

**UNIVERSITE KASDI MERBAH OUARGLA**

**Faculté des Nouvelles Technologies de l'Information et la Communication**

**Département D'Informatique et des Technologies de l'Information**

**Spécialité :**



**Informatique Fondamentale**



# **Mémoire**

Pour l'obtention du diplôme de **MASTER FONDAMENTALE**

Thème

*Systeme Distribué Pour La Gestion De Production*

Présenté par :

Bentaleb Wahid

Ferhat Hmida Salim

Soutenu publiquement le : 14/06/2014

Devant le jury :

Mr	BEKARI Fouad	Président	UKM Ouargla
Mr	BELMIR Abd-el-Kader	Examineur	UKM Ouargla
Mr	MAHJOUB Mohamed Bachir	Encadreur	UKM Ouargla

Année Universitaire 2013 - 2014

## *DEDICATION*

*I dedicate my master's dissertation work to my family and many friends. A special*

*Feeling of gratitude to my loving parents, whose Words of encouragement and push for tenacity ring in my ears. My sisters and my brother*

*Have never left my side and are very special.*

*I also dedicate this dissertation to my many friends who have Supported me throughout the process. I will always appreciate all they have done,*

*Especially Dr.Mahjoub for helping me develop my technology skills, Dr. Hocine who gives me several advises, and otman for the many hours of proofreading,*

*I dedicate this work and give special thanks to my best Friends LaaniniSadak, Fares, Otman, Taki, Salah, Baker, Moussa,nassim,zouher,mohi,mohamed,khlifa,adel*

*Salim, Ilyes, Halim.*

*And my wonderful brother Mahdi.*

# Table de matière

Dédication.....	1
Table des matières.....	2
Liste des figures.....	9
Liste des tableaux.....	11
Résumé.....	12
Abstract.....	13
Abréviation .....	14
Introduction générale .....	15

## Chapitre 1 : Etude de l'existant

<b>I.1. Introduction</b> .....	17
<b>I.2. Les systèmes de production</b> .....	17
<b>I.2.1 Production</b> .....	17
<b>I.2.2 Gestion de production</b> .....	17
<b>I.2.3 Historique des systèmes de gestion de production</b> .....	17
<b>I.2.4 Généralités sur les systèmes de production</b> .....	18
<b>I.2.5 Les caractéristiques des systèmes de productions</b> .....	19
<b>I. 2.6 Composant d'un système de gestion de production</b> .....	19
<b>I.2.7 Classification des systèmes de productions</b> .....	19
I.2.7.1 La production unitaire .....	19
I.2.7.2 La production en série .....	20
I.2.7.3 La production continue : .....	20
<b>I.3. Présentation de l'entreprise d'accueil</b> .....	21
<b>I.3.1 Historique de l'entreprise</b> .....	21
<b>I.3.2 Définition de l'entreprise</b> .....	22
<b>I.3.3 Les missions de l'entreprise</b> .....	22
<b>I.3.4 Produits développés par l'entreprise</b> .....	22

<b>I.3.5 Capacité de l'entreprise.....</b>	<b>22</b>
I.3.5.1 En moyens humains.....	22
I.3.5.2 Superficie .....	23
<b>I.3.6 Présentation de l'organisme d'accueil .....</b>	<b>24</b>
<b>I.3.7 Document.....</b>	<b>25</b>
I.3.7.1 Forme de bordereau d'envoi .....	25
I.3.7.2 Forme de technique de résultat microbiologique .....	26
I.3.7.3 Forme fiche de lait des collecteurs .....	27
I.3.7.4 Forme fiche traitement d_eaux .....	28
<b>I.3.8 Etude de document.....</b>	<b>29</b>
I.3.8 .1 Ensemble de documents utilisés dans le service.....	29
I.3.8.2.1 Fiche d'étude du document n° 1 .....	30
I.3.8.2.2 Fiche d'étude du document n° 2 .....	31
I.3.8.2.3 Fiche d'étude du document n° 3.....	32
I.3.8.2.4 Fiche d'étude du document n° 4.....	33
<b>I.3.9 Circuits d'information de l'entreprise .....</b>	<b>34</b>
<b>I.3.10 Flux d'informations .....</b>	<b>34</b>
I.3.10.1 Synthèse d'informations .....	35
<b>I.3.11 Les différentes structures de l'entreprise .....</b>	<b>35</b>
I.3.11.1 Direction administration et finance .....	35
I.3.11.1.1 Département finance comptabilité .....	35
I.3.11.1.2 Département administration générale .....	36
I.3.11.1.3 Département informatique et organisation .....	36
I.3.11.2 Direction commerciale .....	36
I.3.11.2.1 Département d'achat.....	36
I.3.11.2.2 Département gestion de stock.....	37
I.3.11.2.3 Département de vente.....	37
I.3.11.3 Département agro-élevage.....	38
I.3.11.4 Département qualité .....	38

I.3.11.5 Département maintenance .....	38
I.3.11.6 Département production.....	38
<b>I.4 Critiques de l'existant .....</b>	<b>39</b>
<b>I.5 Conclusion .....</b>	<b>39</b>

## Chapitre 2 : Les outils de modélisation Et

### Analyse et Conception

<b>II.1 Introduction .....</b>	<b>40</b>
<b>II.2 Les applications distribuées .....</b>	<b>40</b>
<b>II.2.1 Définition .....</b>	<b>40</b>
<b>II.2.2 Motivations des applications distribuées .....</b>	<b>41</b>
<b>II.2.3 Les exigences des applications distribuées .....</b>	<b>42</b>
II.2.3.1 Extensibilité « Mise à l'échelle » .....	42
II.2.3.2 Hétérogénéité .....	42
II.2.3.3 Accès aux ressources partagés.....	42
II.2.3.4 Tolérance aux pannes .....	42
<b>II.3 les outils de programmations réparties .....</b>	<b>43</b>
<b>II.3.1 TCP et UDP.....</b>	<b>43</b>
<b>II.3.2 Socket.....</b>	<b>43</b>
<b>II.3.2.1 Définition .....</b>	<b>43</b>
<b>II.3.2.2 Déroulement d'une communication avec sockets .....</b>	<b>44</b>
<b>II.3.3 RMI (Remote Method Invocation).....</b>	<b>45</b>
<b>II.3.3.1 Définition .....</b>	<b>45</b>
<b>II.3.3.2 Structure des couches RMI .....</b>	<b>45</b>
<b>II.3.3.3 Architecture de RMI .....</b>	<b>47</b>
<b>II.3.3.4 Mise en œuvre de RMI .....</b>	<b>47</b>
<b>II.4 Etude préliminaire .....</b>	<b>49</b>
<b>II.4.1 Présentation du projet a réalisé .....</b>	<b>49</b>
<b>II.4.2 Définition des grands choix technique .....</b>	<b>49</b>

