

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Master en Informatique

Option : Génie Logiciel (GL)

Thème

**Développement d'une application mobile
Pour la gestion de stock basée sur Odoo**

Réalisé par :

- Taleb chahrazed abir
- Moussaoui sara firdaous

Présenté le 06 Juillet 2019 devant le jury composé de :

- Sidi Mohamed chouiti (Président)
- Souad khitri (Examinatrice)
- Fazilet Benmansour (Encadreur)
- Noredine Brahmi (Maitre de stage)

Année universitaire :2018-2019

REMERCIEMENT

☺ *Nous tenons tout d'abord à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce Modeste travail, à nos familles et nos amis qui par leurs prières et leurs encouragements, on a pu surmonter tous les obstacles.*

☺ *En second lieu, nous tenons à remercier notre encadreur Madame : **Fazilet Benmansour**, ses précieux conseils et son aide durant toute la période du travail.*

☺ *Nous tenons à remercier Monsieur **Noureddine Brahmí** pour son accueil comme des stagiaires au sein de son entreprise et pour son soutien et ses conseils*

☺ *Nos vifs remerciements vont également aux membres du jury Monsieur **Sidi Mohamed Chouiti** et madame **Souad Khitri** pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail Et de l'enrichir par leurs propositions.*

☺ *Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.*

Merci 

DÉDICACE

Ce travail, et bien au-delà, je le dois à

• *Mes très chers parents qui m'ont fourni au quotidien un soutien et une confiance sans faille et de ce fait, je ne saurais exprimer ma gratitude seulement par des mots. Que dieu vous protège et vous garde pour nous sans oublier mes grands-mères, mon grand-père, mes tantes mes oncles et toutes ma famille.*

• *Mes chères frères et sœurs à qui je souhaite un avenir radieux plein de réussite*

• *A tous mes amis avec lesquels j'ai partagé mes moments de joie et de bonheur*

• *Ainsi au personnel de la société SOGESI.*

« Ce succès de mon grand-père qui a voulu vivre longtemps pour partager avec moi la joie de ce succès. » " La miséricorde de Dieu sur vous "

Abir

DÉDICACE

Je dédie ce travail

A mes très chers parents

A mes frères Fayçal et Nadir

A ma sœur Wafaa

A mes enseignants

A mes amis

Sara

TABLE DES MATIERES

| | |
|--|----|
| Introduction Générale | 1 |
| Chapitre1 : Les ERP et Odoo..... | 3 |
| 1. Introduction..... | 4 |
| 2. Entreprise Resource Planning (ERP) | 4 |
| 2.1 Définition d'une solution ERP..... | 4 |
| 3. Architecture modulaire des ERP | 5 |
| 4. Etude comparative entre quelques ERP existants sur le marché..... | 6 |
| 5. Odoo..... | 7 |
| 5.1 La plateforme Odoo | 7 |
| 5.2 Architecture fonctionnelle d'Odoo | 7 |
| 6. La gestion des stocks et ses objectifs | 8 |
| 7. La structure logistique de stock dans Odoo | 8 |
| 7.1 Concept de la chaine logistique dans l'entreprise..... | 10 |
| 8. Schéma des flux | 11 |
| 9. Conclusion..... | 12 |
| Chapitre2 : Analyse et Conception du système | 13 |
| 1. Introduction | 14 |
| 2. Présentation de la société | 14 |
| 2.1 Les services de la société | 14 |
| 2.2 Les solutions de la société | 14 |
| 2.3 Les clients de la société : | 15 |
| 3. Méthodologie de travail | 15 |
| 4. Planification de projet | 16 |
| 5. Spécification des besoins | 18 |
| 6. Conception | 19 |
| 6.1. Schéma illustratif d'une réception partielle..... | 20 |
| 6.2. Diagramme de cas d'utilisation | 20 |
| 6.2.1. Cas d'utilisation « gestion de stock » | 21 |
| 6.3. Diagrammes de séquence | 21 |
| 6.3.1. Gérer les réceptions | 22 |

| | |
|--|----|
| 6.3.2 Authentification | 24 |
| 6.4 Diagramme de classe | 25 |
| 6.5 Diagramme de navigation..... | 26 |
| 7. Conclusion..... | 27 |
| Chapitre3 : Réalisation..... | 28 |
| 1. Introduction | 29 |
| 2. Étude technique | 29 |
| 2.1 Odoo..... | 29 |
| 2.2 Service Web | 29 |
| 2.3 Android | 30 |
| 3. Environnements de développement | 30 |
| 3.1 Android Studio..... | 30 |
| 3.2 PostgreSQL..... | 31 |
| 3.3 Visual Paradigm..... | 31 |
| 4. Les langages de développement utilisés | 31 |
| 5. Synchronisation de l'application mobile avec la base de données Odoo..... | 32 |
| 6. Développement de l'application mobile « Gestion De Stock » | 32 |
| 6.1 Modification du module « Gestion d'entrepôt »..... | 33 |
| 6.2 Interfaces de l'application sous Android | 34 |
| 7. Conclusion..... | 42 |
| Conclusion Générale..... | 43 |
| Bibliographie et Webographie | 44 |
| Liste des abréviations | 47 |
| Liste des figures..... | 48 |
| Liste des tableaux..... | 49 |

Introduction Générale

La gestion des stocks est un problème pour la plupart des entreprises œuvrant dans les divers secteurs de l'économie : manufacturier, distribution, assemblage et autres.

La gestion des stocks peut être considérée comme une discipline de gestion, appartenant à la famille des techniques d'organisation logistique (gestion des flux des entreprises) et dont l'enjeu principal consiste à disposer de ressources suffisantes pour ne pas bloquer les différents processus de production ou de vente.

La gestion des stocks se définit comme l'ensemble des activités se rapportant à la planification, à la constitution, au dénombrement, à l'entreposage des stocks et visant à assurer de façon optimale la disponibilité des matières, des composants, des articles de façon à satisfaire, dans les conditions les plus économiques, les besoins de la production et de la vente [1]. Dans un tel contexte, l'importance de bien gérer les stocks est cruciale.

L'objectif de la gestion logistique est de garantir le niveau de stock requis pour une manipulation adéquate de la production avec un minimum possible des coûts et des risques.

Des ruptures de stock peuvent être provoquées à tout moment par des facteurs tels que la demande fluctuante des clients, les délais variables pour les matériaux achetés ou les pièces fabriquées, ainsi que par des problèmes inattendus dans les processus de la chaîne d'approvisionnement [2].

L'optimisation des niveaux de stock est donc une préoccupation constante d'un gestionnaire en charge de ces questions. Cependant, ce n'est pas toujours chose facile. En effet bien que, de prime abord, la gestion d'un stock paraisse extrêmement simple (il s'agit d'une zone tampon entre un flux entrant et un flux sortant), gérer un stock requiert une parfaite maîtrise des composants qui le constitue.

Les problématiques de gestion auxquelles sont confrontées les entreprises poussent les décideurs à réfléchir à l'optimisation de leur organisation.

Avant que les systèmes ERP (Enterprise Resource Planning) ne soient devenus le standard dans de nombreuses organisations, les entreprises traitaient généralement les

données et les informations en utilisant plusieurs systèmes d'information et packages dispersés [3].

Aujourd'hui, les systèmes ERP prennent en charge différentes fonctions métier à l'aide d'une base de données logique qui relie et connecte les diverses fonctions dans l'organisation.

Au fil du temps, les solutions de gestion intégrées se sont peu à peu standardisées donnant naissance aux premières solutions ERP.

C'est dans ce cadre que s'inscrit notre projet de fin d'étude pour l'obtention d'un Master en Génie Logiciel. Notre travail consiste à proposer une application mobile pour la gestion des stocks. Ce sujet nous a été proposé par l'entreprise SOGESI (Société de Gestion des Solutions Informatiques). Le stage s'est déroulé du 24 Février au 23 Juin 2019.

L'objectif de ce travail est de proposer une solution qui soit adaptée et dédiée aux entreprises pour la gestion des stocks en customisant les solutions existantes. Cette contribution devra permettre d'éviter les erreurs commises lors de la saisie des informations à l'aide d'une application mobile (Android).

Pour bien comprendre la chronologie de ce travail, ce manuscrit est organisé en trois chapitres suivis d'une conclusion générale.

Le premier chapitre présente les ERP, plus particulièrement Odoo et s'attarde aussi à définir la gestion des stocks et ses activités.

Le deuxième est très important puisqu'il présente et explique le cadre conceptuel spécifique utilisé pour structurer notre recherche. Il contient l'ensemble des diagrammes nécessaires à la bonne compréhension du système d'information, du diagramme de cas d'utilisation au diagramme de classe.

Le troisième est destiné à la phase de réalisation de l'application, explique ses différentes fonctionnalités et propose des aperçus de celle-ci.

Chapitre 1 : Les ERP et Odoo

1. Introduction

Chaque entreprise ou organisation est dotée d'un système de gestion caractérisé par une certaine unité et une certaine cohérence et est conduite à organiser son système de gestion et à le remanier périodiquement en fonction de l'évolution de ses orientations, de la nature de ses ressources et des relations qu'elle doit établir avec son environnement national et international.

Les entreprises considèrent la mise en œuvre d'une solution ERP comme une véritable opportunité de se structurer, d'appliquer les bonnes pratiques de gestion et de disposer d'un outil de pilotage qui lui permettrait de soutenir sa stratégie. L'expression Progiciel de Gestion Intégrée, en abrégé PGI, est l'équivalent français du terme anglais Enterprise Resource Planning ou ERP.

L'ERP, est le terme le plus couramment utilisé, procure d'importants avantages aux organisations comme la réduction des coûts, l'augmentation de la productivité, l'amélioration de l'efficacité opérationnelle et la réalisation des avantages concurrentiels.

Les ERP ont été développés afin de répondre à la problématique d'hétérogénéité des systèmes d'information et offrent aux entreprises la possibilité de gérer l'ensemble de leurs fonctions dans une seule et même solution de gestion [4].

La gestion des stocks est une discipline très technique car elle fait appel à de nombreux concepts de gestion, de statistiques et nécessite une bonne vision du fonctionnement de la chaîne logistique de l'entreprise. Le stock est en réalité le résultat d'un grand nombre de composantes qui rendent l'optimisation de la performance plus complexe [5].

L'Open ERP propose un module de gestion de stock à double entrées. Il permet de gérer facilement les besoins complexes comme le suivi des stocks fournisseurs/clients, une traçabilité complète et aussi des liens vers d'autres modules.

2. Entreprise Resource Planning (ERP)

2.1 Définition d'une solution ERP

Millet définit l'ERP comme étant : « Une offre progicielle regroupant des applications paramétrables, modulaires, intégrées et ouvertes, s'appuyant sur un

référentiel unique de données, de procédures et de règles de gestion. Configuré et adapté au contexte d'une entreprise, il devient le support d'une stratégie d'intégration qui vise à fédérer et à optimiser les processus de gestion de l'entreprise et de relation avec ses partenaires » [6]. Ce dernier insiste sur l'intégration de l'ERP dans l'organisation de l'entreprise et sur l'impact de la performance des processus de gestion de celle-ci.

3. Architecture modulaire des ERP

Le terme « modulaire » signifie la possibilité pour l'entreprise de sélectionner les modules fonctionnels qu'elle désire utiliser parmi ceux proposés dans l'offre de l'ERP.

Le terme « fonctionnel » fait référence aux fonctions de l'entreprise. De ce fait, les « modules fonctionnels » ont pour finalité la gestion informatisée d'une fonction bien définie de l'entreprise.

L'ensemble de ces modules utilise et partage en temps réel, les informations contenues dans la base de données unique de l'ERP. Ces informations sont également appelées : « données de base ». La base de données garantit l'unicité et la fiabilité des informations qui sont échangées entre les modules fonctionnels.

Grace à l'intégration des fonctions de l'entreprise offerte par l'ERP, l'entreprise va gérer ses processus métiers avec plus de transversalité. Les échanges entre les fonctions seront quasi-automatisés. Les utilisateurs en charge des fonctions effectuent davantage de contrôle que de saisie d'informations [4].

