



الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET
POPULAIRE
وزارة التعليم العالي والبحث العلمي
MINISTÈRE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
جامعة 20 أوت 1955 - سكيكدة
UNIVERSITÉ 20 Aout 1955 – Skikda



Faculté des Sciences

Département d'INFORMATIQUE

**Mémoire présenté en vue de l'obtention d'un Diplôme de
Master 2 académique**

Spécialité : Génie Logiciel Avancée et Application « GLAA »

THEME :

***GESTION DE LA MAINTENANCE ASSISTEE PAR
ORDINATEUR***

***GESTION DES STOCKS DE PIECES DETACHEES ET ACHATS
T.V.E. SIKKDA***

Réalisé par :

▪ MAAMCHIA Ghoulam

Encadré par :

TOUIL Ghassen

Année universitaire : 2023-2024

Remerciements

On tient à la fin de ce travail à remercier ALLAH le tout puissant de nous avoir donné la foi et de nous avoir permis d'en arriver là Malgré les circonstances difficiles auxquelles nous avons été confrontés.

En préambule à ce mémoire, on souhaitait adresser nos remerciements les plus sincères aux professeurs qui nous ont étudiés durant notre parcours académique et aux personnes qui nous ont apporté leurs aides et qui ont contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire.

On tient à remercier sincèrement notre encadreur Mr TOUIL GHASSANE, d'avoir accepté de nous encadrer, proposer notre thème et nous avoir guidé tout au long de notre projet de master par ses précieuses directives, pour sa confiance et sa gentillesse, sa disponibilité ainsi pour l'inspiration, l'aide et le temps qu'elle a bien voulu nous consacrer.

On exprime également nos gratitude à nos examinateurs, qui vont nous honorer en évaluant ce travail.

Par la même occasion, nous tenons à remercier les responsables administratifs pour nous avoir accueillies chaleureusement au sein de leur structure.

Ghoulem



Dédicaces

A mes très chers parents

*Pour tout l'amour dont vous m'avez entouré, pour tout ce que
vous avez
fait pour moi.*

*Je ferai de mon mieux pour rester un sujet de fierté à vos yeux
avec*

l'espoir de ne jamais vous décevoir.

*Que ce modeste travail, soit l'exaucement de vos vœux tant
formulés et de*

Vos prières quotidiennes.

A mes très chers Sœurs

A mes très chers frères

A mes très chères amies

*En souvenir de nos éclats de rire, des bons moments et nuits
blanches.*

En souvenir de tout ce qu'on a vécu ensemble.

J'espère de tout mon cœur

Que notre amitié durera

éternellement.

Ghoulem

ملخص

حققت الحوسبة اليوم تطورًا تكنولوجيًا مذهلاً في مجالات مختلفة (شبكات الكمبيوتر، وقواعد البيانات، والويب، وما إلى ذلك). وهذا التطور ضروري لمعالجة المشاكل التي نواجهها في حياة اليوم. في الواقع، تعد إدارة المخزون نشاطًا أساسيًا يظل في قلب اهتمامات أي مدير يرغب في توجيه شركته نحو الأداء. نظرًا للحجم الكبير من المعلومات المتعلقة بإدارة المخزون في شركة الإنتاج، فإن استخدام نظام إدارة المخزون أمر ضروري. ويرجع ذلك إلى مرونتها في الاستخدام ووظائف المراقبة والإدارة الغنية التي تقدمها. الهدف من هذا المشروع هو تطوير نظام إدارة المخزون والإمدادات لشركة T.V.E المتخصصة في نقل الركاب عن طريق البر. تم نمذجة نظامنا باستخدام لغة UML. تم تنفيذ التطبيق بلغة (HTML و CSS و PHP) باستخدام برنامج Visual Studio و XAMPP.

الكلمات المفتاحية: إدارة المخزون والمشتريات، UML، HTML، Visual Studio.

Abstract

Today, computing has achieved prodigious technological evolution in different areas (computer networks, databases, WEB, etc.). This development is necessary to remedy the problems encountered in today's life. Indeed, inventory management is an essential activity which remains at the heart of the concerns of any manager wanting to steer their company towards performance. Given the large volume of information related to inventory management in a production company, the use of an inventory management system is essential. This is thanks to their flexibility of use and the rich monitoring and management functions that they offer. The objective of this project is to develop a stock and supply management system for the T.V.E company specializing in the transport of passengers by road. The modeling of our system was done using the UML language. The implementation of the application was used in (HTML, CSS, PHP) language using Visual Studio and XAMPP software.

Keywords: inventory management, UML, HTML, Visual Studio.

Résumé

Aujourd'hui, l'informatique a atteint une prodigieuse évolution technologique dans différents domaines (réseaux informatiques, bases de données, WEB, etc.). Cette évolution est nécessaire pour remédier aux problèmes rencontrés dans la vie actuelle. En effet, La gestion de stock est une activité incontournable qui reste au cœur des préoccupations de tout gestionnaire voulant piloter son entreprise vers la performance. Vu le volume important des informations liées à la gestion de stock dans une entreprise de production, l'utilisation d'un système de gestion de stock est indispensable. Ceci est grâce à leur souplesse d'utilisation et les fonctions riches de suivi et de gestion qu'ils offrent. L'objectif de ce projet est de développer un système de gestion de stock et l'approvisionnement pour l'entreprise T.V.E spécialisé dans le transport de voyageurs par route. La modélisation de notre système a été faite par le langage UML. L'implémentation de l'application a été utilisé le langage (HTML, CSS, PHP) réalisé en utilisant le logiciel Visual Studio et XAMPP.

Mots-clés : stock management and procurement, UML, HTML, Visual Studio.

Table des matières

Liste des Figures.....	I
Liste des Tables.....	II
Liste des Abréviations.....	III
Introduction Générale.....	1
Chapitre 1 : Etude de l'Existant	3
1. Introduction	4
2. Présentation de l'entreprise TVE	4
2.1. Historique de l'entreprise	4
2.2. Situation Géographique	5
2.3. Organigramme de l'entreprise	5
2.4. Présentation du service maintenance	6
2.5. Organigramme du service maintenance	7
2.6. Les taches du service maintenance	7
2.7. Les taches de section techno administratives	11
3. Présentation du sujet d'étude	11
3.1. Problématique	11
3.2. Objectifs du projet	11
4. Conclusion	12
Chapitre 2 : Généralités sur la GMAO et le stock, le réapprovisionnement	13
1. Introduction	14
Partie I : Généralités sur le GMAO	14
I.1. Introduction	14
I.2. La définition du GMAO	14
I.3. Caractéristiques d'un logiciel GMAO	15
I.4. Les Conditions du GMAO	16
I.4.1. Conditions de la mise en œuvre de la GMAO	16
I.4.2. Les conditions de réussite d'un SGMAO	16
I.5. Les différents modules fonctionnels d'une GMAO	17
I.5.1. Module "gestion des équipements"	17
I.5.2. Module "gestion du suivi opérationnel des équipements"	17
I.5.3. Module "gestion des interventions"	17

I.5.4. Module "gestion du préventif"	17
I.5.5. Module "gestion des stocks"	17
I.5.6. Module "gestion des approvisionnements et des achats"	17
I.5.7. Module "analyse des défaillances"	17
I.5.8. Module "budget et suivi des dépenses"	18
I.5.9. Module "gestion des ressources humaines"	18
I.5.10. Module "tableaux de bord et statistiques"	18
I.6. Le guide du choix de GMAO	18
I.7. Exemple GMAO pour le domaine de transport	19
I.7.1. CARL Source Transport	19
I.7.2. OpenGST	19
Partie II : La gestion de stock et l'approvisionnement	20
II.1. Introduction	20
II.2. Définition	20
II.2.1. La gestion de stock	20
II.2.2. La gestion d'approvisionnement	21
II.2.3. La gestion d'achat	21
II.3. Les types de la gestion du stock	21
II.4. Les méthodes de la gestion du stock	22
II.4.1. Réapprovisionnement calendaire	22
II.4.2. Méthode de rechargement	23
II.4.3. Méthode du point de commande	24
II.4.4. Le réapprovisionnement à la commande	24
II.4.5. Le "drop shipping"	25
II.5. Les principales méthodes de réapprovisionnement des stocks en entrepôt	26
II.5.1. La méthode du réapprovisionnement des stocks à la demande	26
II.5.2. Le réapprovisionnement des stocks avec rupture de stock minimum programmée	27
II.5.3. La stratégie de réapprovisionnement des stocks « top off », basée sur le Lean time	27
II.6. Les outils de gestion de stock	27
II.6.1. Gestion de stock sur Excel (tableau de gestion de stock)	28
II.6.2. Logiciel ou application de gestion de stock	28

II.7. Le fonctionnement de la gestion du stock	28
2. Conclusion	28
Chapitre 3 : Conception Et Description De Notre Système	29
1. Introduction	30
2. Les besoins	30
2.1. Les acteurs du système	30
2.2. Besoins fonctionnels	30
2.3. Besoins non fonctionnels	31
3. UML	32
3.1. Définition	32
3.2. Historique d'UML	32
3.3. Les caractéristiques d'UML2.0	35
3.4. Les diagrammes d'UML2.0	36
3.5. Les Points Forts et Faibles d'UML	37
4. Les diagrammes d'UML	38
4.1. Diagramme de cas d'utilisation	38
4.2. Description textuelle de chaque cas d'utilisation	39
4.3. Conception	43
5. Conclusion	49
Chapitre 4 : Implémentation Et Réalisation Du Projet	50
1. Introduction	51
2. Environnement de développement	51
2.1. Environnement matériel	51
2.2. Environnement logiciel	51
3. Présentation du site web	54
3.1. Extrait du code	55
3.2. Capture des interfaces de site web	56
4. Conclusion	58
Conclusion générale.....	59
Bibliographie & Webographie.....	61

Liste des figures

Chapitre 1 : Etude de l'Existant

Figure 1.1 : Situation géographique du TVE	05
Figure 1.2 : Organigramme unité 37 Skikda	06
Figure 1.3 : Organigramme service maintenance	07
Figure 1.4 : Processus d'entretien curatif	08
Figure 1.5 : Processus d'intervention préventive	10

Chapitre 2 : Généralités sur la GMAO et le stock, le réapprovisionnement

Figure 2.1 : CARL Source Transport	19
Figure 2.2 : OpenGST	20

Chapitre 3 : Conception Et Description De Notre Système

Figure 3.1 : Le diagramme de cas d'utilisation global	39
Figure 3.2 : Le diagramme de classe de l'application	45
Figure 3.3 : Diagramme de séquence du cas « S'Authentifier »	46
Figure 3.4 : Diagramme de séquence du cas « Gérer les fournisseurs »	47
Figure 3.5 : Diagramme de séquence du cas « Gérer les sociétés »	47
Figure 3.6 : Diagramme de séquence du cas « Gérer le stock »	47
Figure 3.7 : Diagramme de séquence du cas « Voir les commandes de société »	48
Figure 3.8 : Diagramme de séquence du cas « Voir répondre de fournisseur »	48
Figure 3.9 : Diagramme de séquence du cas « Répondre à la disponibilité du pièce »	48
Figure 3.10 : Diagramme de séquence du cas « Voir les commandes d'admin »	49
Figure 3.11 : Diagramme de séquence du cas « Demander des commandes »	49

Chapitre 4 : Implémentation Et Réalisation Du Projet

Figure 4.1 : Image du pc Sony VAIO	51
Figure 4.2 : Code HTML du projet	55
Figure 4.3 : Code PHP et MYSQL du projet	55
Figure 4.4 : Code JQuery du projet	56
Figure 4.5 : Capture d'écran accueil du notre site web	56
Figure 4.6 : Capture d'écran d'interface de gérer le stock	57

Figure 4.7 : Capture d'écran d'interface de gérer les sociétés 57

Figure 4.8 : Capture d'écran d'interface du profil du login 57

Liste des tableaux

Chapitre 1 : Etude de l'Existant

Aucune Tableaux

Chapitre 2 : Généralités sur la GMAO et le stock, le réapprovisionnement

Aucune Tableaux

Chapitre 3 : Conception Et Description De Notre Système

Tableau 3.1 : Historique d'UML	35
Tableau 3.2 : Description textuelle du cas d'utilisation « S'Authentifier »	40
Tableau 3.3 : Description textuelle du cas d'utilisation « Gérer les fournisseurs »	40
Tableau 3.4 : Description textuelle du cas d'utilisation « Gérer les sociétés »	41
Tableau 3.5 : Description textuelle du cas d'utilisation « Gérer le stock »	41
Tableau 3.6 : Description textuelle du cas d'utilisation « Voir les commandes de société »	42
Tableau 3.7 : Description textuelle du cas d'utilisation « Voir répondre de fournisseur »	42
Tableau 3.8 : Description textuelle du cas d'utilisation « Répondre à la disponibilité du pièce »	42
Tableau 3.9 : Description textuelle du cas d'utilisation « Voir les commandes d'admin » .	43
Tableau 3.10 : Description textuelle du cas d'utilisation « Demander des commandes » ...	43

Chapitre 4 : Implémentation Et Réalisation Du Projet

Aucun Tableaux

Liste des abréviations

Chapitre 1 : Etude de l'Existant

- 1.1. **COGS** = Cost of goods sold
- 1.2. **RFID** = Radio Frequency IDentification
- 1.3. **BTP** = Bâtiment et travaux publics
- 1.4. **UE** = Union Européenne
- 1.5. **TVE** = Transport De Voyageurs De L' Est
- 1.6. **SPA** = La Société Protectrice des Animaux
- 1.7. **PV** = Procès-Verbal

Chapitre 2 : Généralités sur la Maintenance et la GMAO

- 2.1. **GMAO** = Gestion de Maintenance Assistée par Ordinateur
- 2.2. **NF X 60-000** = Norme aFnor X 60-000
- 2.3. **BT** = Bon de Travail
- 2.4. **OT** = Ordre de Travail
- 2.5. **SAV** = Service Après Vente
- 2.6. **DI** = Demandes d'Intervention
- 2.7. **MTTR** = Mean time to repair
- 2.8. **MTBF** = Mean Time Between Failures
- 2.9. **DTE** = Dossier Technique d'Equipement
- 2.10. **MES** = Manufacturing Execution System
- 2.11. **ERP** = Enterprise Resource Planning
- 2.12. **IOT** = Internet of Things
- 2.13. **EAM** = La gestion des actifs d'entreprise

Chapitre 3 : Conception Et Description Du Projet

- 3.1. **UML** = Unified Modeling Language
- 3.2. **OMT** = Object Modeling Technique
- 3.3. **OOSE** = Object Oriented Software Engineering
- 3.4. **RAM** = Random Access Memory
- 3.5. **CRC** = Class Responsibility Collaboration

3.6. **UM** = Unified Method

3.7. **ISO** = International Organization for Standardization

3.8. **IEC** = International Electrotechnical Commission

3.9. **BDD** = Base De Données

Chapitre 4 : Implémentation Et Réalisation Du Projet

4.1. **DDR** = **D**ouble **D**ata **R**ate

4.2. **PCI** = Est un bus utilisé pour la connexion de cartes de contrôle de périphériques externes

4.3. **PC** = Portable Computer

4.4. **SGML** = Standard Generalized Markup Language

4.5. **MySQL** = My Structured Query Language

4.6. **GPL** = **GNU** = General Public License

4.7. **VB** = Visual Basic

4.8. **DOM** = Document Object Model

4.9. **JSON** = JavaScript Object Notation

4.10. **IDE** = Integrated Development Environment

Introduction Générale

Introduction Générale

L'informatique est au cœur de l'entreprise, quel que soit son secteur d'activité. On peut facilement comparer la place que joue l'informatique au sein d'une entreprise à celle que joue le système nerveux chez l'être humain. En effet, l'informatique est au centre de l'activité, et doit fonctionner pleinement et en permanence. Les activités des entreprises reposent essentiellement sur leurs systèmes informatiques. Ces derniers sont liés de façon intrinsèque au bon fonctionnement de l'entreprise.

La continuité dans les services offerts est un facteur prépondérant dans la survie d'une entreprise parce que la rupture dans l'offre de ces services constitue une perte, souvent énorme pour une entreprise.

Il est alors important de prévenir et de gérer les problèmes qui surviennent, afin de garantir, non seulement la qualité des services fournis mais aussi et surtout la vie de l'entreprise.

La gestion des stocks, élément clef majeur de la performance d'une entreprise, peut être définie comme une activité permettant de garantir la maîtrise des marchandises en termes de quantité, qualité, coûts et respect des approvisionnements. Pour optimiser l'efficacité de leur fonction logistique, les entreprises doivent choisir la politique et les paramètres de gestion du stock de façon optimum. Les diverses méthodes de gestion des stocks peuvent être regroupées en deux catégories : le suivi continu et le suivi périodique.

Dans ce mémoire nous allons décrire le processus de création de notre application qui est une plateforme commune à l'entreprise, aux chauffeurs et les réparateurs de la panne. L'administrateur qui y a toute la responsabilité pour gérer notre application. Cette application combine différents composants tels que des graphiques, des scripts, des fichiers texte et des images offrant une interface est très facile à utiliser pour tous les utilisateurs, avec une grande flexibilité lors de l'utilisation des différentes fonctionnalités offertes par ce dernier.