<b>II.4.3</b>	<b>Présentation générale du problème</b>	49
<b>II.4.4</b>	<b>Recueil des besoins fonctionnels</b>	50
<b>II.4.5</b>	<b>Recueil des besoins opérationnels</b>	50
<b>II.4.6</b>	<b>Description du contexte du système</b>	50
II.4.6.1	L'identification des acteurs	51
II.4.6.2	L'identification des cas d'utilisations	51
II.4.6.2.1	Suivi des consommations des matières premières	51
II.4.6.2.2	Suivi de la répartition de consommation de lait de vaches sur les produits	51
II.4.6.2.3	Contrôle de la qualité des produits	51
II.4.6.2.4	Gestion des réalisations du produit fini	51
II.4.6.2.5	Répartition des charges sur les produits	52
II.4.6.2.6	Suivi des arrêts de la production	52
II.4.6.2.7	Gestion du personnel de la production (ressources humaines)	52
II.4.6.2.8	Edition des états et statistiques sur la production	52
II.4.6.2.9	Vente des produits fini	52
<b>II.5</b>	<b>Modélisation</b>	52
<b>II.5.1</b>	<b>Langage de modélisation UML</b>	52
II.5.1.1	Introduction	52
II.5.1.2	Définition d'UML et les caractéristiques	53
II.5.1.3	La modélisation UML	53
II.5.1.3.1	Les vues statiques	53
II.5.1.3.2	Les vues dynamiques	54
<b>II.6</b>	<b>Partie d'analyse détaillée</b>	55
<b>II.6.1</b>	<b>Présentation diagramme de cas d'utilisation</b>	55
<b>II.6.2</b>	<b>Les fiches descriptives</b>	56
II.6.2.1	Cas d'utilisation « Authentifier »	56
II.6.2.2	Cas d'utilisation « Suivre la consommation des MP »	57

II.6.2.3 Cas d'utilisation « Suivre les arrêts » .....	58
II.6.2.4 Cas d'utilisation « Gérer les personnels » .....	59
II.6.2.5 Cas d'utilisation « Démarrer le serveur » .....	60
II.6.2.6 Cas d'utilisation « Contrôler la qualité de produit » .....	61
II.6.2.7 Cas d'utilisation « Suivre des réalisations » .....	62
II.6.2.8 Cas d'utilisation « Commander matière première » .....	63
<b>II.6.3 Les Diagrammes de séquence système</b> .....	<b>64</b>
II.6.3.1 Diagramme de séquence système de « Authentifier ».....	64
II.6.3.2 Diagramme de séquence système de « Démarrer le serveur ».....	65
II.6.3.3 Diagramme de séquence système de « Suivre consommation MP »....	66
II.6.3.4 Diagramme de séquence système de « Gérer le personnel ».....	67
II.6.3.5 Diagramme de séquence système de « Suivre des réalisations ».....	68
II.6.3.6 Diagramme de séquence système de « Contrôler la qualité de produit » .....	69
<b>II.6.4 Les Diagrammes d'activité</b> .....	<b>69</b>
II.6.4.1 Diagramme d'activité « Authentifier ».....	70
II.6.4.2 Diagramme d'activité « Gérer les droit d'accès ».....	70
II.6.4.3 Diagramme d'activité « Commander matière première ».....	71
II.6.4.4 Diagramme d'activité « demander et rejeter des produits ».....	71
<b>II.6.5 Diagramme de classe</b> .....	<b>72</b>
<b>II.6.6 Diagramme état transition</b> .....	<b>73</b>
<b>II.6.7 Diagramme de déploiement</b> .....	<b>74</b>
<b>II.7 L'architecture Client/serveur</b> .....	<b>75</b>
II.7.1 Présentation d'architecture client serveur .....	75
II.7.2 Caractéristiques d'Architecture Client/ Serveur .....	75
II.7.3 Les Avantages et les inconvénients de cette architecture .....	76
II.7.4 les types des Architectures Client/serveur .....	77
II.7.4.1 Architecture à 2 niveaux .....	77

II.7.4.2	Architecture à 3 niveaux .....	77
II.7.4.3	Comparaison entre les deux types .....	78
<b>II.7.5</b>	<b>Bases de données en client-serveur.....</b>	<b>78</b>
<b>II.8</b>	<b>Conclusion .....</b>	<b>80</b>

### Chapitre 03 : réalisation

<b>III.1</b>	<b>Introduction.....</b>	<b>81</b>
<b>III.2</b>	<b>Environnement de réalisation de notre application .....</b>	<b>81</b>
<b>III.2.1</b>	<b>Choix du langage de programmation.....</b>	<b>81</b>
III.2.1.1	Le JDK.....	81
III.2.1.2	NetBeans .....	81
III.2.1.3	JDBC .....	82
III.2.1.4	Apache POI .....	85
III.2.1.4.1	Les composantes de POI .....	85
III.2.1.4.2	L'API POI-HSSF .....	86
III. 2.1.4.3	La liste des packages de HSSF.....	87
III.2.1.4.4	Les fonctionnalités de l'API « HSSF ».....	87
III.2.1.4.5	RMI .....	88
<b>III.2.2</b>	<b>Choix du SGBD .....</b>	<b>88</b>
III.2.2.1	SGBD .....	88
III.2.2.2	SQL .....	88
<b>III.2.3</b>	<b>Environnement matériel.....</b>	<b>89</b>
III.2.3.1	Caractéristiques techniques du serveur .....	89
III.2.3.2	Caractéristiques techniques du Client.....	89
<b>III.3</b>	<b>Mapping objet-relationnel.....</b>	<b>89</b>



III.3.1	PowerAMC.....	89
III.3.2	Les règles de passage de l'orienté objet au relationnel .....	90
III.3.3	Equivalence entre les concepts objets et relationnels.....	91
III.3.4	Le modèle relationnel .....	91
III.4	Architecture de réseaux de notre système distribué .....	93
III.5	Structure de logiciel .....	93
III.5.1	La page d'authentification de l'administrateur.....	93
III.5.2	La fenêtre de l'espace administrateur.....	94
III.5.3	La fenêtre d'authentification des services .....	94
III.5.4	La fenêtre des mises à jour de MP (Espace magasinier).....	95
III.5.5	La fenêtre de création un bon de livraison (magasinier).....	95
III.5.6	La fenêtre de suivi la consommation de matière première par jour.....	96
III.5.7	La fenêtre de suivi mensuel de la consommation de matière première par équipe.....	96
III.5.8	La fenêtre de l'affectation de lait cru sur déférent produits .....	97
III.5.9	La fenêtre des statistiques de consommation MP par jour .....	97
III.5.10	La fenêtre de planning des équipes de travail.....	98
III.5.11	La fenêtre de suivi MP au niveau de chef département.....	98
III.5.12	La fenêtre de l'espace membre de laboratoire.....	99
III.6	Conclusion .....	100
	Conclusion générale.....	101

