteint le but souhaité.

4.2. Le modèle métier global d'EPS:

L'entreprise EPS gère plusieurs fonctionnalités à savoir les navires, le suivi de marchandise, la gestion des taxes, les factures, etc. Pour mieux comprendre le fonctionnement de l'entreprise EPS on va présenter le modèle métier global d'EPS qui met l'accent sur les différents composants métiers. La figure ci-dessous représente le schéma du modèle métier global d'EPS qui va nous aider à mieux comprendre nos composants métiers: [1]



Figure 20 : Modèle métier global d'EPS. [1]

4.2.1. Décomposition du modèles global d'EPS en sous modèles:

Afin de décomposer notre modèle métier global en plusieurs sous modèles métier On va regrouper toutes les fonctionnalités métiers qui appartiennent aux mêmes contextes dans un modèle (nommé modèle A par ex.), et cela en suivant le pattern bounded contexte de la conception pilotée par le domaine (DDD) ensuite on va définir les liens entre les composants métiers de même modèle.

Ici on prend les composants qui sont en relation et qui partagent le même contexte on va les regrouper ensemble dans le même modèle c'est-à-dire on va limiter notre modèle à qui on va associer un nom pour le distinguer des autres et pour comprendre de quel modèle s'agit '-il par exemple on va créer un modèle qui s'occupe juste de la taxation un autre pour la facturation ainsi de suite.

Maintenant passant à la décomposition et la définition de chaque modèle de l'entreprise EPS (on a pu décomposer notre modèle métier grâce aux réunions que nous avons faite avec les experts métiers de l'EPS)

✓ Le composant Manifeste on va le mètre dans un modèle qu'on va nommer gestion manifeste.

✓ Le composant gérer navire on va le mètre dans un modèle qu'on va nommer

Administrateur.

- ✓ Les composants suivi marchandise, pointage débarquement, pointage embarquement, débarquement, embarquement vont être dans le même modèle qu'on nomme pointage
- ✓ Les composants sortie navire, arriver navire, suivi navire, placement, mouvement navire, prévision vont être regroupés dans le même modèle escale
- ✓ Les composants pilotage, moyens de servitude, chargement, remorquage, déchargement, gardiennage sont dans le modèle prestation
- ✓ Les composants taxation pilotage et taxation remorquage sont regroupés dans le même modèle nommé taxation
- ✓ Le composant poussage est seul dans le modèle nommé poussage
- ✓ Le composant facturation est dans le modèle facturation

La figure suivante représente les différents modèles de l'EPS et la relation entre les composants métiers d'un modèle ainsi que le nom de chaque modèle : [1]

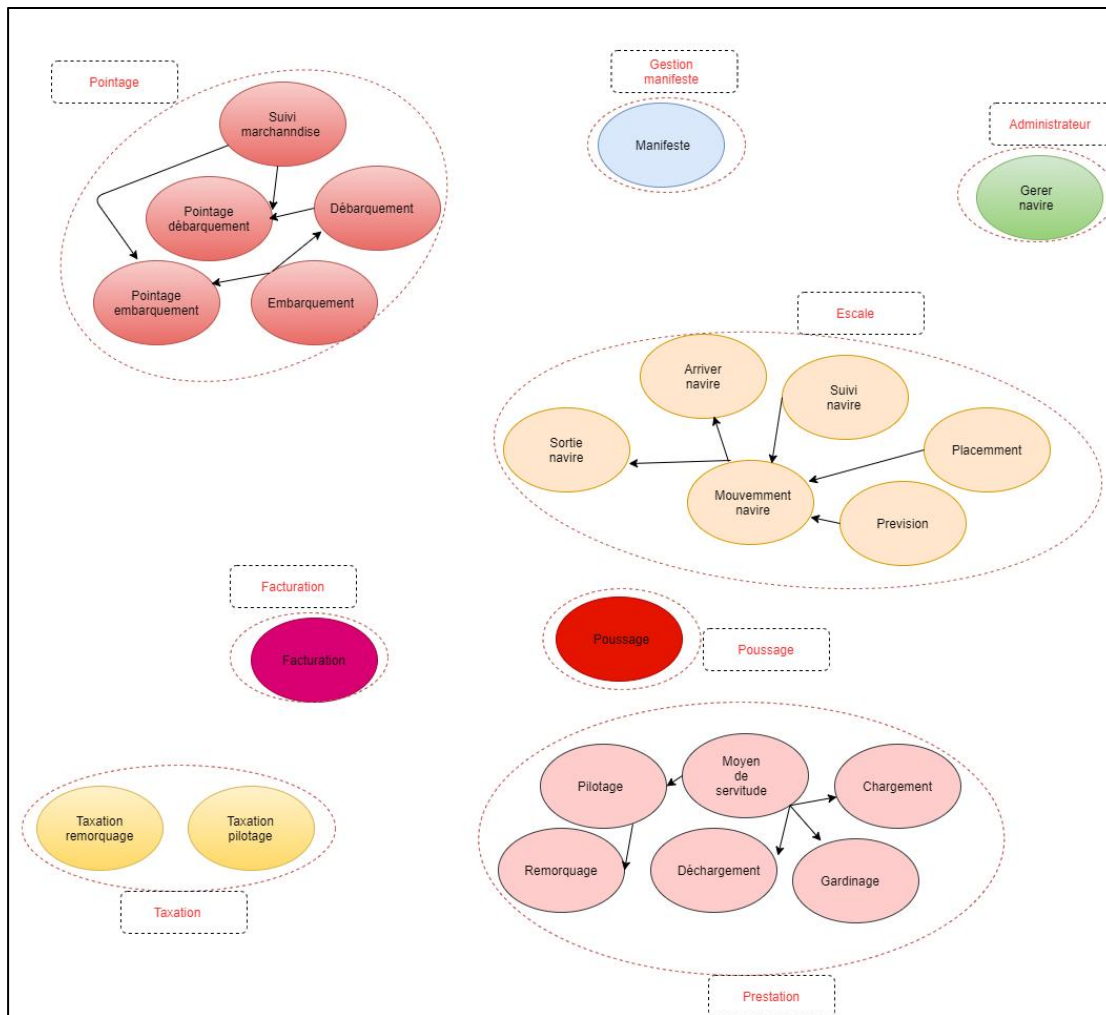


Figure 21: Le pattern bounded contexte[1]

4.2.2. Réalisation de la carte de contexte d'EPS:

Après avoir défini les modèles on va essayer de faire les liens entre les composants des modèles définis auparavant (c'est-à-dire on définit les liens entre les composants de modèle A avec les composants de modèle B). Il est nécessaire de définir les frontières et les relations entre ces modèles et cela en se basant sur la carte de contexte du Domaine Driven Design.

La figure suivante représente notre carte de contexte :[1]

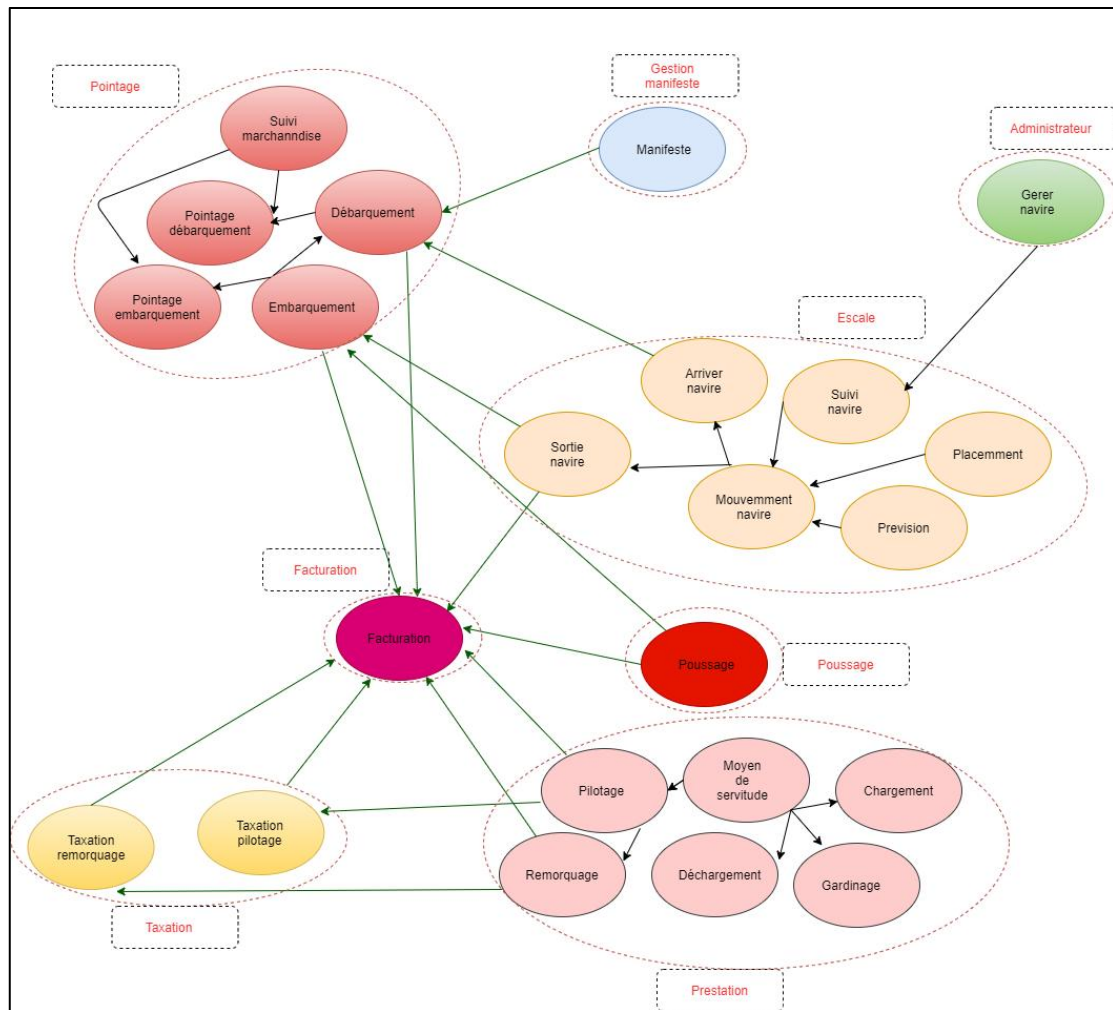


Figure 22: La carte de contexte [1]

Chaque contexte présente un microservice autonome tel que la relation inter microservices est forte et le couplage entre les microservices est faible.

Le patronne bounded contexte nous a permis d'extraire huit (8) microservices :

- Microservice Gestion Manifeste.
- Microservice Prestation.
- Microservice Escale.
- Microservice Pointage.
- Microservice Taxation.
- Microservice Facturation.
- Microservice Administrateur.

- Microservice Poussage.

Après avoir décomposé notre plate-forme en microservices nous exposerons une démonstration de la solution proposée qui concernera les microservices de la capitainerie à savoir Escale.[1]

4.3. Identification des acteurs :

4.3.1. Acteur principaux:

Les acteurs principaux de notre projet sont :

- **Communauté EPS** : Il s'agit de toutes les personnes qui ont accès à l'application, ils peuvent consulter les données publiques sans authentification.

- **Membre** : c'est toutes personnes à un compte pour s'authentifier, il a l'accès à consulter l'historique des escales des navires, consulter les notifications et de les créer.

- **Directeur** : Son rôle principale est de gérer les membres (Ajouter, modifier, désactiver), et de consulter leur historique.

- **Gestionnaire des escales (acteur)** : son rôle est de s'assurer que les annonces soit bien récupérer et valider, il peut aussi modifier des données d'une escale en cas de changement d'informations.

- **Officier du Port** : Les officiers de port sont chargés de la gestion de la CPN (Ajouter, modifier, supprimer des prévisions ou mouvements).

- **Officier de Radio** : Il ajoute le navire en rade, et il exécute les prévisions et mouvements de la CPN, de plus il peut jouer le rôle de Gestionnaire à tout moment.

4.3.2. Acteur secondaire:

Contrairement aux acteurs principaux, les acteurs secondaires n'utilisent pas le système mais le système à besoins d'eux pour fonctionner correctement. Cela est généralement un autre système avec lequel le nôtre doit échanger des informations. Notre application sollicite l'acteur secondaire suivant :

- **APCS.dz** : C'est une plate-forme nationale communautaire portuaire d'échange de données numériques, qui permet de publier ou de récupérer toutes les données concernant l'annonce d'escale des navires.

4.4. La spécification de besoins:

Notre analyse des besoins comportera deux parties, en premier lieu nous parlerons sur les besoins non fonctionnels de notre projet, ensuite, nous détaillerons les besoins fonctionnels de celui-ci et cela grâce à l'identification des différents acteurs et de leurs fonctionnalités requises.

4.4.1. Besoins fonctionnels:

Notre application doit satisfaire les besoins fonctionnels suivants :

- ✓ Recréer l'API de l'APCS pour Récupérer les annonces des navires.
- ✓ Authentification et gestion des rôles.
- ✓ Gestion d'escale : récupération d'annonce, ajout de navires en attendu, ajout de navires en rade, exécution des prévisions (Ajout de navire en quai si c'est une prévision d'entrée ou rendre navire en état fini si c'est une prévision de sortie) et des mouvements sur le navire.
- ✓ Gestion de navire : ajout, modification et/ou suppression des prévisions et mouvements.
- ✓ Archivage l'historique des membres.
- ✓ Intégration de système des notifications.
- ✓ Consultation des différents listes (Liste des navires attendus, liste des navires en rade, liste des navires en quai, liste historique des navires).
- ✓ Pouvoir imprimer les différentes listes.

4.4.2. Besoin non fonctionnels:

Ce sont les besoins permettant l'amélioration de la qualité des services offerts par une application. Dans notre travail nous avons considéré la liste des besoins non fonctionnel présentés ci-dessous :

- ✓ Sécurité : l'application doit être extrêmement sécurisée, et les données ne doivent pas être accessibles facilement à tous, mais plutôt au moyen d'un identifiant et d'un mot de passe attribués à l'utilisateur.
- ✓ Convivialité : l'application doit être facile à utiliser, et elle doit présenter un enchaînement logique entre les interfaces.
- ✓ L'extensibilité : l'application devra être extensible c'est-à-dire qu'il serait possible d'ajouter ou de modifier des fonctionnalités.
- ✓ La performance : l'application devra être performante c'est-à-dire que le système
- ✓ doit réagir dans un délai minimal quel que soit l'action de l'utilisateur.
- ✓ L'ergonomie : l'application doit être intuitif.

5. La conception de microservice <ESCALE>:

5.1. diagramme de contexte:

Le diagramme de contexte met en évidence le champ d'application et les acteurs intervenant:

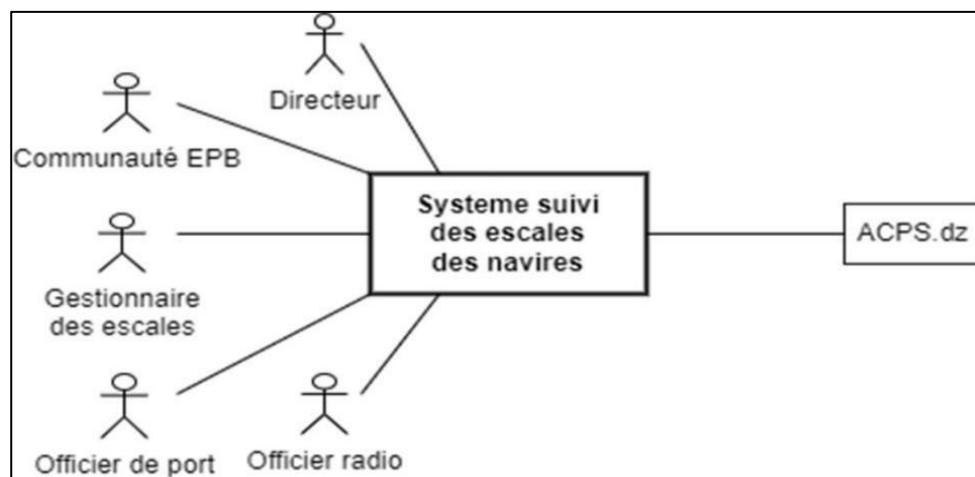


Figure 23 : Diagramme de contexte

5.2. Diagramme de cas d'utilisation:

Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système logiciel. Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (Humain ou Machine) et un système.

Il est une entité significative de travail .Dans un diagramme de cas d'utilisation il existe des acteurs qui interagissent avec des cas d'utilisation (use case) UC [29]

- **Cas d'utilisation** : Un cas d'utilisation est un ensemble d'actions réalisées par le système en réponse à une action d'un acteur.
- **Relation « include »** : Cas d'utilisation source incorpore explicitement et de manière obligatoire le comportement décrit dans le cas d'utilisation destination.
- **Relation « extend »** : Le cas d'utilisation source incorpore implicitement et de manière facultative un autre cas d'utilisation à l'endroit spécifié. Ceci veut dire que le cas d'utilisation source « étend » (en anglais « extend ») ou précise le cas d'utilisation destination.

5.2.1. Diagramme de cas d'utilisation de microservice <ESCALE>:

Le microservice escale doit gérer les prévisions des navires et le suivi des escales donc nous allons présenter les diagrammes de cas d'utilisation pour les prévisions ainsi que pour le suivi d'escales.

5.2.1.1. Cas d'utilisation associés à Communauté EPS:

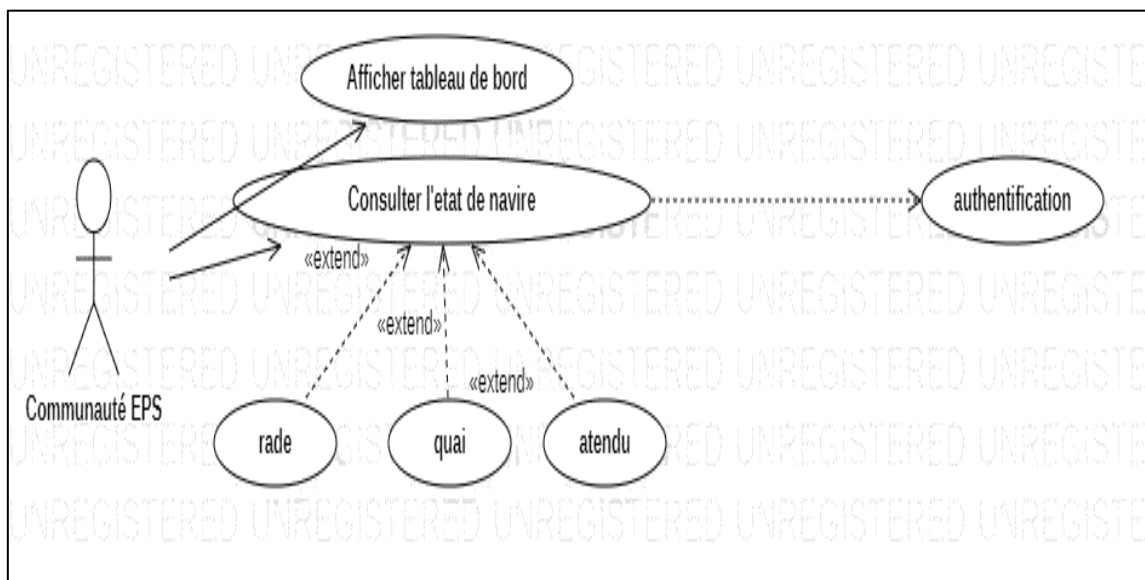


Figure 24 : Cas d'utilisation associés à Communauté EPS

Ce diagramme illustre les cas d'utilisation associés à communauté EPS et cela après l'authentification.