La figure 1 représente un exemple d'architecture modulaire qui tend à représenter tous les ERP :

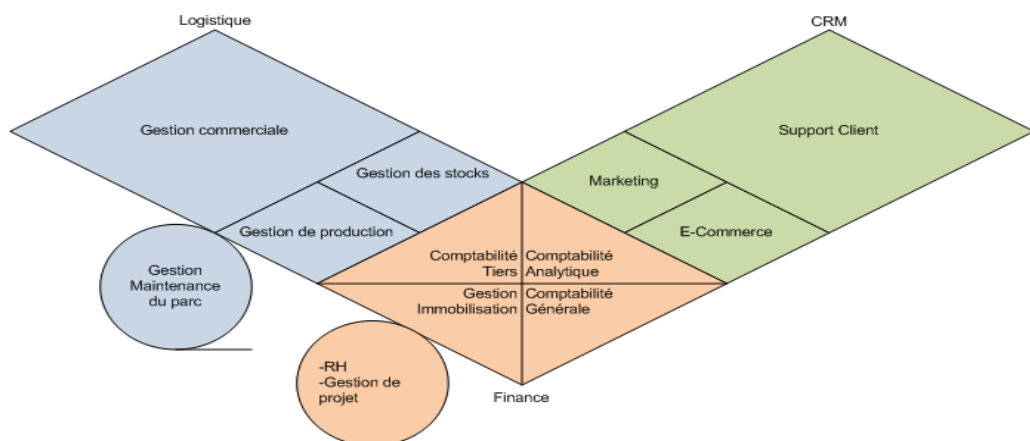


Figure 1 : Architecture modulaire des ERP [25].

4. Etude comparative entre quelques ERP existants sur le marché

Les logiciels ERP sont particulièrement adaptés aux PME qui veulent maîtriser et optimiser leur chaîne de production et avoir de la visibilité sur le projet qu'elles veulent concevoir et développer.

Côté secteurs, les logiciels ERP sont particulièrement prisés par les entreprises du domaine des services, des domaines de l'informatique et de l'Internet.

Elles doivent en effet gérer des projets sur le long terme.

De manière globale, les logiciels ERP s'adresse aux entreprises menant des projets nécessitant de multiples compétences et devant coordonner le travail de plusieurs équipes ou au moins de plusieurs personnes.

On peut considérer qu'un logiciel ERP est nécessaire pour conduire un projet où une dizaine de personnes doivent intervenir [7].

Le tableau 1 représente une comparaison de quelques ERP open source.

| ERP Open Source | Année de création | Couverture fonctionnelle | Environnement de travail |
|-----------------|-------------------|--|----------------------------|
| Open Bravo | 2001 | Comptabilité, finance, achat, vente, stock, production, CRM, point de vente | J2EE, oracle PostgreSQL |
| Dolibarr | 2003 | Comptabilité, finance, achat, vente, stock, production, CRM | PHP, PostgreSQL |
| Odoo | 2002 | Comptabilité, finance, achat, vente, stock, production, CRM, point de vente, GED, e-commerce, RH | Python, XML, PostgreSQL |
| WebERP | 2003 | Comptabilité, finance, achat, vente, stock, production, RH | J2EE, JBoss |
| ERP NEXT | 2008 | Comptabilité, finance, achat, vente, stock, production, RH | J2EE, JBoss |

Tableau 1 : Comparaison de quelques ERP open source [7].

5. Odoo

5.1 La plateforme Odoo

Créée en 2002 par Fabien PINCKAERS, anciennement connu sous le nom TinyERP, Odoo est un ERP open source. Il permet de construire des applications informatiques (gestion des commandes, des stocks, de la paie, de la comptabilité, etc.) de manière modulaire et intégrée au niveau des traitements offerts (les différents modules qui le composent sont indépendants mais parfaitement compatibles entre eux), de manière rigoureuse et cohérente au niveau des données gérées (partage d'une base de données unique et commune). Odoo fournit à l'ensemble des acteurs de l'entreprise une image unique, en plus il permet de fédérer l'ensemble des processus de l'entreprise dans chacun de ses domaines, il optimise sa productivité puisque le code source est à la disposition du grand public, généralement un effort de collaboration améliorent l'ensemble [7].

5.2 Architecture fonctionnelle d'Odoo

Odoo possède une structure modulaire qui permet d'ajouter de nouveaux modules facilement pour étendre les fonctionnalités.

Un module est un dossier avec une structure prédéfinie contenant du code python et des fichiers XML, qui permet de définir la structure de données, formulaires, rapports, menus, procédures, flux de travail, etc.

La figure 2 ci-dessous représente les différents modules locaux disponibles dans Odoo 8.

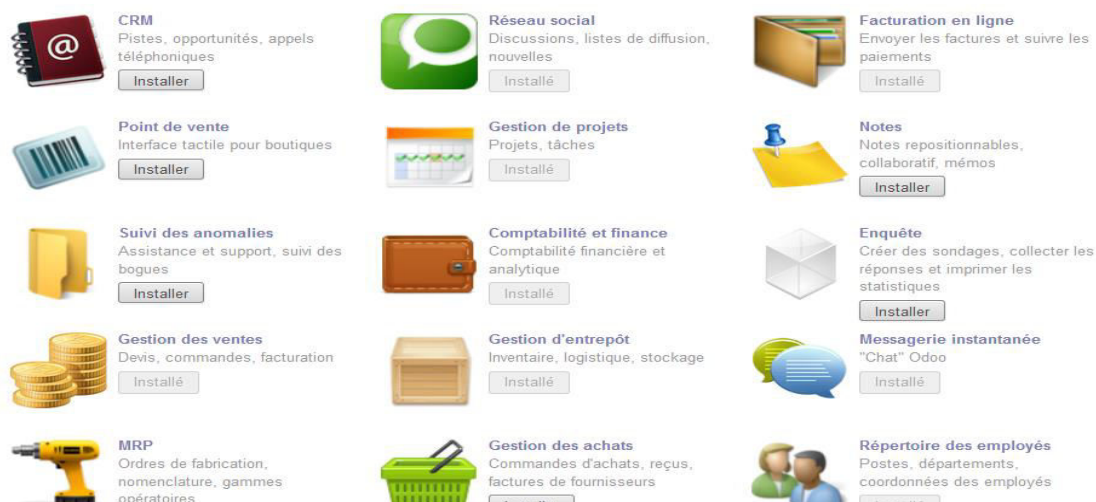


Figure 2 : Les modules locaux disponibles dans Odoo 8

Lors de la première installation, on installe le noyau d'Odoo avec un certain nombre de modules selon le profil d'installation choisit.

6. La gestion des stocks et ses objectifs

Le maintien d'un système de contrôle des stocks précis est vital pour toute entreprise. La précision des informations concernant le niveau des stocks est un prérequis critique pour l'ensemble des décisions subséquentes. Le traitement des informations recueillies sur les stocks permet au gestionnaire d'obtenir les informations pertinentes à l'égard des articles maintenus en stock (numéro de l'article, composants, poids, etc.). Dans une optique d'efficacité, les coûts liés à ce système de contrôle doivent être minimisés. La classification des stocks devient ainsi très importante.

L'objectif de la gestion des stocks est de satisfaire une demande, avec un taux de service élevé, et au moindre coût [8].

7. La structure logistique de stock dans Odoo

La chaîne logistique décrit les emplacements de stocks standards dans Odoo ainsi que la relation qui existe entre eux, dont chaque emplacement a un rôle bien précis [9] :

- **Entrepôt (Warehouse)**

Un entrepôt est le bâtiment où les articles sont stockés.

- **Emplacement de stock (Stock Location)**

Un emplacement est un espace spécifique dans l'entrepôt. Il peut être considéré comme une sous-localisation de l'entrepôt, ça peut être une étagère, un plancher, une allée, etc...

Par conséquent, un emplacement fait partie d'un seul entrepôt et il est impossible de relier un emplacement à plusieurs entrepôts. On peut avoir autant d'emplacements que possible dans un entrepôt. On peut avoir plusieurs types d'emplacements dans un entrepôt.

- **Emplacement fournisseur**

Un emplacement virtuel qui correspond à l'emplacement d'origine des produits issus des fournisseurs.

- **Vue**

Un emplacement virtuel utilisé pour créer des structures hiérarchiques pour l'entrepôt, en agréant ses emplacements enfants ; ne peut pas directement contenir de produits.

- **Emplacement interne**

L'emplacement physique dans l'entrepôt.

- **Emplacement client**

Un emplacement virtuel qui correspond à l'emplacement de destination des produits envoyés aux clients.

- **Perte de stock**

Un emplacement virtuel servant de contre-valeur aux opérations d'inventaire utilisées pour corriger les niveaux de stock (inventaires physiques).

- **Approvisionnement**

Un emplacement virtuel servant de contre-valeur temporaire pour les opérations d'approvisionnement lorsque la source (fournisseur ou fabrication) n'est pas encore connue. Normalement, cet emplacement est vide une fois que le planificateur d'approvisionnement a fini de s'exécuter.

- **Fabrication**

Un emplacement virtuel de contre-valeur pour les opérations de fabrication. Cet emplacement consomme des matières premières et fabrique des produits finis.

- **Emplacement de transit**

Un emplacement physique de contre-valeur à utiliser pour les opérations inter-entreprises et inter-entrepôts.

7.1 Concept de la chaine logistique dans l'entreprise

Les modules qui constituent la chaine logistique utilisée dans Odoo sont « Purchase Management », « Warehouse Management », « Sales Management » dont le scénario réel est achat, inventaire, vente.

- **Réception**

Le processus de la réception commence d'abord dans le module Achat par la création d'une demande de prix définie par une référence créée automatiquement, une date, un magasin, un fournisseur et des lignes d'achat. Après la confirmation de la commande, l'opération dans l'état prêt à transférer peut être consultée dans le module entrepôt.

- **Livraison**

Le processus de la livraison commence dans le module Vente par la création d'un Devis défini par une référence créée automatiquement, une date, un magasin et un client. Une fois la commande confirmée, une opération dans l'état en attente de disponibilité sera générée dans le module entrepôt. Après la vérification de la disponibilité, l'état de cette dernière devient prêt à transférer.

- **Transfert interne**

Le transfert interne s'effectue à partir du module entrepôt, en choisissant l'article, la quantité et l'emplacement de destination.

Dans les trois types de transfert (réception, livraison, transfert interne), on peut effectuer soit un transfert complet, soit un transfert partiel des articles, un reliquat est généré automatiquement dans ce cas.

- **Inventaire**

L'inventaire permet de corriger le stock théorique de l'ERP à partir des quantités observées sur le terrain (les entrepôts). Il est également possible de renseigner les emplacements des produits.

La figure suivante représente le tableau de bord du module entrepôt, c'est à partir de ce tableau qu'on peut faire les différents mouvements de stock.

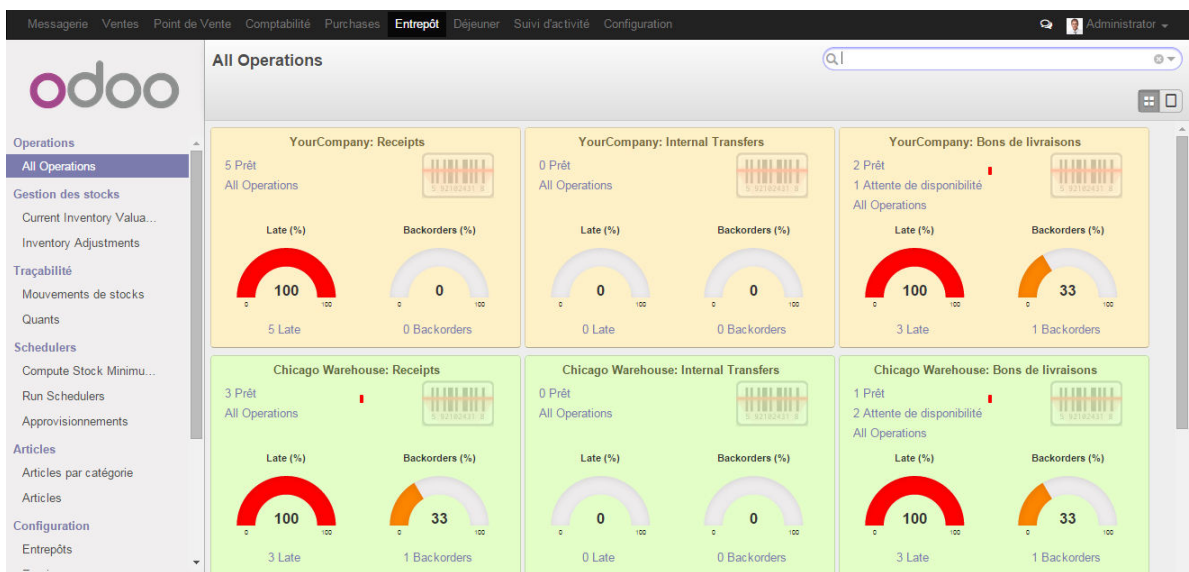


Figure 3 : Tableau de bord du module Entrepôt

8. Schéma des flux

Ce schéma décrit les emplacements de stocks standards du système Odoo de base, avec leurs parents respectifs (Société, Entrepôt, Emplacement parent). Les routes entre ces emplacements sont également spécifiées.

Deux entrepôts de la même société peuvent échanger des articles du stock en passant par une zone de transit, appartenant à la société, mais hors entrepôt.