## Liste des figures

<b>Figure I-1</b>	Système de gestion de production .....	18
<b>Figure I.2</b>	Organigramme de « Laiterie Numidia » .....	24
<b>Figure I.3</b>	Bordereau d'envoi.....	25
<b>Figure I.4</b>	Technique de résultat microbiologique.....	26
<b>Figure I.5</b>	Lait des collecteurs.....	27
<b>Figure I.6</b>	Traitement d'eaux.....	28
<b>Figure I.7</b>	Fiche d'étude du document n° 1.....	30
<b>Figure I.8</b>	Fiche d'étude du document n° 2.....	31
<b>Figure I.9</b>	Fiche d'étude du document n° 3.....	32
<b>Figure I.10</b>	Fiche d'étude du document n° 4.....	33
<b>Figure I.11</b>	Flux d'information.....	34
<b>Figure II.1</b>	Présentation de l'application distribuée .....	41
<b>Figure II.2</b>	Schéma d'une communication en mode connecté .....	44
<b>Figure II.3</b>	Schéma d'une communication en mode non connecté .....	45
<b>Figure II.4</b>	Structure des couches RMI .....	46
<b>Figure II.5</b>	L'architecture de RMI .....	47
<b>Figure II.6</b>	Diagramme de cas d'utilisation .....	55
<b>Figure II.7</b>	DSS d'authentifier .....	64
<b>Figure II.8</b>	DSS démarrer le serveur .....	65
<b>Figure II.9</b>	DSS suivre la consommation MP .....	66
<b>Figure II.10</b>	DSS gérer les personnels .....	67
<b>Figure II.11</b>	DSS Suivre des réalisations .....	68
<b>Figure II.12</b>	DSS Contrôler la qualité de produit .....	69
<b>Figure II.13</b>	Diagramme d'activité d'authentifier .....	70
<b>Figure II.14</b>	Diagramme d'activité gérer les droit d'accès .....	70
<b>Figure II.15</b>	Diagramme d'activité commander matière première .....	71
<b>Figure II.16</b>	Diagramme d'activité demander et rejeter des produits .....	71
<b>Figure II.17</b>	Diagramme de calasse .....	72
<b>Figure II.18</b>	Diagramme d'état transition de la classe produit .....	73
<b>Figure II.19</b>	Diagramme de déploiement .....	74
<b>Figure II.20</b>	Architecture a 2 niveaux.....	77
<b>Figure II.21</b>	Architecture Client/serveur à 3 niveaux.....	78
<b>FigureIII-1</b>	Communication entre NetBeans et les bases de données.....	82
<b>FigureIII.2</b>	Pilotes JDBC de type 1.....	83
<b>FigureIII.3</b>	Pilotes JDBC de type 2.....	83
<b>FigureIII.4</b>	Pilotes JDBC de type 3.....	84
<b>FigureIII.5</b>	Pilotes JDBC de type 4.....	84
<b>FigureIII.6</b>	Architecture JDBC.....	85
<b>FigureIII.7</b>	Le modèle relationnel.....	92
<b>FigureIII.8</b>	Architecture de réseaux de notre système.....	93
<b>FigureIII.9</b>	Fenêtre d'authentification.....	93
<b>FigureIII.10</b>	Fenêtre de l'espace administrateur.....	94
<b>FigureIII.11</b>	Fenêtre d'authentification service.....	94
<b>FigureIII.12</b>	Onglet de MAJ matière.....	95

<b>FigureIII.13</b>	Onglet de création un bon de livraison.....	95
<b>FigureIII.14</b>	Onglet de suivi la consommation matière première par jour.....	96
<b>FigureIII.15</b>	Onglet de suivi mensuel de la consommation matière par équipe.....	96
<b>FigureIII.16</b>	Onglet de l'affectation de lait cru.....	97
<b>FigureIII.17</b>	Fenêtre de Statistiques de consommation matière première .....	97
<b>FigureIII.18</b>	Fenêtre de planning des équipes de travail .....	98
<b>FigureIII.19</b>	Fenêtre de suivi matière première dans le chef département .....	98
<b>FigureIII.20</b>	Fenêtre de l'espace de membre de laboratoire .....	99

## Liste des tableaux

<b>Tableau I.1</b>	synthèse d'information.....	35
<b>Tableau II.1</b>	La fiche descriptive de cas « Authentifier » .....	56
<b>Tableau II.2</b>	La fiche descriptive de cas « Suivre de consommation matière premier » .....	57
<b>Tableau II.3</b>	La fiche descriptive de cas « Suivre des arrêtes de la production » .....	58
<b>Tableau II.4</b>	La fiche descriptive de cas «Gestion du personnel» .....	59
<b>Tableau II.5</b>	La fiche descriptive de cas «Démarrer le serveur » .....	60
<b>Tableau II.6</b>	La fiche descriptive de cas «Contrôle la qualité de produit» .....	61
<b>Tableau II.7</b>	La fiche descriptive de cas «Suivre des réalisations».....	62
<b>Tableau II.8</b>	La fiche descriptive de cas «Commander matière première» .....	63
<b>Tableau II.9</b>	Comparaison entre l'architecture 2 niveaux et 3 niveaux.....	78
<b>TableauIII.1</b>	Les packages de l'API 'HSSF' .....	87
<b>TableauIII.2</b>	Les caractéristiques techniques de serveur .....	89
<b>TableauIII.3</b>	Les caractéristiques techniques de client .....	89
<b>TableauIII.4</b>	Les règles de passage de l'orienté objet au relationnel .....	90
<b>TableauIII.5</b>	La réalisation d'association .....	91

## ***Résumé***

Notre travail a permis de définir la conception et le développement d'une application d'un système distribué d'une entreprise « Laiterie Numidia ». Les entreprises algériennes ont besoin de ces applications est de plus en plus jour après jour, car Aujourd'hui l'utilisation des applications distribués est trop large dans le monde. On trouve ça dans plusieurs domaines, comme le vol-réservation en ligne, Machines de l'argent distribué, Applications de vidéoconférence, par exemple « MBone » et dans le World Wide Web.

Il y a plusieurs facteurs contribuent à l'importance croissante des systèmes distribués comme : le Développement de la coopération et de l'utilisation des ressources partagées par les utilisateurs géographiquement dispersés causés par la mondialisation des marchés et des entreprises, par exemple l'application de la coopération de la télé (groupware, TCAO) et communication mobile afin d'améliorer le travail d'équipe distribuée.

Pour cela, nous avons réalisé une application qui facilite la tâche et accélère le rondement de production de l'entreprise, on utilise des techniques pour le développement comme : le langage UML pour les applications java à l'aide d'une API comme RMI, basé sur l'architecture : client/serveur. Plus d'autres outils on va trouver dans le fond de l'application.

**Mots clés** : système distribué, application distribué, RMI, java.

## **Abstract**

Our works allowed to define the design and the development of an application of a distributed system of a company “Laiterie Numidia”, the Algerian companies need these applications is more and more day after day, because today the use of the applications distributed is too broad in the world. One finds that in several fields, like the flight-reservation in line, Machines of the money distributed, Applications of videoconference, for example “MBONE” and in the World Wide Web.