5.2.1.2. Cas d'utilisation associé au directeur capitainerie:

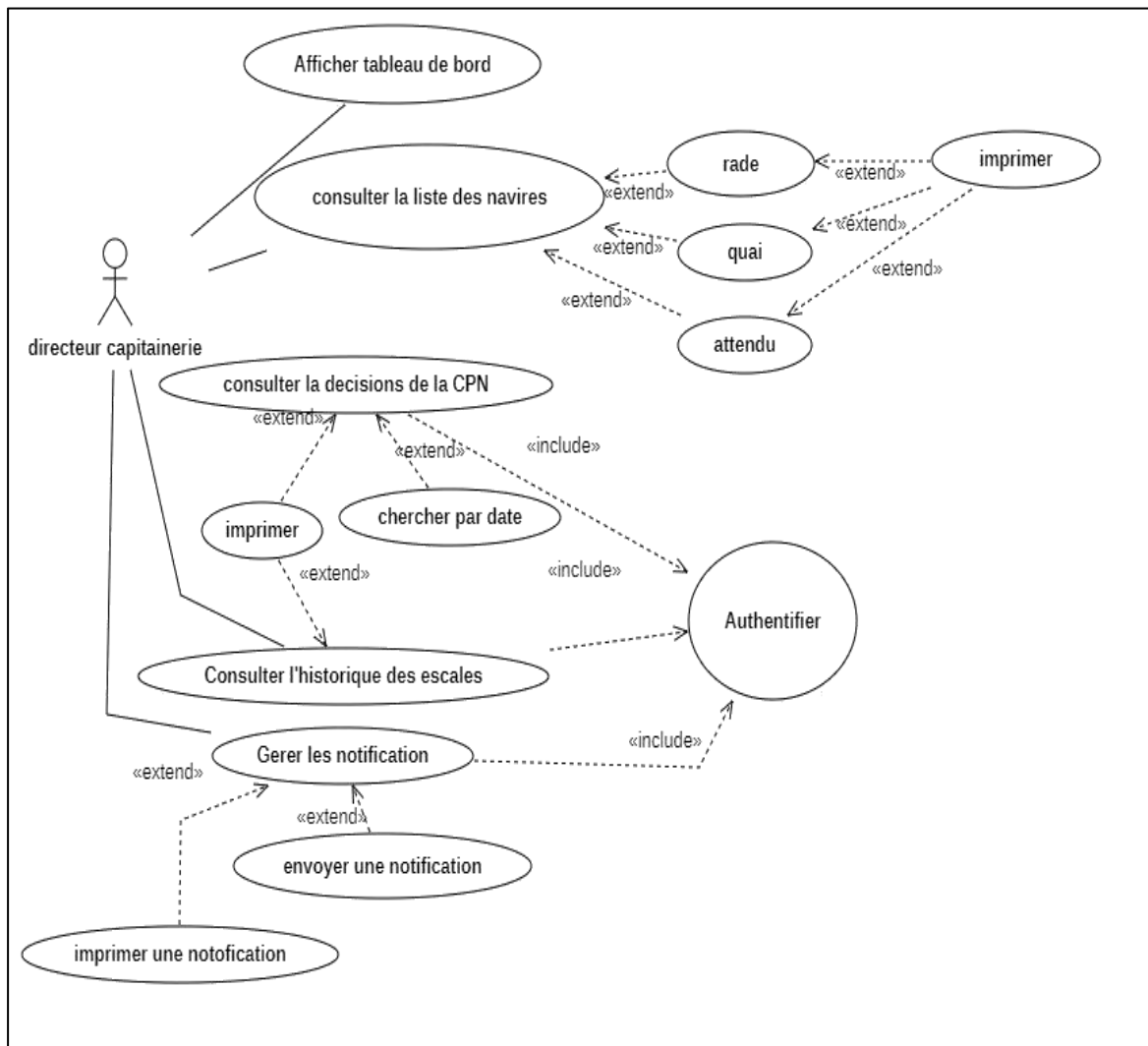


Figure 25 : Cas d'utilisation associés au directeur capitainerie

Ce diagramme illustre les cas d'utilisation associés au directeur et cela après l'authentification.

5.2.1.3. Cas d'utilisation associés au Gestionnaire des escales:

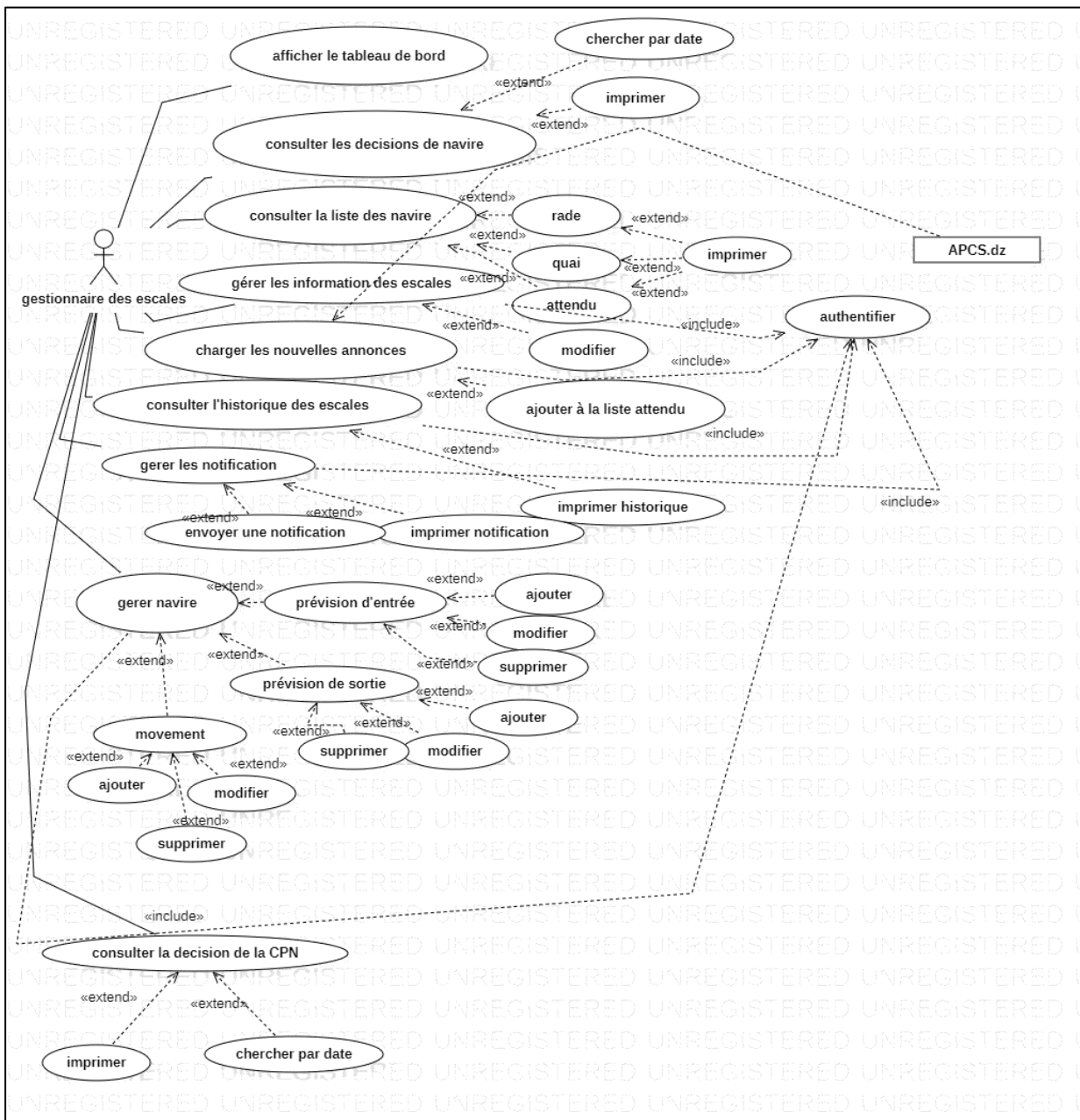


Figure 26 : Cas d'utilisation associés au Gestionnaire des escales

5.2.1.4. Cas d'utilisation associés à l'officier de port :

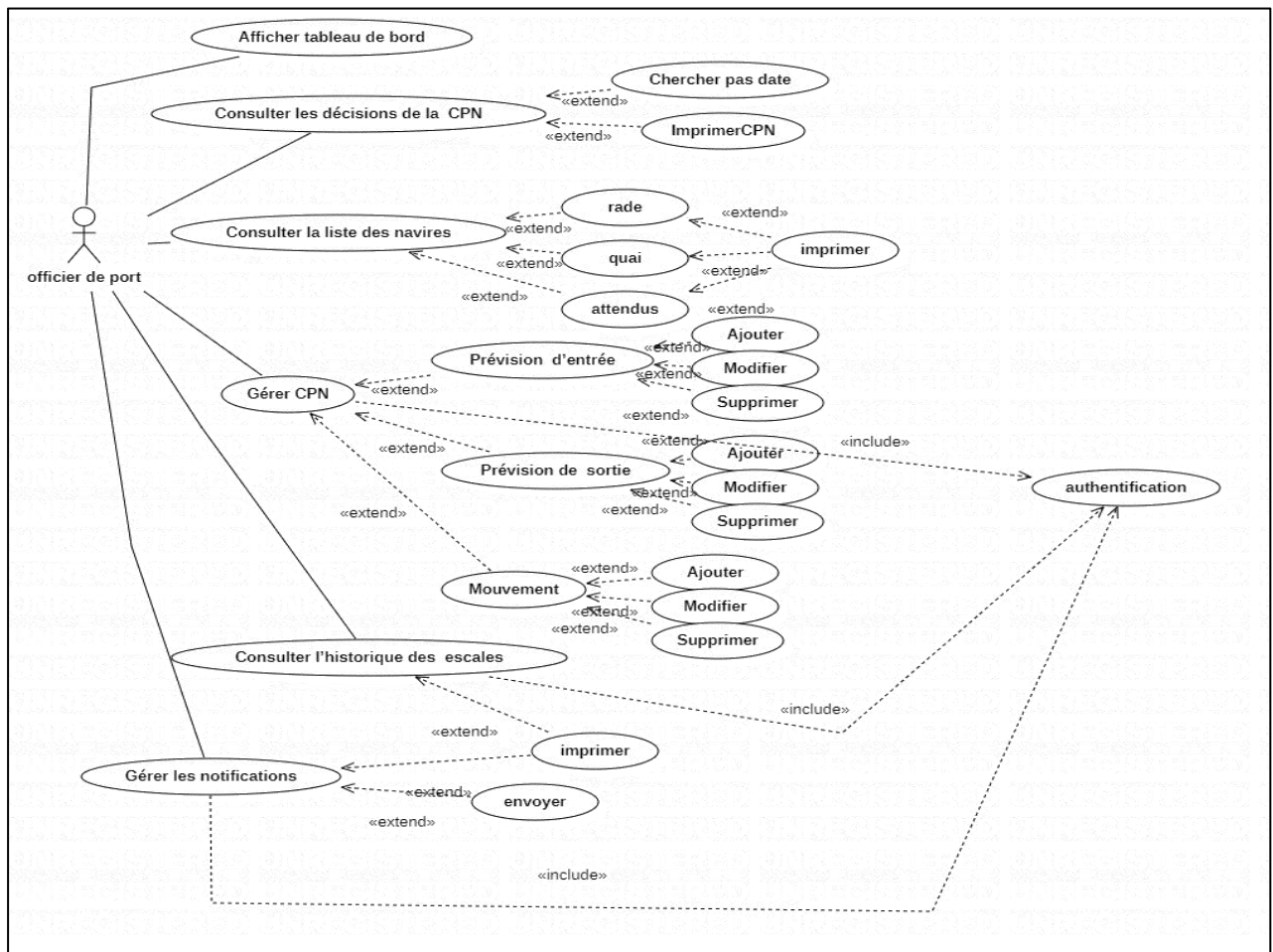


Figure 27 : Cas d'utilisation associés à l'officier de port

5.2.1.5. Cas d'utilisation associés à l'officier radio :

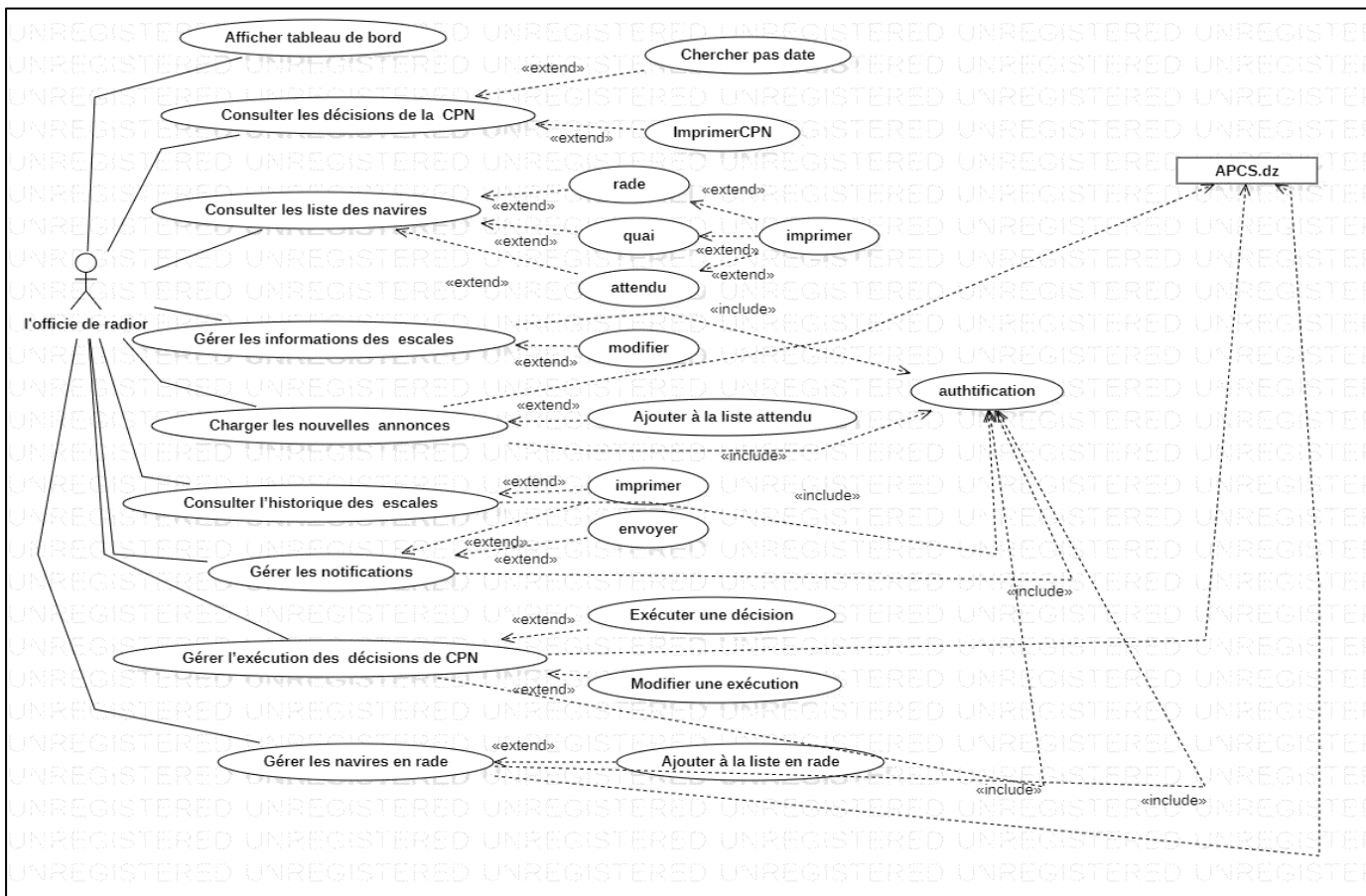


Figure 28 : Cas d'utilisation associés à l'officier radio

5.2.2. Description textuelle des cas d'utilisation:

Une autre définition de cas d'utilisation est ensemble de scénarios de réussite ou d'échec qui décrivent comment un système est utilisé par un acteur particulier pour atteindre un objectif.

L'approche la plus évidente pour détailler la dynamique d'un cas d'utilisation est de lister toutes les interactions entre les acteurs et le système sous forme textuelle. Un cas d'utilisation doit avoir un début et une fin bien définis. Vous devez également ordonner la description pour améliorer la lisibilité et spécifier les variantes possibles.