L'organisation de notre mémoire est comme suit : le **chapitre 1** c'est la partie de l'étude de l'existant où présenter l'entreprise du T.V.E, sa définition, historique, et l'organigramme. Dans le **chapitre 2**, étude générale sur la GMAO, définition, les caractéristiques, les conditions et le guide du choix, Et d'autre part une étude générale sur le stock, l'achat et le réapprovisionnement. Dans le **chapitre 3**, nous allons présenter l'analyse des besoins ainsi que les étapes de conception et de modélisation de notre logiciel. Dans le **chapitre 4**, nous présentons les étapes de mise en œuvre du logiciel, ainsi que les moyens logiciels et les langages utilisés. Ce mémoire se termine par une conclusion générale. Nous indiquons aussi quelques perspectives futures.

Chapitre 1

Etude de l'Existant

1. Introduction

L'étude de l'existant est une phase importante pour bien comprendre le système actuel et définir ses objectifs. Pour chaque module, Il sera question d'effectuer une description précise de l'existant en énumérant les principaux acteurs impliqués, les principaux documents manipulés et les moyens de traitements utilisés.

La gestion des stocks est l'ensemble des mesures qu'utilise une entreprise pour savoir quelle quantité commander et à quel moment, dans l'optique d'atteindre l'équilibre entre un coût de stockage faible et une capacité de réponse élevée face aux clients.

Nous allons aborder dans ce chapitre deux partie principales, la première partie on étudier une généralité de l'entreprise du stage (TVE), sa définition et la localisation, l'organigramme générale et d'autre part on présenter le sujet d'étude, problématique et l'objectif.

2. Présentation de l'entreprise TVE

T.V.E est une Entreprise Publique, issue de la restructuration de la Ex. S.N.T.V, créée sous forme de S.P.A. en mars 1989 avec un capital social actuel de 370 Millions de dinars. Le patrimoine de l'Entreprise (T.V.E.), outre le siège de la Direction Générale sis à Constantine, est constitué de trois unités de production, implantées à : Skikda, Sétif et Biskra. Son activité essentielle est le transport de voyageurs par route. L'Entreprise T.V.E. couvre vingt-trois Wilayas du pays et transporte neuf à douze Millions de voyageurs par an, sur un réseau routier de plus de 09 Millions de Kms, pour un parc circulant récent de cent-trente-cinq véhicules.

2.1. Historique de l'entreprise

L'entreprise publique économie des transports de voyageurs existe depuis le 17/11/ 1971. Elle est issue de la restructuration organique et financière de la société mère, à savoir l'ex-SNTV qui a été divisé en mai 1983 sur 5 entreprises publiques géographique régionales (Centre, Est, Ouest sud-est et sud-ouest) par le degré numéro 83-306 à compétence régionale. EPE TVE est passé à l'autonomie de gestion en mars 1989. Avec un capital social actuel de 370 Millions de dinars. Le patrimoine de l'Entreprise (T.V.E.), outre le siège de la Direction Générale sis à Constantine, est constitué de trois unités de production, implantées à : Skikda, Sétif et Biskra [1].

Son activité essentielle est le transport de voyageurs par route. L'Entreprise T.V.E. couvre vingt-trois Wilayas du pays et transporte neuf à douze Millions de voyageurs par an, sur un réseau routier de plus de 09 Millions de Kms, pour un parc circulant récent de cent-

trente-cinq véhicules.

Ainsi, l'objet social de l'entreprise a été complété par les activités suivantes [1] :

- ❖ Agence de voyage et tourisme ;
- ❖ Réparation mécanique de matériel lourd de transport routier ;
- ❖ Mécanique générale ;
- ❖ Ajustage, tournage et fraisage ;
- ❖ Equilibrage, parallélisme des roues ;
- ❖ Réparation électricité auto ;
- ❖ Tôlerie et peinture automobile ;
- ❖ Contrôle technique de véhicule.

2.2. Situation Géographique

EPE TVE Skikda est située dans la ville de Skikda, en Algérie, plus précisément, elle se trouve à l'adresse suivante : Zone Industrielle, route de Filfla, BP 17, Skikda Algérie [1].

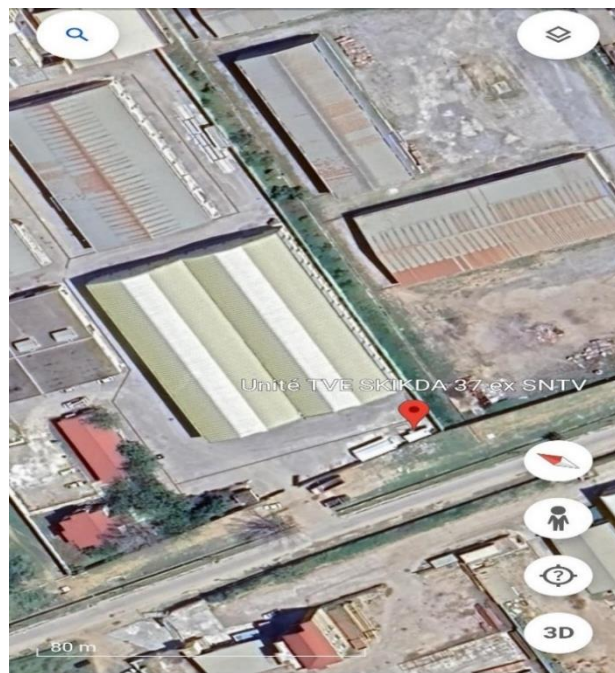


Figure I.1 : Situation géographique du TVE

2.3. Organigramme de l'entreprise

L'Unité 37 de Skikda est une composante essentielle de l'EPE TVE, dédiée à la réalisation de divers projets de travaux publics et de voies de communication dans la région Est de l'Algérie. Cette unité est spécialisée dans la construction, l'exploitation et la maintenance des infrastructures routières, ferroviaires, et autres installations nécessaires au développement des infrastructures de transport [1].

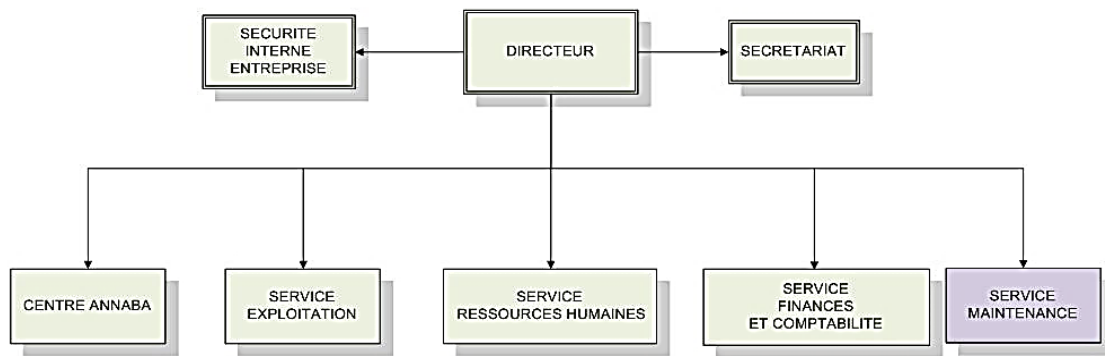


Figure I.2 : Organigramme unité 37 Skikda

2.4. Présentation du service maintenance

Le service de la Maintenance de l'unité est chargé, principalement des attributions suivantes [1] :

- Assurer la préservation du parc d'exploitation par la mise en place d'un programme d'entretien préventif et d'entretien curatif.
- Veiller à la disponibilité de la pièce de rechange nécessaire à l'entretien du par l'exploitation.
- Assurer une large prospection pour assurer l'approvisionnement en pièces de rechanges et organes de qualité et un coût normal.
- Assurer une gestion des stocks, conformément à la réglementation en vigueur.
- Respect des clauses des Contrats concernant les véhicules en période de garantie.

2.5. Organigramme du service maintenance

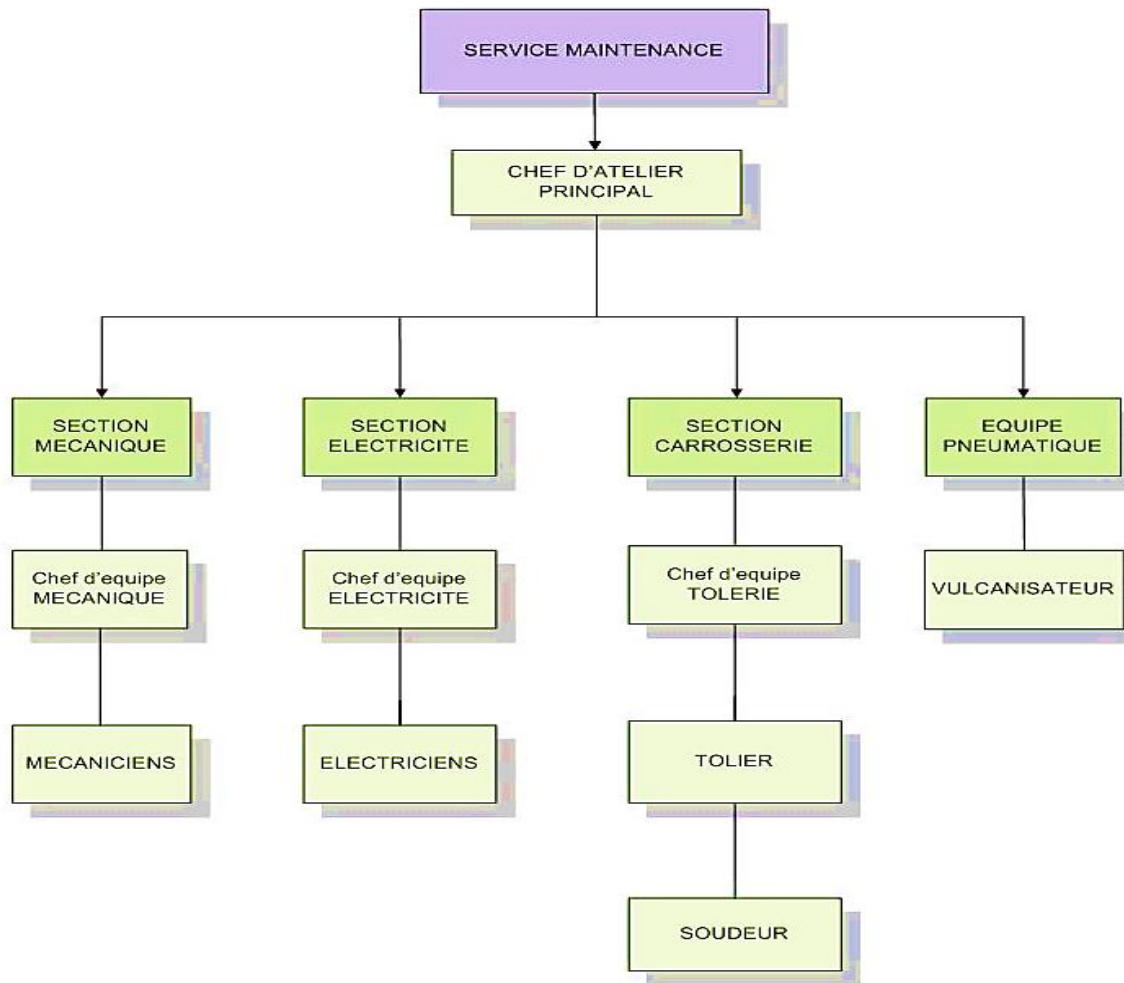


Figure I.3 : Organigramme service maintenance

2.6. Les taches du service maintenance

2.6.1. Processus d'entretien curatif

- Vérification de chaque autocar par un agent d'intervention lors de son entrée au parc.
- Ce dernier remplit une fiche signalétique qui décrit l'état intérieur et extérieur de l'autocar (partie tôle, partie mécanique, partie électricité).
- Si un chauffeur détecte une panne dans l'autocar, il la déclare dans un registre au niveau du poste de garde.
- Le chef de section technique fait un premier diagnostic afin de déterminer la section d'atelier concernée par la réparation de l'autocar.
- L'établissement du bon de travail avec le premier diagnostic et désignation de la section d'atelier responsable de la réparation.

- Le chef de section technique envoie le bon travail au chef d'atelier pour validation puis envoi à la section concernée.
- Si cette panne a besoin d'un changement de pièces ; le mécanicien retourne chez le chef de section technique pour ajouter ces pièces dans le bon de travail afin de retirer les pièces du magasin.

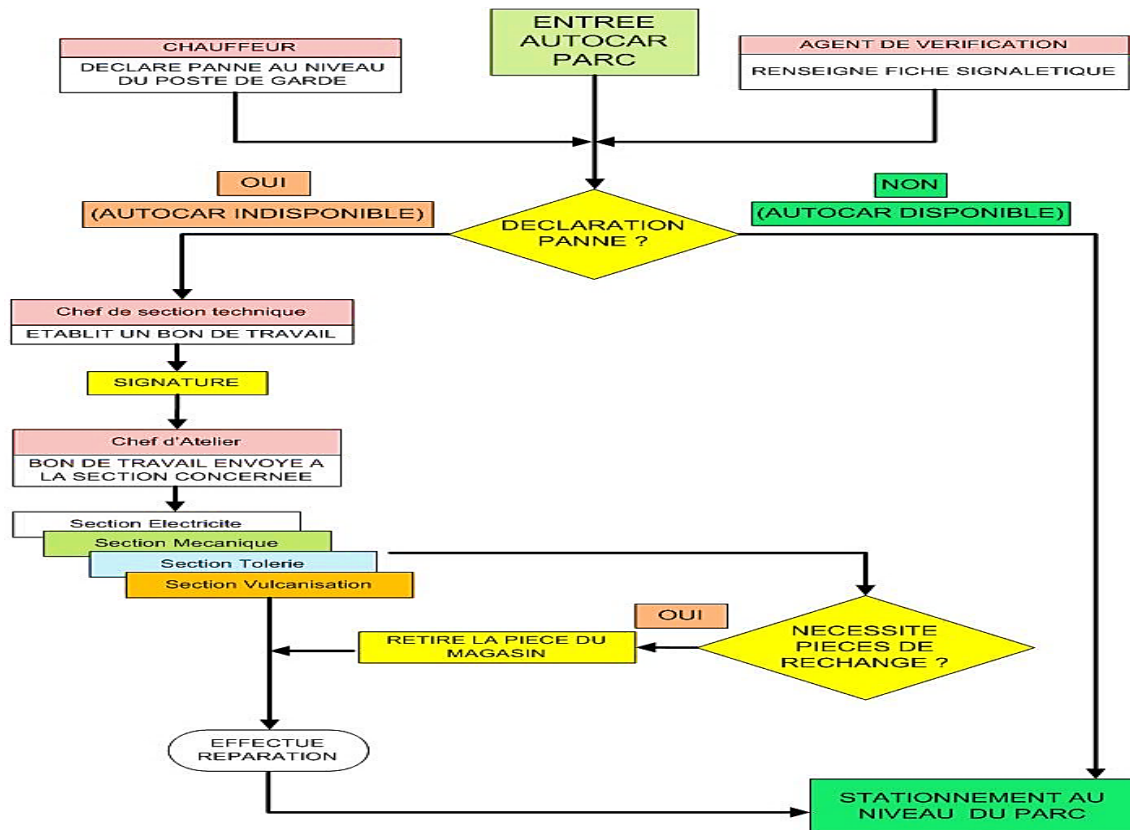


Figure I.4 : Processus d'entretien curatif

2.6.2. Processus d'intervention préventive

- Entrée au parc pour la maintenance programmée, établissement d'une fiche d'intervention préventive.
- Le chef section technique remplit les informations de l'autocar sur la fiche (kilométrage, marque, modèle, numéro de parc, numéro de châssis).
- La vérification de l'autocar se compose de cinq parties (mécanique, électricité, tôle, graissage, pneumatique).
- Le chef d'atelier choisit un mécanicien, un électricien, un tôlier, un spécialiste du graissage et un réparateur de pneu pour cette inspection.

- On dirige l'autocar vers la section mécanique ou vers la section électricité selon la numérotation de la fiche maintenance pour vérifier les pièces affichées sur la fiche.
- Après la fin de l'inspection le chef d'atelier prend la fiche au niveau du section technique.
- Au cas où l'on trouve des éléments ou des pièces non-conformes le chef de section technique établit un bon de travail selon la panne et la section responsable et l'envoie vers la section concernée.
- En cas de changement de pièces, le mécanicien retourne vers le service technique pour remplir le bon de travail avec les pièces de rechanges fin que le mécanicien puisse récupérer ces pièces au niveau du magasin.
- La personne décelant la panne est chargée de la réparation.
- Après la fin de l'intervention préventive la fiche va être signée par le chef de parc et le chef de service technique puis celle-ci sera archivée.