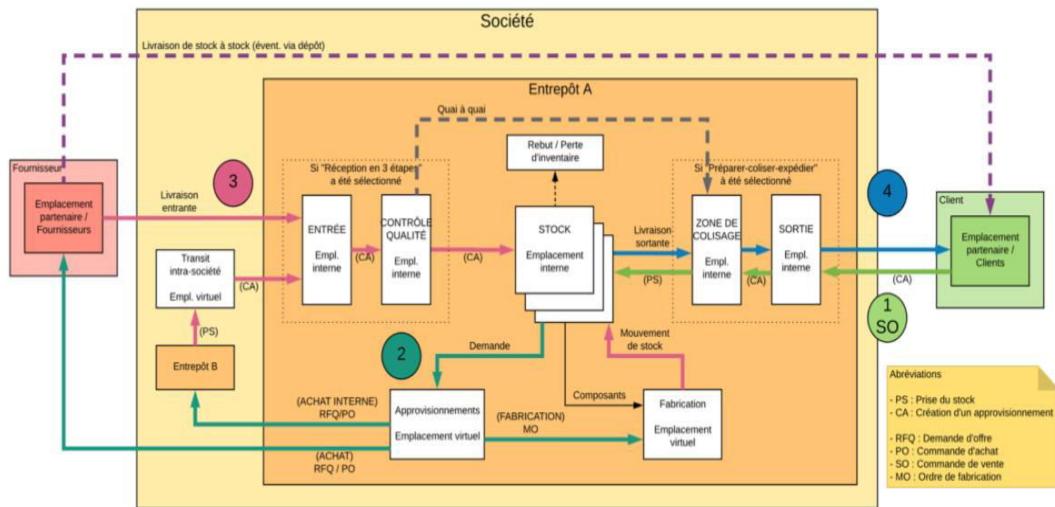


Figure 4 : Schéma des flux [26].

9. Conclusion

Nous avons abordé dans ce chapitre des généralités sur les ERP, les différents types ERP open source et plus particulièrement l'ERP Odoo. Aussi, nous avons donné une vue générale sur la gestion de stock ainsi que ses principales activités.

*Chapitre 2 : Analyse
et Conception du
système*

1. Introduction

L'étape de conception est très importante pour la réussite d'un projet informatique car elle permet de rendre le développement plus fidèle aux besoins du client. Elle vise à définir une feuille de route du projet, le concevoir et le valider avant de passer au développement du système. Elle permet aussi d'avoir une bonne réflexion avant de passer à l'action, une bonne organisation du travail et une bonne communication entre les différents intervenants dans le projet.

Dans le présent chapitre, nous avons commencé par présenter la société dans laquelle nous avons effectué notre stage, après nous nous sommes consacrés à l'étape de la conception qui va contenir d'une part la planification du projet à l'aide d'un diagramme de Gantt, et d'autre part les diagrammes UML pour présenter les cas d'utilisation, les diagrammes de séquence et le diagramme de classe.

2. Présentation de la société

La société SOGESI (Société de Gestion des Solutions Informatique), créée en 2013 spécialisée dans l'intégration des solutions Odoo et le développement des sites web, elle propose aux entreprises des solutions durables et efficaces pour répondre à leurs problématiques [15].

2.1 Les services de la société

- ✓ Odoo et PowerBI.
- ✓ ERP et solutions pour entreprises.
- ✓ Formation Odoo technique et fonctionnelle.
- ✓ Solutions cloud.
- ✓ Développement web.

2.2 Les solutions de la société

- ✓ Formation technique Odoo.
- ✓ Gestion de la maintenance assistée par ordinateur (GMAO).
- ✓ Gestion du dossier d'importation.
- ✓ Gestion de la paie algérienne.
- ✓ Gestion de la comptabilité algérienne.
- ✓ Gestion des ressources humaines.

- ✓ Gestion des ventes et achats.
- ✓ Gestion des stocks.
- ✓ Gestion du métier transitaire.
- ✓ Gestion d'une flotte automobile.
- ✓ Gestion d'une clinique médico-chirurgicale.

2.3 Les clients de la société :



Figure 5 : La liste des clients pour la société SOGESI [15].

3. Méthodologie de travail

La nature du projet incite à suivre une méthode agile et plus précisément la méthode Scrum. En effet les spécifications du projet n'ont pas cessé d'être modifiées tout au long du projet. Scrum permet d'entamer le projet avec un minimum d'informations sur les travaux à faire et qui seront ajustées régulièrement au cours du projet. Le cycle de vie de la méthode Scrum se décompose en plusieurs Sprint successifs.

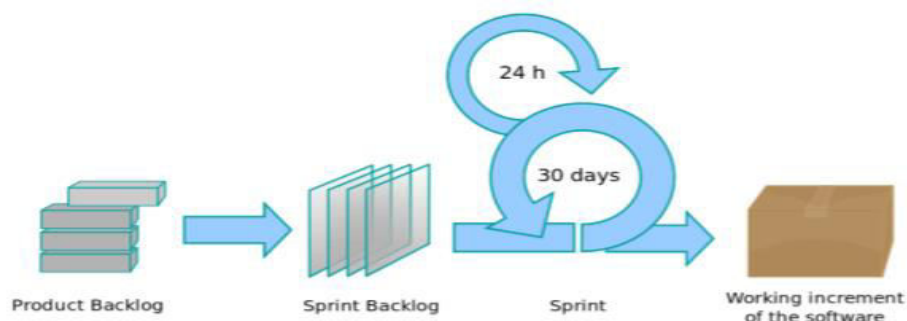


Figure 6 : Exemple de développement en Scrum [27].

4. Planification de projet

Le diagramme de Gantt, couramment utilisé en gestion de projet, est l'un des outils les plus efficaces pour représenter visuellement l'état d'avancement des différentes activités (tâches) qui constituent un projet. La colonne de gauche du diagramme énumère toutes les tâches à effectuer, tandis que la ligne d'en-tête représente les unités de temps les plus adaptées au projet (jours, semaines, mois etc.). Chaque tâche est matérialisée par une barre horizontale, dont la position et la longueur représentent la date de début, la durée et la date de fin. Ce diagramme permet donc de visualiser d'un seul coup d'œil :

- Les différentes tâches à envisager
- La date de début et la date de fin de chaque tâche
- La durée escomptée de chaque tâche
- Le chevauchement éventuel des tâches, et la durée de ce chevauchement
- La date de début et la date de fin du projet dans son ensemble

En résumé, un diagramme de Gantt répertorie toutes les tâches à accomplir pour mener le projet à bien, et indique la date à laquelle ces tâches doivent être effectuées (le planning) [10].

Le planning constitué va permettre de réaliser le suivi de projet durant son déroulement. Il sera ainsi possible de déterminer si les objectifs fixés sont atteints, de suivre l'avancement au travers d'un tableau de bord de suivi de projet et enfin de contrôler l'affectation des ressources à la réalisation des différentes tâches.

5. Spécification des besoins

Les gestionnaires de stock éprouvent beaucoup de difficultés à accomplir leurs opérations de stockage, déstockage et inventaires de marchandises. Aujourd'hui tout est fait à la main, d'où le recensement d'énormes inconvénients principalement la perte de temps dans le traitement des informations et les erreurs commises lors de la gestion des opérations de stockage et de déstockage.

L'acteur impliqué dans l'application est l'opérateur.

Les besoins définis principalement par l'entreprise SOGESI sont :

- **Gestion des réceptions**

Lorsque l'on souhaite stocker la marchandise, l'opérateur imprime le bon de réception à partir du module « entrepôt » dans Odoo. Ce bon spécifié par son nom unique, contient les articles achetés avec leurs quantités.

L'opérateur cherche le bon de réception imprimé en saisissant son nom manuellement ou vocalement dans l'application.

L'opérateur scanne ensuite le code à barre de chaque article.

En cas de réception partielle, l'opérateur crée un reliquat pour qu'il puisse indiquer la quantité reçue réellement.

- **Gestion des livraisons**

Lorsque l'on souhaite déstocker la marchandise, c'est le même principe que les réceptions sauf que le partenaire c'est un client et non un fournisseur.

L'opérateur imprime le bon de livraison à partir du module « entrepôt » dans Odoo. Ce bon spécifié par son nom unique, contient les articles vendus avec leurs quantités.

L'opérateur cherche le bon de livraison imprimé en saisissant son nom manuellement ou vocalement dans l'application.

L'opérateur scanne ensuite le code à barre de chaque article, si la quantité disponible dans le stock n'est pas suffisante, on doit pouvoir déstocker uniquement les quantités disponibles, c'est le cas d'une livraison partielle.

L'opérateur crée un reliquat pour qu'il puisse modifier la quantité (le reliquat représente les quantités réellement livrées).

- **Gestion des transferts internes**

Lorsque l'on souhaite faire un transfert de la marchandise d'un dépôt A vers un dépôt B, l'opérateur imprime le bon de transfert à partir du module « entrepôt » dans Odoo. Ce bon spécifié par son nom unique, contient les articles transférés avec leurs quantités.

L'opérateur cherche le bon de transfert imprimé en saisissant son nom manuellement ou vocalement dans l'application.

L'opérateur scanne ensuite le code à barre de chaque article. Si la quantité disponible dans le stock n'est pas suffisante, on doit pouvoir transférer uniquement les quantités disponibles, c'est le cas d'un transfert interne partiel.

L'opérateur crée un reliquat pour qu'il puisse modifier la quantité (le reliquat représente les quantités réellement transférées).

- **Gestion des inventaires**

L'opérateur doit pouvoir réaliser des ajustements de stock sur son dépôt. L'opérateur consulte la liste des inventaires avec une possibilité d'ajustement de stock en cas de nécessité.

- **Gestion de la traçabilité**

L'opérateur consulte la liste des numéros de série avec une possibilité de modification de ce numéro en cas de nécessité.

6. Conception

Afin de répondre aux besoins et aux exigences initiales décrites dans le cahier des charges, une étude conceptuelle est élaborée sous forme de diagrammes UML (Unified Modeling Language) [12] qui est un langage de modélisation orientée objet développée pour fournir une méthode normalisée pour visualiser la conception d'un système.

6.1. Schéma illustratif d'une réception partielle

Le schéma suivant illustre le cas d'une réception partielle (le cas où la quantité des produits reçus réellement est inférieure à la demande initiale). Ce même schéma peut être appliqué pour les trois types de transfert (réceptions, livraisons, transferts internes).

L'opérateur peut consulter la liste de tous les bons de réception. Il peut effectuer une recherche soit en saisissant le nom du bon concerné, soit par reconnaissance vocale. Les détails de chaque bon peuvent être consultés, ou le bon contient tous les produits commandés. L'opérateur peut générer un reliquat pour saisir la quantité de l'article réellement reçue.

Une fois l'article scanné, sa quantité peut être consultée et modifiée.

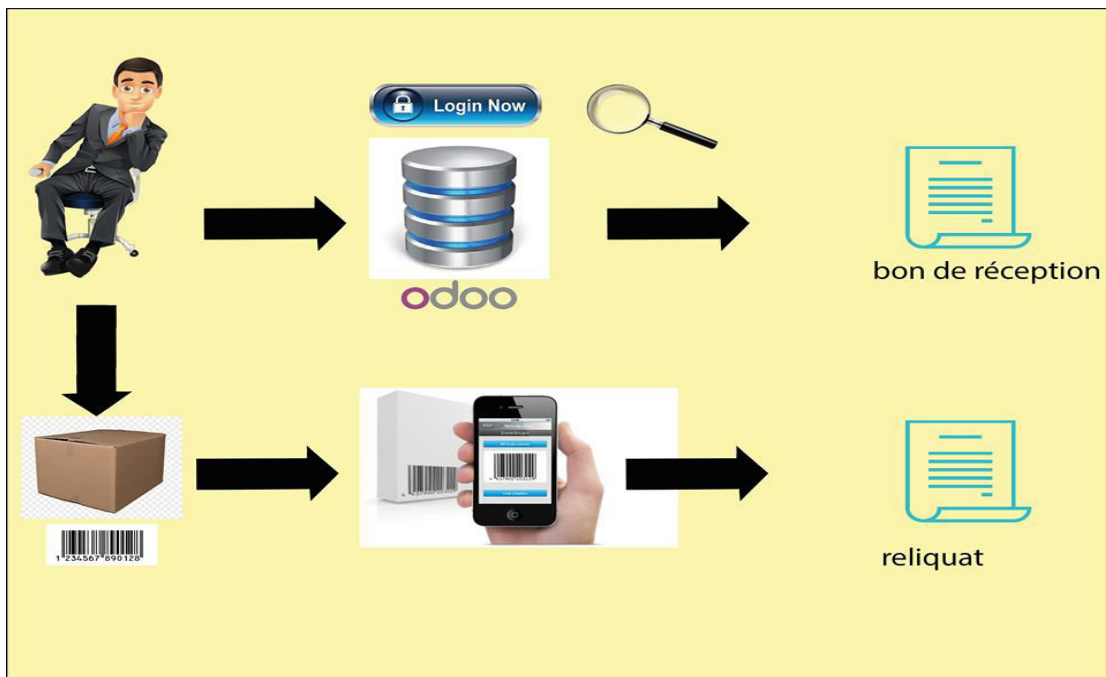


Figure 8 : Schéma illustratif d'une réception partielle

6.2. Diagramme de cas d'utilisation

Un cas d'utilisation est la description d'un ensemble de séquences d'actions qu'un système effectue pour produire un résultat observable à un acteur. Un cas d'utilisation représente une exigence fonctionnelle du système dans son ensemble. Les diagrammes de cas d'utilisation décrivent ce qu'un système fait du point de vue d'un observateur externe [11].