There are several factors contribute to the increasing importance of the systems distributed like: the Development of the co-operation and the use of the resources shared by the geographically dispersed users caused by the universalization of the markets and the companies, for example the application of the co-operation of TV (groupware, TCAO) and mobile communication in order to improve team work distributed.

For that, we carried out an application which facilitates the task and accelerates roundly production of the company, one uses techniques for the development like: language UML for the java applications using API like RMI, based on architecture: client/server. More other tools one will find in the content of the application.

**Key words:** distributed system, application distributed, RMI, java.

## Abréviation

ANSI	American National Standard Institute
API	Application Programming Interface
ATM	Asynchronous Transfer Mode
CAO	Conception Assisté par Ordinateur
CIP	Service Cleaning In Place
CORBA	Common Object Request Broker Architecture
DNS	Domain Naming Service
FDDI	Fiber Data Distribution Interface
FTP	File Transfer Protocol
GPAO	Gestion Production Assistée par Ordinateur
http	Hyper Text Transfer Protocol
IDE	Integrated Development Environment
IIOF	Internet Inter-ORB Protocol
IP	Internet Protocol
ISO	International Standards Organization
J2EE	Java 2 Platform Enterprise Edition
JAR	Java Archive
JDBC	Java Data Base Connectivity
JRMP	Java Remote Method Protocol
JVM	Java Virtuelle Machine
MFC	Microsoft Foundation Classes
ODBC	Open Data Base Connectivity
OLE	Object Linking and Embedding
OMG	Object Management Group
OS	Operating System
POI	Poor Obfuscation Implementation
POO	Programmation orientée objet
RMI	Remote Method Invocation
RRL	Remote Reference Layer
SPA	Société Par Action
SQL	Structured Query Language
TCP	Transport Control Protocol
UDP	User Datagram Protocol
UML	Unified Modeling Language
XNS	Xerox Network System

## Introduction générale :

Avant l'invention de l'ordinateur, on enregistrait toutes les informations manuellement sur des supports en papier ce qui engendrait beaucoup de problèmes tel que la perte de temps considérable dans la recherche de ces informations ou la dégradation de ces dernières. ..etc.

Il ne fait désormais plus aucun doute que l'informatique est la révolution la plus importante et la plus innovante qui a marqué la vie de l'humanité moderne. En effet, les logiciels informatiques proposent maintenant des solutions à tous les problèmes de la vie, aussi bien dans les domaines de gestion (production, stock, personnel, finance...ect) .

Aujourd'hui, la fonction Production est l'une des fonctions importantes de l'entreprise, dont l'objectif est de produire des biens et des services, afin de dégager une rémunération du capital engagé, qui pourra être ou non réaffecté à de nouveaux investissements.

L'amélioration des performances de l'entreprise est tributaire des méthodes d'organisation et d'exploitation des ressources dont elle dispose. C'est pourquoi, une bonne gestion de production est aujourd'hui une nécessité de plus en plus préoccupante pour les entreprises : bien gérer une entreprise, comme bien gérer la production, c'est trouver une solution admissible par rapport aux objectifs fixés, tout en réglant un ensemble de conflits.

D'autre part, pour faire face à la mondialisation de l'enjeu économique, les industriels doivent être diminués les coûts et la durée de production. Dans ce sens, l'implantation des systèmes informatiques distribués, une meilleure répartition du travail sur les machines ou les machines peuvent communiquer, négocier et coopérer entre elle, et une réactivité accrue face aux perturbations. Tout cela contribue à réduire directement les coûts et gagner le temps.

L'informatique représente un outil d'aide à la décision et aussi aux planifications de la production.

L'objectif de notre projet est la modélisation et le développement d'une application distribuée fiable pour le suivi de la production de l'entreprise « **Laiterie Numidia** », pour atteindre notre objectif, nous avons effectué une bonne étude de l'existant et collecter toute les informations nécessaires et suffisantes.

L'entreprise « laiterie Numidia » utilise une méthode classique et simple au niveau du service suivi production elle consiste à collecter manuellement les informations et saisir des feuilles de calcul Excel.

Notre contribution consiste à mettre en œuvre un logiciel confiant et sécurisé via un système facile et fiable d'authentification pour nous permettre de gérer d'une façon idéale la gestion de production de l'entreprise depuis l'acquisition des matière premières jusqu'à livraison du produit fini, Y compris le suivi des consommations matières premières, contrôle de la qualité des produits par rapport aux normes, suivi des contraintes à la production, gestion des ressources humaines .....ect.



Pour la modélisation de notre application on a utilisé le langage de modélisation UML, Et notre réalisation pratique fait l'appel à des outils technologiques récents tels que JAVA, JDBC, Access, Edraw Max, Excel.....ect.

Pour mieux élaboré le cadre de notre travail, nous exposons dans le premier chapitre introductif une grand projection sur les systèmes de gestion de production en donnant des définitions et des caractéristiques. Et aussi la présentation de l'établissement d'accueil à savoir la société « laiterie Numidia » avec ses organisations, fonctionnement, organigramme et missions.

Dans le second chapitre, nous illustrerons les différentes parties des systèmes distribués à savoir les exemples et les outils de programmations qui permettent la mise en œuvre à celle-ci, ainsi que les outils technique de développement de notre application comme le langage de modélisation UML et son évolution et l'architecture Client/serveur laquelle a été utilisé pour notre application.

aussi on est passé à l'analyse préliminaire et détaillée et compléter la conception de notre système de façon plus approfondie, ensuite on a essayé d'éclaircir les idées fortes auxquelles se rattache la démarche pratique adoptée dans la suite de la mémoire.

Le dernier chapitre, est consacré à La réalisation et l'implémentation d'un cas réel en utilisant les outils de développement d'un logiciel, ou on a parlé du langage objet purs et durs Java, JDBC, MySQL, RMI.

En terminant par une conclusion générale.

## I.1. Introduction :

Les systèmes de production sont des conséquences d'une demande sociale aux objectifs limités, et progressivement ils sont devenues les systèmes dominants sur le marché, et que cette évolution est le résultat d'une dynamique autonome et impérieuse de l'esprit humain tout autant que l'informatique qui la suit.

Dans ce chapitre introductif, nous nous limiterons essentiellement à une discussion de quelques notions de base sur des systèmes de gestion de production avec ses différentes caractéristiques et composantes, ensuite nous allons présenter notre lieu de stage « **Laiterie Numidia** » en spécifiant son historique, son organisation et en détaillant chacune de ses structures.

## I.2. Les systèmes de production :

### I.2.1 Production :

La production c'est l'ensemble des opérations permettant de Créer à partir de consommations des ressources, des biens ou des services ayant une utilité. Il s'agit en fait de la création d'un bien économique ou addition de la valeur à un bien existant.

### I.2.2 Gestion de production :

La gestion de production est l'action ou manière de gérer, d'administrer, de diriger et d'organiser un ensemble d'activité de conception, de planification, d'organisation, de direction et de contrôle de flux au niveau de la production, depuis l'acquisition des matières premières jusqu'à livraison du produit fini, auxquelles on ajoute des activité d'assurance qui consistent à une vérification générale visant à s'assurer que les cinq activités précédentes ont été correctement.