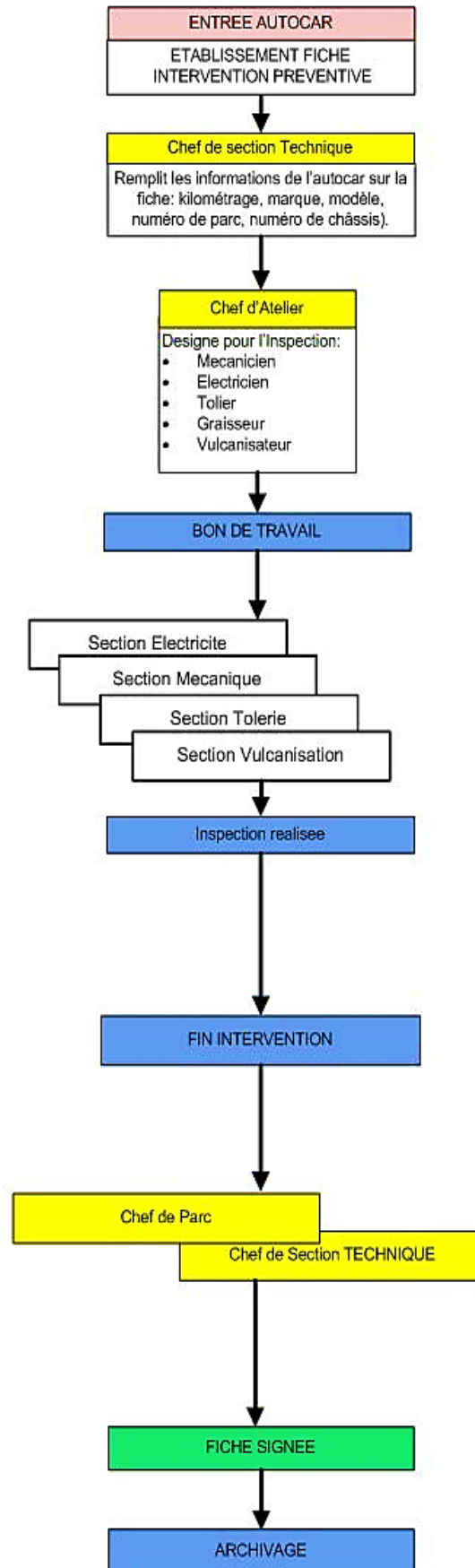


Figure I.5 : Processus d'intervention préventive

2.7. Les taches de section techno administratives

- Rapport d'atelier quotidien a la direction,
- Situation de parc quotidien autorisé à la direction.
- PV d'essai.
- Fax.
- Autorisation d'achat.
- Rapport d'activités mensuel.

3. Présentation du sujet d'étude

Le sujet qui nous a été présenté par l'organisme d'accueil porte sur le développement d'une application pour la gestion des stocks et de réapprovisionnement du l'entreprise TVE, pièce détachée.

3.1. Problématique

Durant la période de notre stage au sein de TVE de Skikda et plus précisément au niveau du magasin, on s'est rendu compte des problèmes suivants :

- Risque de tomber en rupture de stock (difficulté de constater l'arrivée au stock minimum) ;
- Absence d'un moyen efficace de vérification des entrées, sorties du stock ;
- La prestation de service lente ;
- Mauvaise gestion des équipements informatique
- Documents rédigés manuellement.

Tous ces facteurs engendrent la perte de temps et d'erreurs dans les quantités.

3.2. Objectifs du projet

Compte tenu des problèmes déduits de notre analyse et dans le souci de trouver les solutions appropriées, nous avons proposé d'implémenter une application permettant d'automatiser tous les processus de la gestion des stocks et le réapprovisionnement et rendre les tâches plus faciles à l'utilisateur à savoir :

- Automatiser toutes les procédures de la gestion des entrées et des sorties du stock (Achat et le vente des commandes) ;
- Calcul de l'autonomie des produits stockés en fonction de leur consommation ;

- Alerte automatique lorsque le stock atteint le seuil minimum ;
- Gestion informatique rapide ;
- Faciliter la recherche et la consultation des commandes réçit ou envoyées ;
- La rapidité dans l'utilisation ;
- Génération des documents du processus de gestion des stocks.

4. Conclusion

Au cours de ce chapitre, nous avons présenté d'une manière générale l'entreprise de TVE de Skikda ainsi que son fonctionnement, puis les problèmes rencontrés au sein du magasin du TVE Skikda et nos objectifs afin de leurs fournir un système de gestion des stocks qui répond à leurs attentes.

Cette étape sera le point de départ d'une étude détaillée et une conception que nous présenterons dans le chapitre de la conception.

Dans le prochain chapitre, nous passerons à la partie de définit le GMAO, le stock et le réapprovisionnement.

Chapitre 02

**Généralités Sur La GMAO
Et
Le Stock, Le Réapprovisionnement**

1. Introduction

Avec la croissance des parcs de machines, la GMAO, ou Gestion de la Maintenance Assistée par Ordinateur, revêt une importance croissante pour les acteurs de tous les secteurs.

L'évolution et la complexité des systèmes de production ainsi que le besoin de produire vite et bien, ont obligé les industriels à structurer et à organiser les ateliers maintenance, ils ont surtout créé de nouveaux concepts d'organisation et de nouvelles manières d'intervenir.

Nous allons aborder dans ce chapitre deux parties principales, la première partie on étudie une généralité sur la GMAO, sa définition, des caractéristiques, des conditions et le guide du choix d'un part, et d'autre part on étudie le stock, l'achat et le réapprovisionnement.

Partie I : Généralités sur le GMAO

I.1. Introduction

Le service maintenance cherche à maintenir un bien afin que celui-ci soit en mesure d'assurer un service déterminé. Ainsi, une GMAO permet la gestion complète du parc machines, l'analyse du curatif, l'organisation des interventions préventives et réglementaires, la gestion des stocks et des achats, le reporting à travers les tableaux de bord et les statistiques, en prenant en compte les réalités du terrain.

Dans ce qui suit, nous allons décrire les caractéristiques de GMAO, puis nous allons décrire les avantages, les inconvénients, les conditions et le guide du choix.

I.2. La définition du GMAO

Un système informatique de management de la maintenance est un progiciel organisé autour d'une base de données permettant de programmer et de suivre sous les trois aspects techniques, budgétaire et organisationnel, toutes les activités d'un service de maintenance et les objets de cette activité (services, lignes d'atelier, machines, équipements, sous-ensembles, pièces...etc.) à partir de terminaux disséminés dans les bureaux techniques, les ateliers, les magasins et bureaux d'approvisionnement [2].

La GMAO est une méthode / outil de gestion de la maintenance par le biais d'un logiciel permettant de gérer les différentes tâches la maintenance des équipements au sein d'une entité (entreprise, collectivité, administration...).

L'ensemble des fonctionnalités et tableaux de bord de la GMAO permet de dresser une arborescence complète des équipements ou des sites à maintenir, et d'orchestrer ainsi ses opérations de maintenance. Adaptable à tous les utilisateurs, la GMAO devient un vrai assistant au service d'une activité continue, et donc d'une performance accrue des machines.

Initialement dédiée à l'industrie, durant les années 1970 à 1990, la GMAO n'a, depuis, cessé d'évoluer. L'apparition de solutions SaaS a apporté de la flexibilité aux outils de maintenance assistée, permettant de déployer la GMAO sur tous types d'infrastructures et dans des secteurs très variés [2].

Outre les opérations de maintenance, la GMAO couvre aujourd'hui de nombreuses tâches périphériques. Sa capacité à exploiter la donnée permet d'en tirer des analyses pointues au service de la productivité.

Un tel outil informatique, au service de la maintenance, permet de répondre aux besoins de différentes configurations techniques, à savoir :

- ❖ La maintenance de moyens de productions ;
- ❖ La maintenance tertiaire de bâtiments, ou de parcs immobiliers ;
- ❖ La maintenance de parcs de véhicules ;
- ❖ La gestion du SAV (Service Après Vente), en connexion avec différents sites, différents clients.

I.3. Caractéristiques d'un logiciel GMAO

Un Logiciel GMAO ou Logiciel de Gestion de Maintenance assistée par Ordinateur est destiné aux différents secteurs de l'industrie, du tertiaire, des institutions publiques... Son intérêt est d'assister quotidiennement les services maintenance dans leurs missions, en adéquation avec les nouvelles technologies.

La GMAO peut avoir de nombreuses utilités ; voici une liste non exhaustive des différents rôles qu'elle peut avoir pour faciliter la gestion d'entreprise [3] :

- **Gestion des équipements** : inventaire, localisation, gestion d'information dédiée par type d'équipement.
- **Gestion de la maintenance** : corrective (avec BT : bon de travaux), préventive (systématique, conditionnelle, prévisionnelle), curative.
- Gestion des demandes d'intervention (DI).
- **Gestion des stocks** : magasins, réapprovisionnements, valorisation des stocks.
- **Gestion des achats** : demandes d'achats, commandes, achats de fournitures, pièces et prestations, facturation fournisseurs, etc.
- **Gestion du personnel et planning** : activités, métiers, planning de charge, prévisionnel, etc.
- **Gestion des coûts et budget** : main d'œuvre, stocks, achats, location de matériel, etc., préparation des budgets, suivi périodique, rapports d'écart, etc.

- **Application Mobile** : connectée ou en hors connexion pour gagner en productivité.
- **Indicateurs clés de performance** : tableau de bord (requêtes de base de données concernant des statistiques, des alertes, MTTR, MTBF, Pareto, etc.).

Ce qui permet une souplesse d'implantation au sein de l'entreprise en fonction des objectifs de sa politique de maintenance :

- D'exploitation conviviale en offrant des possibilités d'apprentissage rapides et simples ;
- D'intégration aisée au sein de l'entreprise en prenant facilement en compte la terminologie et ses règles de gestion [3].

I.4. Les Conditions du GMAO

I.4.1. Conditions de la mise en œuvre de la GMAO

La mise en œuvre d'un système de GMAO nécessite la création de fichiers de base qui regroupent, par nature, toutes les informations nécessaires à la gestion de la maintenance.

Par exemples :

- Fichier fournisseur.
- Fichier articles.
- Fichier nomenclature des matériels.

Cet ensemble de fichiers de base constitue une base de données ou banque de données qui doit être en permanence actualisée. Cette création et mise à jour des fichiers nécessite une codification préalable des informations avec la définition de leur format [4].

I.4.2. Les conditions de réussite d'un SGMAO

- ❖ La convivialité de l'outil informatique : Le système va être donc utilisé par tout le personnel du service maintenance, et non réservé à certains. Il faut donc bannir l'ésotérisme, exiger absolument que la totalité de l'information soit disponible en langage connu, s'efforcer de trouver un bon compromis entre langage normalisé et terme de métier habituel de l'entreprise.
- ❖ **Volonté de réussite de consensus** : C'est une question de motivation de la part de tous les intéressés, qu'il convient donc de faire participer à la définition générale du projet GMAO.
- ❖ **Facteur temps** : Le facteur temps doit être pris en considération. La mise en place d'un système de GMAO peut durer jusqu'à 2 ans.

I.5. Les différents modules fonctionnels d'une GMAO

Tous les logiciels de GMAO ont en commun la même structure modulaire proposant les mêmes fonctions. Mais, selon les logiciels, les fonctions remplies sont diversement dénommées, diversement réparties et diversement organisées. Dans les bureaux techniques d'une entreprise (méthodes, ordonnancement, logistique et travaux neufs), on pourra effectuer la gestion par exploitation des 10 modules suivants [3] :

I.5.1. Module "gestion des équipements"

Il s'agit de décrire et de coder l'arborescence du découpage allant de l'ensemble du parc à maintenir aux équipements identifiés et caractérisés par leur DTE (dossier technique d'équipement) et leur historique, puis à leur propre découpage fonctionnel.

I.5.2. Module "gestion du suivi opérationnel des équipements"

Ce module permettra de suivre les performances d'un équipement à partir des indicateurs de fiabilité, de maintenabilité, de disponibilité.

I.5.3. Module "gestion des interventions"

Ce module doit permettre un enregistrement rapide de la durée, de la localisation, et de la nature d'une intervention.

I.5.4. Module "gestion du préventif"

Ce module doit permettre de gérer la maintenance systématique à travers un planning calendaire par équipement, les dates doivent être déterminées à partir d'un relevé de compteur (ou d'une mesure dans le cas de la maintenance conditionnelle). Ce qui fait que le déclenchement sera automatique, par listing hebdomadaire des opérations prévues dans la semaine. Chaque opération sera prévue par sa gamme préventive.

I.5.5. Module "gestion des stocks"

Le système repose sur le fichier des articles en magasin comprenant les "lots de maintenance" par équipement et sur les mouvements entrées/sorties du magasin.

I.5.6. Module "gestion des approvisionnements et des achats"

Les caractéristiques de la fonction de maintenance sont beaucoup de références et de fournisseurs pour des quantités faibles et des délais courts. Ce module doit permettre la gestion des achats.

I.5.7. Module "analyse des défaillances"

La base de ce module est constituée des historiques automatiquement alimentés par chaque saisie de BT (bons de travaux) ou OT (ordre de travaux). Ce qui permet une analyse quantitative ou qualitative des défaillances.

I.5.8. Module "budget et suivi des dépenses"

L'objectif de ce module est le suivi de l'évolution des dépenses par activité dans un budget donné.

I.5.9. Module "gestion des ressources humaines"

Spécifiquement adapté au service maintenance, ce module sera principalement une aide à l'ordonnancement.

I.5.10. Module "tableaux de bord et statistiques"

Les tableaux de bord concernent la mise en forme de tous les indicateurs techniques, économiques, et sociaux sélectionnés pour assurer la gestion et le management du service maintenance.

I.6. Le guide du choix de GMAO

Lors du choix de votre GMAO, certains critères sont à considérer avec attention, notamment [4] :

- **Le périmètre fonctionnel** : le logiciel choisi devra s'adapter au type d'installation et au secteur de l'entreprise. Il est recommandé d'étudier les solutions (génériques ou sectorielles) pour s'assurer que la solution de GMAO répondra bien au besoin.
- **La mobilité** : la maintenance est un travail de terrain. Les informations fournies par le logiciel de maintenance doivent pouvoir être consultées et enrichies depuis n'importe quel site.
- **La connectivité** : la GMAO devra peut-être communiquer avec un certain nombre d'outils métier (MES, ERP, IoT...). L'existence de connecteurs, natifs ou supplémentaires, doit être vérifiée à l'avance.
- **La collaboration** : le partage d'éléments (documentation technique, actualisation des stocks...) par les différents intervenants de la maintenance doit pouvoir se faire de manière intuitive et en temps réel. L'aspect collaboratif, mais aussi la disponibilité de l'application sont des critères essentiels.

Lors de la rédaction du cahier des charges, on s'interrogera également sur le besoin en fonctionnalités supplémentaires. Le besoin des entreprises va en effet souvent plus loin que la seule maintenance. La gestion avancée des stocks ou l'analyse prédictive font partie des outils pouvant s'avérer très importants pour faire progresser son activité.

I.7. Exemple GMAO pour le domaine de transport

I.7.1. CARL Source Transport

Accessible en ligne, CARL Source transport conçoit des solutions informatiques visant à relever les défis de l'exploitation des actifs de transport .

Dans le milieu des solutions de gestion d'exploitation maintenance depuis 40, les solutions ont évolué avec l'expérience et les besoins des clients. Maîtrise des installations fixes et linéaires et des matériels roulants. Pensé pour maîtriser un parc roulant, des installations fixes et des actifs linéaires, il est aussi et surtout important pour garantir la disponibilité des équipements et la sécurité du personnel [5].



Figure II.1 : CARL Source Transport

Fonctionnalité phare : connecté avec les plateformes des principaux organismes de contrôle afin de piloter la maintenance réglementaire.

Faiblesse : Profondeur technique assez importante, cela peut prendre du temps de maîtriser, voir simplement de connaître toutes les fonctionnalités.

I.7.2. OpenGST

OpenGST est pensé pour les collectivités qui veulent optimiser la gestion et le suivi de leurs services techniques.

La solution veut moderniser le travail des agents et l'implication des usagers en centralisant les informations et leur traitement pour faciliter la gestion au quotidien. Faite de différents modules, la solution peut être facilement ajustée aux besoins de la collectivité [5].

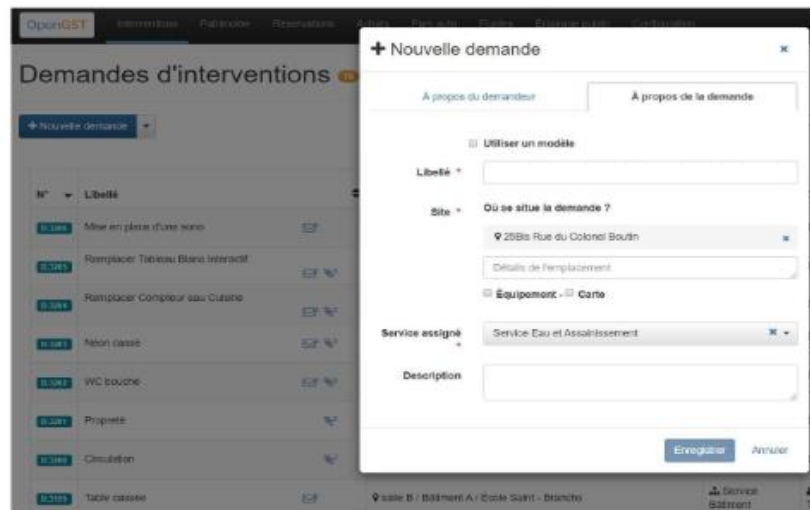


Figure II.2 : OpenGST

Fonctionnalité phare : Application Agents : saisie de demandes d'intervention avec photo et site géolocalisé / L'application citoyenne : permet aux usagers de signaler des problèmes sur les sites.

Faiblesse : Nous savons que les utilisateurs sont illimités, mais au-delà de cela pas de transparence sur les prix.