6.2.1. Cas d'utilisation « gestion de stock »

Ce diagramme représente les fonctionnalités principales que peut effectuer un opérateur dont la phase d'authentification est nécessaire avant toute action. L'opérateur a la possibilité de gérer les réceptions, les livraisons, les transferts internes comme il peut faire des ajustements de stock dans les inventaires. La traçabilité est réalisée en utilisant la technologie de code à barre.

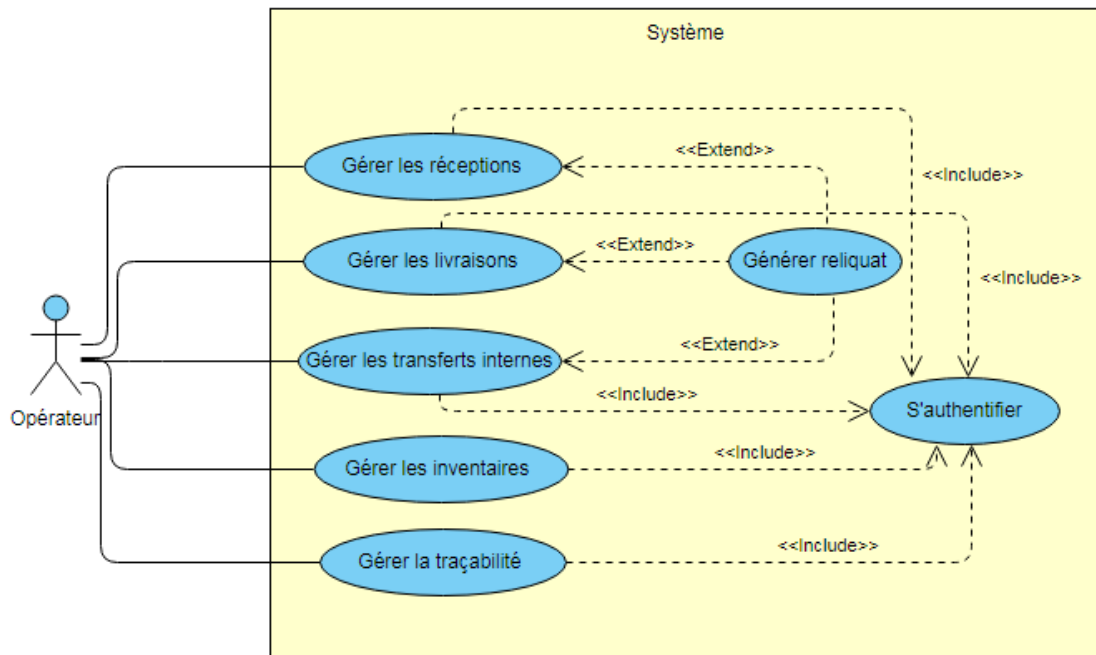


Figure 9 : Diagramme de cas d'utilisation « gestion de stock »

6.3. Diagrammes de séquence

Un diagramme de séquence est un diagramme UML (Unified Modeling Language) d'interaction mettant l'accent sur la chronologie de l'envoi des messages entre objets. Un diagramme de séquence comprend un groupe d'objets, représentés par des lignes de vie, et les messages que ces objets échangent lors de l'interaction [12].

Les diagrammes des cas d'utilisations « gérer les réceptions », « gérer les livraisons » et « gérer les transferts internes » possèdent le même principe et se déroulent de la même façon. Pour cette raison, nous avons décidé de montrer l'un d'entre eux.

6.3.1. Gérer les réceptions

Ce diagramme explique les étapes pour effectuer un mouvement de réception:
l'opérateur doit :

- se connecter
- saisir le nom du bon pour trouver le bon désiré (la recherche peut se faire manuellement ou vocalement), le nom du bon s'affiche par conséquent.
- L'opérateur peut consulter les détails du bon et peut afficher ensuite les articles associés en scannant article par article.
- Dans le cas où la quantité reçue de chaque article est égale à la quantité affichée dans le bon, l'opérateur enregistre la réception.
- Dans le cas contraire, c-à-d le cas de transfert partiel : la quantité reçue de chaque article est inférieure à la quantité affichée dans le bon, l'opérateur génère un reliquat et modifie la quantité de chaque article.

La figure suivante montre le diagramme de séquence du cas d'utilisation « gérer les réceptions »

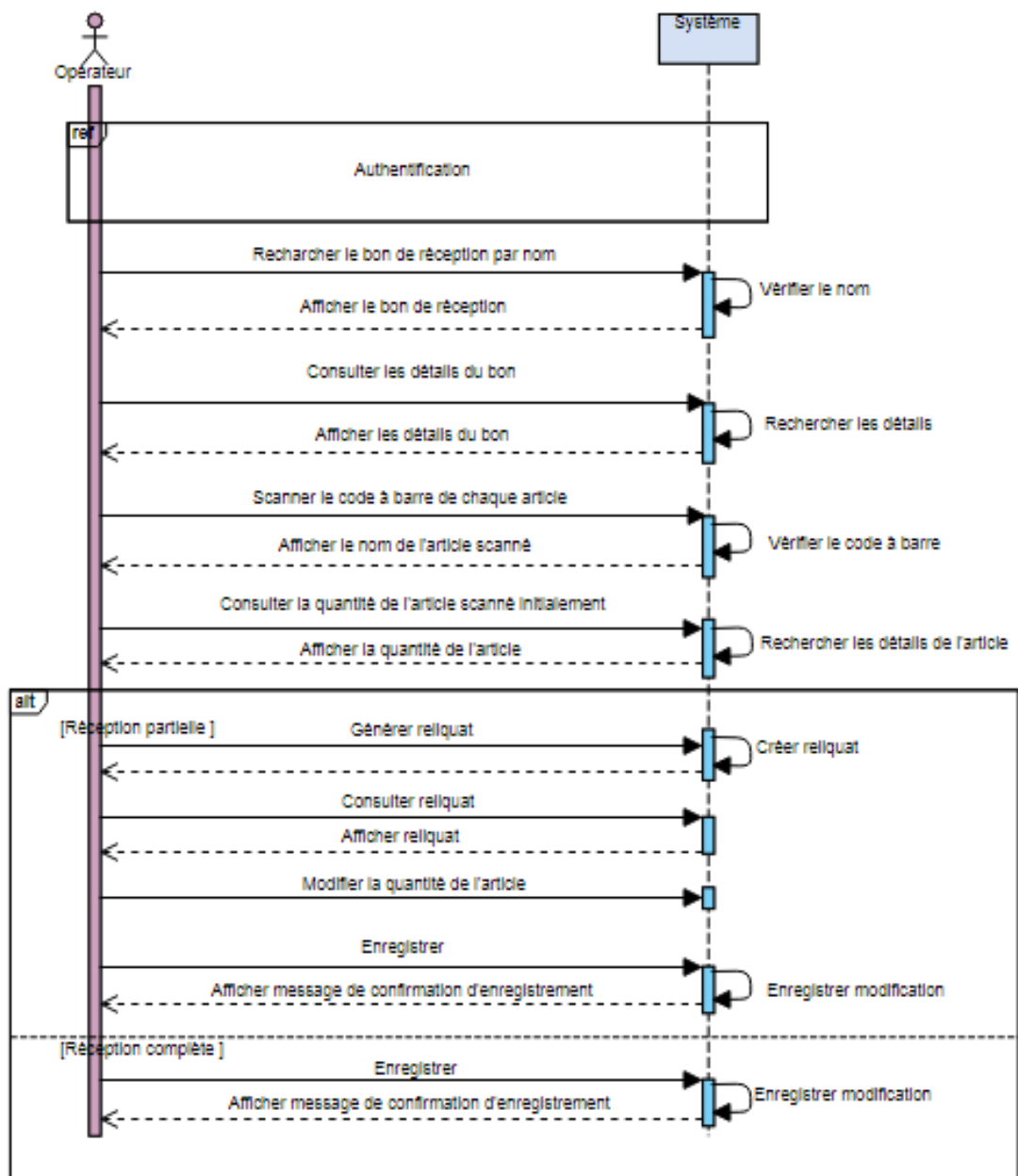


Figure 10 : Diagramme de séquence du cas « gérer les réceptions »

6.3.2 Authentification

Le diagramme d'authentification explique la référence mentionnée dans le diagramme de séquence précédant. Avant toute action, l'opérateur doit d'abord s'identifier en saisissant le nom de l'utilisateur, le mot de passe, le nom de la base de données et l'adresse IP de l'ordinateur sur lequel se trouve le serveur. Si les données sont valides le système autorise l'opérateur à accéder à l'application, sinon l'accès est refusé et l'opérateur est obligé de ressaisir les bonnes informations pour se connecter.

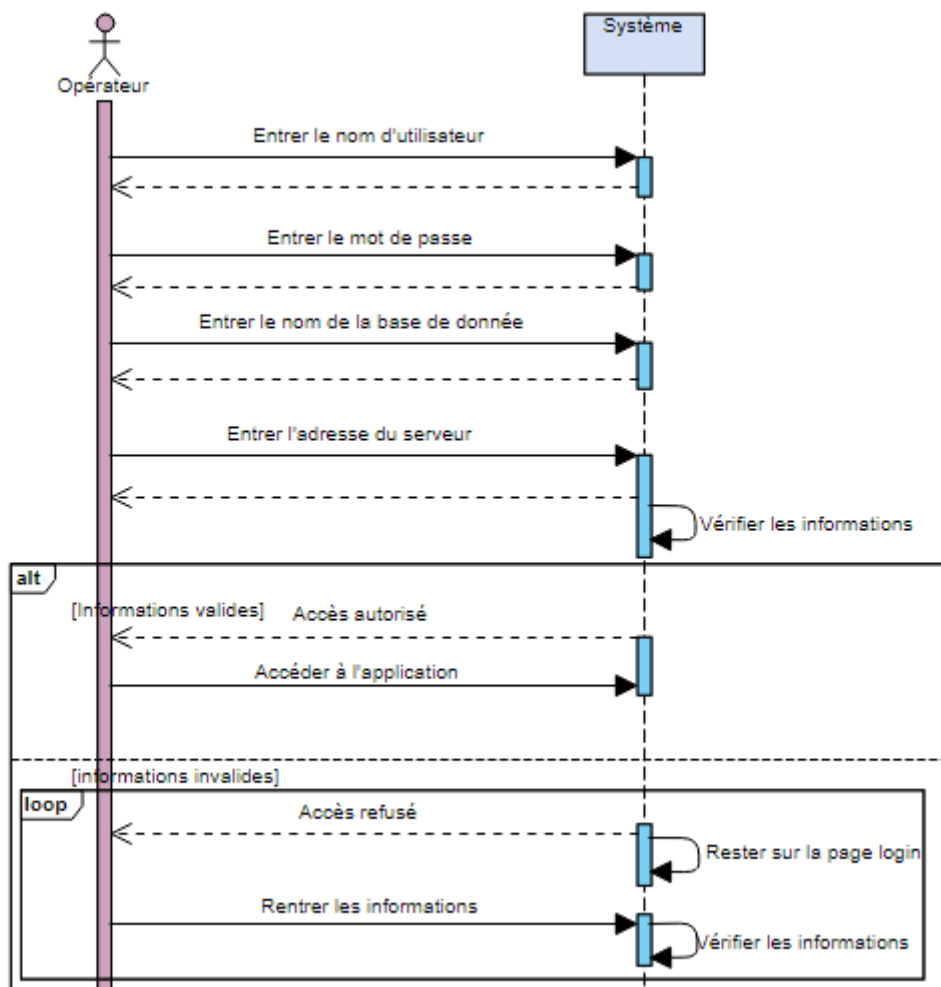


Figure 11 : Diagramme de séquence du cas « s'authentifier »

6.4 Diagramme de classe

Dans le langage UML, les diagrammes de classes appartiennent à l'un des six types de diagramme structurel. Un diagramme de classes est fondamental pour le processus de modélisation des objets, il modélise la structure statique d'un système. Suivant la complexité d'un système, il est possible d'utiliser un seul diagramme de classes pour modéliser un système complet ou bien il est possible d'utiliser différents diagrammes de classes pour modéliser les composants d'un système [13].