5.2.2.1. Cas d'utilisation « Authentifier »:

Cas d'utilisation	Authentifier
Objectif	Permettre aux utilisateurs d'accéder à ses privilèges dans le système
Acteurs	membre
Précondition	L'utilisateur doit avoir un compte.
Scénario nominal	<ol style="list-style-type: none"> 1. Cliquer sur bouton de Login. 2. Le système affiche la fenêtre d'authentification. 3. L'utilisateur introduit ses données de connexion et valide. 4. Le système vérifier l'existence du compte. 5. Le système vérifier que le compte est actif. 6. Le système confirme l'authentification.
Scénario alternatif	<p>A1. Si le compte n'existe pas</p> <p>A2. Si le compte est inactif</p> <p>Dans les deux exceptions le système affiche un message d'erreur et reprend l'étape 2 du scénario nominal.</p>
Post-condition	L'utilisateur est authentifié et diriger vers le tableau de bord

Table 3 :Description textuelle du cas d'utilisation "Authentifier"

5.2.2.2.Cas d'utilisation « Chargement des nouvelles annonces »:

Cas d'utilisation	Chargement des nouvelles annonces
Objectif	Récupérer les annonces des navires de la plate-forme APCS vers l'application
Acteurs	Gestionnaire des escales, Officier de radio
Précondition	L'acteur s'est bien authentifié
Scénario nominal	<ol style="list-style-type: none"> 1. Cliquer sur le bouton Annonces. 2. Le système récupère les nouvelles annonces de APCS.(Nous n'avons pas l'accès à APCS.dz pour l'instant) 3. Le système récupère les annonces non validées de sa BDD. 4. Le système affiche les annonces.
Scénario alternatif	<p>A1. Récupération de APCS échouée</p> <ol style="list-style-type: none"> 1. Rediriger vers l'étape 3 du scénario nominal. <p>A2. Récupération de la BDD échouée</p> <ol style="list-style-type: none"> 1. Afficher un message d'erreur.
Post-condition	Récupération des annonces

Table 4 :Description textuelle du cas d'utilisation "Chargement des nouvelles annonces"

5.2.2.3. Cas d'utilisation « Ajouter à la liste attendu »:

Cas d'utilisation	Ajouter à la liste attendu
Objectif	Valider l'annonce de navire et l'ajouter à la liste attendu
Acteurs	Gestionnaire des escales, Officier de radio
Précondition	L'acteur s'est bien authentifié et une annonce est affichée dans l'interface d'Annonces
Scénario nominal	<ol style="list-style-type: none"> 1. Cliquer sur le bouton Valider sur une annonce afficher. 2. Le système affiche la fenêtre de validation. 3. L'utilisateur clique sur valider. 4. Le système ajoute le navire à la liste attendu. 5. Le système envoi l'état (attendu) du navire à APCS.dz.(Nous n'avons pas l'accès à APCS.dz pour l'instant) 6. Le système ré affiche l'interface des annonces.
Scénario alternatif	<p>A1. Ajout à la liste attendu échouée</p> <ol style="list-style-type: none"> 1. Rediriger vers l'étape 2 du scénario nominal. 2. Afficher un message d'erreur.
Post-condition	Navire ajouter à la liste Attendu

Table 5 :Description textuelle du cas d'utilisation "Ajouter à la liste attendu"

5.2.2.4. Cas d'utilisation « Valider des navires en Rade » :

Cas d'utilisation	Valider des navires en Rade
Objectif	L'ajout de navire à la liste en Rade
Acteurs	Officier de radio

Précondition	L'acteur s'est bien authentifié
Scénario nominal	<ol style="list-style-type: none"> 1. Sélectionner le bouton Valider en Rade. 2. Le système récupère les navires dans la liste Attendu. 3. Le système affiche les navires dans l'interface de validation. 4. Cliquer sur valider sur un navire affiché. 5. Le système affiche la fenêtre de validation. 6. Saisir les données et valider. 7. Le système ajoute le navire à la liste en Rade. 8. Le système envoi l'état (en rade) du navire à APCS.dz.(Nous n'avons pas l'accès à APCS.dz pour l'instant) 9. Le système ré affiche l'interface de validation.
Scénario alternatif	<p>A1. Récupération des navires dans la liste Attendu échouée</p> <ol style="list-style-type: none"> 1. Afficher message d'erreur. <p>A2. Ajout à la liste en Rade échouée</p> <ol style="list-style-type: none"> 1. Afficher un message d'erreur. 2. Rediriger vers l'étape 2 du scénario nominal.
Post-condition	Navire ajouter à la liste en Rade

Table 6 :Description textuelle du cas d'utilisation "Valider des navires en Rade"

5.2.2.5. Cas d'utilisation « Ajouter - Prévision d'entrée »:

Cas d'utilisation	Ajouter - Prévision d'entrée
Objectif	L'ajout d'une prévision d'entrée d'un navire
Acteurs	Officier du port

Précondition	L'acteur s'est bien authentifié
Scénario nominal	<ol style="list-style-type: none"> 1. Cliquer sur le bouton Ajouter dans la section Prévision d'entrée. 2. Le système affiche la fenêtre d'ajout. 3. Saisir les données et valider. 4. Le système ajoute la prévision à la BDD. 5. Le système ré affiche l'interface de Gérer CPN.
Scénario alternatif	<p>A1. L'ajout de prévision a échoué</p> <p>Afficher un message d'erreur.</p> <ol style="list-style-type: none"> 1. Rediriger vers l'étape 5 du scénario nominal.
Post-condition	Prévision ajouter avec succès

Table 7:Description textuelle du cas d'utilisation "Ajouter - Prévision d'entrée"

5.2.2.6. Cas d'utilisation « Modifier - Prévision d'entrée »:

Cas d'utilisation	Modifier - Prévision d'entrée
Objectif	Modification d'une prévision d'entrée du navire
Acteurs	Officier du port
Précondition	L'acteur s'est bien authentifié et une décision non exécuter est affichée dans l'interface Gérer CPN
Scénario nominal	<ol style="list-style-type: none"> 1. Cliquer sur le bouton Modifier sur une prévision afficher. 2. Le système affiche fenêtre de modification. 3. Modifier les données et valider. 4. Le système modifie la prévision dans la BDD. 5. Le système ré affiche l'interface Gérer CPN.

Scénario alternatif	A1. Modification de prévision échouée 1. Afficher un message d'erreur. 2. Rediriger vers l'étape 5 du scénario nominal.
Post-condition	Prévision modifier avec succès

Table 8:Description textuelle du cas d'utilisation "Modifier - Prévision d'entrée"

5.2.2.7. Cas d'utilisation « Exécuter une décision »:

Cas d'utilisation	Exécuter une décision
Objectif	Ajouter le navire à la liste des navires en quai (Entrée), ou le mettre en état Finie (Sortie) ou changer son poste en quai (mouvement).
Acteurs	Officier de radio
Précondition	L'acteur s'est bien authentifié et une décision est affichée dans l'interface d'Exécutions
Scénario nominal	<ol style="list-style-type: none"> 1. Cliquer sur le bouton Valider sur une décision affichée. 2. Le système affiche fenêtre de validation. 3. Remplir les données et valider. 4. Cliquer sur valider. 5. Le système modifie l'état du navire <ul style="list-style-type: none"> - Si c'est une entrée : Ajouter à la liste des navires en quai. - Si c'est une sortie : Mettre en état Finie. - Si c'est un mouvement : Changer son poste. 6. Le système envoie l'état ou le poste change du navire à APCS.dz.(Nous n'avons pas l'accès à APCS.dz pour

	l'instant) 7. Le système ré affiche l'interface d'exécutions.
Scénario alternatif	A1. Modification de l'état du navire a échouée 1. Afficher un message d'erreur. 2. Rediriger vers l'étape 7 du scénario nominal.
Post-condition	Décision valider avec succès

Table 9:Description textuelle du cas d'utilisation "Exécuter une décision"

5.3. Diagramme de séquence :

Le diagramme de séquence suit le diagramme de cas d'utilisation car ils sont complémentaires. Il permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets. En particulier, il montre également les objets participant aux interactions via les lignes de vie et les messages qu'ils échangent [30].

5.3.1. Diagramme de séquence du cas d'utilisation « Authentifier » :

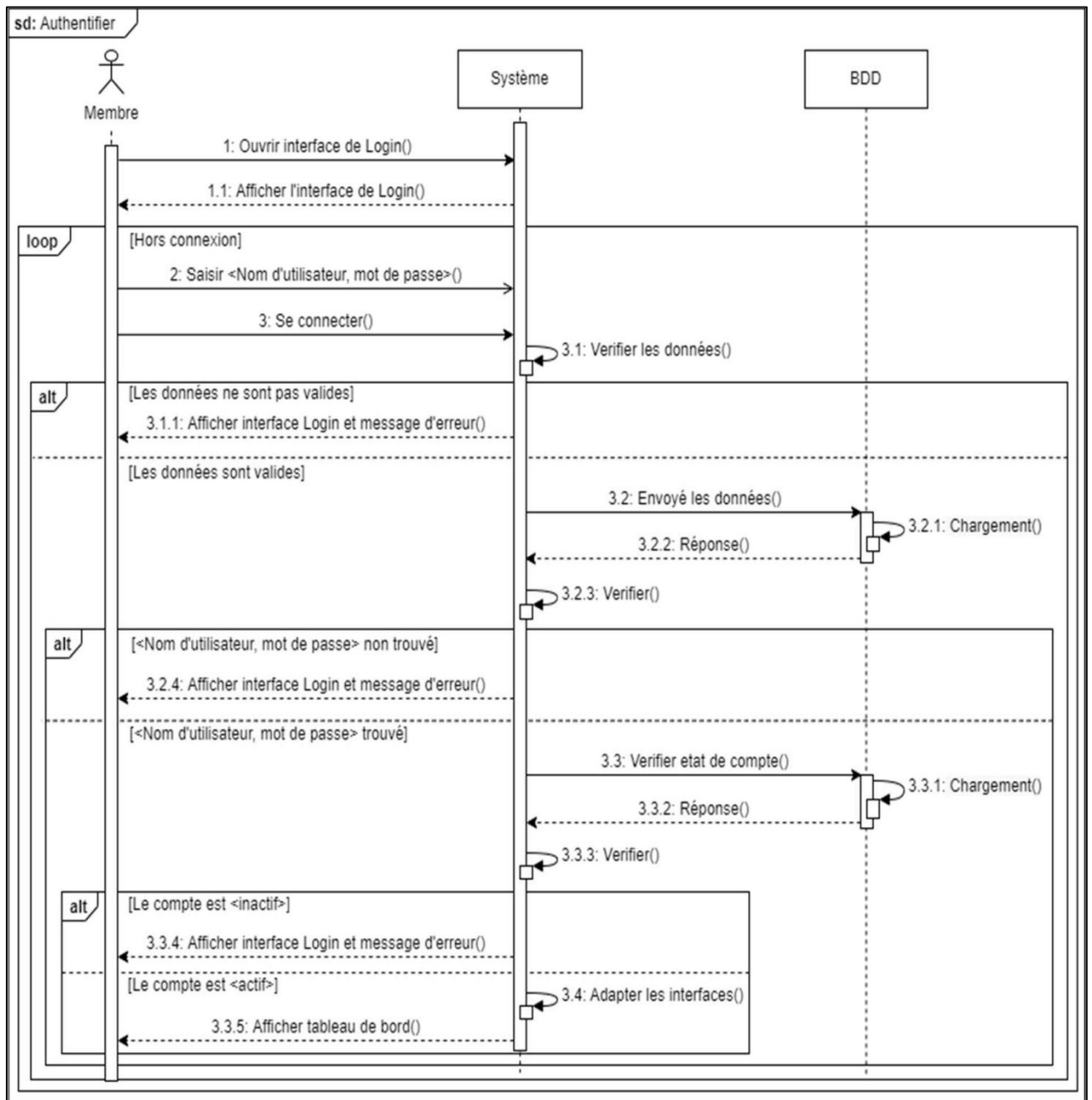


Figure 29 : Diagramme de séquence du cas d'utilisation « Authentifier »

5.3.2. Diagramme de séquence du cas « Charger les nouvelles annonces - Ajouter à la liste attendu »:

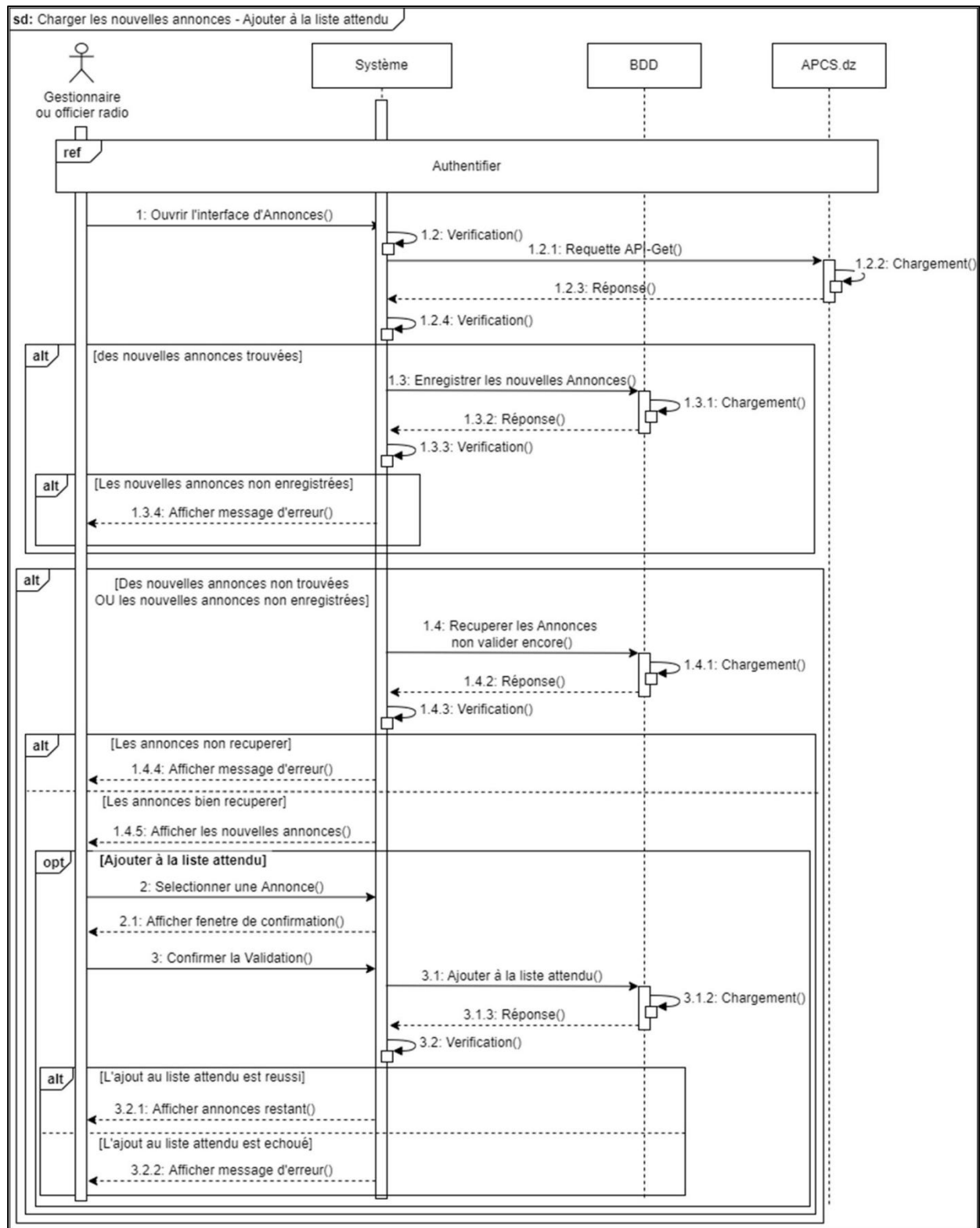


Figure 30 : Diagramme de séquence du cas « Charger les nouvelles annonces - Ajouter à la liste attendu »

5.3.3. Diagramme de séquence du cas « Gérer les navires en rade - Ajouter à la liste en rade » :

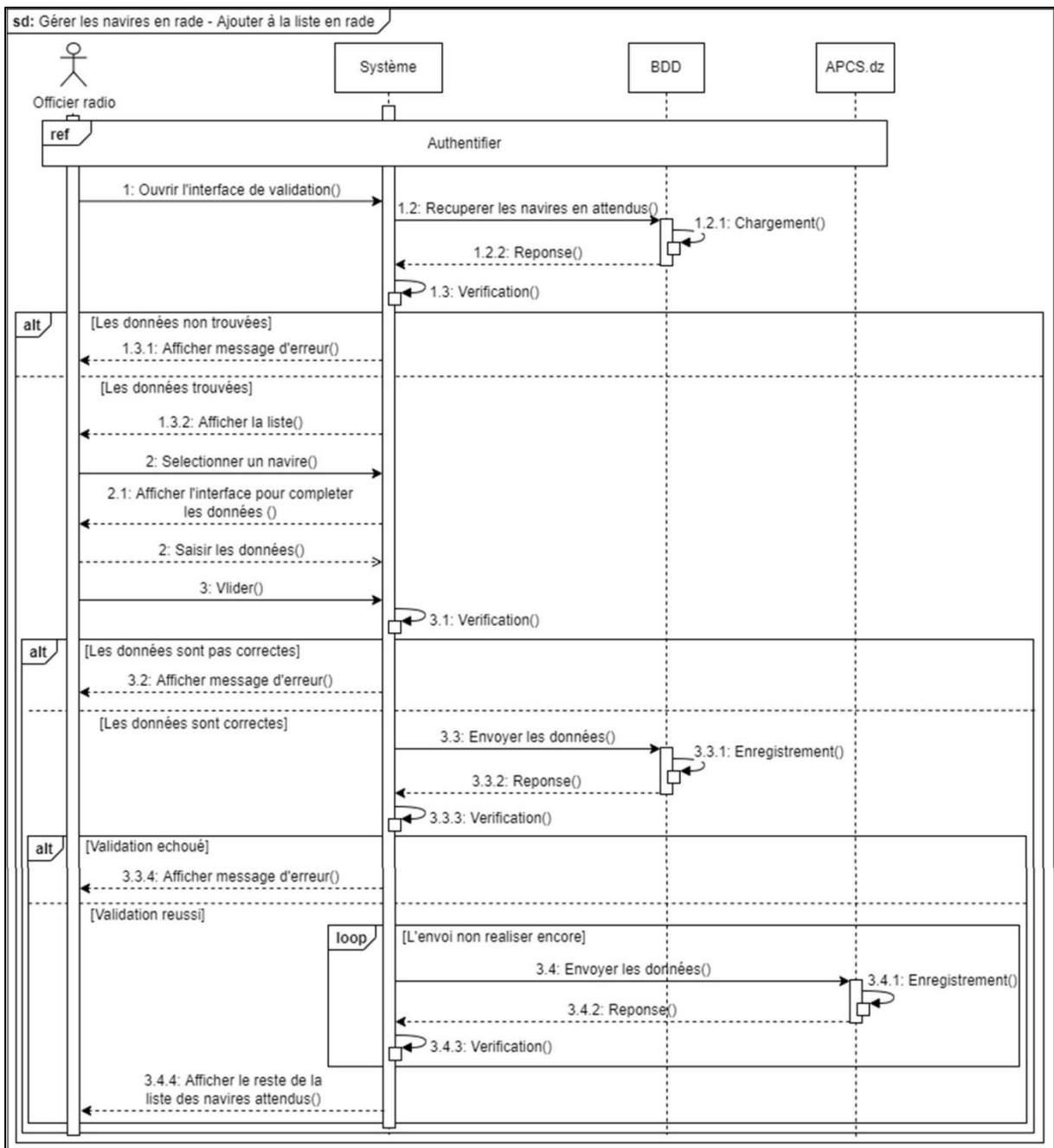


Figure 31 : Diagramme de séquence du cas « Gérer les navires en rade - Ajouter à la liste en rade »