Partie II : La gestion de stock et le réapprovisionnement

II.1. Introduction

La gestion de l'approvisionnement, la gestion des achats et des stocks est une composante critique et essentielle du système de gestion de la qualité.

II.2. Définition

II.2.1. La gestion de stock

La gestion des stocks regroupe les activités d'achat et de stockage et se propose d'établir les références à tenir en magasin, de déterminer les quantités associées à ces références et de fixer les modes et les échéances de réapprovisionnement en respectant les conditions de coût, de délai et de qualité [6].

La gestion des stocks est le processus de gestion du flux des biens au sein d'une entreprise dans un cycle continu de commandes, de stockage, de production, de vente et de biens de restockage. La gestion des stocks s'effectue généralement à deux niveaux : gestion des stocks agrégés et emplacement de stockage ou gestion des stocks par niveau d'articles. Pour déterminer leur stratégie de gestion des stocks, les entreprises font des compromis entre la réduction du montant des liquidités liées aux stocks et la détention de stocks supplémentaires afin d'optimiser l'efficacité du service à la clientèle ou de la production [6].

II.2.2. La gestion d'approvisionnement et de réapprovisionnement

La gestion des approvisionnements désigne les activités d'une entreprise consistant à maîtriser son flux d'approvisionnement, à acquérir de nouvelles marchandises ou pièces, qu'il s'agisse [7] :

- De produits destinés à la revente directe au client final,
- Des composants et matières premiers nécessaires à la fabrication de produits ou services,
- De marchandise vouée à être stockée.

La gestion des approvisionnements fait donc partie intégrante de la gestion de stock.

Donc, Le réapprovisionnement des stocks se réfère au processus de ravitaillement des rayonnages après la réception de nouvelles marchandises envoyées par les fournisseurs ou le centre de production. Il peut également s'agir du transfert des matériaux, des rayonnages de stockage vers les rayonnages de picking.

Le réapprovisionnement des stocks doit être effectué en tenant compte des différentes tâches exécutées au sein des installations. Il est primordial de trouver le juste équilibre entre les exigences imposées par le service fourni et les caractéristiques de l'entrepôt (notamment l'espace de stockage).

II.2.3. La gestion d'achat

La gestion des achats est également appelée le processus de la source au règlement. Elle englobe l'évaluation, la sélection et la création d'accords contractuels formels ainsi que la gestion des relations permanentes de l'entreprise avec ses fournisseurs [8].

Bien qu'il y ait de nombreuses interférences entre gestion des stocks et achats, les deux sont différenciés par le fait que la gestion des stocks est à dominante interne (se focalise sur le coût et la relation avec les acteurs internes) et managériale alors que la gestion des achats est à dominante externe (se focalise sur le prix et la relation avec les partenaires externes) et procédurale.

II.3. Les types de la gestion du stock

Les entreprises utilisent divers systèmes de gestion des stocks, selon leurs opérations, leurs complexités ou leurs besoins. Les trois principaux systèmes de gestion des stocks sont par exemple manuels, périodiques et permanents. Les systèmes permanents sont les systèmes de gestion des stocks les plus avancés et les plus précis, tandis que la méthode manuelle est le moyen le moins sophistiqué de superviser les opérations d'inventaire [9].

- ❖ **Gestion périodique des stocks** : Le système de gestion périodique des stocks est une méthode d'évaluation des stocks conçu pour l'établissement de rapports financiers, et qui effectue un inventaire physique des stocks à intervalles précis. Cette méthode de comptabilité prend les stocks au début d'une période, ajoute les nouveaux achats au cours de cette période et déduit le stock final pour calculer le coût des biens vendus (COGS).
- ❖ **Gestion des stocks par code-barre** : Les entreprises utilisent des systèmes de gestion des stocks par code-barre pour attribuer un numéro à chaque produit qu'elles vendent. Elles peuvent associer plusieurs points de données à ce numéro, y compris le fournisseur, les dimensions du produit, le poids, et même des données variables telles que le nombre de produits en stock.
- ❖ **Gestion des stocks avec RFID** : RFID (radio frequency identification) est un système qui transmet via une connexion sans fil l'identité d'un produit sous la forme d'un numéro de série unique pour effectuer le suivi des articles et fournir des informations détaillées sur le produit. Le système de gestion d'entrepôt basé sur la technologie RFID améliore l'efficacité, accroît la visibilité des stocks et assure l'enregistrement automatique rapide des réceptions et des livraisons

II.4. Les méthodes de la gestion du stock

Il n'existe pas une méthode universelle pour gérer son stock et sa rotation des stocks. Mais toutes les manières de procéder tournent autour des deux mêmes variables : la date de commande et la quantité [10].

II.4.1. Réapprovisionnement calendaire

Cette méthode de gestion des stocks est plutôt rigide puisque l'entreprise qui l'utilise va commander des quantités fixes de marchandises à date fixe. Généralement, elle est mise en place dans le cadre d'un contrat que l'entreprise acheteuse signe avec son fournisseur : il prévoit les jours de livraison précis pour les mois ou l'année à venir, ce qui facilite l'organisation pour les deux parties.

Cette méthode est adaptée pour des entreprises déjà bien implantées et pour des marchandises qui sont vendues, transformées ou consommées à intervalles réguliers. Pour des produits dont la fréquence d'écoulement est plus "aléatoire", cette méthode n'est pas recommandée.

Avantages

- C'est l'une des méthodes de gestion des stocks les plus simples à utiliser : elle ne demande pas beaucoup d'investissement en temps pour fonctionner et entraîne des frais administratifs relativement bas.
- Possibilité de réaliser des économies non négligeables sur les commandes si ces dernières sont de grande taille (achats groupés par exemple).
- Les fournisseurs peuvent savoir très en avance les dates auxquelles ils doivent livrer l'entreprise, ce qui leur facilite le travail d'organisation, de préparation et d'expédition.

Inconvénients

- Convient uniquement à des entreprises déjà développées disposant d'une capacité d'achat élevée.
- Les dates de commandes sont renseignées dans le contrat et si pour une quelconque raison l'entreprise doit absolument changer une date, des difficultés administratives sont à prévoir avec le fournisseur dans certains cas.
- Du fait de son principe de fonctionnement, c'est une méthode peu flexible conçue majoritairement pour des produits ayant une fréquence de rotation élevée.

II.4.2. Méthode de reapprovisionnement

Elle est optimale pour des produits onéreux et / ou périssables qui sont vendus ou consommés de manière régulière. Chaque fois que le responsable en charge des achats doit passer une commande, il regarde la quantité restante dans le stock et commande autant que nécessaire pour revenir au niveau de stock maximum défini au préalable.

Avantages

- Tant que la rotation du stock suit les prévisions de l'entreprise, la quantité de produits entreposés reste optimale.
- Quand le responsable analyse son stock avant de passer la commande à la date fixée, il peut définir la quantité dont il a besoin et adapter en conséquence, ce qui peut faire réaliser de belles économies.
- Le montant des immobilisations financières est facilement maîtrisable par le responsable des achats.

Inconvénient

Si la consommation du mois qui suit n'est pas adaptée à la quantité du stock, l'entreprise peut vite se retrouver en surplus ou rupture de stock.

II.4.3. Méthode du point de commande

Cette troisième méthode de gestion des stocks fonctionne à l'inverse de la précédente. Également connue sous le nom de méthode "juste-à-temps", elle rend la date de commande variable mais la quantité achetée fixe. Pour ce faire, le gestionnaire des stocks s'appuie sur le stock critique. Dès que ce seuil est atteint, une commande est passée pour réapprovisionner l'entreprise avec une quantité prédéfinie et toujours identique. Attention, une fois le seuil critique atteint, il est indispensable de déterminer si l'entreprise aura de quoi produire et donc satisfaire ses clients jusqu'à l'arrivée de la cargaison. Si la réponse est négative, le responsable doit s'empresse de remonter le seuil critique pour que l'entreprise puisse continuer à répondre à la demande le temps d'être livrée.

Avantages

- C'est une méthode de gestion des stocks très intéressante si la marchandise a tendance à être vendue ou transformée de manière irrégulière. Cela évite de recommander alors que le stock est encore plein ou de se retrouver en rupture pendant une période de forte affluence.
- Puisque les quantités commandées sont identiques, le gestionnaire a la possibilité de commander de manière plus économe : un moins grand nombre de commandes diminue les frais d'expédition et la quantité à stocker invariable permet de prévoir un espace de stockage de taille optimale.

Inconvénients

- L'entreprise peut être amenée à commander à tout moment, ce qui est parfois problématique si le fournisseur n'est pas en mesure de répondre favorablement à la demande pour une raison quelconque.
- Cette méthode demande un suivi administratif constant et minutieux, parfois chronophage. C'est pourtant nécessaire afin de s'assurer que les commandes seront passées au bon moment.
- Avec cette méthode, l'entreprise va continuellement conserver un stock de sécurité (qui lui permet de continuer à fonctionner entre le moment où la livraison est passée et celui où elle réceptionne la commande). Ce sont donc des frais supplémentaires auxquels l'entreprise doit faire face de manière permanente.

II.4.4. Le réapprovisionnement à la commande

C'est la technique la plus souple mais en même temps la plus difficile à maîtriser. Ici, le responsable des stocks va commander des quantités variables à des dates variables elles

aussi. Cette méthode de gestion des stocks est adaptée pour certains cas particuliers, comme pour des projets sur une durée définie ou la commande se fait d'une seule traite en amont du démarrage par exemple (Le secteur du BTP utilise cette méthode pour un projet de construction dans certains cas). C'est également une option intéressante dans le cas d'une entreprise ayant besoin de commander de la marchandise onéreuse et/ou rare si elle ne peut pas vraiment déterminer à quel moment son stock sera écoulé ni des quantités dont elle aura besoin.

Avantages

- Pour de la marchandise qui n'est pas commandée de manière récurrente, c'est une bonne méthode : elle évite des immobilisations de capitaux inutiles.
- Comme évoqué plus haut, si l'entreprise n'a pas ou peu de visibilité sur sa fréquence de commande ou de la quantité dont elle va avoir besoin à court terme, cela peut être la meilleure option à sélectionner.

Inconvénients

- Compliquée à utiliser, c'est une méthode qui demande une bonne connaissance de la gestion des stocks de l'entreprise. Elle requiert par ailleurs beaucoup d'attention et d'investissement pour être réellement efficace (analyse des stocks, nécessité de recommander ou non ...).
- Les fournisseurs peuvent être frileux envers cette méthode, puisqu'elle ne leur permet pas d'avoir une visibilité suffisante. Par ailleurs, l'entreprise peut se retrouver en difficulté si son fournisseur n'est pas en mesure de lui procurer ce dont elle a besoin en cas de commande non prévue.

II.4.5. Le "drop shipping"

Cette méthode, assez répandue outre-Atlantique, le drop shipping se développe en Europe depuis quelques années. Mais que signifie cet anglicisme ? Concrètement, quand une entreprise utilise le drop shipping, elle ne possède aucun stock ! Impossible ? Et si, c'est tout à fait réalisable. Petit tour d'horizon.

Lorsque la société enregistre une commande client, elle s'occupe d'encaisser le client et de récupérer sa marge. Ensuite, elle envoie le récapitulatif de commande à un fournisseur qui possède le stock nécessaire. C'est lui, qui par la suite, se charge de préparer et d'expédier le colis.

Avantages

- Se libérer de toute la partie gestion des stocks représente un sérieux gain de temps !

- Le fonds de roulement de l'entreprise sera positif l'immense majorité du temps. Intéressant pour les dirigeants qui démarrent leur activité et ne sont pas forcément avec ces notions.
- Toute la partie logistique est gérée par le fournisseur. Les coûts (emballages, expédition) sont par conséquent supprimés (sauf si l'entreprise est chargée de participer), et c'est un autre gain de temps pour le l'entreprise.

Inconvénients

- C'est une méthode de gestion des stocks plus adaptée pour du e-commerce. Elle est relativement peu conseillée pour du commerce physique.
- Risque de devoir faire face à du mécontentement client accru. En effet, c'est le fournisseur qui s'occupe de l'emballage ainsi que de l'expédition. Pour autant, c'est l'entreprise que le client appellera si jamais son colis lui parvient en mauvais état : elle devra en assumer les conséquences. Il est donc très important de bien choisir son fournisseur au préalable !
- Les délais de livraison peuvent être particulièrement longs, notamment dans le cas où le fournisseur est situé à l'extérieur de l'UE.

II.5. Les principales méthodes de réapprovisionnement des stocks en entrepôt

L'objectif final du processus de réapprovisionnement des stocks est d'éviter que l'opérateur se déplace pour aller chercher un produit et qu'il se retrouve face à des rayonnages vides. Il existe trois principales méthodes [11] :

II.5.1. La méthode du réapprovisionnement des stocks à la demande

Avec le réapprovisionnement des stocks à la demande, les produits faisant partie des commandes quotidiennes sont sélectionnés et placés sur les rayonnages. Cette technique de réapprovisionnement des stocks est généralement utilisée lorsque le picking suit la méthode du wave picking (picking par vague). Ainsi, les références ultérieurement emballées, préparées et expédiées, sont choisies lors du réapprovisionnement.

Le réapprovisionnement des stocks à la demande est dynamique et est réajusté continuellement lors de chaque vague. Cette technique est surtout utilisée :

- Si la zone de préparation de commandes ne dispose pas de suffisamment de place pour entreposer les produits correspondants à chacune des références.
- Si la demande est très variable et donc difficile à prévoir.

II.5.2. Le réapprovisionnement des stocks avec rupture de stock minimum programmée

Avec cette technique, le réapprovisionnement des stocks est enclenché dès qu'un produit atteint la limite du stock minimum. De plus, les mouvements dédiés au réapprovisionnement sont minimisés et réservés aux situations strictement nécessaires.

Cette méthode est recommandée si :

- La demande est stable ou qu'il s'agit de produits à faible rotation. En effet, il n'est pas nécessaire de réapprovisionner instantanément le stock de ce type de références, il est possible de programmer un stock de sécurité qui vous indique à quel moment le reconstituer.
- Vous disposez de suffisamment de place sur vos rayonnages.

II.5.3. La stratégie de réapprovisionnement des stocks « top off », basée sur le Lean time

Le réapprovisionnement des stocks top off tire profit des fluctuations de l'activité de préparation de commandes de l'entrepôt pour remplir les rayonnages de picking.

Il s'agit d'une technique très souple, qui optimise les flux de travail. Nous l'utilisons également pour ravitailler les stocks à l'avance lorsque l'entrepôt est en période de faible activité et qu'une nouvelle vague de commandes est prévue.

La grande différence entre ce type de réapprovisionnement des stocks et le précédent est que les marchandises sont approvisionnées en continu (pas besoin d'attendre que le stock minimum soit atteint. Nous avons recours au réapprovisionnement top off principalement dans les cas suivants:

- Pour les références au taux de rotation très élevé et dont il est nécessaire d'avoir du stock à n'importe quel moment pour satisfaire la demande.
- Lorsque l'activité de préparation de commandes est très importante, cela permet d'éviter l'étape de ravitaillement avant l'extraction des marchandises et d'optimiser les temps en suivant la logique du Lean logistics.

II.6. Les outils de gestion de stock

Vous l'aurez compris, la gestion des stocks est une tâche complexe qui requiert l'utilisation d'outils appropriés. Il existe des logiciels de gestion des stocks ainsi que d'autres outils comme les codes-barres, les scanners et appareils de collectes de données portables [12].

II.6.1. Gestion de stock sur Excel (tableau de gestion de stock)

Certaines sociétés utilisent encore Excel pour gérer leurs stocks : cela demande un suivi très fin et une actualisation manuelle rigoureuse et régulière.

II.6.2. Logiciel ou application de gestion de stock

Pour les entreprises qui font face à un gros volume de stocks, il est incontournable de gérer les stocks d'une entreprise grâce à un logiciel dédié. Ils aident en effet les entreprises à gérer efficacement leurs stocks, à prévoir la demande future, à mieux gérer les commandes et les livraisons et à réduire les coûts de stockage.

II.7. Le fonctionnement de la gestion du stock

Le stock correspond aux marchandises ou aux matériels qu'une entreprise a l'intention de vendre à ses clients dans un but de rentabilité. La gestion des stocks, élément stratégique de la chaîne d'approvisionnement, désigne le suivi des stocks depuis les fabricants jusqu'aux entrepôts, puis jusqu'à un point de vente. L'objectif de la gestion des stocks est de disposer des bons produits au bon endroit et au bon moment. Cela nécessite une visibilité des stocks : savoir quand passer une nouvelle commande, combien commander et où entreposer le stock. Les étapes de base de la gestion des stocks sont les suivantes [13] :

- ❖ **Achat du stock** : les marchandises prêtes à vendre sont achetées et livrées à l'entrepôt ou directement au point de vente.
- ❖ **Entreposage du stock** : le stock est entreposé jusqu'à utilisation. Les marchandises ou les matériels sont transférés via votre réseau d'exécution jusqu'à ce qu'ils soient prêts à être expédiés.
- ❖ **Bénéfices du stock** : Le montant du produit à vendre est contrôlé. Les produits finis sont extraits pour exécuter les commandes. Les produits sont expédiés chez les clients.

2. Conclusion

Ce chapitre nous a permis en premier lieu de découvrir et de mieux comprendre une généralité sur la GMAO, où nous avons discuté sur les caractéristiques, les avantages, les inconvénients, les conditions et le guide du choix, puis nous avons discuté sur le stock, l'achat et le réapprovisionnement.