Le diagramme suivant montre les classes impliquées dans la conception de l'application, les classes colorées en bleu représentent les modèles existants dans Odoo, tandis que les classes restantes, sont des classes impliquées dans le fonctionnement de l'application.

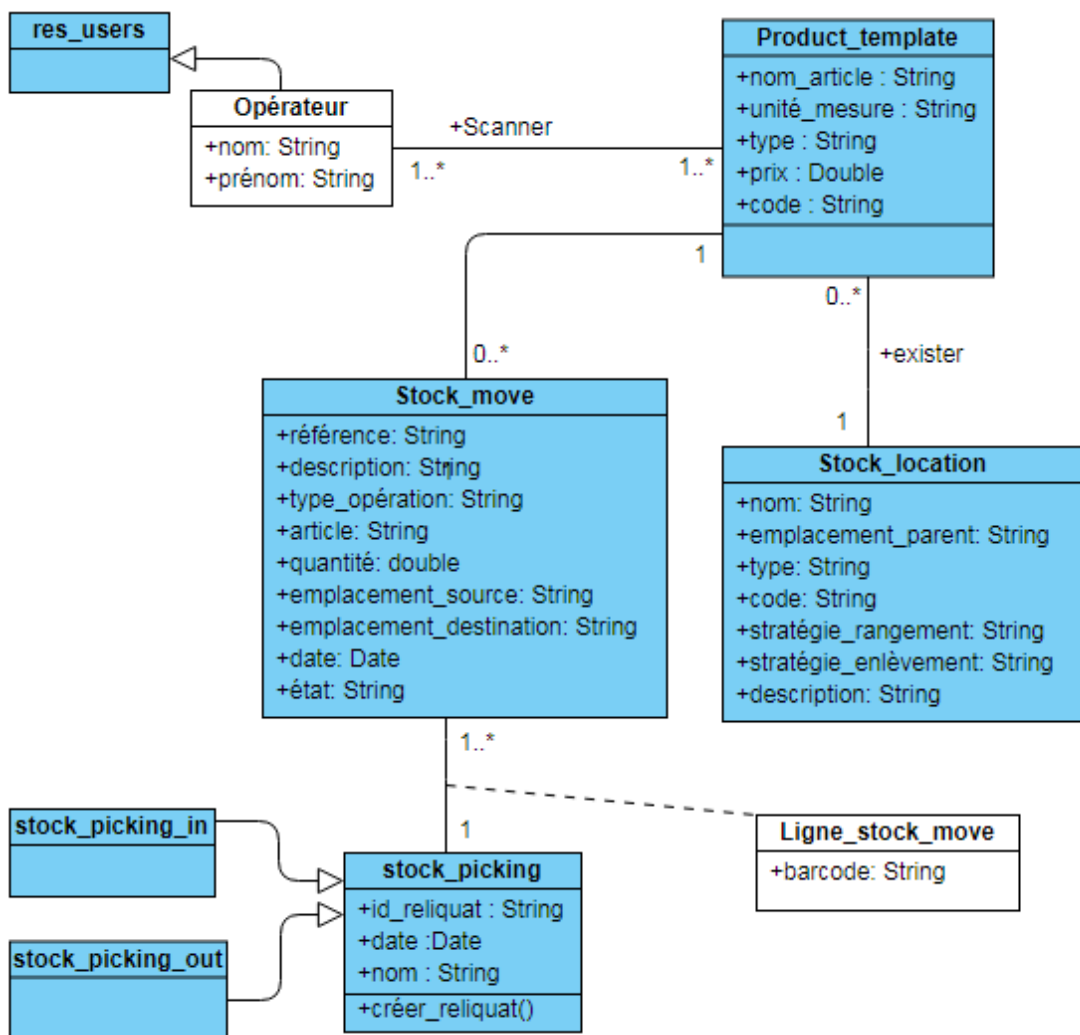


Figure 12 : Diagramme de classe

6.5 Diagramme de navigation

UML offre la possibilité de représenter graphiquement cette activité de navigation dans l'interface en produisant des diagrammes dynamiques. On appelle ces diagrammes des diagrammes de navigation [14].

Il a pour rôle de modéliser la navigation entre les différents liens des pages web. Il représente de manière formelle l'ensemble de chemins possibles entre les principales pages proposées à l'utilisateur, bref, l'aspect dynamique de l'interface utilisateur.

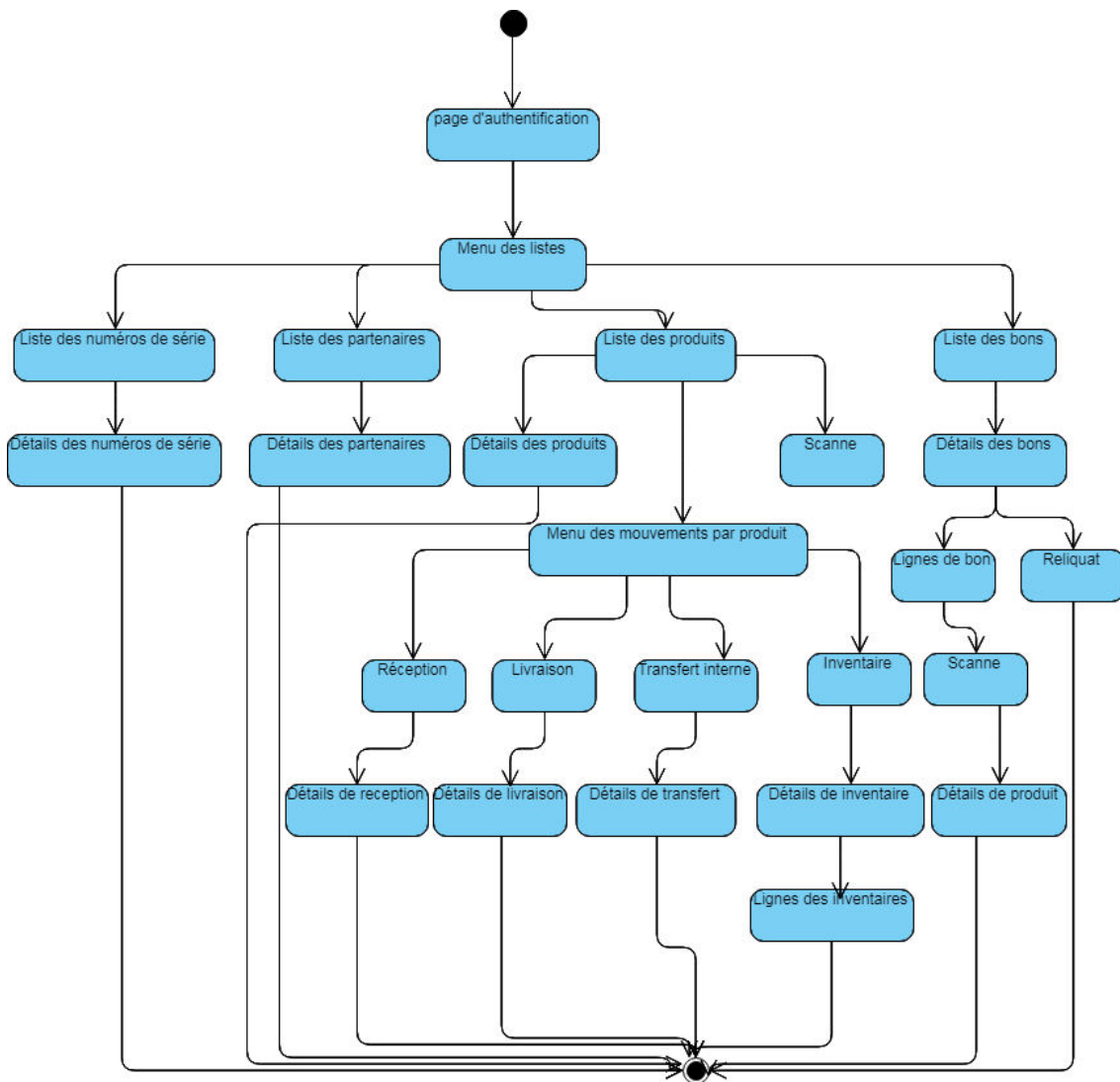


Figure 13 : Diagramme de navigation

7. Conclusion

Ce chapitre a abordé la phase de conception qui a un rôle primordial dans la compréhension des objectifs et des fonctionnalités de l'application. Nous avons commencé par recenser les besoins de l'application, pour aller par la suite à la représentation des différents diagrammes qui permettent l'interaction entre les différents composants du système. Le chapitre suivant sera consacré à la réalisation de la dite application.

Chapitre 3 : *Réalisation*

1. Introduction

Le projet consiste à concevoir et à développer une application mobile sous Android destinée à toutes les entreprises pour la bonne gestion de stock.

Pour bien mener ce travail, il est judicieux de suivre une démarche structurée pour produire des solutions informatiques de qualité répondant aux besoins des utilisateurs tout en respectant les délais.

Dans ce chapitre nous présentons la méthodologie de travail utilisée pour réaliser le projet ainsi que les différents outils utilisés pour l'implémentation de l'application. Nous abordons par la suite les différentes fonctionnalités de notre application et nous proposons des aperçus de celle-ci.

2. Étude technique

L'étude technique met en relief les principales plates-formes utilisées et les différentes techniques utilisées pour la réalisation de cette application.

2.1 Odoo



Fondé en 2005 en Belgique par Fabien Pinckaers, Odoo anciennement OpenERP et Tiny ERP, est un progiciel open source de gestion intégré qui contient un grand nombre de modules qui permettent de simplifier la gestion d'entreprise en générale [16].

Figure 14: Odoo logo

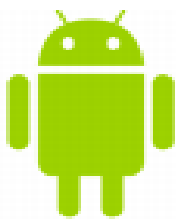
2.2 Service Web

Au sens large, les web services sont des systèmes logiciels, permettant l'interopérabilité entre plusieurs systèmes logiciels (agents) sur un réseau informatique. Les web services sont normalisés car ils utilisent les standards XML et HTTP pour transférer des données et sont compatibles avec de nombreux autres environnements de développement. Ils sont indépendants des plates-formes [17]. « XML-RPC » est un type de service web.

XML-RPC :

XML-RPC est un protocole simple utilisant XML pour effectuer des messages RPC. Les requêtes sont écrites en XML et envoyées via HTTP POST. Les requêtes sont intégrées dans le corps de la réponse HTTP. XML-RPC est indépendant de la plate-forme, ce qui lui permet de communiquer avec diverses applications. Par exemple, un client Java peut parler de XML-RPC à un PerlServer [17].

2.3 Android



Android est un système d'exploitation mobile basé sur le noyau Linux, développé et lancé par Google en 2005. Celui-ci met à disposition un kit de développement complet (SDK), il s'appuie sur deux piliers langage java et le Sdk [18].

Figure 15 : Android logo

3. Environnements de développement

3.1 Android Studio



Android Studio est l'environnement de développement que Google propose à ses développeurs pour créer des applications Android. Android Studio permet principalement d'éditer les fichiers Java et les fichiers de configuration d'une application Android [19].

Figure 16 : Android studio logo

3.2 PostgreSQL



PostgreSQL est un système de gestion de bases de données relationnelles objet (ORDBMS) basé sur POSTGRES, Version 4.2, développé à l'université de Californie au département des sciences informatiques de Berkeley [20].

Figure 17 : PostgreSQL logo

3.3 Visual Paradigm



Visual Paradigm est un logiciel de création de diagrammes dans le cadre d'une programmation. Tout en un, il possède plusieurs options permettant une large possibilité de modélisation en ULM. [21].

Figure 18 : Visual Paradigm logo

4. Les langages de développement utilisés

- **Python :** Python est un langage de programmation de haut niveau, orienté objet, totalement libre et terriblement efficace, conçu pour produire du code de qualité, portable et facile à intégrer. Ainsi la conception d'un programme Python est très rapide et offre au développeur une bonne productivité [22].
- **XML :** Le XML ou eXtensible Markup Language est un langage informatique de balisage générique, qui permet de structurer des données afin qu'elles soient lisibles aussi bien par les humains que par des programmes de toute sorte. Il est souvent utilisé pour faire des échanges de données entre un programme et un serveur ou entre plusieurs programmes [23].

- **Java Android :** Java est un langage de programmation orienté objet, interprété et compilé, créé par James Gosling et Patrick Naughton et présenté le 23 mai 1995 [24].

5. Synchronisation de l'application mobile avec la base de données Odoo

L'application mobile est synchronisée à Odoo via le service web grâce au protocole XML-RPC.