5.3.4. Diagramme de séquence du cas « Prévission d'entrée »:

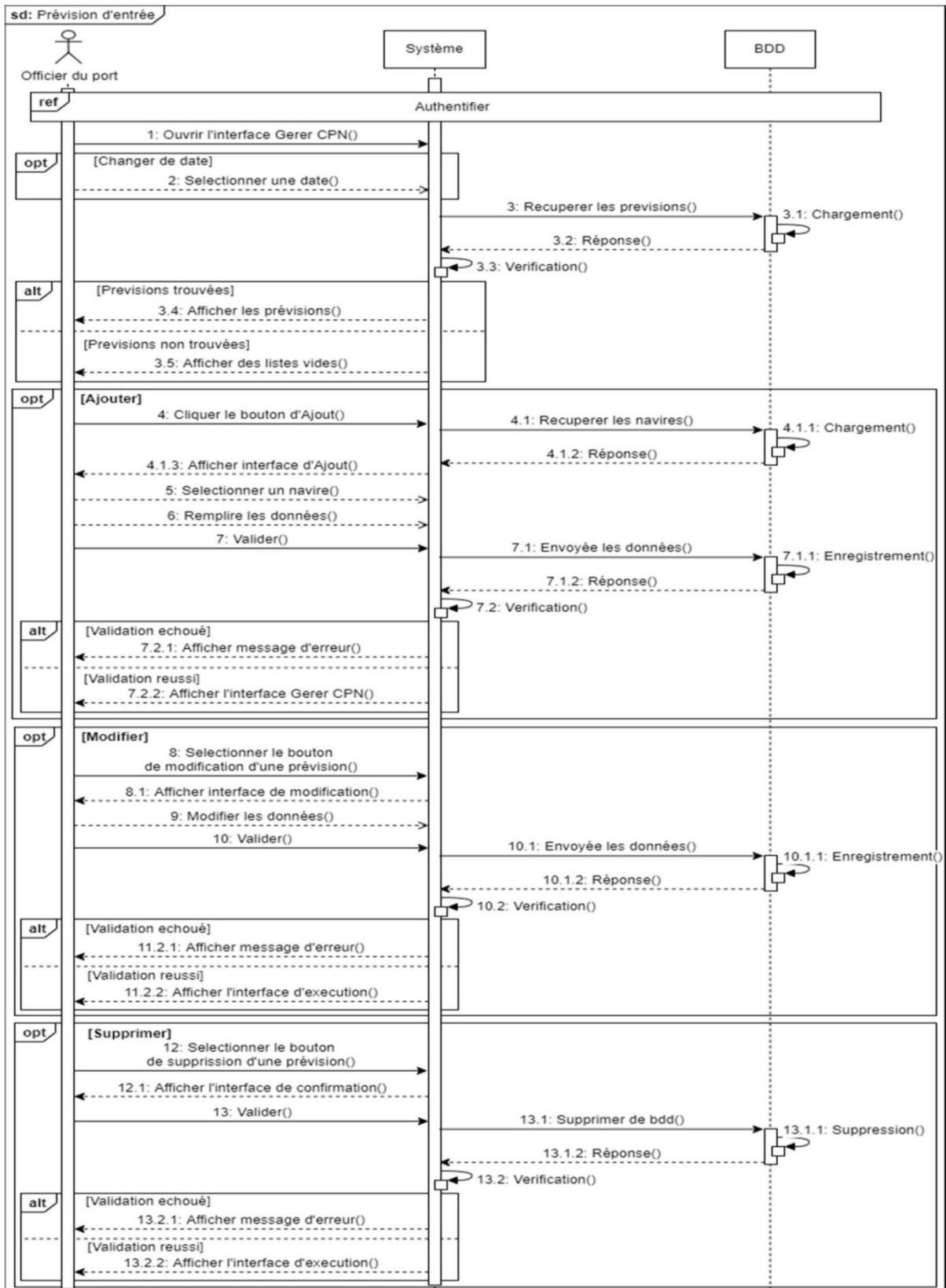


Figure 32 : Diagramme de séquence du cas « Prévission d'entrée »

5.3.5. Diagramme de séquence du cas « Exécuter une décision » :

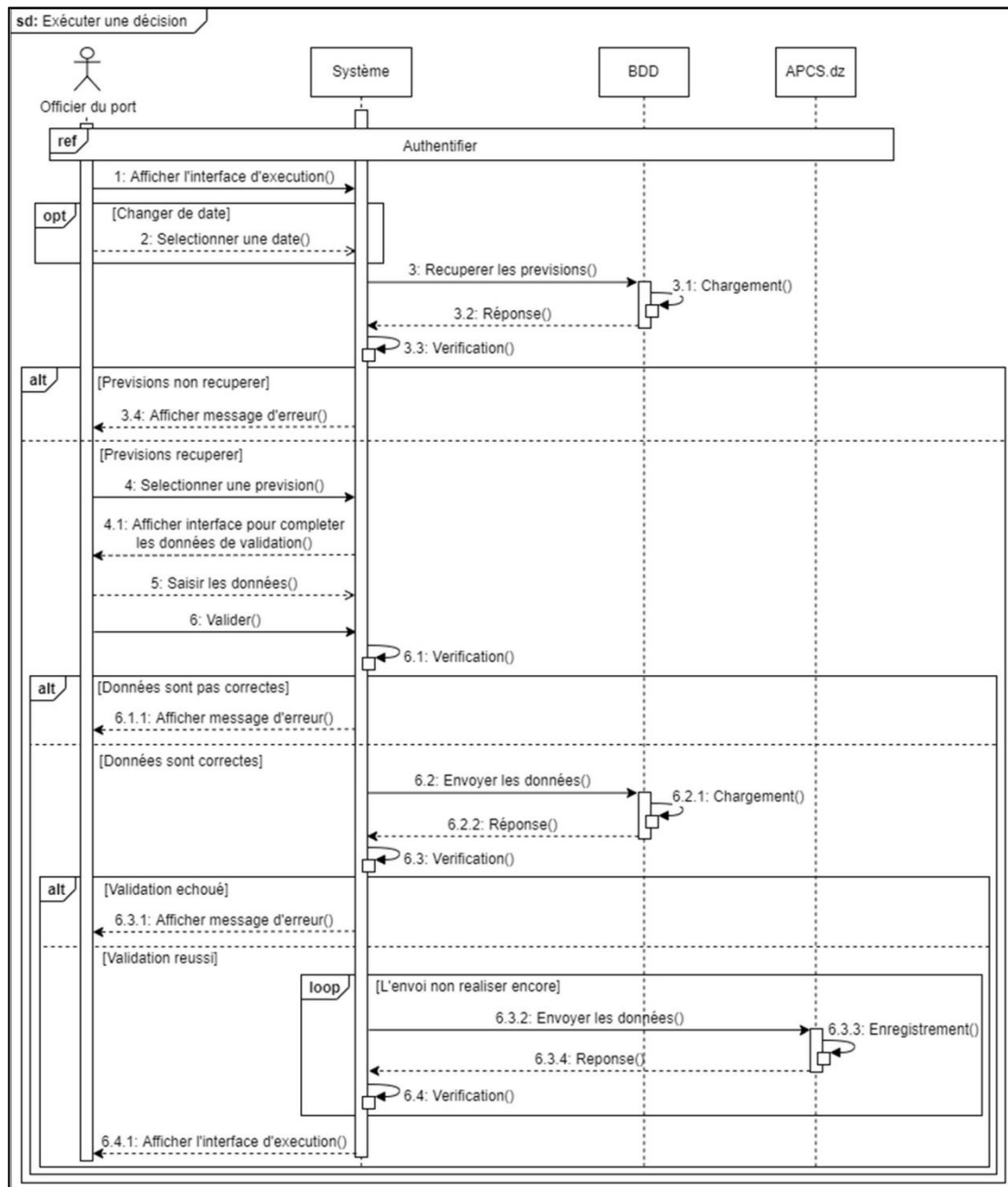


Figure 33 : Diagramme de séquence du cas « Exécuter une décision »

5.4. Diagramme de classe:

Les diagrammes de classes expriment de manière générale la structure statique d'un système, en termes de classes et de relations entre ces dernières. La figure 34 , illustre le diagramme de classe de notre application.

	adresse	Varchar	Adresse de l'utilisateur	-consulte_ historique_ navire()
	email	Varchar	Email de l'utilisateur	
	num	Varchar	Numéro de téléphone de l'utilisateur	
	etat	Boolean	Etat de l'utilisateur	
	role	Varchar	Rôle de l'utilisateur	
Directeur Capitainerie	/	/	/	
Gestionnaire	/	/	/	- valider_ annonce() -modifier_ listes
Officier_port	/	/	/	- crée_cpn() -modifier_ cpn() -supprimer_ cpn()
Officier_radio	/	/	/	- valider_ executions() - modifier_ execution()
Historique_	idhistorique	Int	Identifiant de l'historie	
	realiserle	DateTime	Date et heure de tache	

utilisateur	tache	Text	Détaille sur la tache	
Cpn	idCPN	Int	Identifiant de la CPN	
	journee	Date	Jour de création de la prévision	
	heure	Time	Heure de création de la prévision	
	typeDecision	Varchar	Type de prévision	
	dateExecution	Date	Date de réalisation de la prévision	
	heureExecution	Time	Heure de réalisation de la prévision	
Sorties	details	Varchar	Consignes	/
Entrees	details	Varchar	Instructions à suivre	
	auPoste	Int	Numéro du quai	
Movements	auPoste	Int	Numéro du quai	
	numEscale	Varchar	Code d'escale	
	Type	tinyint	Type d'un escale	
	Position_Est	Varchar	Position	
	Nb_Passagerse	Int	Nombre de Passagerse	
	Situation	tinyint	Situation d'un navire (Rade, Attendu, Quai)	
	Navire_Affret	bit	La location d'un navire selon un contrat	
	Port_Provence	Varchar	Le port de provenance	
	Port_Destination	Varchar	Le port de destination	

Escalaes	Port_Dernier	Varchar	Le port d'escale
	Code_Produit	Varchar	Code de produit
	Tonnage	real	c'est la capacité cubique intérieure
	Code_Consignataire	Varchar	Code de consignataire
	Notes	Varchar	Notes
	Datep_Arrivee	smalldatetime	Date de l'arrivee d'un navire
	Heurep_Arrivee	Varchar	Hrure de l'arrivee d'un navire
	Dateheure_Pconference	smalldatetime	Date & heure de pconference
	Dateep_Depart	smalldatetime	Date de départ de navire
	Heurep_Depart	Varchar	Heure de départ de navire
	Produit_Danger	bit	Les produit danger
	Type_Marchandise	tinvente	Type de marchandise
	Code_Armateur	Varchar	Code armateur
	Dateheure_Rade	datetime	Date & heure de navire en Rade
Dateheure_Entree	datetime	Date & heure de navire Entree	

	Dateheure_Sortie	datetime	Date & heure de navire Sortie
	Tonnage_Embarque	float	Tonnage Embarque
	Classe_Statistique	Int	Classe Statistique
	Tonnage_Embarque_Mois_Suivant	float	Tonnage Embarque Mois Suivant
	Tonnage_Débarque_Mois_Suivant	float	Tonnage Débarque Mois Suivant
	Escale_Principale	varchar	Escale Principale
	Dateheure_Entree_Passe	smalldatetime	Date & heure de navire Entree Passe
	ATP	Varchar	
	Etablissement	datetime	Etablissement
Navire	Code_Navire	Int	CODENAVIRE
	Designation	Varchar	Désignation
	Code_Type	Varchar	Type de navire
	Code_Pavillon	Varchar	Code de Pays de navire
	TJB	Varchar	tonneaux de jauge brute (désigne le volume des espaces fermés du navire à l'exception du double fond)
	TJN	Varchar	tonnage net qui désigne le volume de la capacité

			commerciale du navire	
	TPL	Varchar	désigne la capacité de chargement d'un navire, la charge à bord qu'il peut transporter, incluant équipage, soute, vivres, etc	
	imo_navre	Varchar	Numéro IMO de navire	
	largeur_navire	Float	La largeur de navire	
	longueur_navire	Float	La longueur de navire	
	pavillon	Varchar	Pays de navire	
	eta	Datetime	Date et heure estimer de l'arrivée denavire	
	etd	Datetime	Date et heure estimer de sortie denavire	
	Tirant_Eau_Avant	Float	La hauteur de la partie avant immergée du navire	
	Tirant_Eau_Arriéré	Float	La hauteur de la partie arrière immergée du navire	
	Tirant_Eau_Ete	Float	La hauteur de la partie immergée du navire	
	Nb_Cales	Int	Nombre de Cales	
	Nb_Gruess	Int	Nombre de Gruess	
	Nb_Mats	Int	Nombre de Mats	
	Code	Varchar	Code port	

Port	designation	Varchar	Designation	
Poste	Code-Poste	Varchar	Code Poste	
	Designation	Varchar	Designation	
	Code-Specialite	Varchar	Code-Specialite	
	Tirant-eau	Varchar	Tirant eau	
	X1	Varchar	Longueur de poste	
	Y1	Varchar	Largeur de poste	
	X2	Varchar	Longueur de poste	
	Y2	Varchar	Largeur de poste	
	Lieu	Varchar	Lieu de poste	
Consignataire	Code- Consignataire	Varchar	Code de consignataire	
	Nom- Consignataire	Varchar	Nom de consignataire	
	Libellé- Consignataire	Varchar	Libellé de Consignataire	
	Adresse	Varchar	Adresse de Consignataire	
	Tlf	Varchar	Num de tlf de Consignataire	
	E-mail	Varchar	Email de Consignataire	
	Code-Cargaison	Varchar	Code marchandise	

Cargaison	Libellé	Varchar	Libellé marchandise
	Mouvement	Varchar	Mouvement marchandise
	Type	Varchar	Type marchandise
	Libellé-Type	Varchar	Libellé type marchandise
	Tonnage	Varchar	Tonnage marchandise
	Num-Colis	Varchar	Nombre de colis
	Num-Escale	Varchar	Numéro escale
Type-Navire	Code-Type-Navire	Varchar	Code Type Navire
	Designation	Varchar	Designation
	Categorie	Varchar	Categorie
Avoir-Situation	Code-Avoir-Situation	Varchar	Code de Avoir situation
	Heure-Situation	date	Heure situation
	Date-Situation	date	Date-Situation
	Observation	Varchar	Observation
	Code-Navire	Varchar	Code de Navire
Reacostage	Code-Reacostage	Varchar	Code Reacostage
	Code-Escale	Varchar	Code Escale
	Date-Reacostage	date	Date de Reacostage

	Heure-Reacostage	date	Heure de Reacostage	
Opération	Code-Opération	Varchar	Numéro opération	
	Code-Type-Navire	Varchar	Code Type Navire	
	Code-Escale	Varchar	Code Escale	
Prestation	Code-Prestation	Varchar	Code Prestation	

Table 10 : Présentation des classes et leurs attributs.

5.5. Présentation d'un modèle relationnel:

En appliquant les règles de passages au diagramme de classes [13], nous obtenons le schéma relationnel suivant :

- **Membres:** (**Code mem**, username, password, nom, prenom, dateNaissance, lieuNaissance, adresse, email, num, etat, role).
- **Historique_utilisateur:** (**idhistorique**, realiserle, tache).
- **CPN:** (**idCPN**, journee, heure, typeDecision, dateExecution, heureExecution).
- **Escale:** (**numEscale**, **#Code Consignataire**, **#Code Navire** Type, Position_Est, Nb_Passagerse, Situation, Navire_Affret, Port_Provence, Port_Destination, Port_Dernier, Code_Produit, Tonnage, Notes, Datep_Arrivee, Heurep_Arrivee, Dateheure_Pconference, Dateep_Depart, Heurep_Depart, Produit_Danger, Type_Marchandise, Code_Armateur, Dateheure_Rade, Dateheure_Entree, Dateheure_Sortie, Tonnage_Embarque, Classe_Statistique, Tonnage_Embarque_Mois_Suivant, Tonnage_Débarque_Mois_Suivant, Escale_Principale, Dateheure_Entree_Passe, ATP, Etablissement).
- **Navire:** (**Code Navire**, **#Code Type**, **#Code Pavillon**, Designation, TJB, TJN, TPL, imo_navre, largeur_navire, longueur_navire, pavillon, eta, etd, Tirant_Eau_Avant, Tirant_Eau_Arrière, Tirant_Eau_Ete, Nb_Cales, Nb_Grues, Nb_Mats).
- **Port:** (**Code Port**, designation).

- **Poste:** (Code-Poste, Designation, Code-Specialite, Tirant-eau, X1, Y1, X2, Y2, Lieu).
- **Consignataire:** (Code-Consignataire, Nom-Consignataire, Libellé-Consignataire, Adresse, Tlf, E-mail).
- **Cargaison:** (Code-Cargaison, #Num-Escale, Libellé, Mouvement, Type, Libellé-Type, Tonnage, Num-Colis).
- **Type-Navire:** (Code-Type-Navire, Designation, Catégorie).
- **Avoir-Situation:** (Code-Avoir-Situation, #Code-Navire, Heure-Situation, Date-Situation, Observation).
- **Reacostage:** (Code-Reacostage, #Code-Escale, Date-Reacostage, Heure-Reacostage).
- **Opération:** (Code-Opération, #Code-Type-Navire, #Code-Escale).
- **Prestation :** (Code-Prestation).

5.6. Règles de gestion:

- Une escale peut avoir un à plusieurs navires.
- Un navire concerne une et une seule escale.
- Une escale peut être planifié par un a plusieurs consignataires.
- Un consignataire planifie une et une seule escale
- Une escale à un seul port de départ et un seul port d'arrivé.
- Un port départ ou d'arrivé est concerné par plusieurs escales.
- Un navire peut effectuer une et une seule opération
- Une opération concerne un et un seul navire.
- Un navire peut avoir une et une seule situation.
- Une situation concerne un et un seul navire.
- Une opération utilise un et un seul post.
- Un post est utilisé pour une ou plusieurs opérations.

6. Conclusion :

Ce chapitre nous a permis de décrire d'une manière globale et détaillée le fonctionnement du système afin d'en faciliter sa réalisation. Nous avons présenté les diagrammes de séquences, terminer par l'élaboration du diagramme de classe et le passage au modèle relationnel lui correspondant.

Nous entamons la partie implémentation qui fera l'objet du chapitre suivant.

1. Introduction :

Dans ce chapitre, nous présenterons la réalisation de notre application en exploitant la spécification et la conception décrite dans le chapitre précédent.

Nous présentons d'abord, l'environnement de travail et les différents outils et langages de programmation qui ont été utilisés pour la réalisation de notre système. Après nous décrivons les étapes de développement ainsi quelques maquettes des interfaces du module de gestion des escales avec explications.

2. Les langages et environnements de développement :

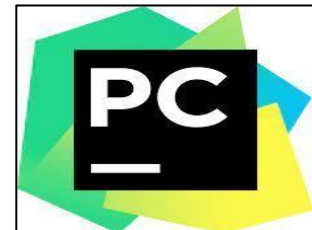
Dans cette partie nous allons présenter les outils utilisés pour réaliser ce projet, la programmation sur Odoo est une combinaison entre Python et XML.

- **Système d'exploitation Windows 10 :**

Windows 10 est un système d'exploitation de la famille Windows NT développé par la société américaine Microsoft. Officiellement présenté le 30 Septembre 2014. Il est disponible publiquement depuis le 29 Juillet 2015.[30]

- **PyCharm:**

Est un environnement de développement intégré (EDI) utilisé pour programmer en Python. Il permet l'analyse de code et contient un débogueur graphique. Il permet également la gestion des tests unitaires, l'intégration de logiciel de gestion de versions.[31]



- **Git/GitHub:**

Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2[32]. Les objectifs que Git devait atteindre à sa conception étaient les suivants : être rapide, avoir un design simple, permettre le développement non-linéaire, être distribué et pouvoir gérer efficacement de gros projets.[33]



- **Python :**

Python est un langage portable, dynamique, extensible, gratuit, qui permet une approche modulaire et orientée objet de la programmation. Python est développé depuis 1989 par Guido van Rossum et de nombreux contributeurs bénévoles.[34]



- **PostgreSQL :**

PostgreSQL est un système de gestion de base de données relationnelle et objet (SGBDRO) [6]. C'est un outil libre disponible selon les termes d'une licence du type BSD. Ce système est concurrent d'autres systèmes de gestion de base de données , qu'ils soient libres (comme MariaDB, MySQL et Firebird), ou propriétaires (comme Oracle, Sybase, DB2, Informix et Microsoft SQL Server). Comme les projets libres Apache et Linux, PostgreSQL n'est pas contrôlé par une seule entreprise, il est fondé sur une communauté mondiale de développeurs et d'entreprises. [35]



- **Langage XML:**

XML (eXtensible Markup Language) est un langage simple et puissant de description et d'échange de documents structurés de n'importe quel domaine de données grâce à son extensibilité. Il décrit cette structure à l'aide d'un système de balises .[36]



- **ODOO: [37]**

Le principe de base d'odoo est d'utiliser une structure modulaire, et indépendante qui permet à la fois d'améliorer régulièrement les modules existants, et d'autre part, faciliter les modifications ou de suppressions des modules dont nous n'avons pas besoin , sans avoir à toucher tout le système.