Dans le prochain chapitre, nous passerons à l'analyse des besoins et la conception utilisée dans le projet.

Chapitre 03

**Conception Et Description
Du Projet**

1. Introduction

La réalisation d'un système suit généralement des méthodes qui conduisent à modéliser et construire des produits de manière fiable. C'est dans cet objectif que nous abordons dans ce chapitre les différentes étapes pour modéliser notre système en utilisant UML 2.0 (Unified Modeling Language) qui est le plus prépondérant actuellement en étant un standard industriel de modélisation orientée objet.

Nous considérons deux étapes principales au cycle de développement : la spécification et la conception. La première correspond à l'analyse des besoins et la modélisation d'un point de vue logique du système, et la deuxième permet d'adapter et d'affiner la structure objet pour l'environnement d'implantation.

2. Les besoins

Dans cette partie, nous allons définir les différents acteurs, les besoins fonctionnels et non fonctionnels de notre système.

2.1. Les acteurs du système

Le système de la gestion de stock et approvisionnement met en interaction trois acteurs avec le système :

- **L'Administrateur** : c'est le responsable de la gestion des fournisseurs et la gestion sociétés et des stock (achat et vente).
- **Le Fournisseur** : c'est une personne qui recevoir les commandes de l'admin et envoyer un message si la commande disponible ou non.
- **La Société** : c'est une petite entreprise qui envoyer les commandes à l'admin s'il y a un manque du stock.

2.2. Besoins fonctionnels

Les besoins fonctionnels doivent répondre aux exigences du futur système en termes de fonctionnalités. Ils constituent une sorte de promesse ou de contrat au comportement du système. De ce fait, La solution proposée doit répondre aux besoins fonctionnels suivants :

- ❖ Afficher une application claire et facile à lire par un utilisateur quelconque ;
- ❖ Permettre à l'utilisateur (Admin) de gérer (Ajouter, modifier, supprimer) les fournisseurs ;
- ❖ Permettre à l'utilisateur (Admin) de gérer (Ajouter, modifier, supprimer) les sociétés ;

- ❖ Permettre à l'utilisateur (Admin) de gérer (Ajouter, modifier, supprimer) le stock (l'achat et le vente) ;
- ❖ Permettre à l'utilisateur (Administrateur) d'envoyer les commandes au fournisseur d'un part et à la société d'autre part ;
- ❖ Permettre à l'utilisateur (Société) du déclarer le manque des pièces en envoyant des commandes à l'administrateur ;
- ❖ Permettre à l'utilisateur (Fournisseur) du répondre sur l'admin s'il y a ou non ;
- ❖ Permettre à l'utilisateur (Fournisseur) du voir les commandes envoyées par l'admin ;
- ❖ Faciliter la communication et la connectivité entre les utilisateurs ;
- ❖ Permettre de faciliter l'utilisation pour l'utilisateur sans déplacement géographique.

2.3. Besoins non fonctionnels

Les besoins non fonctionnels peuvent être considérés comme des besoins fonctionnels spéciaux. Parfois, ils ne sont pas rattachés à un cas d'utilisation particulier, mais ils caractérisent tout le système (l'architecture, la sécurité, le temps de réponse, etc.). Le système doit garantir les besoins opérationnels suivants :

- **Besoins matériels** : Le système doit s'exécuter de la même façon sur les différents systèmes d'exploitation et différents navigateurs.
- **Besoins de déploiement** : Le système doit assurer la facilité de la mise en place et du déploiement d'application du stock et de réapprovisionnement.
- **Besoins de performance** : Le système devra répondre rapidement au besoin de l'utilisateur.
- **Précision** : La précision du système doit être bornée par rapport à internet.
- **Besoins de disponibilité /fiabilité** : Le système doit :
 - ✓ Être disponible pour les utilisateurs lorsqu'il le demande donc une utilisation 24/24 heures, 7/7 sauf période de maintenance et mise à jour.
 - ✓ Fonctionner correctement lors de la demande de l'utilisateur.
- **Besoins de robustesse** : Le système doit être capable de s'adapter aux changements de l'environnement.
- **Besoins de maintenance** : Le système doit être facile à utiliser et à maintenir.
- **Besoins de flexibilité** : Le système doit être souple pour une extension future.

- **L'ergonomie des interfaces** : Doit présenter une interface claire, facile, ergonomique et intuitive.

3. UML

3.1. Définition

UML2.0 est un Langage de modélisation objet unifié, est une démarche orientée objet. Elle est née de la fusion de trois méthodes orientées objet Booch, OMT (Object Modeling Technique) et OOSE (Object Oriented Software Engineering), conçues respectivement par Grady Booch, James Rumbaugh et Ivar Jacobson [14].

UML est intrinsèquement orienté vers la construction de logiciels orientés objet ce langage est utilisé pour faire des diagrammes pour décrire les programmes informatiques en termes de leurs éléments constitutifs ou le déroulement des opérations que le programme exécute ; plus précisément, la qualité logicielle est la programmation orientée objet.

UML ne se limite pas au génie logiciel, mais est également utilisé dans l'ingénierie des systèmes et la représentation des structures organisationnelles avec UML vous pouvez créer des modèles et créer une conception intégrée pour votre projet logiciel cela peut également être fait par des dessins (schémas) en produisant un code initial pour le programme.

3.2. Historique d'UML

Date	Description
Au début des années 1980	Les objets commencent à quitter les laboratoires de recherche et à faire leurs premiers pas dans le monde réel; entre autres, le langage de programmation Small Talk, stabilisé, devient une plate-forme utilisable et le C++ voit le jour. Les méthodes objets commencent à émerger pour remplacer les méthodes structurée et fonctionnelles, trop liés à la machine.
1989 à 1994	Le nombre de méthodes orientées objet passe de dix à plus de cinquante ; toutes ces méthodes ont de nombreux points communs (objets, méthodes, paramètres, etc.). Ces méthodes s'orientant sur l'abstraction des composants matériels, se basent sur des notions de classe, d'association, de partition en sous-

	<p>systèmes et autour de l'étude de l'interaction entre utilisateur et le système. Les principaux auteurs de ces méthodes sont James Rumbaugh, Grady Booch et Ivar Jacobson. Parmi ces méthodes, deux s'imposent : la méthode de Booch et la méthode OMT (Object Modeling Technique). Les deuxièmes versions des méthodes de Booch et OMT font leur apparition : Booch'93 et OMT-2. Ces méthodes sont assez semblables, mais Booch'93 insiste plus sur la construction tandis qu'OMT-2 insiste plus sur l'analyse et l'abstraction.</p>
1989 et 1991	<p>Publication de deux ouvrages, par Sally Shlaer (en) et Steve Mellor sur l'analyse et la conception, qui débouchent sur une approche qu'ils nomment conception réursive.</p>
1989 à 1990	<p>Développement à Portland, par la communauté Smalltalk, de la conception pilotée par les responsabilités et les cartes CRC (Class-Responsability-Collaboration).</p>
1991 à 1996	<p>James Rumbaugh dirige aux laboratoires de recherche de General Electric une équipe de recherche qui publie un ouvrage très apprécié sur OMT.</p>
1991	<p>Publication d'ouvrages, par Peter Coad et Ed. Yourdon, qui développent les approches « allégées » et « orientées prototypes ».</p>
1992 et 1995	<p>Publication des livres d'Ivar Jacobson fondés sur son expérience des commutateurs téléphoniques chez Ericsson. Le premier introduit le concept de cas d'utilisation (use-case).</p>
1994 à 1996	<p>Grady Booch effectue un travail important chez Rational Software en développant des systèmes en Ada.</p>
1994	<p>Le nombre important de méthodes et le fait que les différences entre-elles se réduisent, font reculer la technologie objet au point que James Rumbaugh et Grady Booch s'unissent afin d'unifier leur travaux. Ils proposent une « méthode unifiée ».</p>
1994	<p>Les livres de Jim Odell, écrits avec James Martin, se fondent sur sa longue expérience des systèmes d'information et du génie logiciel et sont, parmi tous ces ouvrages, les plus conceptuels.</p>

Octobre 1994	Début des travaux de la méthode unifiée (<i>unified method</i> (UM)). James Rumbaugh rejoint Grady Booch chez Rational Software.
1995	Ivar Jacobson, créateur des <i>use cases</i> , rejoint James Rumbaugh et Grady Booch.
1995	Les auteurs de la méthode unifiée (UM) publient le document intitulé Unified Method V0.8 .
Octobre 1995	Ivar Jacobson arrive chez Rational Software.
Octobre 1995	UML 0.8 inclut OOD/Booch '93 de Grady Booch et OMT de James Rumbaugh.
1996	Publication d'une nouvelle révision du document, Unified Method V0.9 , à la suite des commentaires des utilisateurs. La révision 0.9.1 est la version la plus aboutie de la méthode unifiée (réorientation de la portée de l'effort d'unification). La méthode change de nom et se transforme en UML (Unified Modeling Language for Object-Oriented Development). Un consortium de grandes entreprises se crée (Microsoft, IBM, Oracle, etc.) qui permettra de faire passer la méthode à sa version 1.0.
Juin 1996	UML 0.9 inclut OOSE d'Ivar Jacobson.
Octobre 1996	UML 0.91
Janvier 1997	Normalisation d'UML 1.0 par l'Object Management Group (OMG).
Août 1997	Proposition des spécifications d'UML 1.1 à l'OMG par un groupe de travail d'analystes et de concepteurs dirigé par Cris Kobryn et administré par Ed Eykholt.
14 novembre 1997	Adoption des spécifications d'UML 1.1 par l'OMG. Différentes améliorations continuent d'être apportées au standard UML, donnant naissance à quatre révisions : UML 1.2, 1.3, 1.4, 1.5. UML 1.5 est la dernière révision avant le passage à la version UML 2.0. Les standards UML 1.x, encore largement influencés par la notation OMT, sont critiqués comme manquant d'intégration sémantique.
Juin 1998	Adoption d'UML 1.2 par l'OMG.
Octobre 1998	Adoption d'UML 1.3 par l'OMG.

Mars 2000	Publication de la spécification UML 1.3 complète.
Septembre 2001	UML 1.4.
6 mars 2003	UML 1.5 (recommandations)
Août 2003	UML 2.0 Superstructure Specification (recommandation)
1 ^{er} septembre 2003	UML 2.0 Diagram Interchange Specification (recommandation)
14 octobre 2003	UML 2.0 OCL Specification
Décembre 2003	UML 2.0 (recommandation)
Juillet 2005	Adoption d'UML 2.0 par l'OMG.
4 avril 2006	UML 2.0 Diagram Interchange Specification
1 ^{er} juin 2006	<i>deployment view</i>
6 octobre 2006	UML 2.1.1 - XMI file
6 février 2007	UML 2.1.1 Infrastructure Specification
3 février 2007	UML 2.1.1 Superstructure Specification
2007	UML 1.4.2 devient une spécification ISO (ISO/IEC 19501).
Novembre 2007	Diffusion d'UML 2.1.2 par l'OMG.
Janvier 2009	Diffusion d'UML 2.2 par l'OMG.
Mai 2010	Diffusion d'UML 2.3 par l'OMG.
Juillet 2011	Diffusion par l'OMG d'UML 2.4.1. Infrastructure et Superstructure sont révisées en août 2011.
Décembre 2017	Diffusion par l'OMG d'UML 2.5.1. Le métamodèle lui-même est inchangé depuis la superstructure d'UML 2.4.1, avec quelques exceptions.

Tableau III.1 – Historique d'UML [15]

3.3. Les caractéristiques d'UML2.0

UML 2.0 est un langage formel qui possède les caractéristiques suivantes :

- Un langage sans ambiguïtés ;
- Un langage universel pouvant servir de support pour tout langage orienté objet ;
- Un moyen de définir la structure d'un programme ;
- Une représentation visuelle permettant la communication entre les acteurs d'un même projet ;
- Une notation graphique simple, compréhensible même par des non informaticiens.

3.4. Les diagrammes d'UML2.0

UML 2.0 comporte treize types de diagrammes représentant autant de vues distinctes pour représenter des concepts particuliers du système d'information. Ils se répartissent en deux grands groupes à savoir les diagrammes structurels (statiques) et les diagrammes comportementaux (dynamiques). Nous allons citer ci-dessous les principaux diagrammes de chaque catégorie :

3.4.1. Diagrammes structurels : qui contiennent :

- ❖ **Diagramme de classes** : structure statique représentant les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci.
- ❖ **Diagramme d'objets (Object Diagram)** : il sert à représenter les instances de classes (objets) utilisées dans le système.
- ❖ **Diagramme de composants (Component Diagram)** : Il permet de montrer les composants du système d'un point de vue physique.
- ❖ **Diagramme de déploiement (Deployment Diagram)** : Il montre les composants sur les dispositifs matériels.

3.4.2. Diagrammes comportementaux

- ❖ **Diagramme de cas d'utilisation** : Il permet d'identifier les possibilités d'interaction entre le système et les acteurs (intervenants extérieurs au système), c'est-à-dire toutes les fonctionnalités que doit fournir le système.
- ❖ **Diagramme d'activités** : Il permet de décrire sous forme de flux ou d'enchaînement d'activités le comportement du système ou de ses composants.
- ❖ **Diagramme d'états-transitions** : affiche les états possibles de l'objet et les transitions qui provoquent un changement d'état.
- ❖ **Diagramme de séquence (Sequence Diagram)** : représentation séquentielle du déroulement des traitements et des interactions entre les éléments du système et/ou de ses acteurs.
- ❖ **Diagramme de communication (Communication Diagram)** : fournit une représentation graphique des interactions entre les objets d'un scénario de cas d'utilisation, l'exécution d'une opération, ou une interaction entre des classes, en mettant l'accent sur la structure du système.

Ces diagrammes, d'une utilité variable selon les cas, ne sont pas nécessairement tous produits à l'occasion d'une modélisation. Les plus utiles pour la maîtrise d'ouvrage sont les diagrammes d'activités, de cas d'utilisation, de classes, d'objets, de séquence et d'états-

transitions. Les diagrammes de composants, de déploiement et de communication sont surtout utiles pour la maîtrise d'œuvre à qui ils permettent de formaliser les contraintes de la réalisation et la solution technique [16].

3.5. Les Points Forts et Faibles d'UML

3.5.1. Les Points Forts

Tout d'abord, les fonctionnalités d'UML sont les suivantes [17] :

- **Lisible et souple** : le meilleur avantage est que les symboles de Schéma sont facilement lisibles par tout programmeur qui comprend ne serait-ce qu'une petite partie du programme. Grâce à cela, le programmeur informatique peut exécuter le programme à l'aide de codes et les modifier ou les réutiliser à tout moment une autre chose qui fait d'UML une programmation de développement logiciel indispensable est sa flexibilité pour le personnaliser en fonction de l'utilisation la technologie d'une entreprise.
- **Bonne communication du génie logiciel** : le schéma du système est en fait l'architecture du logiciel car c'est le cadre dont dépendent le processus et l'efficacité. UML est le langage complet utilisé pour la modélisation de l'architecture logicielle, il aide à évaluer les performances des utilisateurs ainsi qu'au suivi et à la sécurité et fournit des instructions liées à un processus particulier. Parce qu'il a les caractéristiques d'être très largement accessible, UML est langage idéal pour communiquer des informations visuelles sur la structure d'un programme à un grand nombre de travailleurs qui l'utilisent.
- **Outil de planification pré_ programme** : UML peut-vous aider à planifier votre programme avant qu'il ne soit temps de commencer à le programmer. L'outil peut aider à générer du code qui peut préparer un modèle UML, le diagramme est relativement facile à modifier et peut vous aider à réduire les frais généraux lorsque le programme est finalement exécuté.
- **Vous pouvez corriger n'importe quel problème facilement** : lorsqu'il est testé de manière plus approfondie, le logiciel peut prendre jusqu'à des heures et des heures de recherche pour trouver l'erreur, et il peut également causer des problèmes de système plus tard. Ainsi, le programme est bien conçu à l'aide d'UML afin que chaque tâche porte ses propres symboles, faciles à déboguer pour le programmeur.

3.5.2. Les Points Faibles d'UML

Deuxièmement, les inconvénients d'UML sont les suivants [17] :

- **Cela prend un peu de temps :** le principal inconvénient d'UML est qu'il faut beaucoup de temps pour gérer les graphes pour qu'UML vous donne un excellent travail, le programmeur doit synchroniser le code qui nécessite de la maintenance et du temps.
- **C'est juste une langue :** on peut dire que ce n'est qu'un langage de communication et que les gens que par le comprennent peuvent se parler. Cela dépend totalement de la façon dont vous l'utilisez. Si vous l'utiliser pour un système complet basé sur un modèle, vous ne pouvez obtenir que des informations concernant la traçabilité, ce qui est très important dans certaines industries, mais nécessite une discipline sérieuse. Cependant, UML manque également de connotations formelles dans différents domaines tels que les machines et les structures composites sont deux domaines où des travaux sont effectués pour améliorer la qualité des logiciels, donc l'absence de celui-ci et la possibilité d'entrer pleinement parti peuvent vous empêcher d'atteindre les objectifs que votre entreprise a défini.
- **Le design vole tout le temps :** UML consiste en de nombreuses conceptions qui prennent beaucoup de temps à réaliser et peuvent être écrasantes pour certaines entreprises en raison du temps nécessaire à la réalisation et donc des retards de projet, ce qui conduit les entreprises à ignorer le logiciel réel et à trop analyser le vrai problème, en tombant amoureux du code.
- **Sa complexité peut en éloigner le public :** certains programmeurs pensent qu'il est complexe et qu'ils seraient mieux sans lui, ce qui fait que les nouveaux programmeurs hésitent à l'apprendre ou à l'utiliser.