Son rôle est recherché d'une organisation efficace de la production des biens et services. [1]

### I.2.3 Historique des systèmes de gestion de production :

La production a commencé au dix-huitième siècle avec la production artisanale tout en évoluant jusqu'au dix-neuvième siècle ou elle est devenu une production manufacturière qui comporte la production des armes, tabac .....

Mais vingtième siècle un désir de rationalisation des facteurs de production point de départ de la gestion de production est survenu et cela par plusieurs, et on peut citer par exemple :

- ❖ **Taylor (1911)** : organisation du travail basée sur la distinction radicale entre conception et exécution avec la recherche systématique des économies de gestes et mouvement en utilisant le maximum de la machine.

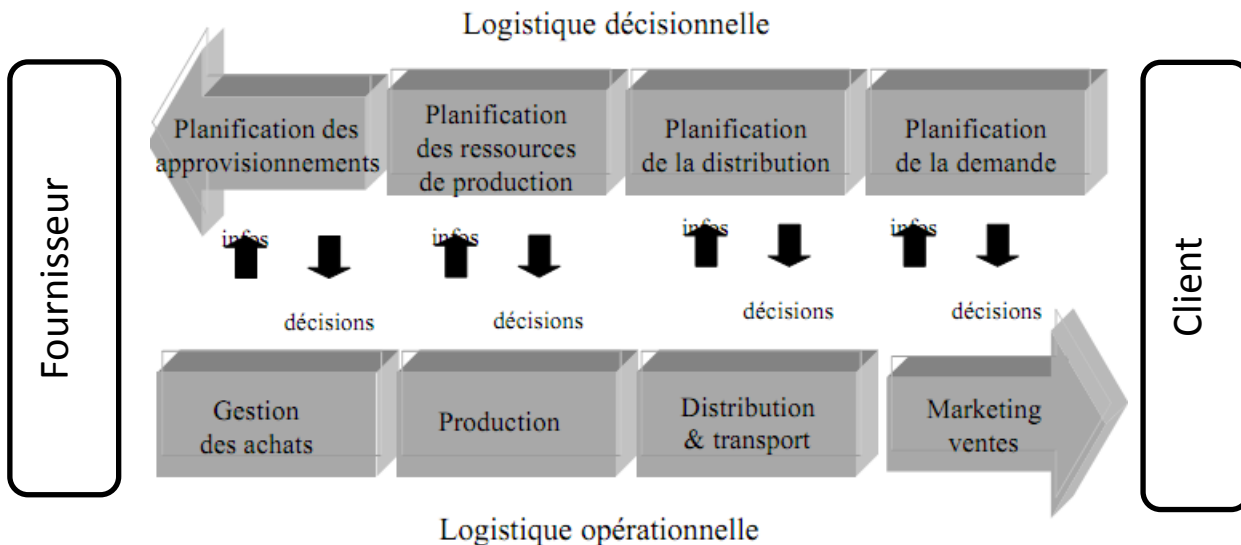
- ❖ **Ford (1913)** : standardisation de la production et travail à la chaîne avec une avancée du concept de flexibilité dans les entreprises.
- ❖ **Harris et Wilson (1913-1924)** : qui visent quantité économique.
- ❖ **Fayol (1916)** : comporte modèle hiérarchique d'organisation de la production en basant sur le savoir, prévention, l'organisation, la commande, la coordination et le contrôle.
- ❖ **Gantt (1917)** : l'ordonnancement [2]

### I.2.4 Généralités sur les systèmes de production

Un système de production ou système logistique est un processus d'addition de valeur à des biens ou à des services répondant à des impératifs de quantité, qualité, prix, délais et cela à travers d'un ensemble d'actions élémentaires en vue d'atteindre des objectifs définis.

Ces actions élémentaires jouent un rôle dans l'évolution et l'optimisation de production en temps et en coût, et généralement on trouve quatre types d'actions élémentaires :

- ✚ **Opération** : action de transformation ajoutant de la valeur
- ✚ **Transport** : action de déplacement agissant sur la localisation de la matière
- ✚ **Stockage** : action d'attente ou de non-flux
- ✚ **Transaction** : action de modification des données de gestion



**Figure I.1** : Système de gestion de production

## I.2.5 Les caractéristiques des systèmes de productions

- Grand nombre de processus dans la fabrication, le transport, ou même le stockage
- Forte interaction entre les éléments comme employés, ressources, stocks... Ce qui rend plus intéressants à prendre.
- Manque d'information pour la gestion (marché, fournisseurs, concurrence...)
- Occurrence des phénomènes aléatoires (pannes, arrivée des approvisionnements...).
- Nature incertaine des décisions à prendre (pas de règle unique et claire).
- Grande variété et hétérogénéité des critères d'appréciation.
- Risques d'échec importants pour les décisions.
- Comportement non linéaire des systèmes.
- Systèmes difficiles à dimensionner et à optimiser (nécessité d'une approche globale). [3]

### I. 2.6 Composant d'un système de gestion de production :

Dans une entreprise industrielle de nombreux services participent au système de production :

- ✓ **Le bureau des études** : il conçoit les produits nouveaux et définit la liste complète des composants entrant dans leur fabrication, il s'appuie sur la CAO (conception assistée par ordinateur) pour l'élaboration des produits.
- ✓ **Le bureau des méthodes** : il définit les différentes opérations et leur ordonnancement en vue d'obtenir le produit il précise en fait comment le produit est réalisé, par quelle machine, avec quels outils et en combien de temps
- ✓ **Le service de planification** : coordonne les activités de production à moyen terme. Il s'agit de planifier les approvisionnements et les fabrications en utilisant les techniques de gestion de stock, de calcul de besoin et de gestion des achats.
- ✓ **Le service d'ordonnancement** : Organise la production au sein des différentes unités il indique la succession des tâches à réaliser en un temps minimum.

### I.2.7 Classification des systèmes de productions :

Les systèmes de production peuvent être classifiés selon la nature et le volume des produits fabriqués.

#### I.2.7.1 La production unitaire :

Elle consiste à produire un seul produit sur commande du client. Il n'existe pas de stock de produit fini. Une fois fabriqué, le produit est directement livré au client. Dans ce mode de production, les produits fabriqués répondent à des besoins précis manifestés par les clients. Le mode de production unitaire est pratiqué dans la production de certains types de produits :

- très onéreux (avions, locomotives, navires...)
- ayant des caractéristiques bien définies qui répondent à un besoin précis (meubles...)
- complexes (montage d'usine ou d'atelier...).

Dans ce type de structure, le problème majeur consiste à réaliser le travail demandé avec un coût compétitif et en respectant les délais.

### **I.2.7.2 La production en série :**

Elle consiste à fabriquer un grand nombre d'articles identiques dont la conception et les caractéristiques ont déjà été définitivement établies. Le processus de fabrication est découpé en opérations élémentaires et répétitives permettant de produire plusieurs fois le même article. La production en série est une production pour le stock. La dimension de la série dépend de deux facteurs : la technologie et la demande.