Par défaut, il contient plusieurs modules prêts à être déployés tel qu'ils sont, ou à les personnaliser selon nos besoins. Les modules installés sur odoo sont inclus dans un répertoire nommé / addons , et il est possible d'y créer nos propres modules en utilisant un nouveau dossier personnalisé.

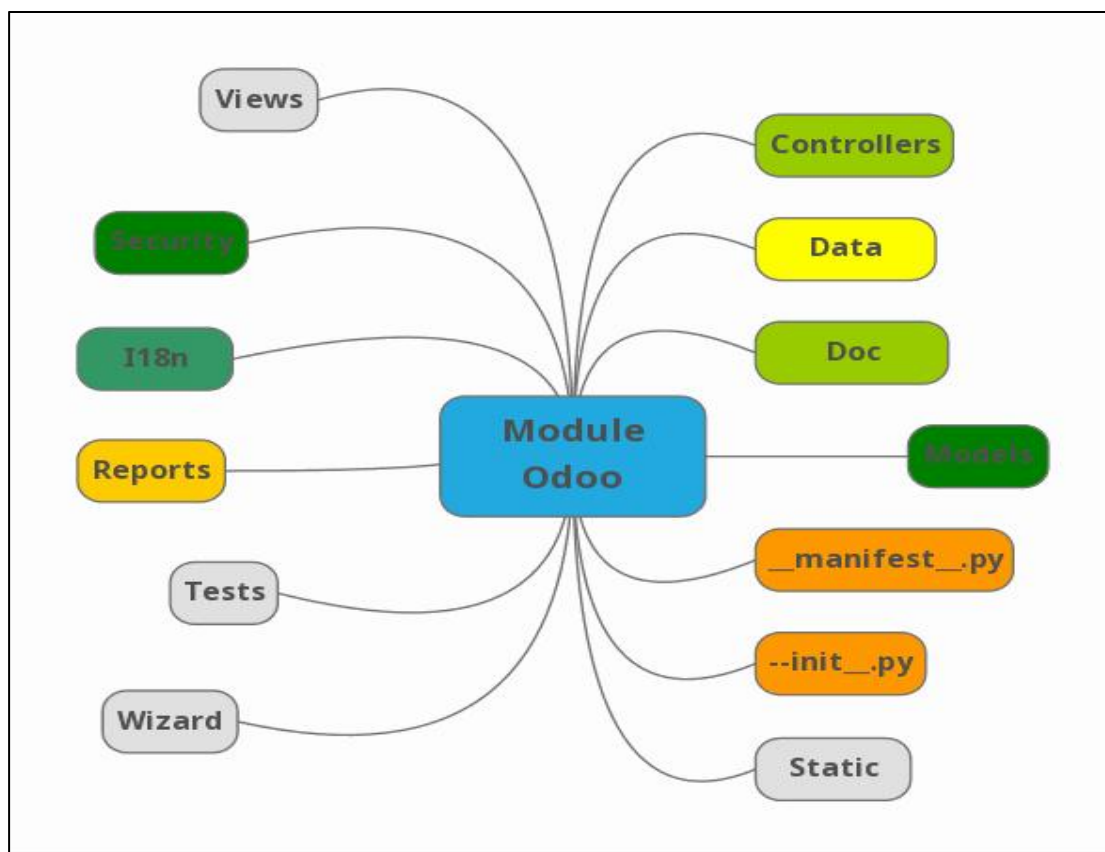


Figure 35 : Structure d'un Module odoo

Un Module sous odoo est représenté comme un répertoire, contenant des sous-répertoires et des fichiers avec une convention bien définie.

Pour pouvoir créer et développer ses propres modules odoo, il est nécessaire de comprendre la signification des fichiers et répertoire d'un module odoo, composant l'architecture technique d'un module :

a) `__init__.py` : (nécessaire pour la création d'un module) C'est le fichier python d'initialisation du module contenant tous les autres fichiers python à importer .

b) `__manifest__.py` : (nécessaire pour la création d'un module) C'est le fichier de manifestation du module , il contient toutes les informations concernant le module comme le nom, une description, la version, l'auteur, le site web, les fichiers de données, démos, sécurité ,...

c) `models` : (nécessaire pour la création d'un module) C'est un répertoire qui contient tous les fichiers pythons du module, ce répertoire doit être déclaré dans le fichier d'initialisation `__init__.py`

d) views : C'est le répertoire qui contient les fichiers xml décrivant la couche présentation ou interface, tel que les vues liste (tree), formulaires (form), graph, pivot, action, menu, rechercher (search),....

e) data: C'est un répertoire qui contient les données prédéfinies nécessaire lors de l'installation du module.

f) démo: Ce répertoire contient les données de démonstrations, qui vont être chargées lors de l'installation du module.

g) static : Ce répertoire renferme tous les fichiers relié au site web, tel que es fichiers img, js (javascript), css, font ,...

- **Src :**

- css:** tous les fichiers css de conception .

- img:** tous les fichiers images .

- js:** tous les fichiers javascript .

- xml:** contient tous les fichiers .xml utilisé pour les view/Qweb templates .

- **description:** Comporte un fichier html nommé "Index.html" qui permet une démonstration et présentation sur votre module , contenant des aperçus et des imprimés écrans des fonctionnalités de votre module .Il contient aussi le fichier icon.png qui sera utilisé comme l'icône de votre module .

h) wizard : C'est un répertoire qui englobe les classes de transitions nécessaire à l'affichage des assistants et des fenêtres qui servent à aider l'utilisateur à introduire les données nécessaires pour l'obtention des résultats souhaités (états de sorties, calcul, ...) Ces modèles seront automatiquement supprimés après utilisation.

i) Report : C'est le répertoire qui renfermerait les descriptions web et xml, des rapports de sortie (en PDF) de votre module .

j) security : Répertoire contenant les fichiers de base définissant les règles de sécurités, les droits d'accès, les groupes, etc.

- le fichier csv des droits d'accès (ir.model.access.csv) : pour assigner les droits de lecture et d'écriture, à un utilisateur d'un groupe.

- security.xml : la création des groupes et les autres règles d'accès aux enregistrements.

k) controller : Ce répertoire tient les fichiers nécessaires des requêtes http et leurs réponses à partir d'un navigateur internet, il concerne la partie développement web qui permet d'étendre ou personnaliser le module de base web site proposé par odoo.

l) i18n : Il comprend la traduction de votre module dans les différentes langues. Principalement, il contient le fichier Template de traduction (.pot) et le fichier .po qui contient la traduction actuelle du module

m) Doc: C'est un répertoire qui inclut la partie documentation concernant votre module les fonctionnalités, les explications et démos,..

n) Test: C'est un répertoire qui comprend les fichiers .py (python) de test.

3. les étapes de développement de module « gestion des escales » :

3.1. Installation Odoo16 sous Windows:

Il y a plusieurs façon d'installer Odoo : Installation online, par programmes d'installation, installation par la source , Docker.

Nous avons choisi l'installation par la source , cette façon offre une plus grande flexibilité : par ex. permet de faire tourner plusieurs versions d'Odoo sur le même système . Bon pour le développement de modules , peut être utilisé pour le déploiement en production .

Installation par la source :

L'installation par la source consiste en fait à ne pas installer Odoo et à l'exécuter directement à partir de la source à la place. Cela peut être plus pratique pour les développeurs de modules, car la source d'Odoo est plus facilement accessible qu'avec une installation de package.

Elle rend le démarrage et l'arrêt d'Odoo plus flexible et plus explicite que les services configurés par les installations de package et permet d'outrepasser les paramètres en utilisant les paramètres de la ligne de commande sans avoir besoin d'éditer un fichier de configuration.

Enfin, il offre un meilleur contrôle sur la configuration du système et permet de conserver (et d'exécuter) plus facilement plusieurs versions d'Odoo côte à côte.

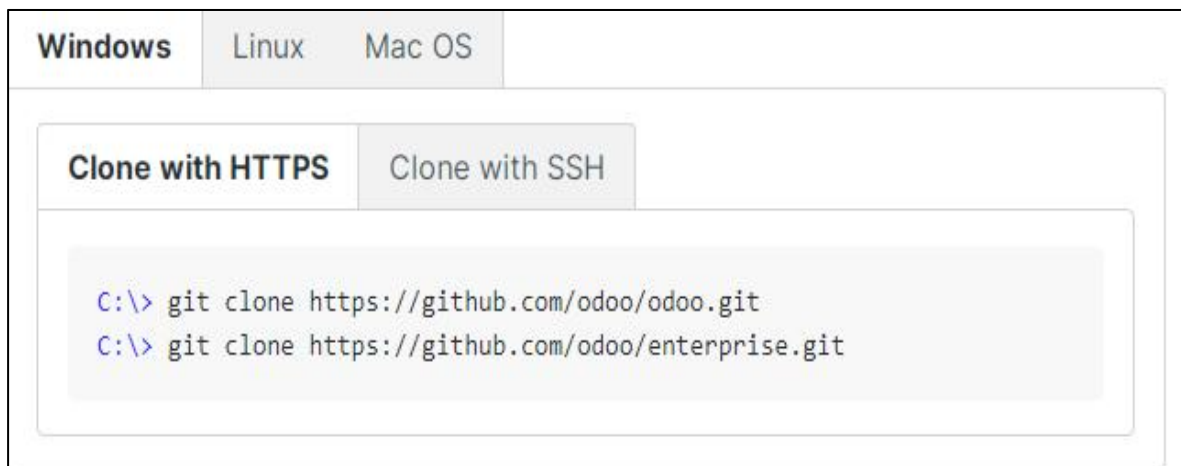
Étape 01 : Récupérer les sources :

Il y'a deux façons d'obtenir le code source d'Odoo : sous forme d'archive Zip ou par Git .

Nous avons utilisé le Git

- **Git :**

La procédure suivante nécessite que Git soit installé sur votre appareil et que vous ayez une connaissance de base des commandes de Git. Pour cloner un dépôt Git, vous devez choisir entre le clonage avec HTTPS ou SSH. Si vous ne connaissez pas la différence entre les deux, la meilleure option est probablement HTTPS. Si vous suivez le tutoriel développeur démarrer ou si vous prévoyez de contribuer au code source d'Odoo, choisissez plutôt SSH.



Étape 02 : Préparation:

- **Python :**

Odoo nécessite Python 3.7 ou une version ultérieure pour fonctionner . Consultez la page de téléchargement de Python pour télécharger et installer la dernière version de Python 3 sur votre appareil.

Pendant l'installation, cochez « Ajouter Python 3 au Chemin » , ensuite cliquez sur « Personnaliser l'installation » et veillez à cocher pip.

- **PostgreSQL:**

Odoo utilise PostgreSQL comme système de gestion de la base de données. Téléchargez et installez PostgreSQL (version prise en charge : 12.0 et ultérieure).

Par défaut, le seul utilisateur est postgres, mais Odoo interdit de se connecter en tant que postgres, vous devez donc créer un nouvel utilisateur PostgreSQL :

1. Ajoutez le répertoire bin de PostgreSQL (par défaut : `C:\Program Files\PostgreSQL|<version>|bin`) à votre PATH.
2. Créez un utilisateur postgres avec un mot de passe en utilisant le guide pg Admin :
 1. Ouvrez **pgAdmin**.
 2. Double-cliquez sur le serveur pour créer une connexion.
 3. Sélectionnez **Objet** ▶ **Créer** ▶ **Rôle Login/Groupe**.
 4. Saisissez le nom d'utilisateur dans le champ **Nom de rôle** (par ex. odoo).
 5. Ouvrez l'onglet **Définition** et saisissez le mot de passe (par ex. odoo), cliquez ensuite sur **Sauvegarder**.
 6. Ouvrez l'onglet **Privilèges** et définissez **Peut se connecter ?** sur **Oui** et **Créer une base de données ?** sur **Oui**.

- **Dépendances :**

Avant d'installer les dépendances, vous devez télécharger et installer les **Build Tools pour Visual Studio**. Lorsque vous y êtes invité, sélectionnez 'C++ build tools' dans l'onglet « Charges de travail » et installez-les.

Les dépendances d'Odoo sont énumérées dans le fichier requirements.txt situé à la racine du répertoire d'Odoo Community.

Naviguez jusqu'au chemin de votre installation Odoo Community (CommunityPath) et exécutez **pip** sur le fichier d'exigences dans un terminal **avec privilèges d'administration**:

```
C:\> cd \CommunityPath
C:\> pip install setuptools wheel
C:\> pip install -r requirements.txt
```

Pour les langues avec une interface de droite à gauche (comme l'arabe ou l'hébreu), le package rtlcss est nécessaire :

1- Télécharger et installer nodejs.

2- Installer rtlcss :

```
C:\> npm install -g rtlcss
```

3- Éditez la variable PATH de l'environnement système pour ajouter le fichier où se trouve rtlcss.cmd (généralement :

C:\Users\<user>\AppData\Roaming\npm)

Étape 03: Exécuter Odoo

Une fois que toutes les dépendances sont installées, Odoo peut être lancé en exécutant odoo-bin, l'interface en ligne de commande du serveur. Elle se situe à la racine du répertoire Odoo Community.

Pour configurer le serveur, vous pouvez spécifier des **arguments de ligne de commande**, ou un **fichier de configuration**.

Les configurations nécessaires les plus courantes sont les suivantes :

- Utilisateur et mot de passe PostgreSQL.
- Chemins d'accès personnalisés au-delà des chemins par défaut, pour charger vos propres modules.

Une façon typique d'exécuter le serveur serait :



The screenshot shows a Windows terminal window with three tabs: 'Windows', 'Linux', and 'Mac OS'. The 'Windows' tab is active. The terminal displays the following commands and their output:

```
C:\> cd CommunityPath/  
C:\> python odoo-bin -r dbuser -w dbpassword --addons-path=addons -d mydb
```

Below the terminal output, there is a text explanation: 'Où CommunityPath est le chemin de l'installation Odoo Community, dbuser est le login PostgreSQL, dbpassword est le mot de passe PostgreSQL et mydb est le nom de la base de données PostgreSQL.'

Une fois que le serveur a démarré (le journal INFO odoo.modules.loading: Modules loaded. est imprimé), ouvrez <http://localhost:8069> dans votre navigateur web et connectez-vous avec le compte administrateur de la base : Utilisez Admin pour l'**Email** et, de nouveau, Admin pour le **Mot de passe**. Ça y est, vous venez de vous connecter à votre propre base de données Odoo !

3.2. Configuration de notre module:

Avant de créer un module on devra d'abord commencer par la création d'une base de données sur Odoo, ensuite en va créer les comptes nécessaires à chaque utilisateur en affectant certain rôles et paramétrage.

La création d'une nouvelle base de donnée:

Les applications Odoo se base sur le patron de conception MVC, comme mentionné précédemment, pour séparer les technologies appropriées pour chaque partie de ce modèle. Pour gérer les données, Odoo utilise le SGBDR robuste et libre PostgreSQL, avec la couche ORM pour sécuriser et faciliter l'accès aux données stockées et au même temps simplifie les différentes requêtes sur les données. La création d'une base de donnée sur Odoo est effectuée après authentification avec l'User *Admin* par défaut.

La figure 36. Représente le formulaire de création d'une base de données Odoo :

The screenshot shows a 'Create Database' window with the following fields and values:

- Master Password: [Redacted]
- Database Name: O14-Formation
- Email: odoo@odooskills.com
- Password: [Redacted]
- Phone number: [Empty]
- Language: French / Français
- Country: Algeria
- Demo data:

At the bottom right, there is a blue 'Continue' button. Below the form, there is a note: 'To enhance your experience, some data may be sent to Odoo online services. See our [Privacy Policy](#).'

Figure 36 : Formulaire de création d'une base de données Odoo.

La figure 37. démontre l'existence de la base de données Odoo16 dans l'instance PostgreSQL.