4. Les diagrammes d'UML

Dans ce qui suit nous allons approfondir la description des besoins précédemment spécifiés par construction des diagrammes de cas d'utilisation correspondants avec une description textuelle associés, par la suite nous proposons quelques diagrammes de séquence pour une meilleure visibilité du système conçue.

4.1. Diagramme de cas d'utilisation

Les cas d'utilisation permettent de définir d'une manière normalisée les relations fonctionnelles entre les acteurs et le système étudié. Ils sont une représentation orientée fonction du système, et permettent de modéliser les attentes des utilisateurs.

La figure ci-dessous illustre différents cas d'utilisation de notre système.

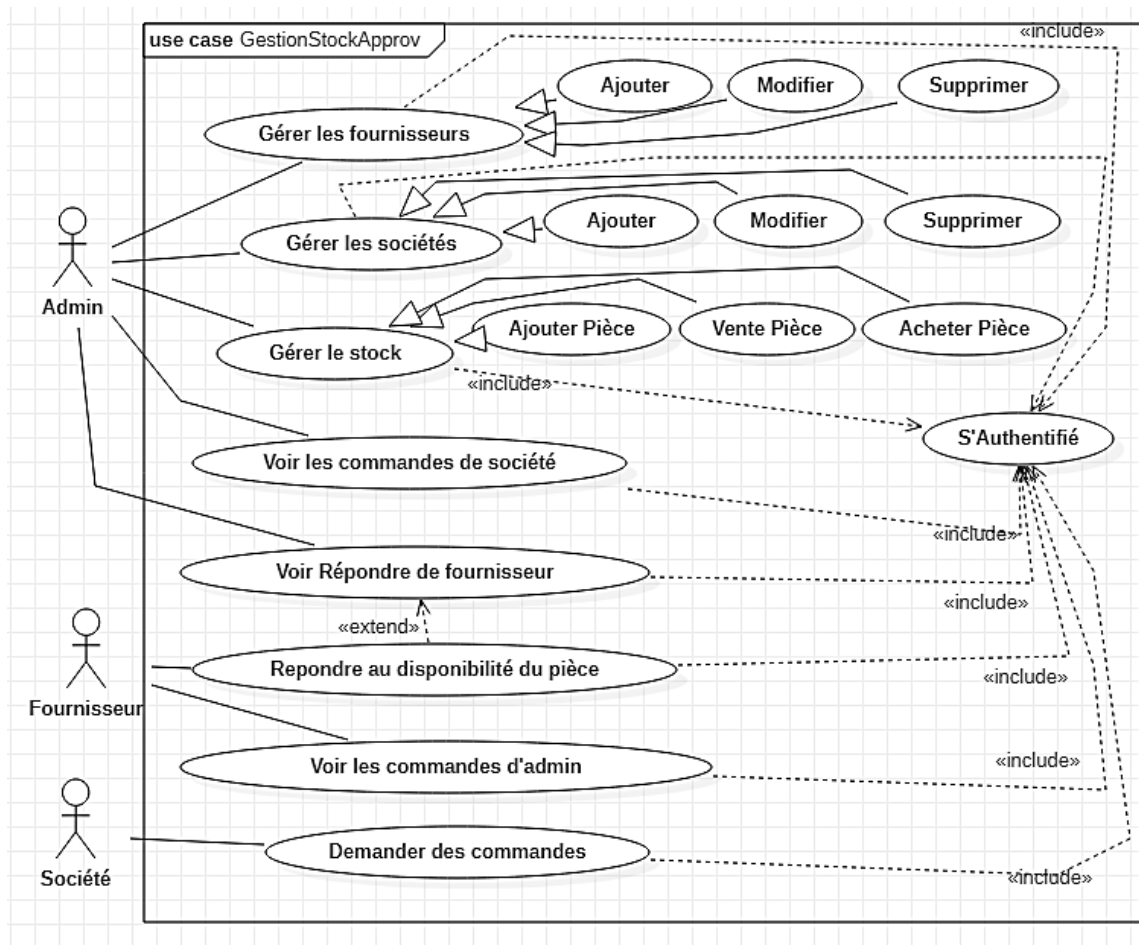


Figure III.1 – Le diagramme de cas d'utilisation global

Ce diagramme décrit une vue générale de l'ensemble des cas d'utilisation relatifs à notre système et décrit éventuellement les différents scénarios susceptibles d'avoir lieu. Une description détaillée de certains cas d'utilisations que nous avons développés sont présentées dans ce qui suit.

4.2. Description textuelle de chaque cas d'utilisation

4.2.1. Description textuelle du cas d'utilisation « s'Authentifier »

Nom	S'Authentifier
Acteur	Administrateur, Fournisseur, Société
Résumé	Décrire le processus utilisé pour se connecter à l'application (soit comme un Admin, Fournisseur ou une Société).
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur ouvre l'interface d'accueil. 2. L'utilisateur se connecte en utilisant le formulaire de connexion en entrant son nom d'utilisateur, son mot de passe et le choix de la catégorie soit comme Fournisseur ou une Société ou un admin. 3. Le système fait l'authentification avec succès.

Alternative	1. En cas de nom d'utilisateur ou de mot de passe erroné, l'authentification échoue et le système le lui signale et lui propose d'essayer d'autres valides. Le cas d'utilisation redémarre à l'étape 2 du scénario nominal.
Pré-condition	L'utilisateur (Fournisseur, Société) n'est pas encore membre.
Post condition	L'utilisateur (Admin, Fournisseur ou Société) fait l'authentification avec succès.

Tableau III.2 – Description textuelle du cas d'utilisation « S'authentifier »

4.2.2. Description textuelle du cas d'utilisation « Gérer les fournisseurs »

Nom	Gérer les fournisseur
Acteur	Admin
Résumé	Décrire le processus utilisé par l'admin pour gérer (Ajouter, modifier, supprimer) des fournisseurs.
Scénario nominal	<ol style="list-style-type: none"> 1. L'admin entre dans le site et authentifié, ensuite entre dans l'interface des fournisseurs. 2. L'admin y'a le droit de gérer (Ajouter, modifier et supprimer) les fournisseurs et notre information. 3. Le système gère les fournisseurs avec succès.
Alternative	
Pré-condition	L'admin ouvre l'interface des fournisseurs.
Post condition	Les fournisseurs gèrent avec succès.

Tableau III.3 – Description textuelle du cas d'utilisation « Gérer les fournisseurs »

4.2.3. Description textuelle du cas d'utilisation « Gérer les sociétés »

Nom	Gérer les sociétés
Acteur	Admin
Résumé	Décrire le processus utilisé par l'admin pour gérer (Ajouter, modifier, supprimer) des sociétés.
Scénario nominal	<ol style="list-style-type: none"> 1. L'admin entre dans le site et authentifié, ensuite entre dans l'interface des sociétés. 2. L'admin y'a le droit de gérer (Ajouter, modifier et supprimer) les sociétés et notre information. 3. Le système gère les sociétés avec succès.
Alternative	

Pré-condition	L'admin ouvre l'interface des sociétés.
Post condition	Les sociétés gèrent avec succès.

Tableau III.4 – Description textuelle du cas d'utilisation « Gérer les sociétés »

4.2.4. Description textuelle du cas d'utilisation « Gérer le stock »

Nom	Gérer le stock
Acteur	Admin
Résumé	Décrire le processus utilisé par l'admin pour gérer (Ajouter, Acheter, Vente) des pièces (stock) du bus.
Scénario nominal	<ol style="list-style-type: none"> 1. L'admin entre dans le site et authentifié, ensuite entre dans l'interface de stock. 2. L'admin y'a le droit de gérer (Ajouter, Acheter, Vente, imprimer et supprimer) les pièces du stock et notre information. 3. Le système gère le stock avec succès.
Alternative	
Pré-condition	L'admin ouvre l'interface de stock.
Post condition	Le stock gère avec succès.

Tableau III.5 – Description textuelle du cas d'utilisation « Gérer le stock »

4.2.5. Description textuelle du cas d'utilisation « Voir les commandes de société »

Nom	Voir les commandes de société
Acteur	Admin
Résumé	Décrire le processus utilisé par l'admin pour Voir les commandes de société qui envoyé.
Scénario nominal	<ol style="list-style-type: none"> 1. L'admin entre dans le site et authentifié, ensuite entre dans l'interface du voir les commandes de société envoyée. 2. L'admin y'a le droit Voir les commandes de société et envoyer la réponse. 3. Le système Voir les commandes de société avec succès.
Alternative	1. Si la commande demandée par la société existe dans le stock, l'admin envoyer un message d'existence, sinon envoyer un message d'attendre pour questionner le fournisseur d'avance. Le cas d'utilisation redémarre à l'étape 2 du scénario nominal.
Pré-condition	L'admin y'a un message de la société pour la commande.

Post condition	L'admin voir les commandes de société avec succès.
----------------	--

Tableau III.6 – Description textuelle du cas d'utilisation « Voir les commandes de société »

4.2.6. Description textuelle du cas d'utilisation « Voir répondre de fournisseur »

Nom	Voir répondre de fournisseur
Acteur	Admin
Résumé	Décrire le processus utilisé par l'admin pour Voir la réponse envoyée de fournisseur pour la commande (les pièces) demandée existe ou non.
Scénario nominal	<ol style="list-style-type: none"> 1. L'admin entre dans le site et authentifié, ensuite entre dans l'interface du Voir répondre de fournisseur envoyé. 2. L'admin y'a le droit Voir la réponse de fournisseur. 3. Le système Voir répondre de fournisseur avec succès.
Alternative	1. Si la commande demandée par l'admin existe au fournisseur, le fournisseur envoyer un message d'existence, sinon envoyer un message d'existe pas. Le cas d'utilisation redémarre à l'étape 2 du scénario nominal.
Pré-condition	L'admin y'a un message de fournisseur pour la commande.
Post condition	L'admin Voir répondre de fournisseur avec succès.

Tableau III.7 – Description textuelle du cas d'utilisation « Voir répondre de fournisseur »

4.2.7. Description textuelle du cas d'utilisation « Répondre à la disponibilité du pièce »

Nom	Répondre à la disponibilité du pièce
Acteur	Fournisseur
Résumé	Décrire le processus utilisé par un fournisseur pour Répondre à la disponibilité de la pièce à l'admin.
Scénario nominal	<ol style="list-style-type: none"> 1. Le fournisseur entre dans le site et authentifié 2. Le fournisseur Répondre à la disponibilité des pièces qui envoyés par l'admin. 3. Le système envoyé la réponse de disponibilité de la pièce avec succès.
Alternative	1. Si les pièces disponibles, le fournisseur envoyer un message d'existe, sinon envoyer un message d'existe pas. Le cas d'utilisation redémarre à l'étape 2 du scénario nominal.
Pré-condition	Le fournisseur y'a un message de l'admin pour la disponibilité des pièces.
Post condition	La réponse envoyées à l'admin avec succès.

Tableau III.8 – Description textuelle du cas d'utilisation « Répondre à la disponibilité du pièce »

4.2.8. Description textuelle du cas d'utilisation « Voir les commandes d'admin »

Nom	Voir les commandes d'admin
Acteur	Fournisseur
Résumé	Décrire le processus utilisé par un fournisseur pour Voir les commandes d'admin demandées par l'admin.
Scénario nominal	<ol style="list-style-type: none"> 1. Le fournisseur entre dans le site et authentifié 2. Le fournisseur Voir les commandes d'admin. 3. Le système affiché les commandes avec succès
Alternative	
Pré-condition	Le fournisseur y'a un message de l'admin pour la commande.
Post condition	Les commandes d'admin voir avec succès.

Tableau III.9 – Description textuelle du cas d'utilisation « Voir les commandes d'admin »

4.2.9. Description textuelle du cas d'utilisation « Demander des commandes »

Nom	Demander des commandes
Acteur	Société
Résumé	Décrire le processus utilisé par la société pour Demander des commandes à l'admin.
Scénario nominal	<ol style="list-style-type: none"> 1. La société entre dans le site et authentifié 2. La société Demander des commandes à l'admin. 3. Le système Demandé les commandes avec succès.
Alternative	
Pré-condition	La société y'a un manque du commande (pièce).
Post condition	Les commandes demandées avec succès.

Tableau III.10 – Description textuelle du cas d'utilisation « Demander des commandes »

4.3. Conception





La conception est un processus créatif et d'importance majeure dans le cycle de développement d'un projet. Ainsi cette partie sera consacré à la présentation des différentes étapes de la conception de notre logiciel, et afin de présenter au mieux cette partie nous allons détailler chacun des modules qui composent le système conformément à la notation UML, en présentant les vues statiques décrivant l'état physique du système via le diagramme de classe, les vues dynamiques montrant le fonctionnement du système via le diagramme de séquence basés sur quelques scénarios bien déterminés.

4.3.1. Le diagramme de classe

A) Présentation

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet. Alors que le diagramme de cas d'utilisation montre un système du point de vue des acteurs, le diagramme de classes en montre la structure interne. Il contient principalement des classes. Une classe contient des attributs et des opérations. Le diagramme de classes n'indique pas comment utiliser les opérations : c'est une description purement statique d'un système.

B) Composants De Base

- **Classe** : Une classe est une description d'un ensemble d'objets ayant une sémantique, des attributs, des méthodes et des relations en commun. Un objet est une instance d'une classe. En notation UML, une classe est représentée sous la forme d'un rectangle divisé en trois cases :
 - ✓ **Nom de la classe** : description abstraite d'un ensemble d'objets
 - ✓ **Les attributs** : définit les propriétés de la classe.
 - ✓ **Les opérations** : énumère les méthodes de la classe.
- **Association** : Une association est une relation générique entre deux classes. Elle est modélisée par une ligne reliant les deux classes. Cette ligne peut être qualifiée avec le type de relation, et peut également comporter des règles de multiplicité pour la relation.
- **Composition** : Si une classe ne peut pas exister par elle-même, mais doit  être un membre d'une autre classe, alors elle possède une relation de composition avec la classe contenante. Une relation de composition est indiquée par une ligne avec un "diamant" rempli.
- **Agrégation** : Les agrégations indiquent une relation de contenant contenu.  Elle est décrite par une relation "possède". Une relation d'agrégation est représentée par une ligne avec un "diamant" creux.
- **Généralisation** : Une relation de généralisation est l'équivalent d'une  relation d'héritage en termes orientés objet (relation "est-un"). Une relation de généralisation est indiquée par une flèche creuse se dirigeant vers la classe "parent".
- **Dépendance** : Quand une classe utilise une autre classe, par exemple  comme membre ou comme paramètre de ces fonctions, elle "dépend" ainsi de cette classe. Une relation de dépendance est représentée par une flèche pointillée.

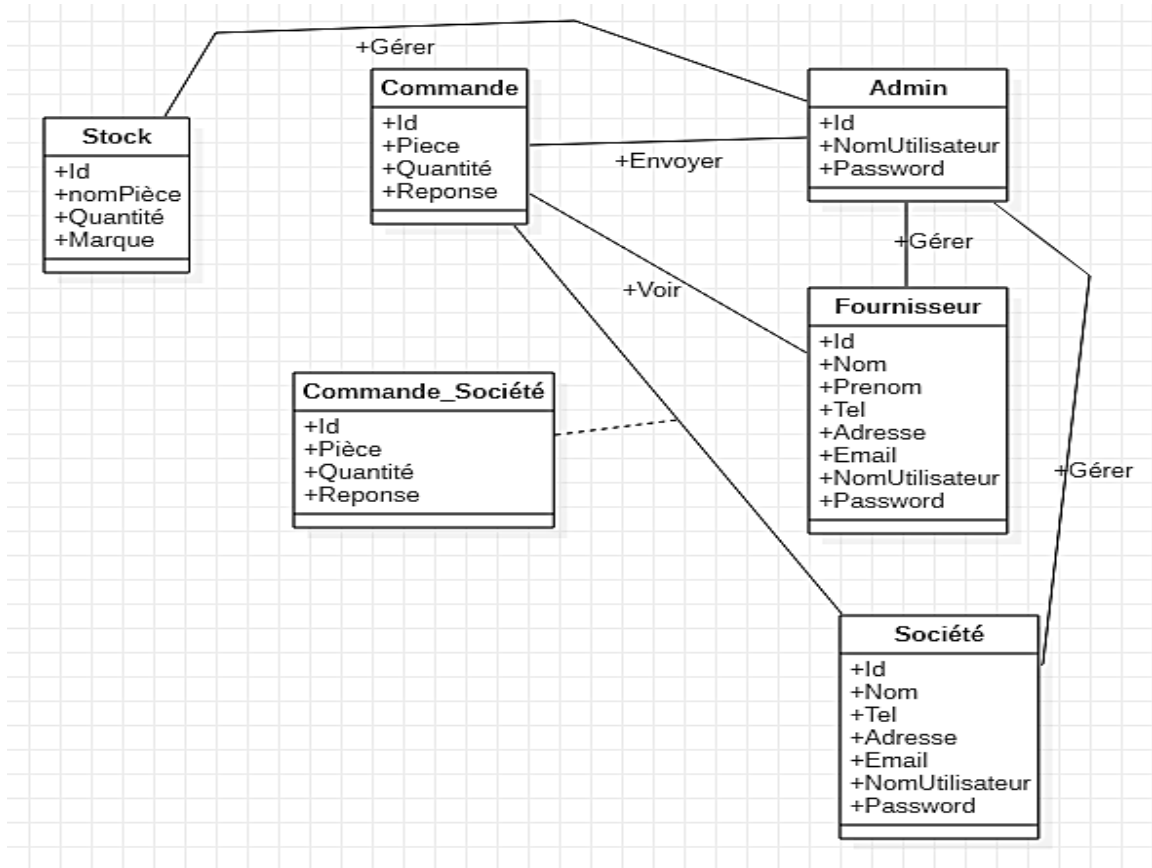


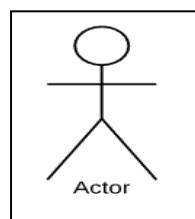
Figure III.2 – Le diagramme de classe de l'application

4.3.2. Diagramme de séquences

Les diagrammes des séquences documentent les interactions à mettre en œuvre entre les classes pour réaliser un résultat, tel qu'un cas d'utilisation. UML étant conçu pour la programmation orientée objet, ces communications entre les classes sont reconnues comme des messages. Le Diagramme des séquences énumère des objets horizontalement, et le temps verticalement. Il modélise l'exécution des différents messages en fonction du temps.