Pour pouvoir accéder au service XML-RPC fourni par Odoo, nous devons pouvoir envoyer une requête et recevoir une réponse au protocole XML-RPC, pour faire cela, la bibliothèque XML-RPC est téléchargée à partir de github dans le lien suivant : <https://github.com/gturri/aXMLRPC/tree/master/src>

Le dossier décompressé de la bibliothèque est copié dans le chemin app/src/main/java de l'application créée dans Android studio.

6. Développement de l'application mobile « Gestion De Stock »

C'est une application mobile pour les appareils Android, dédiée aux entreprises pour leur faciliter le processus de gestion de stock, son but principal est de minimiser les erreurs commises lors de la saisie des informations. Cette application est basée sur la technologie de code à barre qui est un système de traçabilité très apprécié par les entreprises.

L'application permet de scanner les codes à barres des articles à travers l'appareil photo du smart phone, cette méthode facilite les mouvements de stock.



Figure 19 : Scan du code à barre d'un produit à travers l'appareil photo

6.1 Modification du module « Gestion d'entrepôt »

Afin de faciliter le développement de l'application, deux modules se sont ajoutés et installés dans Odoo.

Le premier module s'appelle « Stock Picking Backorder », téléchargé à partir des applications Odoo. Il permet de générer un reliquat en cliquant sur le bouton « Backorder » et sans passer par le wizard.

Le reliquat est utilisé dans le cas où la quantité des produits reçus réellement est inférieure à la demande initiale, c'est le cas d'une réception partielle.

Le document créé dit reliquat, représente la quantité réellement reçue.

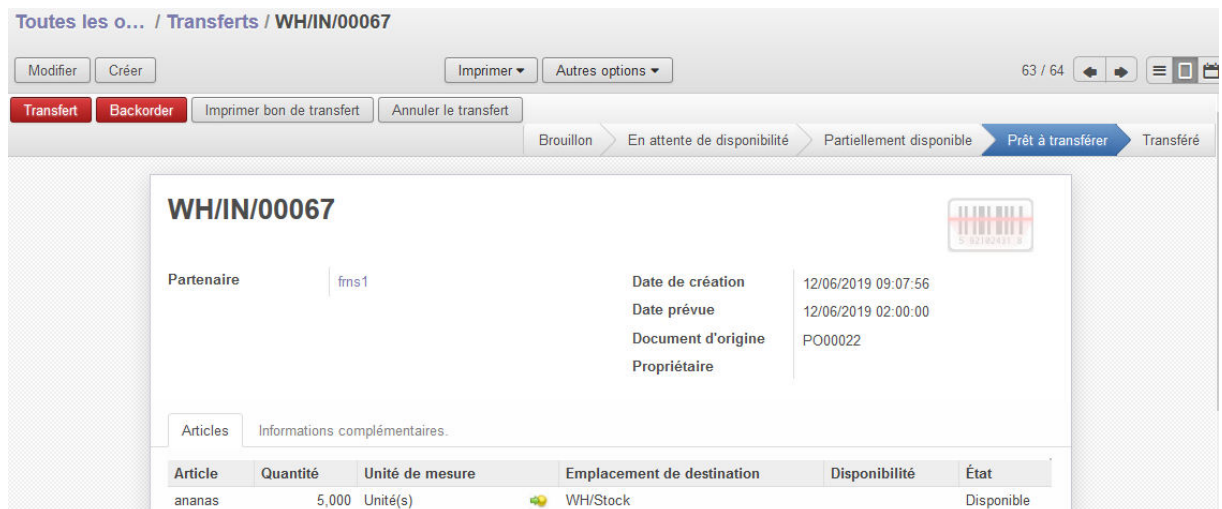


Figure 20 : Création du reliquat par l'intermédiaire du bouton « backorder »

Le second module développé dans Odoo permet d'ajouter le champ code à barre des articles dans les lignes des bons de réception, de commande et de transfert. Ce module permet de faire apparaître la ligne du bon une fois le produit scanné, c'est-à-dire que lorsque l'on scanne le code à barre d'un produit, l'application prends le résultat du scan et fait une requête pour trouver le produit correspondant et par conséquent affiche ses informations.

6.2 Interfaces de l'application sous Android

La figure 21 montre l'interface qui permet l'authentification de l'opérateur en saisissant le nom d'utilisateur, le mot de passe, le nom de la base de données et l'adresse IP de la machine où se trouve le serveur Odoo.

Cette interface permet la synchronisation avec la base de données Odoo et l'accès à l'application en cliquant sur le bouton « Login Now ».

L'application va faire des requêtes en interrogeant la base de données Odoo. Les informations seront affichées et traitées de telle sorte que tous les traitements effectués à travers l'application mobile seront modifiés automatiquement dans la base de données Odoo.

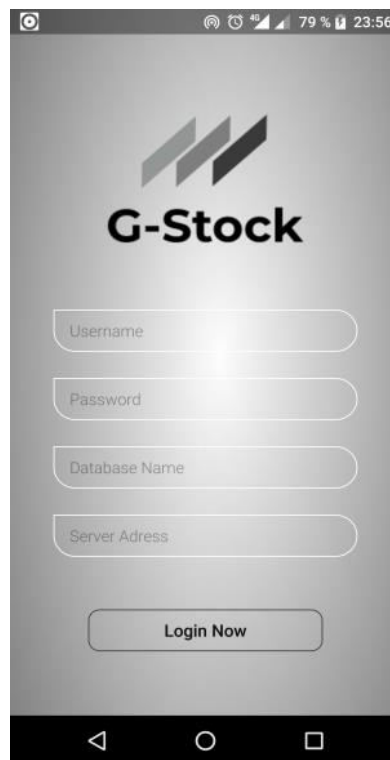


Figure 21 : l'interface de l'authentification

La figure 22 représente l'interface qui apparaît après l'authentification. Elle est composée de quatre icônes et d'un menu pour des enregistrements vocaux. Chaque icône conduit vers une liste (Partenaires, Produits, Bons, N° série). L'explication de chaque icône se trouve dans les figures 27, 28, 29 et 30 respectivement.

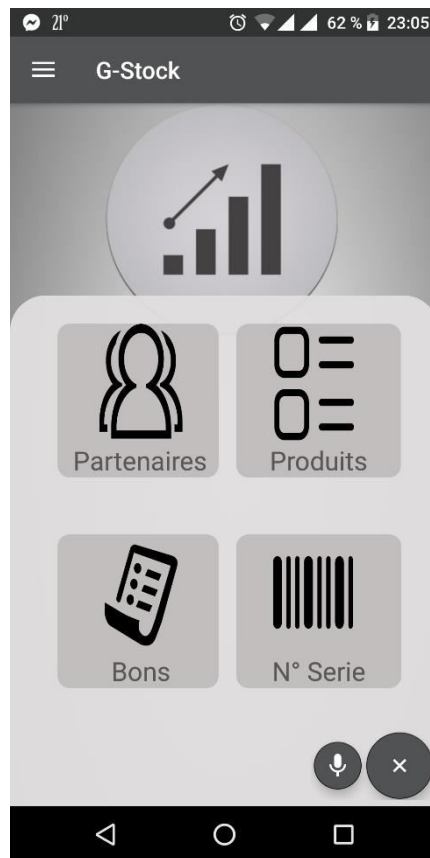


Figure 22 : Menu des listes

Le menu d'enregistrement vocal permet de sauvegarder des remarques vocalement au lieu de les noter par stylo.

Les 3 traits en haut à gauche permettent d'ouvrir un autre menu qui est un endroit privilégié pour placer certaines fonctionnalités tout en économisant notre précieux espace dans l'interface. Ce menu peut être aussi affiché en balayant l'écran vers la droite.

La figure 23 montre le menu qui contient des raccourcis vers le Menu des mouvements montré dans la **figure 24** et vers le formulaire de chaque opération de transfert par produit séparément (**figure 25**) et aussi un accès direct vers la liste des inventaires (**figure 26**).

Le nom d'utilisateur et le nom de la base de données connectée s'affichent en haut de l'interface à gauche.

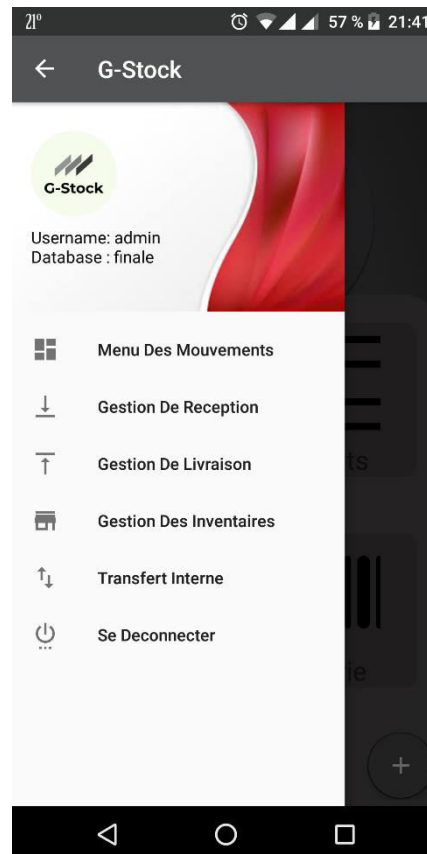


Figure 23 : Navigation menu

La figure 24 représente un tableau de bord composé de quatre parties (réceptions, livraisons, transferts internes, inventaires). L'utilisateur peut scroller horizontalement et choisir la partie désirée.

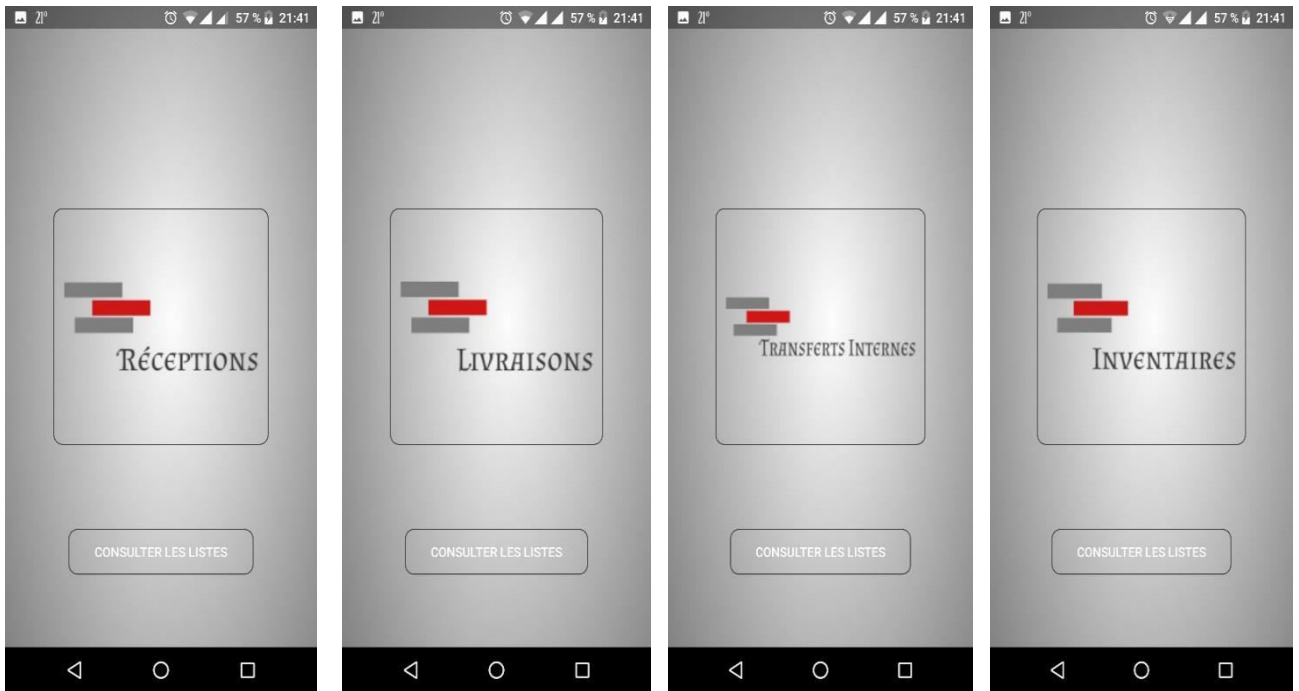


Figure 24 : Menu des mouvements

Les parties réceptions, livraisons et transferts internes concernent les mouvements de stock par produit tandis que la partie inventaire représente les ajustements de stock.

Lorsqu'on clique sur l'une des trois premières parties (réceptions, livraisons, transferts internes), un formulaire pour renseigner les détails du mouvement va apparaître (**figure 25**).