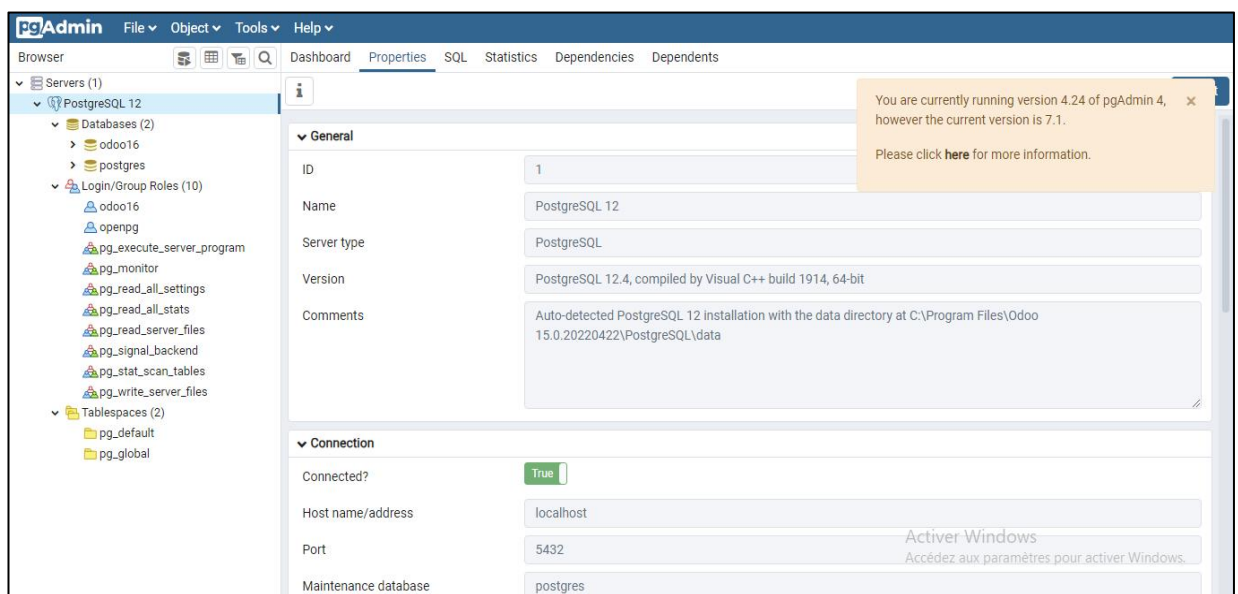
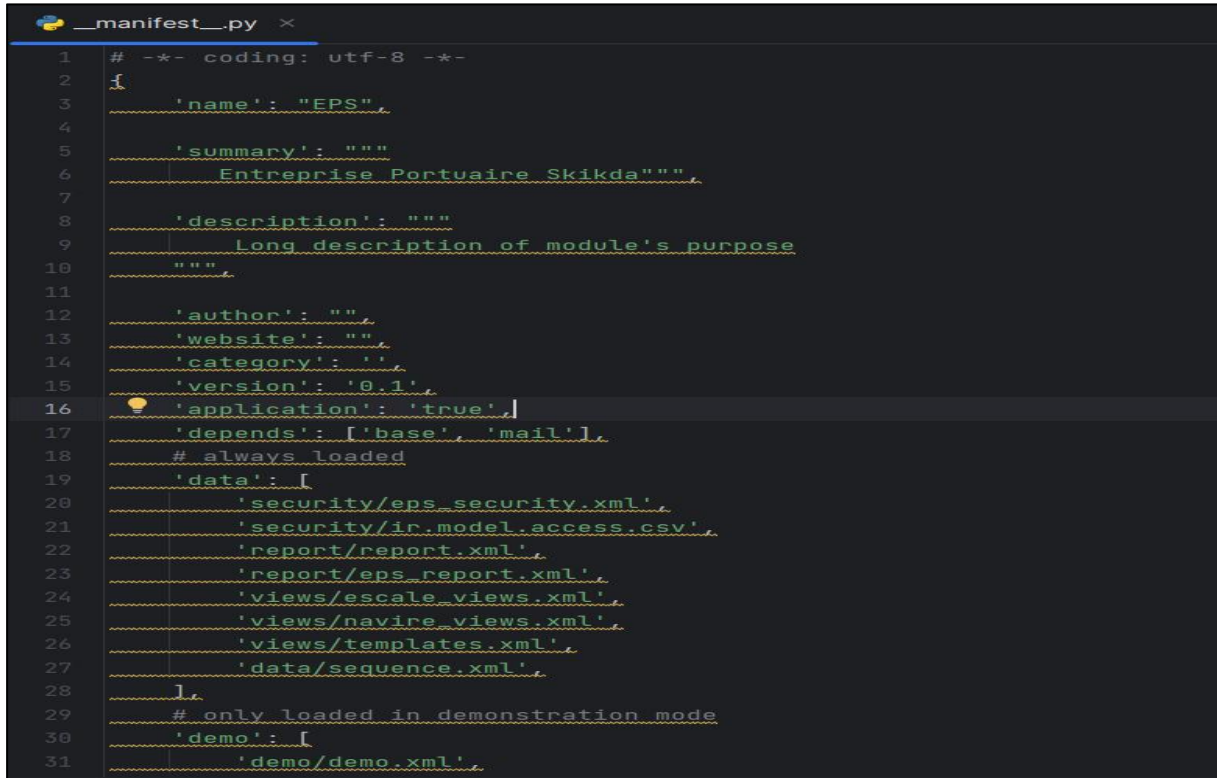


Figure 37 : L'existence de la base de données Odoo16.

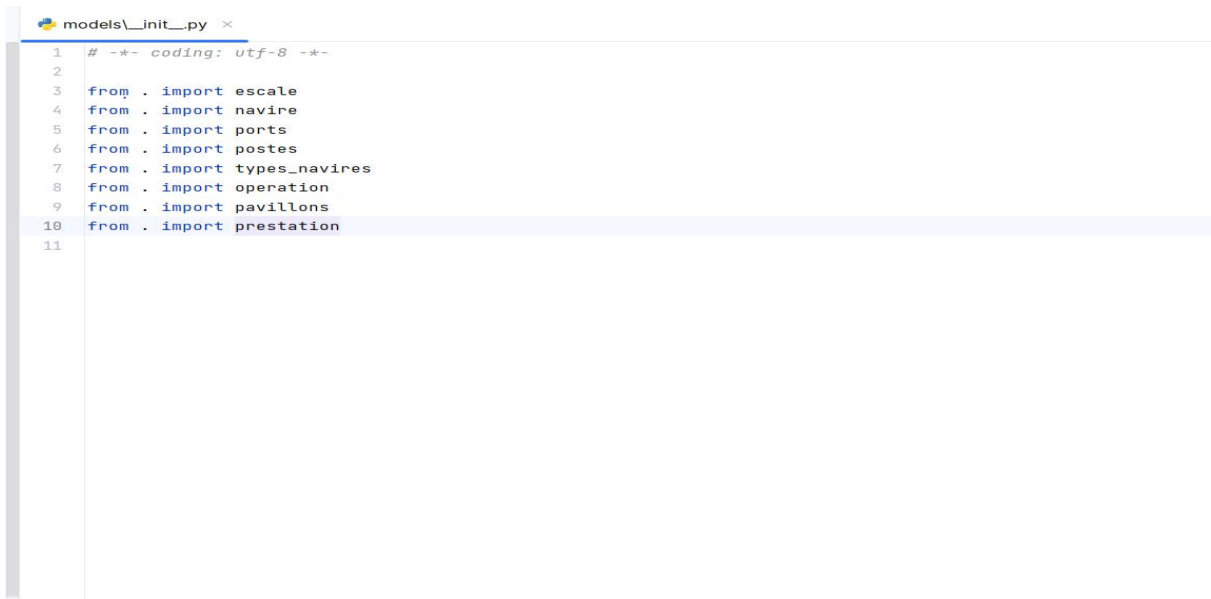
Ensuite la première étape est l'implémentation de fichier `__manifest__.py` et le fichier `init.py` ainsi que l'arborescence des fichiers nécessaires.

La Figure 38 représente l'implémentation de fichier `__manifest__.py` de notre module EPS.



```
1  # -*- coding: utf-8 -*-
2  {
3      'name': "EPS",
4
5      'summary': """
6          Entreprise Portuaire Skikda"""
7
8      'description': """
9          Long description of module's purpose
10         """
11
12     'author': "",
13     'website': "",
14     'category': '',
15     'version': '0.1',
16     'application': 'true',
17     'depends': ['base', 'mail'],
18     '# always loaded
19     'data': [
20         'security/eps_security.xml',
21         'security/ir.model.access.csv',
22         'report/report.xml',
23         'report/eps_report.xml',
24         'views/escale_views.xml',
25         'views/navire_views.xml',
26         'views/templates.xml',
27         'data/sequence.xml',
28     ],
29     '# only loaded in demonstration mode
30     'demo': [
31         'demo/demo.xml',
```

Figure 38 : L'implémentation de fichier `manifest.py` de notre module EPS



```
models\__init__.py
1  # -*- coding: utf-8 -*-
2
3  from . import escale
4  from . import navire
5  from . import ports
6  from . import postes
7  from . import types_navires
8  from . import operation
9  from . import pavillons
10 from . import prestation
11
```

Figure 39: L'implémentation de fichier `init.py` de notre module EPS

La Figure 40 montre le module EPS avec la liste des modules Odoo.

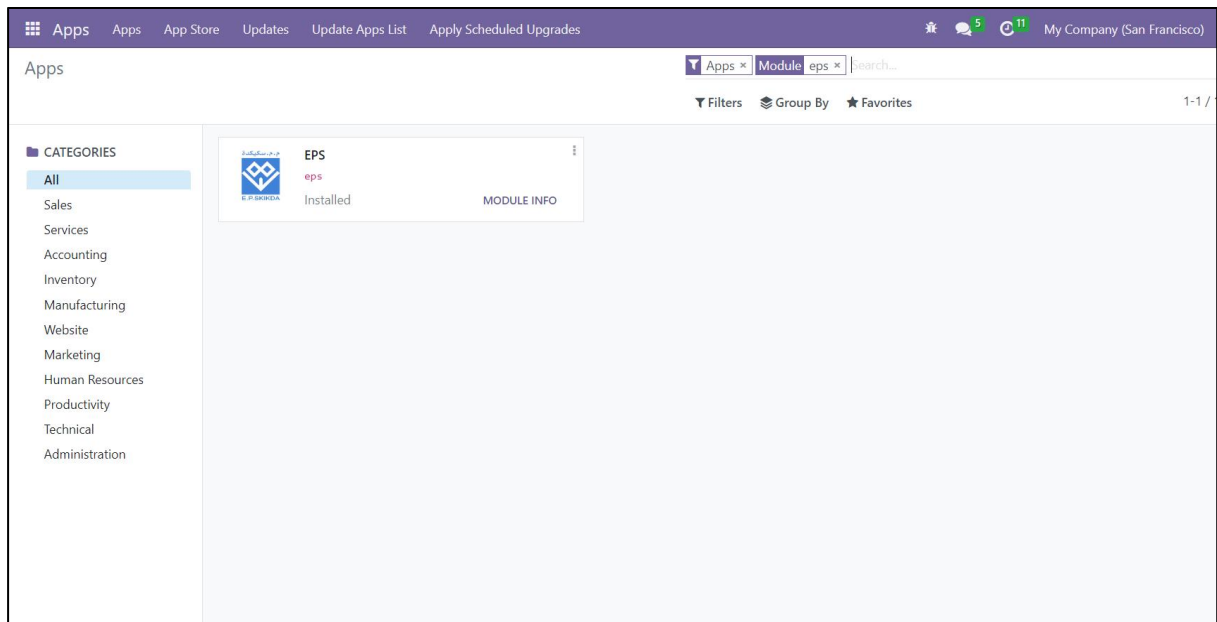


Figure 40 : Le module EPS .

La Figure 41 représente le Login pour accéder à notre module sous Odoo16 .

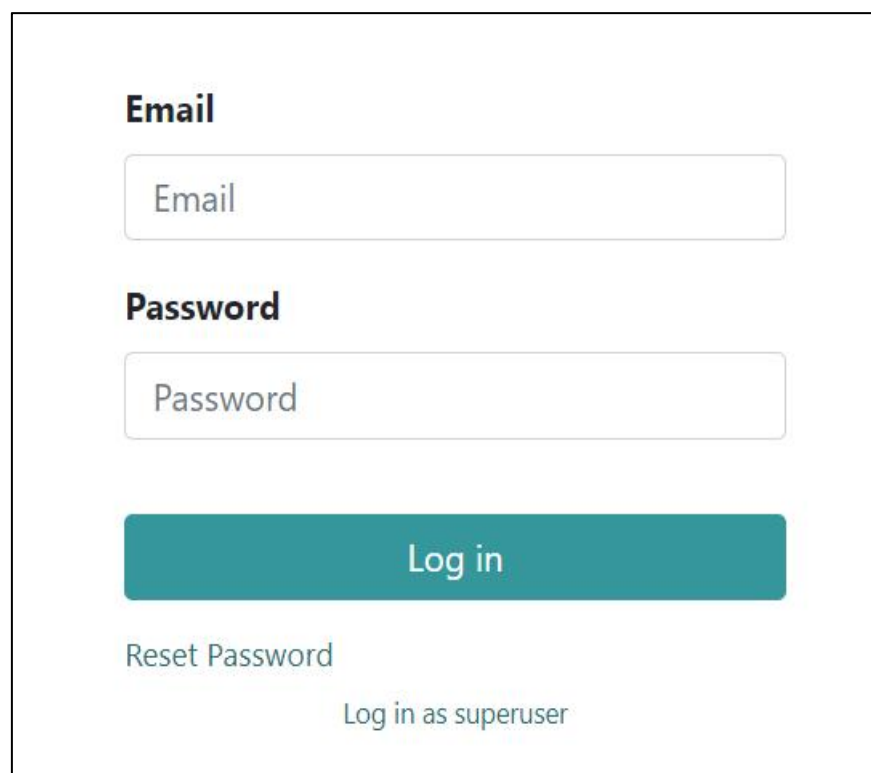
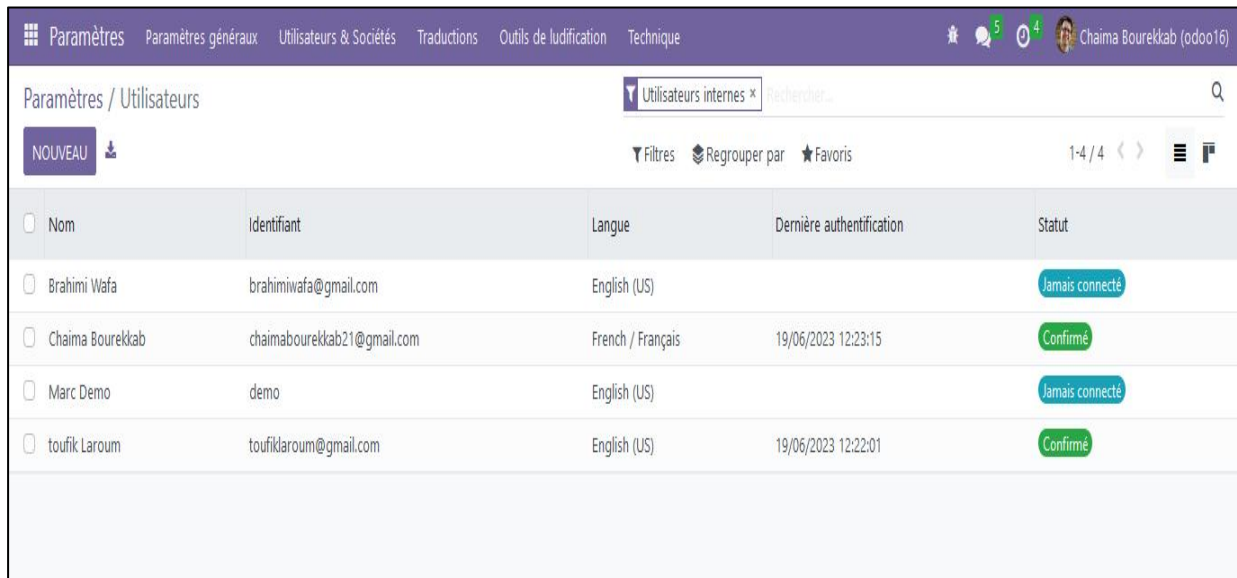
The image shows a screenshot of the Odoo login form. It features two input fields: 'Email' and 'Password'. Below the password field is a teal 'Log in' button. At the bottom, there are two links: 'Reset Password' and 'Log in as superuser'.

Figure 41: Login pour accéder à notre module.

.La création des groupes et comptes:



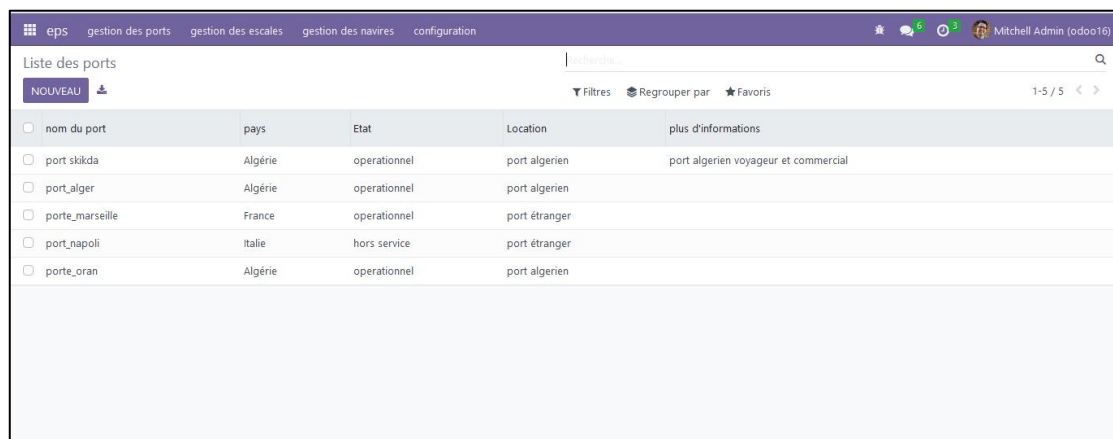
The screenshot shows the 'Utilisateurs internes' (Internal Users) management page in Odoo. The interface includes a search bar, a 'NOUVEAU' button, and a table of users. The table has columns for 'Nom', 'Identifiant', 'Langue', 'Dernière authentification', and 'Statut'. The users listed are Brahimi Wafa, Chaima Bourekkab, Marc Demo, and toufik Laroum. Chaima Bourekkab and toufik Laroum are marked as 'Confirmé', while Brahimi Wafa and Marc Demo are marked as 'Jamais connecté'.

<input type="checkbox"/>	Nom	Identifiant	Langue	Dernière authentification	Statut
<input type="checkbox"/>	Brahimi Wafa	brahimiwafa@gmail.com	English (US)		Jamais connecté
<input type="checkbox"/>	Chaima Bourekkab	chaimabourekkab21@gmail.com	French / Français	19/06/2023 12:23:15	Confirmé
<input type="checkbox"/>	Marc Demo	demo	English (US)		Jamais connecté
<input type="checkbox"/>	toufik Laroum	toufiklaroum@gmail.com	English (US)	19/06/2023 12:22:01	Confirmé

Figure 42 : La création des groupes et compte.

3.3. Développement de module :

Menu principal de l'application :

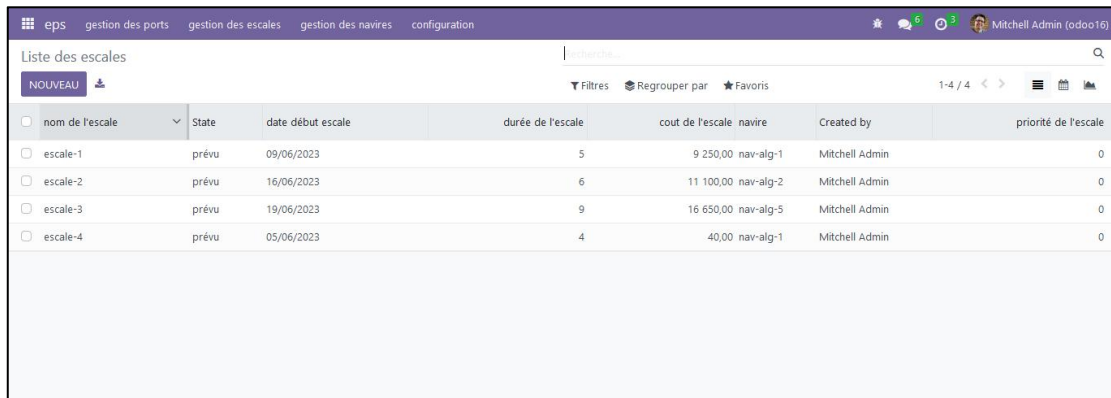


The screenshot shows the main menu of the application. The menu items are 'eps', 'gestion des ports', 'gestion des escales', 'gestion des navires', and 'configuration'. Below the menu is a search bar and a 'NOUVEAU' button. The main content area displays a table titled 'Liste des ports' (List of ports) with columns for 'nom du port', 'pays', 'Etat', 'Location', and 'plus d'informations'. The table lists five ports: port skikda, port_alger, porte_marseille, port_napoli, and porte_oran.