A) Composants De Base

- **Acteur** : Les acteurs peuvent également communiquer avec des objets, ainsi ils peuvent eux aussi être énumérés en colonne. Un acteur est modélisé en utilisant le symbole habituel : « Stickman ».

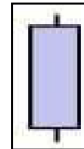


- **Objet** : Les objets sont des instances des classes, et sont rangés horizontalement. La représentation graphique pour un objet est similaire à une classe (un rectangle) précédée du nom d'objet (facultatif) et d'un point-virgule (:).

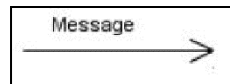
- **Ligne de vie :** Les lignes de vie, *Life Line*, identifient l'existence de l'objet par rapport au temps. La notation utilisée pour une ligne de vie est une ligne pointillée verticale partant de l'objet.



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



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



B) Les diagrammes de séquences

1- Diagramme de séquence du cas « S'Authentifier »

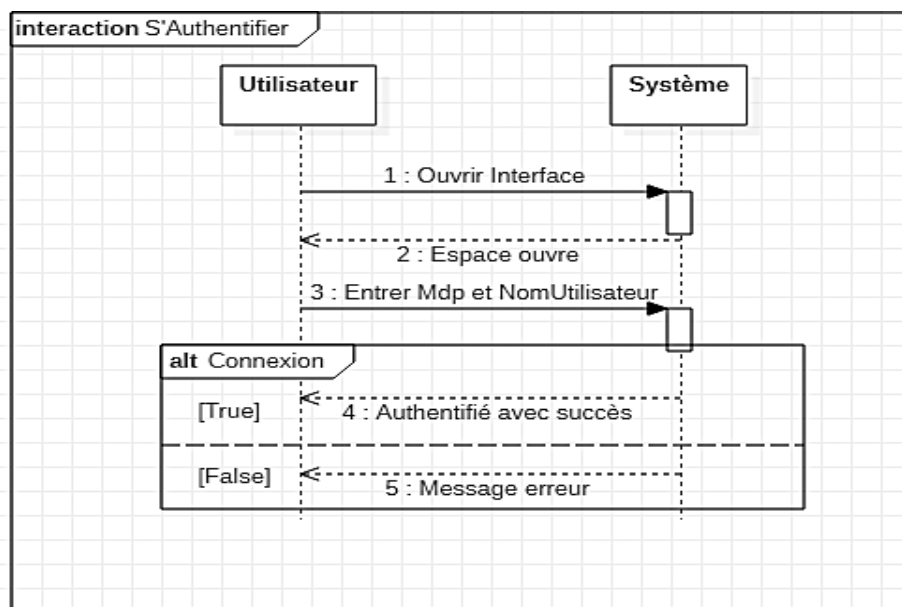


Figure III.3 – Diagramme de séquence du cas « S'Authentifier »

2- Diagramme de séquence du cas « Gérer les fournisseurs »

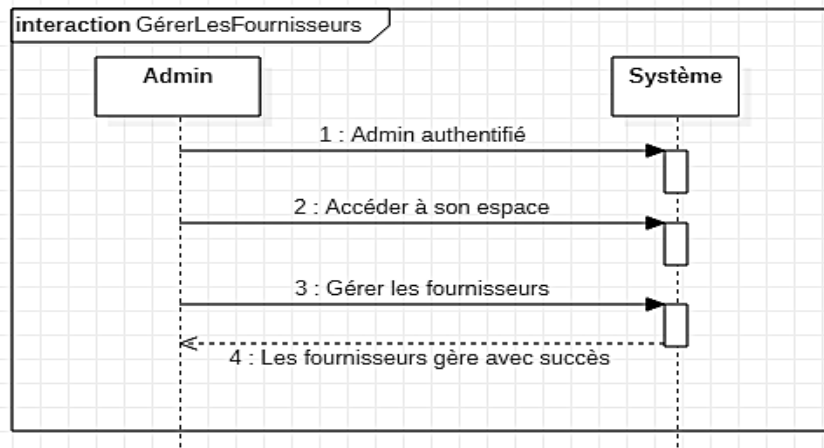


Figure III.4 – Diagramme de séquence du cas « Gérer les fournisseurs »

3- Diagramme de séquence du cas « Gérer les sociétés »

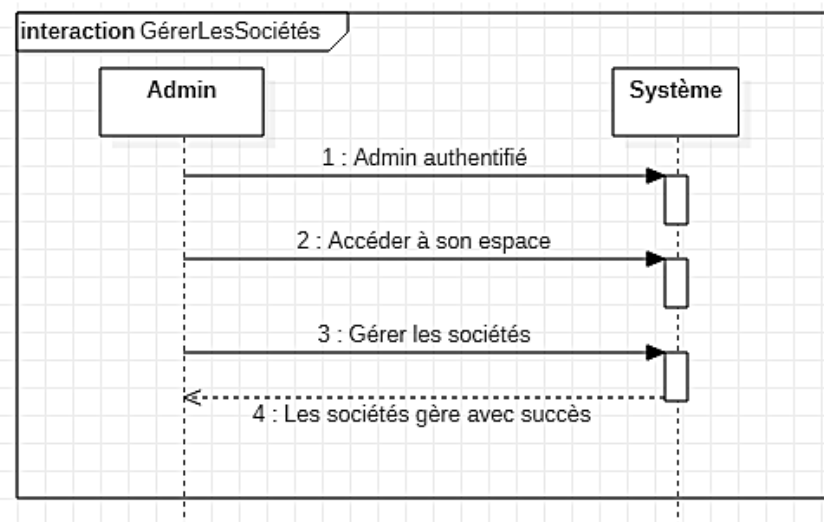


Figure III.5 – Diagramme de séquence du cas « Gérer les sociétés »

4- Diagramme de séquence du cas « Gérer le stock »

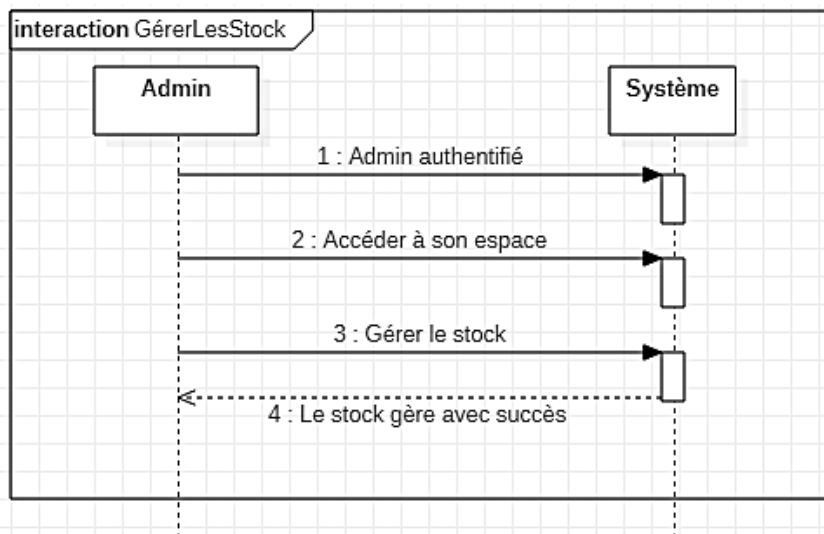


Figure III.6 – Diagramme de séquence du cas « Gérer le stock »

5- Diagramme de séquence du cas « Voir les commandes de société »

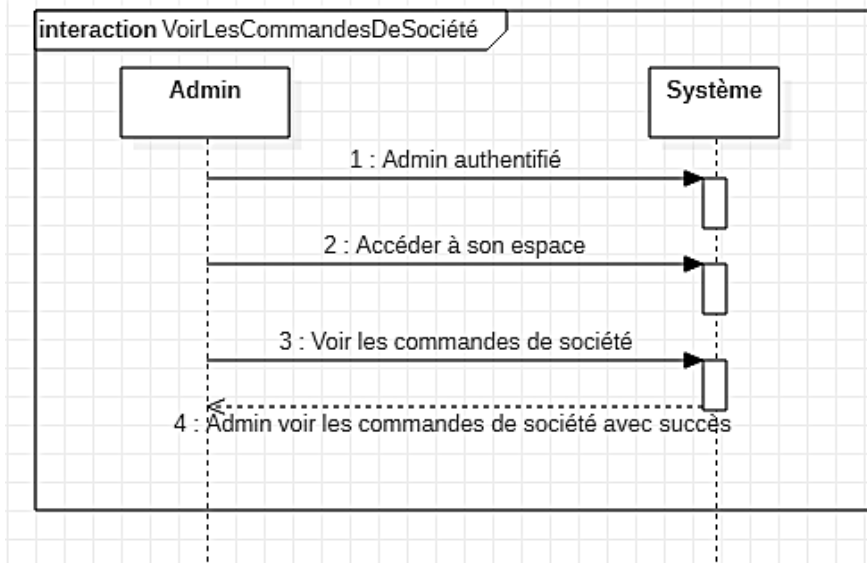


Figure III.7 – Diagramme de séquence du cas « Voir les commandes de société »

6- Diagramme de séquence du cas « Voir répondre de fournisseur »

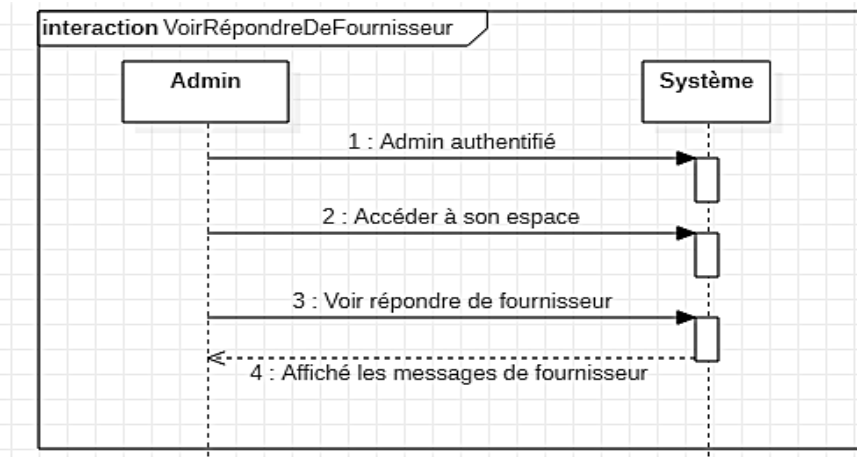


Figure III.8 – Diagramme de séquence du cas « Voir répondre de fournisseur »

7- Diagramme de séquence du cas « Répondre à la disponibilité du pièce »

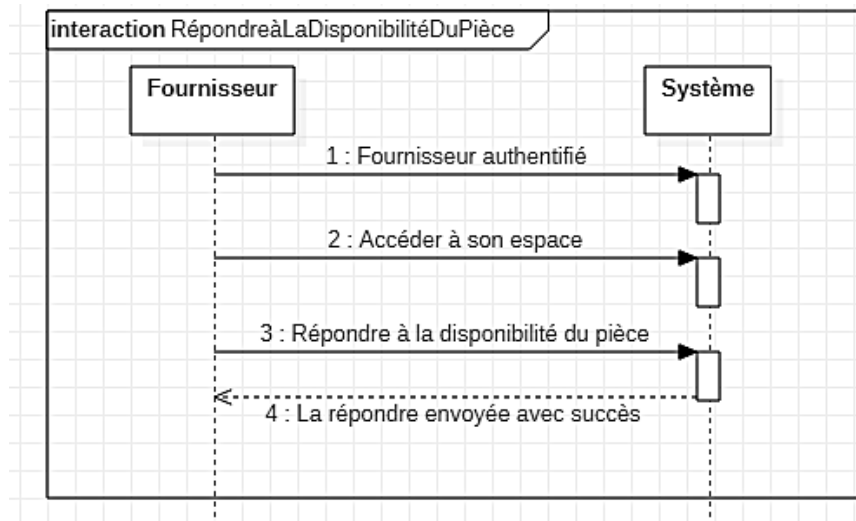


Figure III.9 – Diagramme de séquence du cas « Répondre à la disponibilité du pièce »

8- Diagramme de séquence du cas « Voir les commandes d'admin »

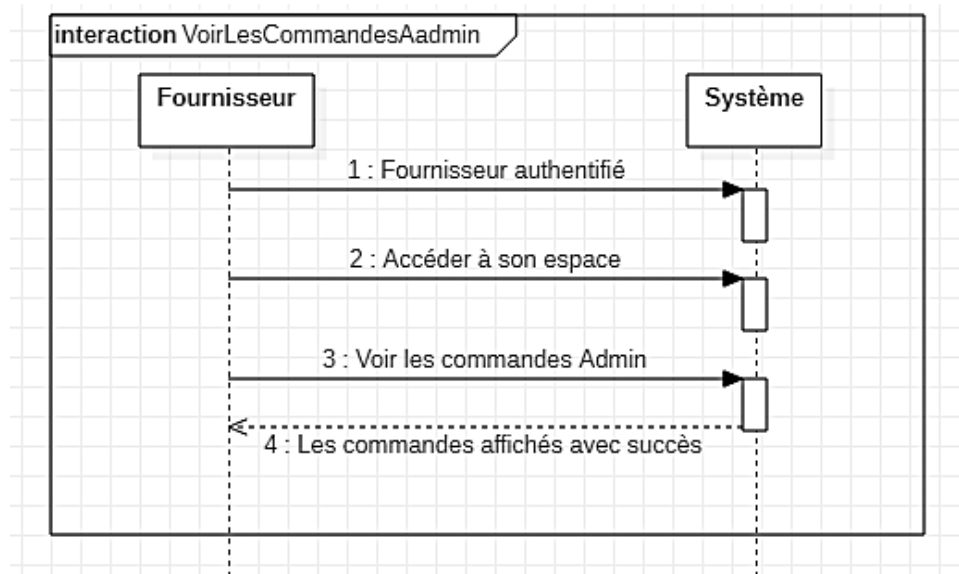


Figure III.10 – Diagramme de séquence du cas « Voir les commandes d'admin »

9- Diagramme de séquence du cas « Demande des commandes »

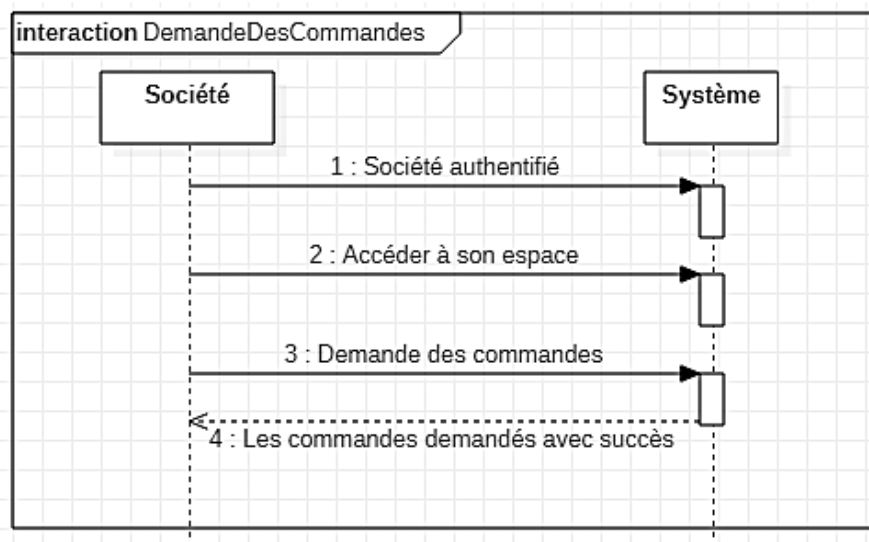


Figure III.11 – Diagramme de séquence du cas « Demande des commandes »

5. Conclusion

Dans ce chapitre, nous avons cerné les objectifs de notre application Gestion de stock et approvisionnement. Cette phase nous a permis de bien élaborer les patterns de conception utilisés pour concevoir notre système. Par la suite, nous avons passé à la conception détaillée pour expliquer les classes de notre système. Ensuite, nous avons conçu quelques aspects dynamiques de notre système en se basant sur le Diagramme de séquence.