La partie inventaires conduit vers la liste et les détails correspondants, l'utilisateur peut par la suite consulter les lignes de l'inventaire et ajuster le stock (**figure 26**).

La figure 25 montre le formulaire qui donne la main à l'utilisateur pour entrer le nom et la quantité du produit et d'enregistrer le mouvement par la suite.



Figure 25 : Les mouvements

La figure 26 représente la liste des inventaires, l'utilisateur peut consulter les détails de chaque inventaire et faire des ajustements de stock.

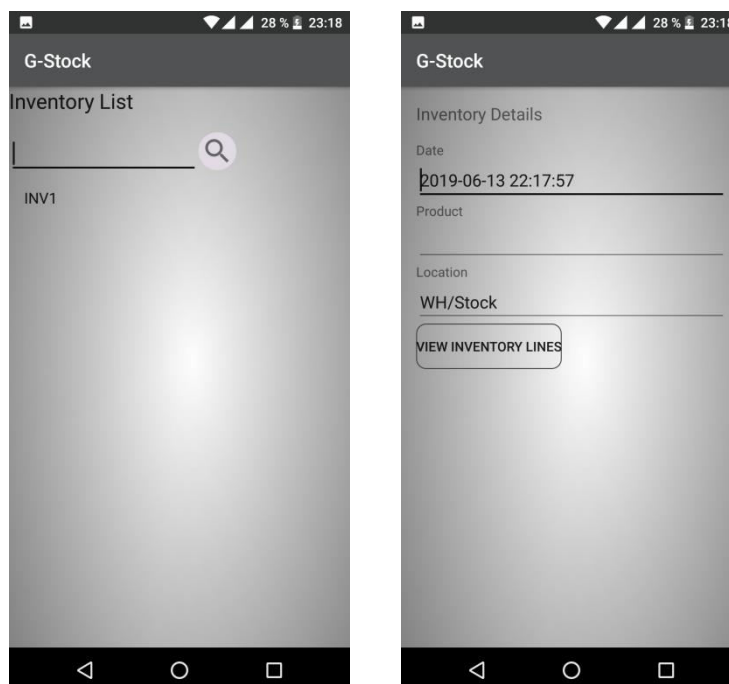


Figure 26 : La liste d'inventaires avec les détails

La figure 27 montre la liste des partenaires avec leurs détails.

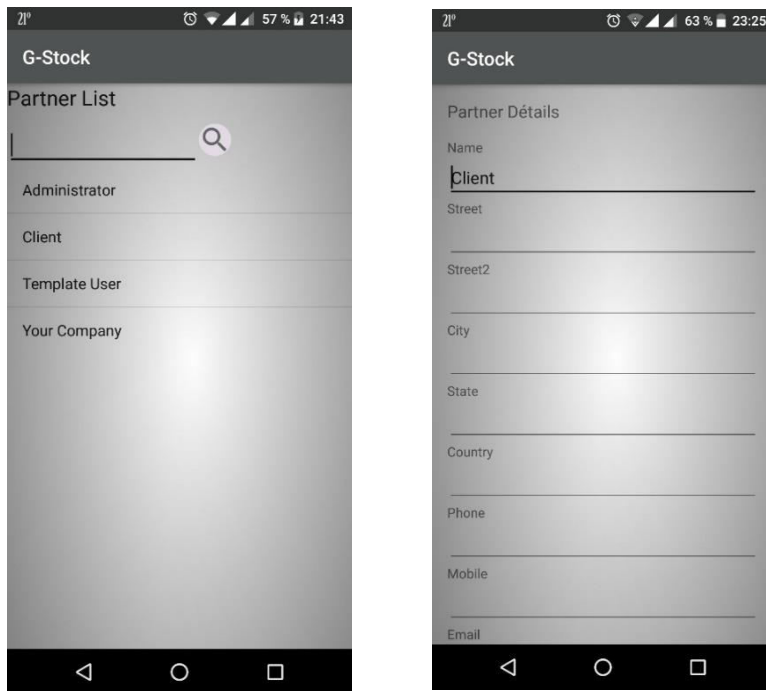


Figure 27 : Interface des listes et détails des partenaires

La figure 28 montre la liste des produits recherchés soit par une recherche manuelle du nom ou par scan du code à barre du produit. Pour faire apparaître les détails, il suffit de cliquer sur le produit.

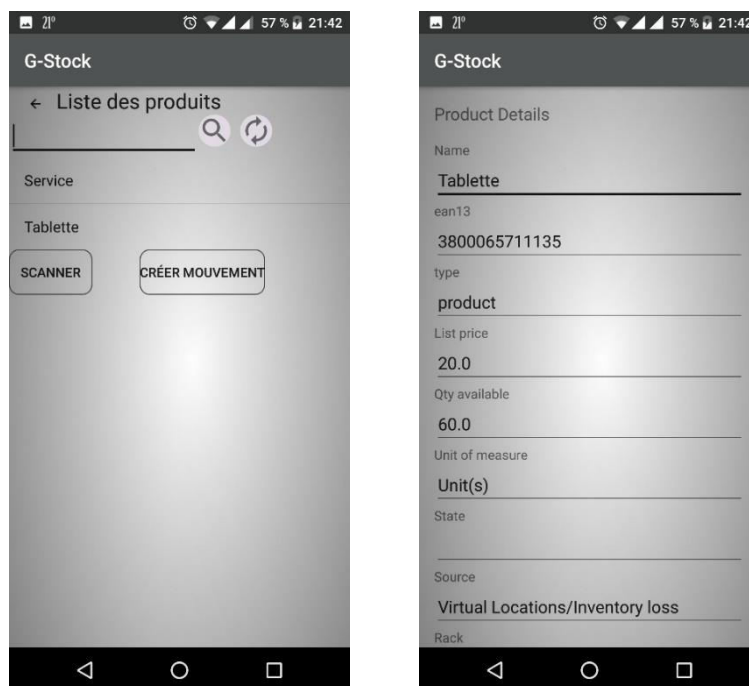


Figure 28 : La liste et détails des produits

La figure 29 représente la liste de tous les bons de réception, de livraison et de transfert interne, la recherche peut être effectuée soit manuellement en saisissant le nom du bon soit par reconnaissance vocale. Les détails de chaque bon peuvent être consultés. Le bouton « backorder » permet de générer un reliquat en cas de transfert partiel.

Une fois l'article scanné, sa quantité peut être consultée et modifiée.

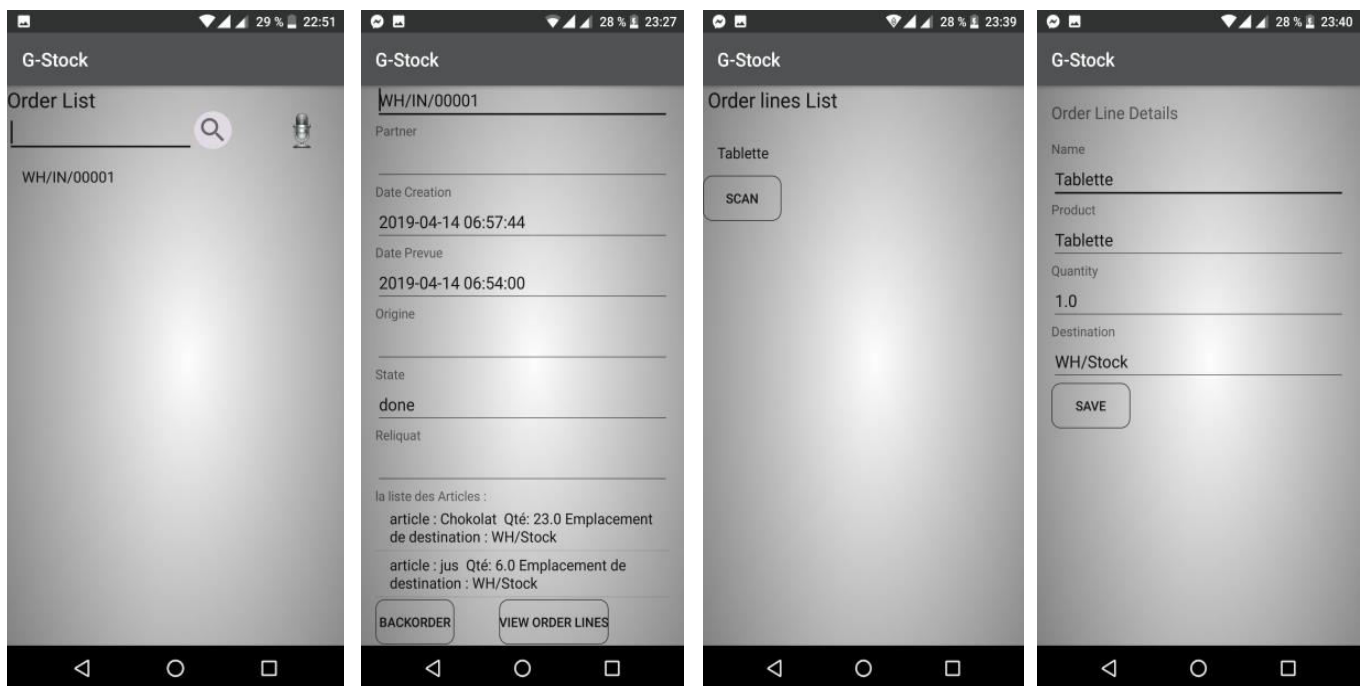


Figure 29 : Liste des bons avec détails

A la différence de la figure 25, ces mouvements sont donc effectués selon la nature du bon, par exemple :

Si le bon recherché est un bon de réception, cela veut dire que l'on vient d'acheter des articles dont la quantité est mentionnée sur ce bon. L'opérateur n'a qu'à comparer donc la quantité reçue réellement avec la quantité mentionnée sur ce bon après le scan de chaque article. La quantité est modifiée en cas de réception partielle.

La figure 30 montre les numéros de série, le produit associé apparaît en cliquant sur le numéro, ce dernier peut être modifié.

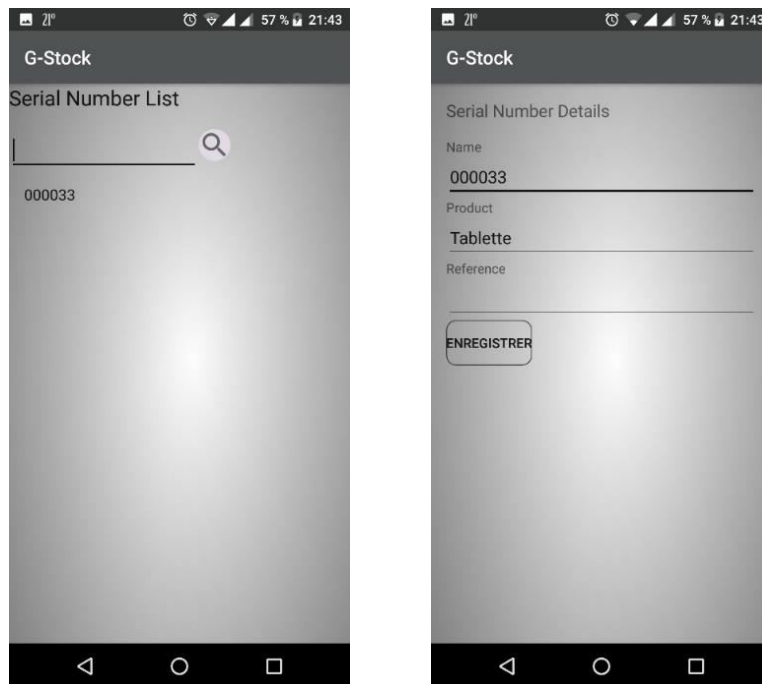


Figure 30 : Liste des numéros de série avec les détails

La figure 31 représente l'interface pour enregistrer vocalement les notes concernant les bons.

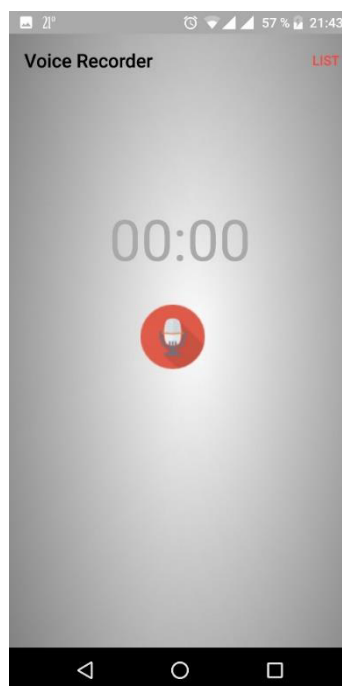


Figure 31 : Enregistrement vocal

7. Conclusion

La phase de réalisation est l'aboutissement des phases précédentes. C'est dans cette phase qu'est réalisé le produit du projet pensé.