Il existe deux types de production en série :

#### **• La production en grande série :**

Les quantités à produire sont importantes et la diversité des produits est limitée. Dans ce cas l'utilisation de chaînes de fabrication est très rentable. Lorsque la chaîne est équilibrée, le taux d'utilisation des ressources est élevé et le temps d'attente par les produits en cours de fabrication est généralement faible.

#### **• La production en petite et moyenne série :**

C'est le cas de la plupart des Petites et Moyennes Entreprises (PME) pour lesquelles la diversité des produits et le faible volume des demandes ne permettent pas une spécialisation des moyens de production. L'ordonnancement, dans ce cas, joue un rôle primordial pour optimiser l'utilisation des ressources et minimiser les pertes de temps.

### **I.2.7.3 La production continue :**

La différence entre la production en série et la production continue est d'ordre technologique. Dans la première, le produit passe, pendant sa fabrication, par des opérations distinctes séparées les unes des autres. Par contre, dans la production continue, le produit circule en flux continu et subit des transformations physiques et/ou chimiques. Ce type de systèmes concerne surtout les industries dont la production nécessite la manipulation de matières liquides ou gazeuses (produit pétrolier, semoule, ...).

La mise en place d'une GPAO diffère d'un mode de production à l'autre. C'est pourquoi, le choix d'un logiciel de GPAO n'est pas une chose simple mais plutôt une

tâche difficile qu'il faut bien étudier afin de choisir le logiciel qui répond parfaitement aux besoins et aux activités de l'entreprise. [4]

### I.3. Présentation de l'entreprise d'accueil

#### I.3.1 Historique de l'entreprise

L'office national du lait et ses dérivés n'était dans les années 50 qu'une coopérative appartenant à la maison de l'agriculture. Suivant l'article 20-11-1969, cette coopérative est devenue alors nommé « ONALAIT » Office National Algérien du LAIT. Ce dernier avait été divisé en trois complexes :

- ✚ Complexe du lait à Constantine COLAC.
- ✚ Complexe du lait à Alger COLAITAL.
- ✚ Complexe du lait à Oran CLO.

En janvier 1981, L'ONALAIT s'est divisé en trois offices :

- ✚ Office Est « ORELAIT ».
- ✚ Office Centre « ORELAC ».
- ✚ Office Ouest « OROLAIT ».

ORELAIT a été créé comme un établissement public à caractère économique, ayant une personnalité morale, et une autonomie financière. En 11 juin 1990, elle a été transformée en une entreprise publique économique, Société Par Action (SPA). Cette dernière joue un rôle important concernant la réponse aux demandes de lait et ses dérivés des habitants de l'Est algérien, et fournit de façon continue à quatre wilayas de l'est, à savoir :

- ✚ Constantine : unité dénommé laiterie «Numidia ».
- ✚ Annaba : unité dénommé laiterie «Edough ».
- ✚ Batna: unité dénommé laiterie «Aurés ».
- ✚ Sétif: unité dénommé laiterie «tell ».

Le projet de l'entreprise Numidia Constantine avait débuté en 1974, dans le cadre du programme de développement de l'industrie nationale.

Des retards ont été enregistrés dans la réalisation du projet dû essentiellement aux manques de financement. Dans les années 1980, l'entreprise a commencé à produire le lait avec une capacité initiale de 282000 litre par jour.

En 1997, l'ONALAIT a été transformée en GIPLAIT (Groupe Industriel des Productions Laitières) qui regroupe tous les filiales des trois offices : ORELAIT, ORELAC et OROLAIT.

### I.3.2 Définition de l'entreprise

Laiterie Numidia est une entreprise avec un cadre commercial et un capital social de 372.650.000.00 DA. Elle est située à Chaab Ersas sur une superficie de 5hectares. Elle est autonome et décentralisée de l'entreprise publique économique GIPLAIT (S.P.A).

Elle a pour objectif la fabrication du lait et des produits laitiers et leur commercialisation dans la zone géographique qui lui a été désignée par la direction générale de l'Entreprise en l'occurrence : Constantine, Mila, Oum El Bouaghi, Khenchela et Skikda.

### I.3.3 Les missions de l'entreprise

Elle a pour mission :

- ✓ La préparation et le conditionnement du lait et des produits laitiers.
- ✓ L'achat de la matière première.
- ✓ La gestion et le contrôle de la qualité de tous les produits et articles entrant dans la fabrication du lait et de ses dérivées.
- ✓ L'approvisionnement régulier des produits finis afin d'assurer la satisfaction des besoins de la population des wilayas concernées.
- ✓ La maintenance préventive et curative des installations, équipements machines et autres moyens matériel.

### I.3.4 Produits développés par l'entreprise

Les produits fabriqués et vendus par l'entreprise sont les suivants :

- Lait pasteurisé conditionné (LPC).
- Lait de vache.
- Lait fermenté conditionné (LFC).
- Crème fraîche.
- Camembert.
- Petit constantinois.
- Smen.

### I.3.5 Capacité de l'entreprise

#### I.3.5.1 En moyens humains

Parmi les tâches que lui incombent la gestion des ressources humaines est de :

- ❖ Gérer ses ressources humaines, leur assurer la paie et une couverture sociale.
- ❖ Préparer les plans de la formation, de recyclage et perfectionnement de ses personnels.

Laiterie Numidia inclut 224 employés, répartis en : 85 cadre, 51 agents de maîtrise et 88 agents d'exécution.

### I.3.5.2 Superficie

Laiterie Numidia se trouve à Chaab Ersas à environ 4km au Sud Est de la ville de Constantine. La superficie totale de la coopérative est de 5 hectares divisés comme suit :

- ✚ 1ha 582m<sup>2</sup> superficie bâtie.
- ✚ 3ha 468m<sup>2</sup> superficie non bâtie.

La superficie bâtie se compose comme suit :

- ✚ Une première partie pour la direction, sous forme d'un bâtiment avec trois (03) étages.
- ✚ Une deuxième partie pour la sécurité (à l'entrée de la Laiterie).
- ✚ Une troisième partie qui consiste en une salle pour le changement de vêtements.
- ✚ Une quatrième partie qui consiste en des ateliers de production et des laboratoires.