Dans le prochain chapitre, nous passerons à une description de la réalisation du projet.

Chapitre 04

Implémentation Du Projet

1. Introduction

Ce chapitre constitue la dernière étape conduisant à la réalisation de notre application web (Gestion de stock et réapprovisionnement) à savoir l'étape d'implémentation. Celle-ci consiste à traduire le résultat obtenu lors de l'étape de conception en un programme exécuté sur une machine en utilisant les outils de programmation adaptés au problème à traiter.

Dans ce qui suit, nous allons décrire les composants de l'environnement de travail, ainsi que les outils d'implémentation et on donnera quelques exemples d'interfaces de notre application web.

2. Environnement de développement

La réalisation de ce projet a nécessité l'utilisation d'un nombre d'outils et de technologies cités ci-dessous.

2.1. Environnement matériel

Afin de réaliser notre système, nous avons utilisé PC portable Sony VAIO et imprimante CANON MF3010 avec les caractéristiques suivantes :

- Processeur : I3-3217U
- RAM : 8 GB DDR3
- Disque dur : 500 GB
- Carte réseau Ethernet PCI-E 10/100/1000 MB



Figure IV.1 – PC Sony Vaio

2.2. Environnement logiciel

Tout le long de la phase de développement, nous avons utilisé l'environnement logiciel suivant :

2.2.1. Langages de programmation

- **HTML** : Le *HyperText Markup Language*, généralement abrégé **HTML** ou dans sa dernière version **HTML5**, est le langage de balisage conçu pour représenter les pages web. Il est souvent utilisé conjointement avec le langage de programmation JavaScript et des feuilles de style en cascade (CSS). HTML est inspiré du *Standard Generalized Markup Language* (SGML). Il s'agit d'un format ouvert [18].

Ce langage permet :

- ✓ D'écrire de l'hypertexte, d'où son nom ;
 - ✓ De structurer sémantiquement la page ;
 - ✓ De mettre en forme le contenu ;
 - ✓ De créer des formulaires de saisie ;
 - ✓ D'inclure des ressources multimédias dont des images, des vidéos, et des programmes informatiques ;
 - ✓ De créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web.
- **PHP** : il a été créé en 1994 par RasmusLerdorf pour les besoins des pages Web personnelles. À cette époque, PHP signifiait Personnal Home Page. PHP est un langage incrusté dans HTML et compilé du côté serveur ce qui veut dire que c'est le serveur qui va interpréter le code PHP et de générer des codes qui pourra être interprété par le navigateur Web « Un navigateur Web est un logiciel conçu pour consulter le World Wide Web ».

Il dérive du C et du Perl dont il reprend la syntaxe. En 1997, PHP devient un projet collectif et son interpréteur est réécrit par Zeev Suraski et Andi Gutmans pour donner la version 3 qui s'appelle désormais PHP : HyperTextPreprocessor (Préprocesseur Hypertexte). PHP est un langage interprété (un langage de script).

- **CSS** : pour cascading Style Sheets (feuille de style en cascade) est un langage informatique qui sert à décrire la présentation des documents HTML et XML. L'un des objectifs majeurs de CSS est de permettre la stylisation hors des documents. Il est par exemple possible de ne décrire que la structure d'un document en HTML, et de décrire toute la présentation dans une feuille de style CSS séparée.

Les styles sont appliqués au dernier moment. Cette séparation fournit un certain nombre de bénéfices, permettant d'améliorer l'accessibilité, de changer plus

facilement de présentation, et de réduire la complexité de l'architecture d'un document.

- **JavaScript** : le **JavaScript** est un langage informatique utilisé sur les pages web. Ce langage a la particularité de s'activer sur le poste client, en d'autres mots c'est votre ordinateur qui va recevoir le code et qui devra l'exécuter. C'est en opposition à d'autres langages qui sont activés côté serveur. L'exécution du code est effectuée par votre navigateur internet tel que Firefox ou Internet Explorer [19].

- **MySQL** : My Structured Query Language (MySQL) ou mon langage de requêtes structuré est un système de base de données (la plus utilisée sur le net partie des technologies utilisées), développé et diffusé sous double licence en fonction de l'utilisation qui est en fait dans un produit libre la GNU (GPL) General Public License ou dans un produit propriétaire (SUN/ORACLE). MySQL fonctionne sur de nombreux systèmes d'exploitation par exemple : Linux, Mac OS X, NetWare, Solaris, Sun OS, Windows (2000, XP, Vista et 7), etc. Les bases de données sont accessibles en utilisant les langages de programmation C, C++, VB, PHP, Java, etc.

MySQL est l'œuvre d'une société suédois fondée par David Axmark, Allan Larsson et Michael Widenius. La première version de MySQL est apparue le 23 mai 1995 et la plus récente est sortie en avril 2007 Version 6.0 (version alpha) [20].

- **Bootstrap** : est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option. C'est l'un des projets les plus populaires sur la plateforme de gestion de développement GitHub [21].

Comme Bootstrap est constitué d'une collection de syntaxes qui remplissent des fonctions spécifiques, il est logique que le framework contienne différents types de fichiers. Les trois principaux fichiers qui gèrent l'interface utilisateur et les fonctionnalités d'un site web sont : Bootstrap.css, Bootstrap.js et Glyphicons.

- **JQuery** : jQuery est une bibliothèque JavaScript libre et multiplateforme créée pour faciliter l'écriture de scripts côté client dans le code HTML des pages web2. La première version est lancée en janvier 2006 par John Resig [22].

Le but de la bibliothèque étant le parcours et la modification du DOM (y compris le support des sélecteurs CSS 1 à 3 et un support basique de XPath), elle contient de

nombreuses fonctionnalités ; notamment des animations, la manipulation des feuilles de style en cascade (accessibilité des classes et attributs), la gestion des évènements, etc. L'utilisation d'Ajax est facilitée et de nombreux plugins sont présents [22].

2.2.2. Outils de réalisation

- **Visual Studio** : Microsoft Visual Studio est une suite de logiciels de développement pour Windows et mac OS conçue par Microsoft. La dernière version s'appelle Visual Studio 2022.

Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications web ASP.NET, des services web XML, des applications bureautiques et des applications mobiles. Visual Basic, Visual C++, Visual C# utilisent tous le même environnement de développement intégré (IDE) [23].

- **XAMPP** : est un ensemble de logiciels permettant de mettre en place un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (X (cross) Apache MariaDB Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne, de plus, sur les systèmes d'exploitation les plus répandus [24].

Il est distribué avec différentes bibliothèques logicielles qui élargissent la palette des services de façon notable : OpenSSL, Expat (parseur XML), PNG, SQLite, zlib, ... mais aussi différents modules Perl et Tomcat. Bon nombre de personnes critiquent la quantité d'extensions ajoutées qui sont pour la majorité inutiles aux débutants. Une version lite à donc été mise en place [24].

- **Photoshop** : c'est un logiciel de traitement et de **retouche d'images** et de photos produit par la société Adobe. **Photoshop** est devenu le standard en matière de gestion des **images matricielles**. Un logiciel tel que Illustrator, lui, gère l'image numérique sous la forme de vecteurs (on parle alors d'images vectorielles). Nous utilisons pour faire le logo de l'application et l'image de menu principal [25].

3. Présentation du site web

La section suivante comporte quelques extraits de code avec des interfaces de notre site web.

3.1. Extrait du code

Afin de réaliser notre site web on a eu recours aux classes suivantes :

3.1.1. Le Code HTML et PHP pour la page d'index.php

Nous avons créé un code qui inclut diverses technologies (Html, CSS, JQuery, ...) pour développer ce site Web :

```

include('../class/database_connection.php');
$message1= ' non supprimer'; |
$id=$_POST['id'];
$quantite=0;
$query1 = "SELECT quantite FROM stock ";
$stmtementP = $connect->prepare($query1);
$stmtementP->execute();
$rslet = $stmtementP->fetchAll();
foreach($rslet as $row){
    $quantite=$row['quantite'];
}
if($quantite>0){
    $output.=' <h2 class="text-dang">' . $message1 . '</h2>';
    echo $output;
}else{
$query1 = " DELETE FROM `stock` WHERE id_stock=$id ; ";
    $stmtement1 = $connect->prepare($query1);
    $stmtement1->execute();
    $row = $stmtement1->fetchAll();

    $output = '';
    if($stmtement1) {
        $query1 = "SELECT count(*)as count FROM stock ";
        $stmtementP = $connect->prepare($query1);
        $stmtementP->execute();
        $rslet = $stmtementP->fetchAll();
        foreach($rslet as $row){
    
```

Figure IV.2 – Code HTML du projet

Ce code est pour la page index.php du notre site de gestion du stock c'est la page d'accueil réalisé par des codes HTML.

3.1.2. Le Code PHP avec MYSQL en connectant la BDD (gestion)

```

<?php
include('../class/database_connection.php');
if(isset($_POST['submit'])){
    $nom=$_POST['nom'];
    $id=$_POST['id'];
    $quantite=$_POST['quantite'];
    $id_fournisseur =$_POST['fournisseur'];
    $query = "UPDATE commande SET piece = '$nom',quantite='$quantite',id_fournisseur = '$id_fournisseur'";
    $statement = $connect->prepare($query);
    $statement->execute();
    $row = $statement->fetchAll();
    header("refresh:0;url=commande.php");
}
?>

```

Figure IV.3 – Code PHP et MYSQL du projet

3.1.3. Le Code JQuery

```

<th width="2%">nom bus</th>
<th width="2%">numero série bus </th>

<th width="1%">nom pièce</th>
<th width="2%"> quantite</th>

<?php
    $query = "SELECT t.nom ,t.prenom ,nom_bus,
numero_serie,so.quantite as quantiteS ,nom_piece FROM
(((sortiepiece so join panne p on id_panne=p.id ) JOIN bus
b on b.id_bus= p.id_Bus) JOIN stock s on s.id_stock= so.
id_stock ) JOIN technicien t on t.id_technicien=so.
id_technicien ;";
    $statementP = $connect->prepare($query);
    $statementP->execute();
    $rslet = $statementP->fetchAll();
    $i=1;
    foreach($rslet as $row){
        ?>

        <tr>
        <td> <?php echo $i ?> </td>
        <td> <?php echo $row['nomT'] ." " . $row['prenomT']?> </td>
        <td> <?php echo $row['nom_bus']?> </td>
        <td> <?php echo $row['numero_serie']?> </td>
        <td> <?php echo $row['nom_piece']?> </td>
        <td> <?php echo $row['quantiteS']?> </td>
    }
}

```

Figure IV.4 – Code JQuery du projet

3.2. Capture des interfaces de site web

- Nous avons adopté une interface simple et facile pour que l'utilisateur accède à tous les services dont nous disposons sur notre site.



Figure IV.5 – Capture d'écran accueil du notre site web

Sa page est la page accueil pour l'admin du notre application web, c'est une page simple et claire pour tous types d'utilisateurs.



Figure IV.6 – Capture d’écran d’interface de gérer le stock



Figure IV.7 – Capture d’écran d’interface de gérer les sociétés

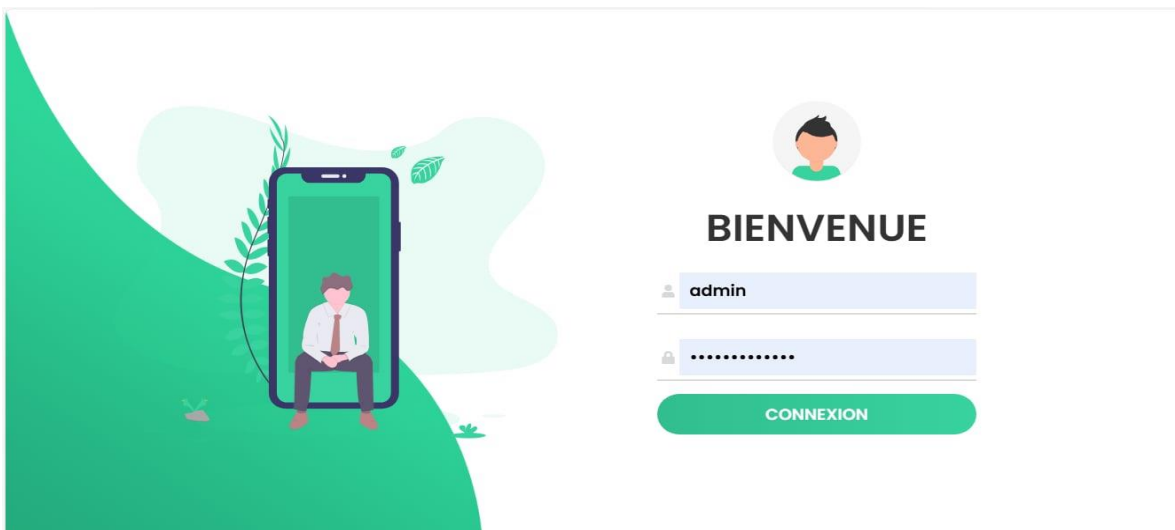


Figure IV.8 – Capture d’écran d’interface du login

5. Conclusion

Dans ce dernier chapitre, nous avons essayé de présenter les notions essentielles relatives à l'implémentation de notre application web gestion de stock et réapprovisionnement.

Pour faciliter la compréhension du processus de réalisation de notre site web, nous avons présenté également les outils, plateformes, et les langages utilisés pour le développement de ce projet.

De plus, nous avons présenté quelques portions de codes ainsi que quelques d'interfaces du logiciel ...

Conclusion générale

Conclusion générale

Tout au long de la préparation de notre projet de fin d'études, nous avons essayé de mettre en pratique les connaissances acquises durant nos études universitaires et cela dans le but de réaliser une application web de gestion de stock et réapprovisionnement en ligne.

Au cours de cette mémoire, nous avons étudié et implémenté une application multifonctions : gérer les fournisseurs, les sociétés et le stock ; Envoyer les commandes des pièces aux fournisseurs ; Demander les commandes par la société ; Répondre par la demande des commandes par le fournisseur ; la sécurité des informations des utilisateurs en utilise plusieurs langages et méthodes.

Ce projet nous a permis d'enrichir nos acquis en conception ainsi qu'en développement. En fait, nous avons acquis une bonne maîtrise des anciens langages comme HTML, CSS, PHP, JavaScript, MySQL ; Encore nous avons appris à connaître des nouveaux langages comme Bootstrap, JQuery, ainsi qu'une solide compétence dans la conception.

Avec la diversité des activités menées, ce projet nous a permis de consolider nos connaissances, essentiellement dans la programmation. Effectivement, l'utilisation de l'approche orientée objet dans le développement de l'application web (gestion de stock et réapprovisionnement) nous a permis d'apprendre à combiner et à transformer un site web en une application web dans un même code.

Comme perspective, nous souhaitons pour les prochains projets de fin d'étude d'élargir notre travail sur la gestion de stock et réapprovisionnement en utilisant le mobile Android ou Mac et Linux, et en réseau internet global.

Bibliographies & Webographies

Bibliographies & Webographies

- [1] Document d'entreprise.
- [2] Benali L (2006). Maintenance industrielle : informatique et maintenance. Alger : Office des publications universitaires. 10-13p.
- [3] <https://www.tribofilm.fr/logiciels/gmao/>
- [4] Monchy F, Vernier JP (2010). Maintenance méthode et organisations : Réussir sa GMAO. Paris : Dunod. 350p.
- [5] Navires et technologie maritime - Spécification pour la gestion de maintenance assistée par ordinateur. Norme NF ISO 23323. Novembre 2021.
- [6] <https://bob-desk.fr/meilleurs-logiciels-gmao/>
- [7] http://elkadiridriss.magix.net/public/Files/polycopie_gestion_des_stocks.pdf
- [8] <https://www.appvizer.fr/magazine/operations/gestion-de-stock/gestion-des-approvisionnements>
- [9] <https://www.oracle.com/dz/erp/procurement/what-is-procurement/>
- [10] <https://agicap.com/fr/article/methodes-de-gestion-des-stocks/>
- [11] <https://www.mecalux.fr/blog/reapprovisionnement-stocks>
- [12] <https://waresito.com/fr/blog/meilleur-outil-gestion-stock/>
- [13] <https://www.ibm.com/fr-fr/topics/inventory-management>
- [14] <https://www.piloter.org/projet/methode/uml.htm>
- [15] <https://www.omg.org/spec/UML/#spec-versions-formal>
- [16] 2013 Laurent AUDIBERT.
- [17] <https://www.clicours.com/les-points-forts-duml/>
- [18] <https://developer.mozilla.org/fr/docs/Web/HTML>
- [19] <https://fr.wikipedia.org/wiki/JavaScript>
- [20] <https://www.mysql.com/fr/>
- [21] <https://www.hostinger.fr/tutoriels/cest-quoi-bootstrap>
- [22] <https://fr.wikipedia.org/wiki/JQuery>
- [23] <https://fr.theastrologypage.com/visual-studio-net>
- [24] <https://fr.wikipedia.org/wiki/XAMPP>
- [25] https://fr.wikipedia.org/wiki/Adobe_Photoshop