Dans ce chapitre, nous avons présenté l'environnement dans lequel nous avons travaillé durant la période de développement du projet, les outils utilisés pour réaliser cette application, ainsi que les différentes interfaces de la dite application avec ses fonctionnalités.

Conclusion Générale

Ce projet de fin d'études que nous avons effectué au sein de l'entreprise SOGESI avait pour but la mise en place d'une application mobile basée sur Odoo qui soit adaptée et dédiée aux entreprises pour la gestion des stocks afin de minimiser les erreurs commises lors de la saisie des informations.

Dans un premier lieu, nous avons commencé par comprendre le contexte général de notre projet à savoir le métier de gestion des entrepôts. Ensuite, nous avons mené une étude technique pour étudier et analyser le progiciel Odoo, ainsi qu'une étude fonctionnelle afin de déterminer les besoins fonctionnels de notre système. Par la suite, nous avons fait une étude conceptuelle pour présenter tous les cas d'utilisation de notre système d'une part et pour réaliser le diagramme de classes d'autre part, qui par la suite nous a permis d'entamer la phase de réalisation.

Le travail au sein de la société SOGESI était sans aucun doute une expérience très enrichissante malgré les difficultés rencontrées au début à causes des technologies utilisées qui étaient pour nous totalement nouvelles ; mais avec l'aide de nos encadreurs nous avons pu surmonter tous les obstacles.

Ce stage a été pour nous un grand pas vers le milieu professionnel. Il nous a permis de bénéficier d'une excellente expérience qui nous a conduits à concrétiser nos connaissances informatiques acquises au cours de nos années d'études.

Finalement, ce projet nous a permis d'acquérir des valeurs indispensables telles que la responsabilité, le travail d'équipe, le sens d'analyse et l'adaptabilité à l'environnement de l'entreprise. Ces valeurs sont sans aucun doute les bases de réussite dans le milieu professionnel.

Ce travail réalisé demande à être complété et poursuivi. Il convient donc de prendre en considération les perspectives suivantes :

- La gestion des colis qui permet l'emballage des articles.
- La gestion des alertes sous forme de notification en cas de rupture de stock ou de dépassement des délais,
- La signature électronique pour valider les différents transferts.

BIBLIOGRAPHIE ET WEBOGRAPHIE

[1] Association canadienne pour la gestion de la production et des stocks (**ACGPS**), Dictionnaire de la gestion de la production et des stocks, Montréal, Édition Québec/Amérique, Presse HEC, **1993**

[2] **Korponai J**, **Tóth A**, **Illés B**, University of Miskolc, H-3515 Miskolc-Egyetemváros, Hungary, 7th International Conference on Engineering, Project, and Production Management Effect of the Safety Stock on the Probability of Occurrence of the Stock Shortage, **2017**

[3] **Elragal A**, **Haddara M**. The impact of ERP partnership formation regulations on the failure of ERP implementations. *Procedia Technology*; 9:527-35, **2013**

[4] **Lacombe C**, Contribution à une méthodologie et une modélisation pour accompagner les petites entreprises dans l'étude de leur organisation afin de spécifier leurs besoins et sélectionner une solution ERP, thèse de doctorat, **2015**

[5] **Mocellin**, Livre Gestion des stocks et des magasins Fabrice Mocellin, **2011**

[6] **Millet**, Une étude de l'intégration organisationnelle et informationnelle. Application aux systèmes d'informations de type ERP. INSA de Lyon, **2008**

[7] **Bensalem M**, Conception et Réalisation d'une Solution Modulaire sous Odoo et d'une Application Android « O'School », thèse d'ingénieur, **2017**

[8] **Blazejewski**, H. et **Nogaro**, J-M. "On peut maîtriser ruptures et sur-stocks", Harvard-L'expansion Management Review, No.68, pp. 65-74, **1993**

[9] La structure logistique de stock dans Odoo : <https://read-the-docs-open.sarl.readthedocs.io/fr/latest/readthedocs/fonctionnement.html> Consulté le **20/04/2019**

[10] Gantt.com : <https://www.gantt.com/fr/> Consulté le 25/06/2019

[11] Diagramme de cas d'utilisation :

http://docwiki.embarcadero.com/RADStudio/Rio/fr/D%C3%A9finition_des_diagrammes_de_cas_d'utilisation_UML_1.5 Consulté le 25/06/2019

[12] Diagrammes de séquence :

https://www.ibm.com/support/knowledgecenter/fr/SSRTLW_9.6.1/com.ibm.xtools.sequenc.e.doc/topics/cseqd_v.html Consulté le 25/06/2019

[13] Diagramme de classe :

https://www.ibm.com/support/knowledgecenter/fr/SS5JSH_9.5.0/com.ibm.xtools.modeler.doc/topics/cclassd.html Consulté le 25/06/2019

[14] Diagramme de navigation :

https://www.ibm.com/support/knowledgecenter/fr/SSRTLW_9.6.1/com.ibm.xtools.topi.cbrowse.doc/topics/cbrowse.html Consulté le 25/06/2019

[15] SOGESI : <http://www.sogesi-dz.com> Consulté le 13/05/2019

[16] **Elouassini O, Khattabi G**, Conception et développement d'une plateforme Odoo et d'une application Android destinée à la gestion d'une société spécialisée dans les métiers de l'offshoring ICT, école nationale des sciences appliquées Tétouan promotion, thèse d'ingénieur, **2015**

[17] Service Web :

<https://openclassrooms.com/courses/les-services-web> Consulté le 24/06/2019

[18] Android : <http://open-prod.com/pourquoi-open-prod/> Consulté le 15/06/2019

[19] Android Studio :

https://fr.wikipedia.org/wiki/Android_Studio Consulté le 24/06/2019

[20] PostgreSQL :

<https://www.postgresql.org/docs/current/intro-what-is.html> Consulté le 23/06/2019

[21] Visual Paradigm

<https://www.commentcamarche.net/download/telecharger-34058460-visual-paradigm>

Consulté le 29/06/2019

[22] **Ziadé, Tarek, and Patrick T**, Programmation Python : [syntaxe, conception et optimisation]. Eyrolles, **2006**.

[23] **Ludovic R**, Structurez vos données avec XML, Simple It ,2014

[24] Java Android [https://fr.wikipedia.org/wiki/Java_\(langage\)](https://fr.wikipedia.org/wiki/Java_(langage)) Consulté le 25/06/2019

[25] Présentation générale des ERP et leur architecture modulaire :

<https://fablain.developpez.com/tutoriel/presenterp/> Consulté le 07/07/2019

[26] Schéma des flux :

<https://read-the-docs-open-net-sarl.readthedocs.io/fr/latest/readthedocs/fonctionnement.html?fbclid=IwAR1sVhMTeH4I-EDUv7QRzHLpCYzxP0DpZp-GruQF80up3OAny-2GZh51cwk>

Consulté le 07/07/2019

[27] Exemple de développement en Scrum :

<https://www.alterest.com/index.php/fr/blog/item/22-mener-un-projet-d-infrastructure-avec-scrum> Consulté le 07/07/2019

LISTE DES ABRÉVIATIONS

- ❖ **API:** Application Programming Interface.
- ❖ **BSD:** Berkeley Software Distribution.
- ❖ **CRM:** Customer Relationship Management.
- ❖ **DB2:** Data Base 2 (SGBD d'IBM).
- ❖ **ERP :** Enterprise Resource Planning.
- ❖ **GED :** Gestion Électronique de Documents.
- ❖ **HTML:** HyperText Markup Language.
- ❖ **HTTP:** Hypertext Transfer Protocol.
- ❖ **IP:** Internet Protocol.
- ❖ **J2EE:** Java 2 Enterprise Edition.
- ❖ **PHP :** Hypertext Preprocessor.
- ❖ **PME :** Petites et Moyennes Entreprises.
- ❖ **RH:** Ressources Humaines.
- ❖ **SDK:** Software Developers Kit.
- ❖ **SOGESI :** SOciété de Gestion et Solution Informatique.
- ❖ **SQL:** Structured Query Language.
- ❖ **UML:** Unified Modeling Language.
- ❖ **XML :** Extensible Markup Language.
- ❖ **XML-RPC:** XML Remote Procedure Call.

LISTE DES FIGURES

| | |
|---|----|
| Figure 1 : Architecture modulaire des ERP | 5 |
| Figure 2 : Les modules locaux disponibles dans Odoo 8..... | 7 |
| Figure 3 : Tableau de bord du module Entrepôt | 11 |
| Figure 4 : Schéma des flux..... | 12 |
| Figure 5 : La liste des clients pour la société SOGESI | 15 |
| Figure 6 : Exemple de développement en Scrum..... | 15 |
| Figure 7 : Diagramme de Gantt | 17 |
| Figure 8 : Schéma illustratif d'une réception partielle..... | 20 |
| Figure 9 : Diagramme de cas d'utilisation « gestion de stock »..... | 21 |
| Figure 10 : Diagramme de séquence du cas « gérer les réceptions » | 23 |
| Figure 11 : Diagramme de séquence du cas « s'authentifier »..... | 24 |
| Figure 12 : Diagramme de classe..... | 25 |
| Figure 13 : Diagramme de navigation | 26 |
| Figure 14 : Odoo logo | 29 |
| Figure 15 : Android logo..... | 30 |
| Figure 16 : Android studio logo..... | 30 |
| Figure 17 : PostgreSQL logo | 31 |
| Figure 18 : Visual Paradigm logo | 31 |
| Figure 19 : Scan du code à barre à travers l'appareil photo..... | 32 |
| Figure 20 : Création du reliquat par l'intermédiaire du bouton « backorder » | 33 |
| Figure 21 : l'interface de l'authentification | 34 |
| Figure 22 : Menu des listes | 35 |
| Figure 23 : Navigation menu | 36 |
| Figure 24 : Menu des mouvements | 37 |
| Figure 25 : Les mouvements..... | 38 |
| Figure 26 : La liste d'inventaires avec les détails | 38 |
| Figure 27 : Interface des listes et détails des partenaires | 39 |
| Figure 28 : La liste et détails des produits | 39 |
| Figure 29 : Liste des bons avec détails | 40 |
| Figure 30 : Liste des numéros de série avec les détails | 41 |
| Figure 31 : Enregistrement vocale | 41 |

LISTE DES TABLEAUX

Tableau 1 : Comparaison de quelques ERP open source6

Résumé

Les progiciels de gestion intégrés (ERP), sont l'une des solutions les plus pratiques. Ils intègrent les principales composantes fonctionnelles de l'entreprise comme la gestion des stocks qui appartient à une famille des techniques de l'organisation logistique. L'enjeu principal consiste à disposer des ressources suffisantes.

L'objectif de ce travail est de proposer une solution qui soit adaptée et dédiée aux entreprises pour la gestion des stocks dans le but est de minimiser les erreurs commises lors de la gestion des opérations de stockage et de déstockage et d'éviter par-là la perte de temps.

Cette solution est une application mobile sous ANDROID qui communique avec le serveur ODOO via le web service XML-RPC pour effectuer les différentes tâches de la gestion des stocks (réception, livraison, transfert interne, inventaire).

Mots-clés : ERP, ODOO, gestion des stocks, ANDROID, XML-RPC, web service.

Abstract

Enterprise Resource Planning are one of the most practical solutions for integrating the main functional components of the enterprise, such as inventory management, which belongs to a family of logistics organization techniques and whose main challenge is to have sufficient resources.

The objective of this work is to propose a solution that is adapted and dedicated to companies for the management of stocks in order to minimize the errors made during the management of the operations of storage and destocking and to avoid by that the loss of time.

This solution is a mobile application under ANDROID that communicates with the ODOO server via the XML-RPC web service to perform the various tasks of inventory management (reception, delivery, internal transfer, inventory).

Keywords: ERP, ODOO, inventory management, ANDROID, XML-RPC, web service.

المخلص

يعد تخطيط موارد المؤسسة أحد أكثر الحلول العملية لدمج المكونات الوظيفية الرئيسية للمؤسسة، مثل إدارة المخزون، والتي تنتمي إلى عائلة من تقنيات تنظيم اللوجستيات والتي يتمثل التحدي الرئيسي في امتلاك موارد كافية.

الهدف من هذا العمل هو اقتراح حل مناسب ومخصص للشركات لإدارة المخزون من أجل تقليل الأخطاء التي تحدث أثناء إدارة عمليات التخزين والاسترجاع ولتجنب ضياع الوقت.

هذا الحل هو عبارة عن تطبيق جوال ضمن ANDROID يتصل بخادم ODOO عبر خدمة الويب XML-RPC لأداء المهام المختلفة لإدارة المخزون (الاستقبال، التسليم، النقل الداخلي، المخزون).

الكلمات المفتاحية: تطبيق، خادم، إدارة المخزون، XML-RPC، ANDROID، ODOO.