<input type="checkbox"/>	nom du port	pays	Etat	Location	plus d'informations
<input type="checkbox"/>	port skikda	Algérie	operationnel	port algerien	port algerien voyageur et commercial
<input type="checkbox"/>	port_alger	Algérie	operationnel	port algerien	
<input type="checkbox"/>	porte_marseille	France	operationnel	port étranger	
<input type="checkbox"/>	port_napoli	Italie	hors service	port étranger	
<input type="checkbox"/>	porte_oran	Algérie	operationnel	port algerien	

Figure 43: Menu principale de l'application.

Liste view des escales :



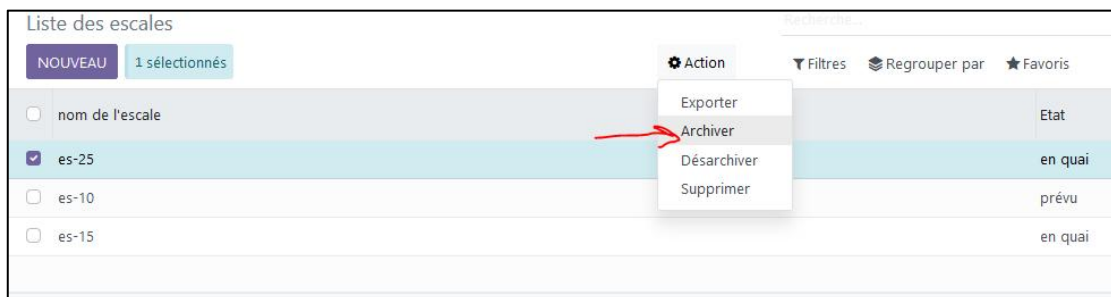
<input type="checkbox"/>	nom de l'escale	State	date début escale	durée de l'escale	cout de l'escale navire	Created by	priorité de l'escale
<input type="checkbox"/>	escale-1	prévu	09/06/2023	5	9 250,00 nav-alg-1	Mitchell Admin	0
<input type="checkbox"/>	escale-2	prévu	16/06/2023	6	11 100,00 nav-alg-2	Mitchell Admin	0
<input type="checkbox"/>	escale-3	prévu	19/06/2023	9	16 650,00 nav-alg-5	Mitchell Admin	0
<input type="checkbox"/>	escale-4	prévu	05/06/2023	4	40,00 nav-alg-1	Mitchell Admin	0

Figure 44: Liste view des escales

3.3.1. Archiver un enregistrement (un escale) :

Pour archiver un enregistrement on ajoute un field active de type boolean dans la classe escale.py

```
active=fields.Boolean(default=True)
```



<input type="checkbox"/>	nom de l'escale	Etat
<input checked="" type="checkbox"/>	es-25	en quai
<input type="checkbox"/>	es-10	prévu
<input type="checkbox"/>	es-15	en quai

Figure 45: archivage de l'enregistrement

3.3.2. Ajouter des bouton et des actions :

Les bouton s'ajoutent dans la partie header de form_view

```
<button  
    name="escale_rade"  
    string="vers etat rade"  
    type="object"  
    class="oe_highlight"  
>
```

L'attribut name représente le nom de la fonction à exécuter dans le model escale.py

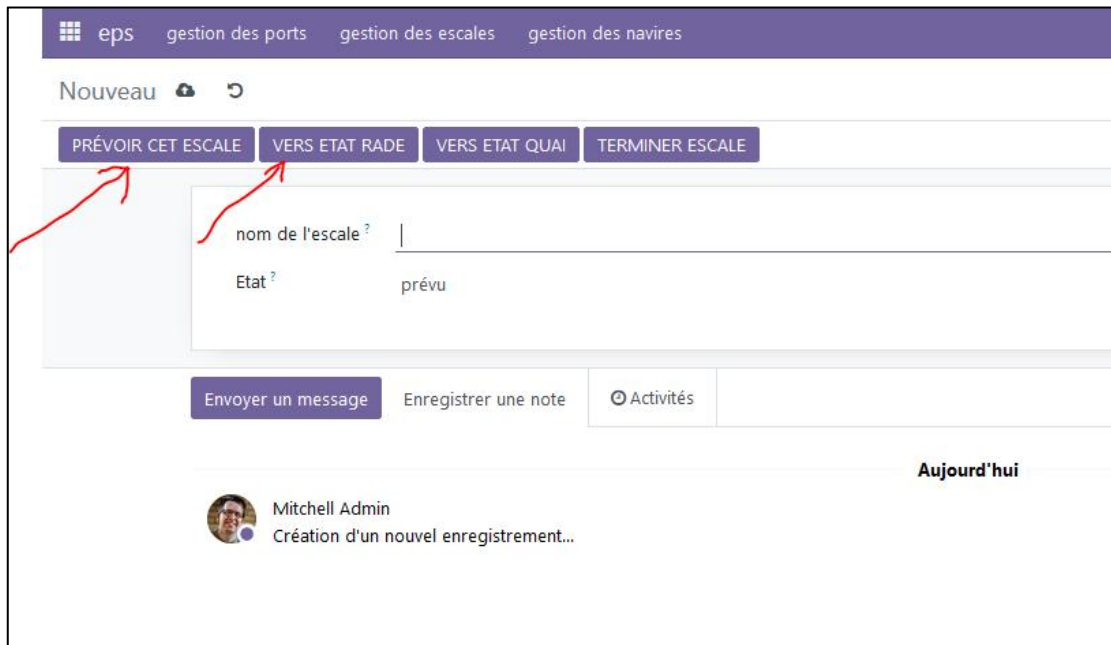


Figure 46: Ajouter des bouton et des actions.

Ça permet de manipuler l'état d'un escale avec :

```
<field name="state" widget="statusbar"/>
```

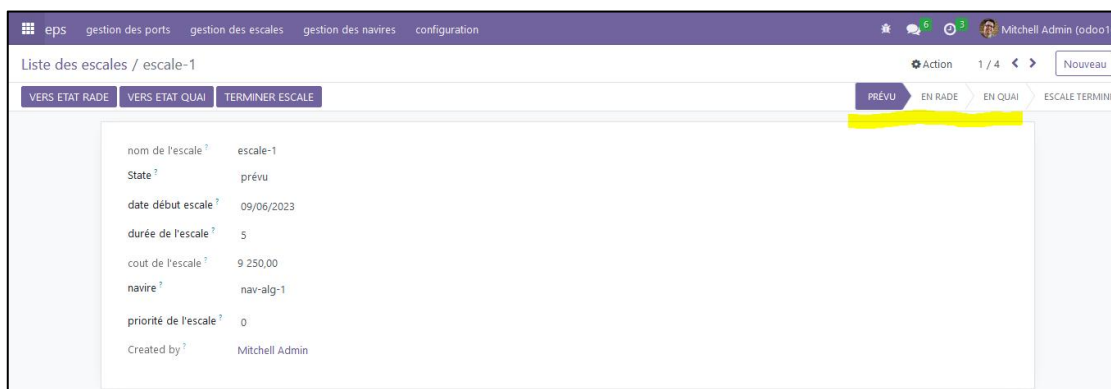


Figure 47: Manipulation de l'état d'un escale .

3.3.3. Afficher ou masquer les boutons selon l'attribut state :

- Il est possible d'afficher ou masquer les boutons selon la valeur de l'attribut **state**. Dans ce cas le model doit avoir un attribut appelé '**state**'.
- Par exemple si l'escale est dans l'état quai on masque le bouton mettre à quai et ainsi de suite.

- Donc pour chacun des boutons on peut définir les valeurs de l'attribut state pour que ce bouton soit affiché.
- Dans le form_view on rajoute l'attribut suivant dans les boutons :

```
<button name="escale_quai"  
    states=" ,prévu ,rade ,terminé"  
    string="vers etat quai"  
    type="object" class="oe_highlight" />
```

3.3.4. Afficher l'utilisateur qui a créé un enregistrement :

- Il est possible d'afficher dans form_view ou dans liste_view l'utilisateur qui a créé un escale donné.
- Odoe crée automatiquement pour chaque enregistrement dans la base de données un attribut appelé **:create_uid**

Il suffit donc de l'ajouter dans le fichier escale_view.xml :

```
<field name="create_uid"/>
```



<input type="checkbox"/> nom de l'escale	Etat	Created by
<input type="checkbox"/> es-10	escale terminé	Mitchell Admin
<input type="checkbox"/> es-15	en quai	Mitchell Admin

Figure 48 : Afficher l'utilisateur qui a créé un enregistrement 01

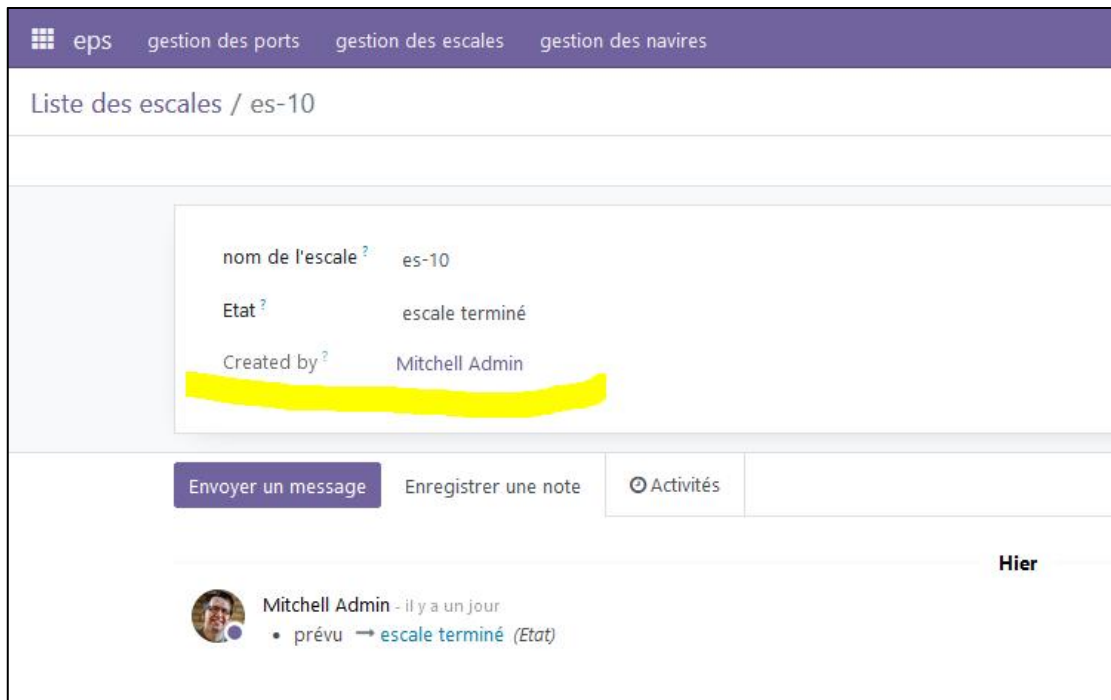


Figure 49 : Afficher l'utilisateur qui a créé un enregistrement 02

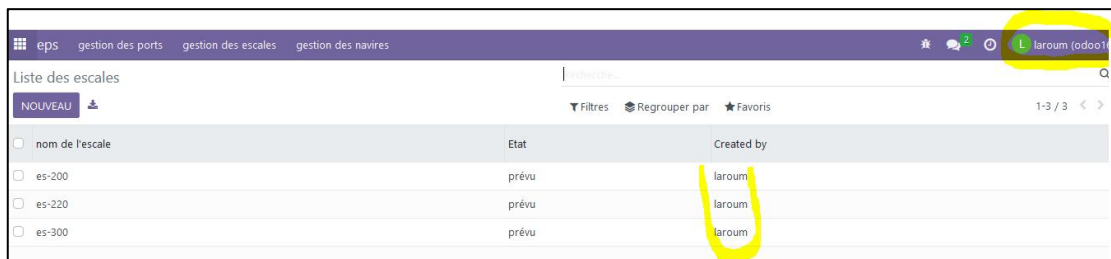


Figure 50 : Afficher l'utilisateur qui a créé un enregistrement 03

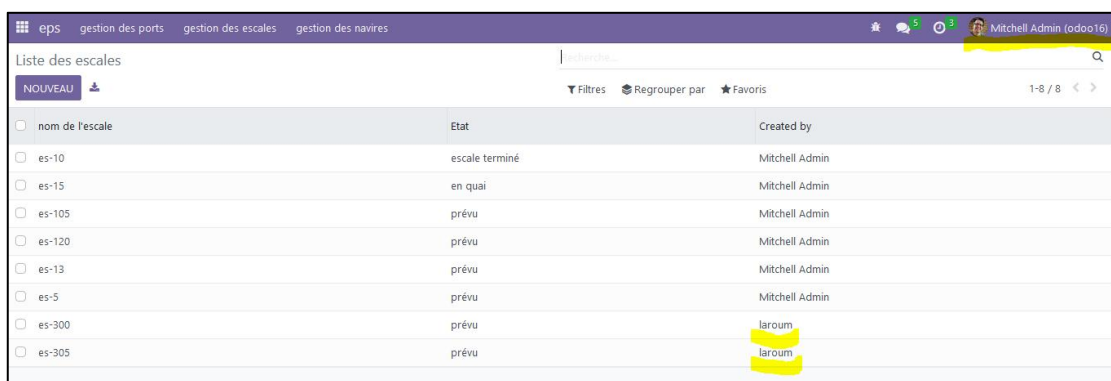


Figure 51 : Afficher l'utilisateur qui a créé un enregistrement 04

3.3.5. Ajouter un search_view :

La list_view permet d'afficher une liste complète es éléments (escales) par exemple.

Parfois il est utile de faire un filtrage des résultats, par exemple afficher uniquement les escale qui sont prévu, ou les escales qui sont en rade. Ou par exemple les escale dont le nom commence par es-10xx ... etc.

Pour ce faire on utilise un search_view dans le fichier escale_view.xml qui comprend : les élément **search** (pour chercher) et **filter** (pour filtrer):

```
<!-- the search view -->
<record id="view_search_escale" model="ir.ui.view">
    <field name="name">filtre escale</field>
    <field name="model">eps.escale</field>
<!-- les champs concernés par la recherche nom et etat -->
<field name="arch" type="xml">
    <search>
        <field name="name"/>
        <field name="state"/>
<!-- pour filtrer les escale prévu -->
<filter name="filter_escale_prévu"
    string="escale prévu"
    domain="[('state','=','prévu')]" />

<!--les escales en cours-->
<filter name="filter_escale_encours"
    string="escale en cours"
    domain="[('state','in',['rade','quai'])]" />

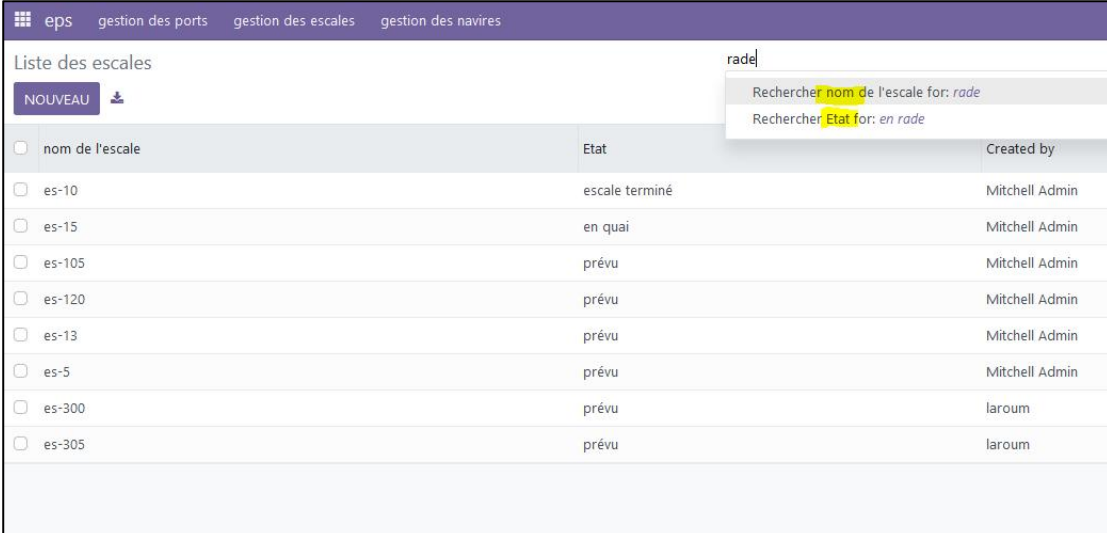
<!--pour filtrer les escale en rade-->
    <filter name="filter_escale_rade"
        string="escale en rade"
        domain="[('state','=','rade')]" />
<filter name="filter_escale_quai"
    string="escale en quai"
    domain="[('state','=','quai')]" />
<filter name="filter_escale_terminé"
```

```
string="escale terminée"
domain="[('state','=', 'terminé')]" />

<!-- groupe by-->
<group expand="0" string="Group by">
  <filter name="group_by_etat"
    string="grouper par etat"
    context="{ 'group_by': 'state' }" />
</group>
</search>
</field>
</record>
```

Dans cet exemple nous avons ajouté une balise `<search>`. Cette balise contient deux type d'éléments :

1. La balise `<field>` : représente les attribut sur lesquels on peut faire la **recherche**. C'est-à-dire quand l'utilisateur tape un mot à chercher la recherche sera effectué dans tous ces éléments fields. Dans cet exemple on a utilisé le name et etat.



The screenshot shows a web application interface with a navigation bar at the top containing 'eps', 'gestion des ports', 'gestion des escales', and 'gestion des navires'. Below the navigation bar, there is a section titled 'Liste des escales' with a 'NOUVEAU' button and a download icon. A table displays a list of scales with columns for 'nom de l'escale', 'Etat', and 'Created by'. A search dropdown menu is open, showing the search term 'rade' and two options: 'Rechercher nom de l'escale for: rade' and 'Rechercher Etat for: en rade'.