I.3.6 Présentation de l'organisme d'accueil :

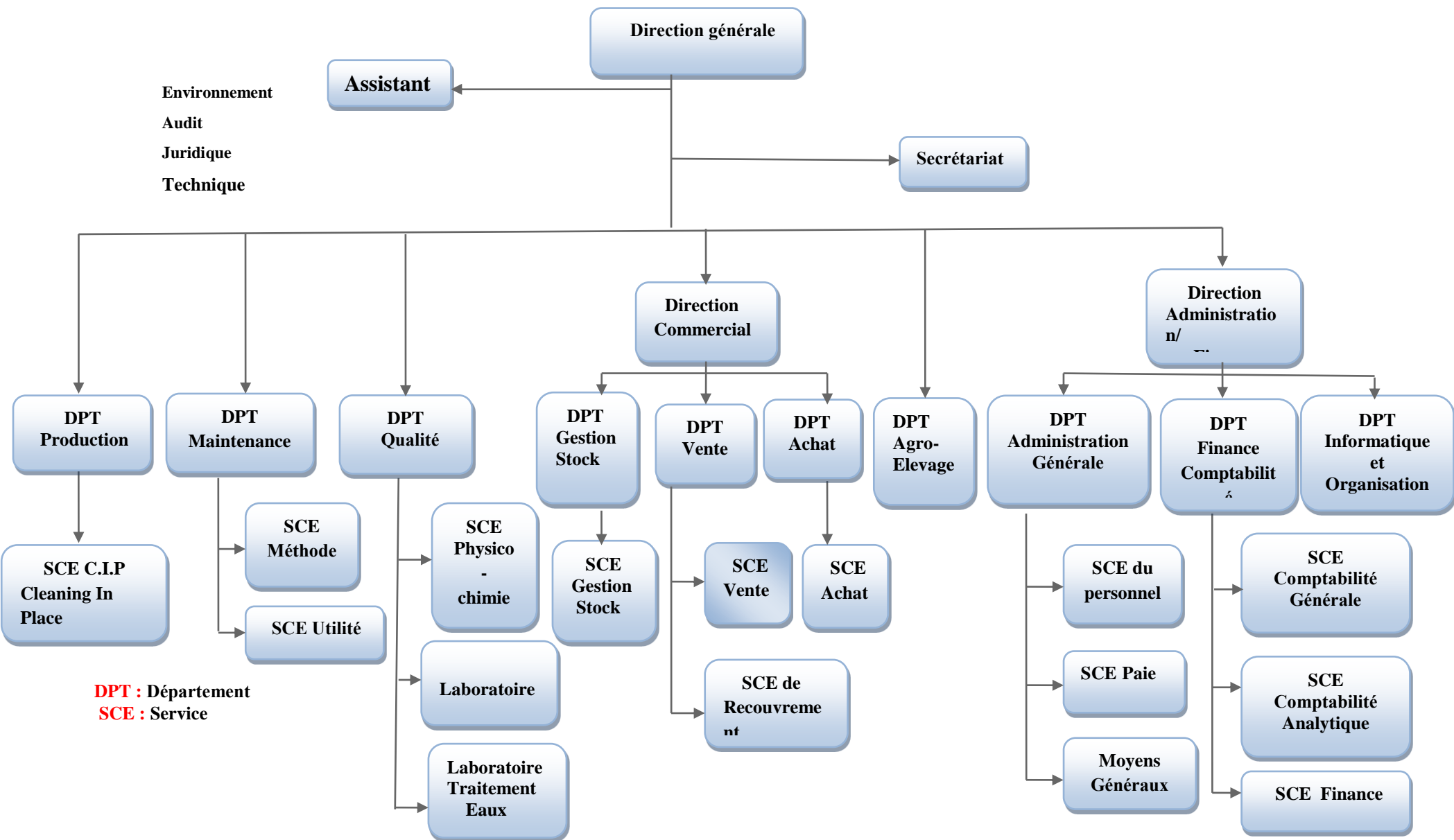


Figure I.2 : Organigramme de « Laiterie Numidia ».

### I.3.7 Document

Nous avons choisi le service de laboratoire qui prend les échantillons de produit pour donne un résultat d'accepté ou refusé le produit

#### I.3.7.1 Forme de bordereau d'envoi:

<u>Laiterie NUMIDIA</u>	<u>Constantine le 21/01/2013</u>	
<u>Département contrôle de la qualité</u>		
<b><u>BORDEREAU D'ENVOI</u></b>		
Réf :485/ MARS(03) /2013		
DESIGNATION	NOMBRE	OBSERVATIONS
<i>1)Fiches d'analyses Bactériologiques :</i>		
-LRPC: /03/2013	00	<u>Réf n° : /03/LRPC/DCQ/2013.</u>
-LMPC: 19/03/(M)/2013 19 /03/ (S)/2013	02	<u>Réf n° 19/03/LMPC/M/DCQ/2013</u> <u>04/ 03/LMPC/S/DCQ/2013</u>
<b>LEPC: 19/03/2013</b>	01	Réf n° 16/03/LEPC/DCQ/2013
<b>-LFC : 19/03/2013</b>	01	<u>Réf n° 6/03/LFC/DCQ/2013.</u>
-C . fraîche :19 /03/2013.	01	<u>Réf n° 04/03/CF/DCQ/2013.</u>
-Petit Constantinois : /03/2013.	00	<u>Réf n° /03/PC/DCQ/2013.</u>
<i>2) Fiches d'analyses Physico-chimiques :</i>		
-Le 20/03/2013	01	<u>Réf n 20/03/BAPC/DCQ/2013</u>
<i>3) Fiches d'analyses des eaux :</i>		
-Le : 21/03/2013	01	<u>Réf n° 15/03/TE/DCQ/2013</u>
<i>4)Fiches d'analyses Physico-chimique</i> <u>Mouillage- Apport en MG et ESD.</u>		
Le : 20/03/2013	01	<u>Réf n° 20 /03/M/DCQ/2013</u>
Reçu le :.....		

**Figure I.3 :** bordereau d'envoi

**I.3.7.2 Forme de technique de résultat microbiologique :**

Laiterie NUMIDIA				Constantine le 01/03/2014				
LABORATOIRE CONTRÔLE DE QUALITE								
DISPOSITIONS REGLEMENTAIRES								
DECRET 98/180 DU 26/06/1998								
ARRETE DU 26/06/1998								
<b>RESULTATS MICROBIOLOGIQUES LAIT DE MELANGE P/C EXPRIMES EN NOMBRE DE GERMES PAR MILLI LITRES</b>								
N° de Ref /25/OCTOBRE (10) / L.M.P.C (M) / D.C.Q/ 2011								
JOURNEE DU 25/03/2014								
ANALYSES SUR MILIEU SOLIDE								
	AN E C H T I			LL O	N S		NORMES	
<b>L.M.P.C</b>	<b>M7</b>	<b>M8</b>	<b>M9</b>	<b>M10</b>	X	X	<b>SATISFAISANTE</b>	<b>ACCEPTABLE</b>
<b>FLORE AEROBIES à 30</b>	1200	1100	900	1100	x	x	<90,000	<300,000
<b>COLIF SORTIE USINE</b>	X	X	X	X	X	X	<03	<10
<b>COLI A LA VENTE</b>	ABS	ABS	ABS	ABS	X	X	<30	<100
<b>COLI FECAUX SORTIE USINE</b>	x	X	x	X	X	X	ABS	ABS
<b>COLI FECAUX A LA VENTE</b>	ABS	ABS	ABS	ABS	X	X	ABS	ABS
<b>STAPHYLOCOQUES AURES</b>	ABS	ABS	ABS	ABS	X	X	<03	<10
<b>PHOSPHATASE</b>	X	X	X	X	X	X	NEG	NEG
<p>M7 LMPC A BASE DE LAIT DE MELANGE TETE T01 à 09h05 PRELEVEMENT DU 25/10/2011</p> <p>M8 LMPC A BASE DE LAIT DE MELANGE TETE T09à 10h00 ANALYSES DU 25/10/2011</p> <p>M9 LMPC A BASE DE LAIT DE MELANGE TETE T12 à 11h00 LECTURES DU 26/10/2011</p> <p>M10 LMPC A BASE DE LAIT DE MELANGE TETE T14 à 11h00 et 27/10/2011</p>								