<input type="checkbox"/>	nom de l'escale	Etat	Created by
<input type="checkbox"/>	es-10	escale terminé	Mitchell Admin
<input type="checkbox"/>	es-15	en quai	Mitchell Admin
<input type="checkbox"/>	es-105	prévu	Mitchell Admin
<input type="checkbox"/>	es-120	prévu	Mitchell Admin
<input type="checkbox"/>	es-13	prévu	Mitchell Admin
<input type="checkbox"/>	es-5	prévu	Mitchell Admin
<input type="checkbox"/>	es-300	prévu	laroum
<input type="checkbox"/>	es-305	prévu	laroum

Figure 52: représente les attribut pour faire la recherche

2. Les éléments `<filters>` : qui sont les filtres qu'on peut utiliser : dans cet exemple nous avons utilisé des filtres sur l'attribut etat pour filtrer les escale dans chaque état :

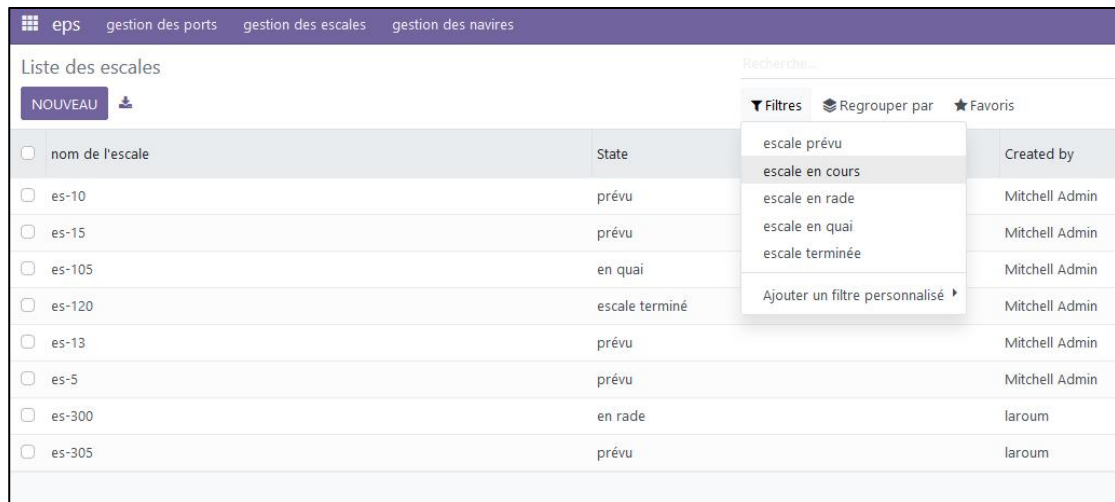


Figure 53 : l'utilisation des filtres sur les attribut

3. Pour grouper les résultats selon un attribut donné par exemple selon l'état de l'escale ou selon l'utilisateur qui l'a créé on ajoute :

```

<!-- groupe by-->
<group expand="0" string="Group by">
  <filter name="group_by_etat"
    string=" grouper par etat "
    context="{ 'group_by': 'state' }"/>
<filter name="group_by_user"
  string="grouper par user"
  context="{ 'group_by': 'create_uid' }"/>
</group>

```



Figure 54 : L'état de l'escale



Figure 55 : Grouper par utilisateur qui a créé l'escale

3.3.6. Les champs calculés (computed fields):

Les champs calculés (`@api.depends`) sont utilisés pour calculer dynamiquement la valeur d'un champ en fonction des valeurs d'autres champs dans le modèle. Le champ calculé est mis à jour automatiquement chaque fois que l'une de ses dépendances est modifiée

Pour ajouter des champs calculer on ajoute l'attribut **cost** dont la valeur sera calculée à partir de la **duration** de l'escale :

Donc, dans le fichier model escale.py, on ajoute les 2 attributs :

```
duration=fields.Integer("durée de l'escale")
cost=fields.Float("cout de l'escale",compute='calculate_cost_from_duration')
```

Où : 'calculate_cost_from_duration' est la fonction qui va calculer automatiquement cette valeur en utilisant le décorateur **@api.depends(duration)**:

```
@api.depends('duration')
def calculate_cost_from_duration(self):
    for record in self:
        record.cost=record.duration*1850
```

nom de l'escale ?	es-145
State ?	prévu
durée de l'escale ?	9
cout de l'escale ?	16 650,00
priorité de l'escale ?	2
Created by ?	

Figure 56 : Les champs calculés

Les champs calculés ne se sauvegarde pas dans la base de données sauf si on indique ça explicitement par l'attribut **store=True**

3.3.7. relations entre les attributs :

Les relations One2Many, Many2One et Many2Many sont des types de relations de base de données utilisées dans Odoo pour établir des liens entre différents modèles.

- **One2Many (un à plusieurs) :**

- ✓ C'est une relation où un enregistrement d'un modèle est lié à plusieurs enregistrements d'un autre modèle.
- ✓ Dans le modèle source (la table principale), vous utilisez un champ de type One2many pour établir cette relation.
- ✓ Dans le modèle cible (la table liée), vous utilisez un champ de type Many2one pour indiquer la relation inverse.
- ✓ Par exemple : un navire peut faire un ou plusieurs escales. Une escale est assurée par un seul navire.
- ✓ Donc dans la table escale on ajoute une relation de type many2one avec la table navire :

```
navire_id=fields.Many2one('eps.navire',string='navire')
```

- ✓ Dans la table navire on ajoute une relation one2many vers la table escale.

```
escale_ids=fields.One2many('eps.escale','navire_id',string='liste escales')
```

- Many2One (plusieurs à un) :

- ✓ C'est l'inverse de la relation One2Many.
- ✓ C'est une relation où plusieurs enregistrements d'un modèle sont liés à un seul enregistrement d'un autre modèle.
- ✓ Dans le modèle source (la table principale), vous utilisez un champ de type Many2one pour établir cette relation.
- ✓ Dans le modèle cible (la table liée), vous utilisez un champ de type One2many pour indiquer la relation inverse.

- Many2Many (plusieurs à plusieurs) :

- ✓ C'est une relation où plusieurs enregistrements d'un modèle sont liés à plusieurs enregistrements d'un autre modèle.
- ✓ Pour établir une relation Many2Many, vous devez utiliser une table de liaison qui relie les deux modèles.
- ✓ Dans le modèle source, vous utilisez un champ de type Many2many pour établir cette relation et vous spécifiez la table de liaison.
- ✓ Dans le modèle cible, vous utilisez également un champ de type Many2many pour indiquer la relation inverse.

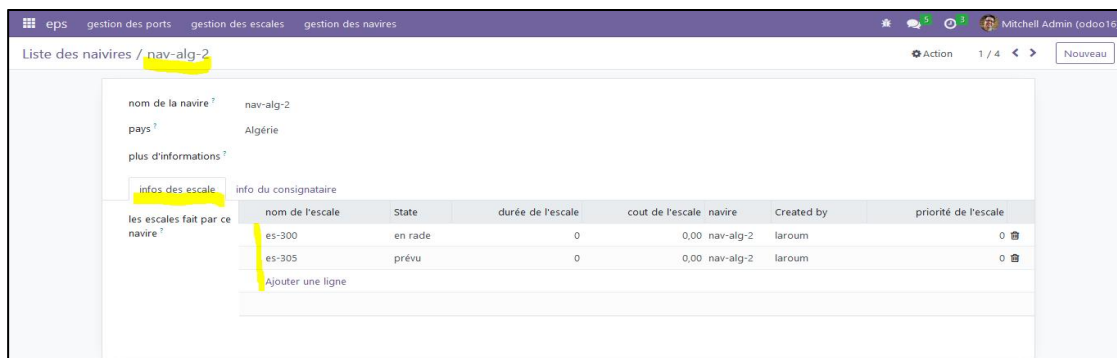


Figure 57 : Présentation des information d'une escale .

3.3.8. Kanban view :

Il est possible d'afficher la liste des escales dans une liste comme nous l'avons déjà fait, ou sous forme de kanban view qui est une sorte de carte d'éléments.

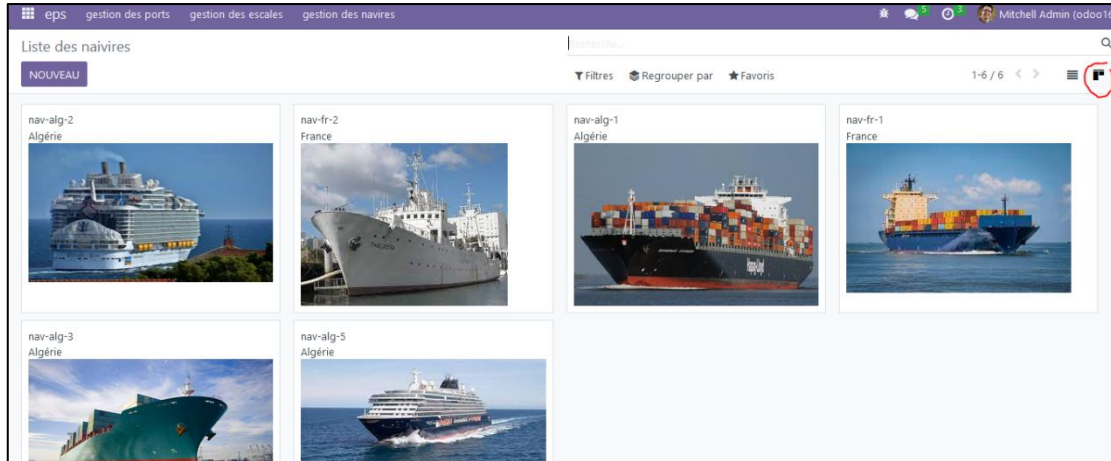


Figure 58 : Le kanban view .

3.3.9. Calendar view :

La vue du calendrier dans Odoo est un outil de gestion des événements et des tâches qui permet d'afficher les informations dans un format de calendrier. Cette vue permet aux utilisateurs d'organiser, de planifier et de suivre les activités dans Odoo de manière visuelle et intuitive.

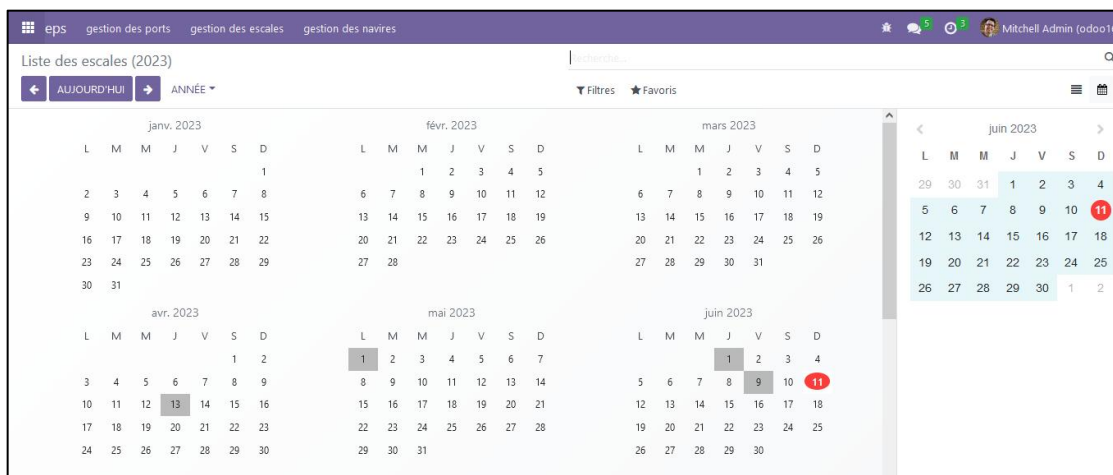


Figure 59 : les escales sous forme de calendrier

3.3.10. Sequences en odoo :

Dans Odoo, une séquence est un mécanisme permettant de générer automatiquement des numéros ou des codes uniques pour les différents types de documents ou d'enregistrements. Les séquences sont couramment utilisées pour générer des numéros de factures, des numéros de commande, des codes de produits, des numéros de lot, etc.

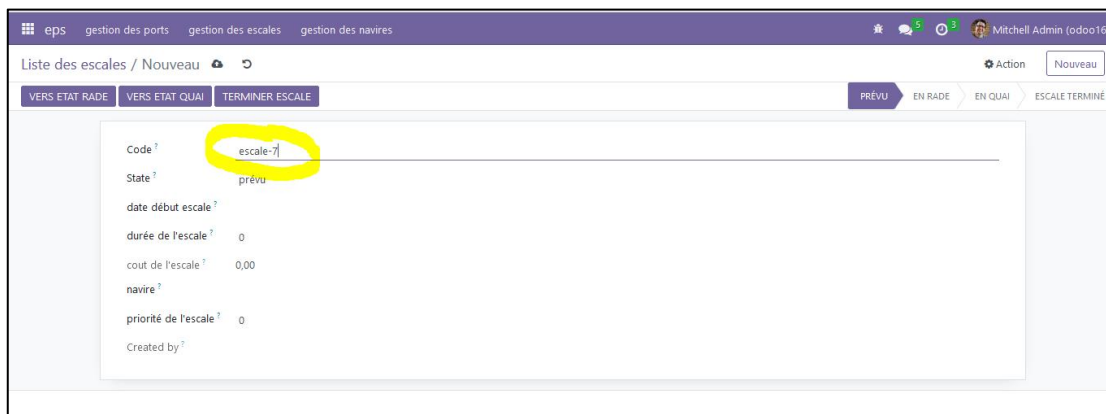


Figure 60 : Sequences en odoo

3.3.11. Créer des paramètres du modèle :

Les paramètres système sont des valeurs stockées dans la base de données d'Odoo et peuvent être utilisées pour stocker des informations globales accessibles dans toute l'application. Par exemple créer un paramètre qui représente le tarif de base de l'escale.



Figure 61 : La création du paramètre de module

- Lire et utiliser le paramètre setting :

Pour lire ce paramètre de configuration que nous avons créé on écrit :

```
tarif_jour_escale=self.env['ir.config_parameter'].sudo().get_param('escale.tarif_escale')
```

Remarque :

Cette valeur est de type string, pour faire des calculs faut faire un cast.

Par exemple le cout de l'escale=nbr-jour*tarif-jour-escal :

si le paramètre tarif-jour-escal=5 alors le cout= 5*duré-escale :



Figure 62: L'utilisation des paramètre setting

4. Conclusion :

Dans ce chapitre nous avons décrit l'environnement de développement, ainsi que les différents outils et techniques informatiques qui ont été utilisés. Nous avons présenté les étapes détaillées d'implémentation, de configuration et d'installation de notre module « gestion des escales » sous Odoo 16. Nous avons illustré également les différentes captures d'écran de notre module et qui présente une partie des fonctionnalités du système conçu.

Conclusion générale

Durant ce projet de fin d'étude, nous avons présentés les étapes de conception et de réalisation d'un ERP dédiée au suivi des escales des navires au sein de l'EPS. Ce projet nous a permis de mettre en pratique toutes les connaissances acquises au cours de notre formation universitaire.

En effet, nous avons fait une étude approfondie du système SIGNE utilisé par le service capitainerie de l'EPS pour la gestion des escales. Nous avons par la suite critiqué ce système et fait ressortir ses failles, faiblesses et limites dans le but de proposer une solution adéquate en accord avec l'organisme d'accueil. Notre logiciel est développé en utilisant un ensemble d'outil et de techniques innovantes .

Contrairement aux solutions existantes, l'ERP proposée permet non seulement d'être informer sur l'état des navires au sein du port (en attente, en rade, en quai) en temps réel, mais aussi d'alerter les membres du personnel de la capitainerie en cas de modification de toute information en relation avec les navires à savoir : le numéro du quai attribué à un navire, données relatives à un navire (le poids, tirant eau, etc.)

Comme tout projet, notre travail peut être compléter et amélioré par l'intégration de nouvelles fonctionnalités. Ainsi, nous envisageons comme perspectives, d'une part de relier le logiciel à la base nationale APCS.dz, définit en tant qu'acteur secondaire dans notre modélisation, à laquelle nous n'avions pas de droit d'accès mais aussi de programmer des alertes sur les smartphones des utilisateurs en cas de création d'une nouvelle notification.

En plus du service capitainerie, nous espérons élargir le champ d'utilisation de notre proposition à d'autres services de l'EPS par l'intégration de nouveaux

modules permettant de prendre en charge la facturation, la gestion des manifestes, la gestion des remorqueurs etc.

Ce projet a fait l'objet d'une expérience très intéressante, car elle nous a permis de nous familiariser avec de nouvelles notions d'une part, et d'améliorer nos connaissances et nos compétences dans le domaine de la programmation, et d'autre part avoir le sens de responsabilité et gestion de projets qui permet et faciliter l'insertion dans le domaine professionnel .

Références

- [1] : source Entreprise Portuaire SKIKDA (EPS) (Accès le 22/03/2023)
- [2] : <https://www.oracle.com/fr/erp/what-is-erp/>
- [3] : <https://www.zdnet.fr/blogs/green-si/erp-digital-quand-scm-et-crm-ne-feront-plus-qu-un-39875681.htm>
- [4] : <https://www.celge.fr/article-conseil/histoire-logiciels-erp>
- [5] : <https://www.cogitime.fr/un-erp-un-outil-pour-les-tpe-pme-leur-permettant-de-sadapter-au-contexte-actuel/>
- [6] : <https://www.flowlineintegration.com/erp/definition/>
- [7] : <https://www.javelin-tech.com/blog/fr/2021/03/is-your-erp-system-manufacturing-focused/>
- [8] : <https://adopteunsoft.com/avantages-et-inconvenients-dun-erp/>
- [9] : <https://www.celge.fr/article-conseil/marche-erp>
- [10] : <https://erp.ooreka.fr/comprendre/logiciel-erp>
- [11] : <https://www.etudier.com/dissertations/Erp/68923747.html>
- [12] : <https://fablain.developpez.com/tutoriel/presenterp/>
- [13] : <https://www.choisirmonerp.com/erp/definition-d-un-erp>
- [14] : <https://fr.wikipedia.org/wiki/Odoo>
- [15] : livre Odoo une vraie opportunité pour les entreprises et les Développeurs de
BENHAMIDA Mustapha – Mars 2021

- [16] : <https://www.celge.fr/editeurs/odoo-logiciel-de-gestion-dentreprise-crm-erp-facturation-comptabilite-gpao-cms-ecommerce>
- [17] : <https://www.blogdumoderateur.com/tools/odoo/>
- [18] : <https://www.cogitime.fr/odoo-lerp-adapte-a-votre-entreprise/>
- [19] : <https://odooskills.com/>
- [20] : <https://atoutpersona.com/odoo/>
- [21] : <https://www.futura-sciences.com/tech/definitions/informatique-uml-3979/>
- [22] : <https://www.altova.com/fr/umodel>
- [23] : <https://www.gleek.io/blog/uml-diagram-types>
- [24] : <https://web.maths.unsw.edu.au/~lafaye/CCM/uml/uml-use-cases.htm>
- [25] : <https://prive.iutenligne.net/iuRxM0CThIXDIjzG/informatique/langages/kettaf/UML/04modeleconceptuel/0301usecase.html>
- [26] : [https://docwiki.embarcadero.com/RADStudio/Sydney/fr/Relations_des_diagrammes_de_classes_\(UML_1.5\)](https://docwiki.embarcadero.com/RADStudio/Sydney/fr/Relations_des_diagrammes_de_classes_(UML_1.5))
- [27] : Cours n°2: Diagramme des cas d'utilisation
- [28] : https://atefsd.weebly.com/uploads/5/0/3/6/503639/csi_02_chap02.pdf
- [29] : R. Pascal, "Uml 2.5 par la pratique", 8eme éd., Eyrolles, Éd., 2018.
- [30] : https://www.coursinfo.fr/decouverte/windows/windows-cest-quoi/?fbclid=IwAR04ejtQoy-Q9aw6XqIMg0LD8NxrcWB771OnUnKehwcIehm29_oUL-aeFdjI
- [31] : <https://fr.wikipedia.org/wiki/PyCharm>.
- [32] : <https://fr.wikipedia.org/wiki/Git>
- [33] : <https://fr.wikipedia.org/wiki/GitHub>.
- [34] : <https://www.python.org/>.

[35] : <https://www.postgresql.org/>

[36] : <https://www.w3.org/XML/>

[37] : <https://odooskills.com/bien-comprendre-architectue-technique-odoo.html>