**Figure I.4 : Technique de résultat microbiologique**

**I.3.7.3 Forme FICHE DE LAIT DES COLLECTEURS :**

Laiterie NUMIDIA														Constantine le 21/03/2013	
LABORATOIRE CONTRÔLE DE QUALITE															
DISPOSITIONS REGLEMENTAIRES															
DECRET 98/180 DU 26/05/1998															
ARRETE DU 26/06/1998															
<b>QUALITE PHYSICO-CHIMIQUE DU LAIT DE COLLECTE</b>															
<b>MOULAGE .APPORT EN MG ET ESD</b>															
N° de REF 17/Mars/(03) /M%- MG-ESD/ D.C.Q/ 2013															
JOURNEE DU 17/03/2013.															
n° V	NOMS	QTE	%	t°C analysé	t°C %	A° D	MG (g/l)	DEN SIT	EST (g)	ESD (g)	MG(t) kg	ESD(t) kg	M% (l)	QTE EAU	NORMES
1	Bouakira,S	696	1,0	10,0	0,103	17	34	1,02	118,0	84,08	23,66	58,52	3,351	23,32	A°D(15-18)
1		896	1,32	12,0	0,159	17	34	1,02	118,09	84,085	30,464	75,340	3,351	30,02	
1		2132	3,15	10,0	0,315	17	34	1,02	118,0	84,08	72,48	179,2	3,351	71,43	
1	Benfadel,H	538	0,79	7,00	0,056	16	34	1,03	120,75	86,750	18,292	46,67	0,287	1,55	DENSITE( ≥1030)
1	Bouakira,R	538	0,79	101.	0,079	17	35	1,02	119,29	84,285	18,830	45,34	3,121	16,79	
1		616	0,91	12,0	0,109	17	35	1,02	119,29	84,285	21,560	51,92	3,121	19,22	
1	Houri,K	548	0,81	10,0	0,081	17	33	1,02	116,89	83,885	18,084	45,96	3,580	19,62	La Densité
1		430	0,63	10,0	0,063	17	33	1,02	116,89	83,885	14,190	36,07	3,580	15,40	
1	Laouar,Mh	1158	1,71	10,0	0,171	17	34	1,02	118,09	84,085	39,372	97,37	3,351	38,80	
1	Laouar,R	2079	3,07	10,0	0,307	17	33	1,02	116,89	83,885	68,607	174,3	3,580	74,44	tolerer au niveau
1	Hassani,R	1104	1,63	9,00	0,147	17	34	1,02	118,09	84,085	37,536	92,83	3,351	36,99	
1		1104	1,63	10,0	0,163	17	34	1,02	118,09	84,085	37,536	92,83	3,351	36,99	
1		599	0,88	12,0	0,106	17	34	1,02	118,0	84,08	20,36	50,36	3,351	20,07	T° :4 -6°C
1		107	0,16	12,0	0,019	17	36	1,02	120,4	84,48	3,852	9,040	2,891	3,09	
1		1684	2,49	6,00	0,149	17	36	1,02	120,4	84,48	60,62	142,2	2,891	48,68	
53	<b>TOTAL</b>	<b>67735</b>	<b>100</b>	<b>9,38</b>	<b>8,64</b>	<b>16,79</b>	<b>34,21</b>	<b>1,0293</b>	<b>119,196</b>	<b>84,989</b>	<b>2317,002</b>	<b>5756,75</b>	<b>2,311</b>	<b>1565,4</b>	

Prélèvement du 17/03/2013

T°C DE RECEPTION 8,64 °C AVEC UN MIN DE 05°C ET MAX DE 12 °C.

**Figure I.5 : LAIT DES COLLECTEURS**

I.3.7.4 Forme fiche traitement d'eaux :

		HEURE	PH	T°C	TH °F	TA	TAC	NORMES
<b>Laiterie NUMIDIA</b>		<b>Constantine le 21/03/2014</b>						
<b>CONSTANTINE</b>								
<b>LABORATOIRE CONTRÔLE DE QUALITE</b>								
<b>DISPOSITIONS REGLEMENTAIRES</b>								
<b>DECRET 2006-141 DU 19/04/2006</b>								
<b>RESULTATS PHYSICO-CHEMIE DES EAUX</b>								
N° Ref /15/MARS(03) TE / (2012)								
JOURNEE : 21/03/2013								
<b>GENIE EAU</b>	<b>AD</b>	09h00	-	-	1			NF T 90-003 AFNOR TH=0°F TH=0°F:9°F
	<b>GENIE EAU</b>							
	<b>EAU ALIMENTAIRE CH1</b>	09h00	-	-	4			
	<b>CONDENSEUR</b>	09h00	-	-	5			
	<b>CHAUDIERE CH1</b>	vanne cassée						
	<b>CHAUDIERE CH2</b>	-	-	-	-	-	-	
<b>STATION PERMO</b>	<b>AD RECOM 1</b>	08h45	-	-	1,6			TH=0°F
	<b>AD RECOM 2</b>	08h45	-	-	0,8			TH=0°F
	<b>MELANGE</b>	08h45	-	-	11			TH=12°F:15°F
	<b>EAU BRUTE+ AD</b>							
	<b>EAU PREPARATION</b>	08h45	-	-	5			TH=12°F:15°F
	<b>AD 1 CIP</b>	EN ARRET						
	<b>AD 2 CIP</b>	EN ARRET						
<b>EFFLUENTS</b>	<b>REJET CIP</b>	<b>HEURE</b>	<b>PH</b>		<b>T°C</b>		<b>NORMES</b> AFNOR PH= 6.5 - 8.5 T°C =<30	
		9H45	7,68		15,0			
		10h30	7,75		9,0			
		-	-		-			
	<b>REJET PASTO</b>	13H00	7,57		15			

**INTERPRETATION:**  
 TH AD1 acceptable  
 TH AD2 acceptable  
 TH AD Sortie Génie-eau acceptable  
 TH eau de condenseur acceptable  
 TH d'eau de mélange acceptable  
 Analyses Effectuées Par : Mme KHEDIRECHE K

**OBSERVATION:** chaudière N°1 : vanne cassé

Figure I.6 : traitement d'eaux

### I.3.8 Etude de document

Dans cette étape, nous étudierons l'ensemble des documents et des procédures utilisées dans le service aux fins de l'information fournie par la connaissance et de la nature de l'étude comprennent les points suivants:

- Etudier les caractéristiques du document en termes de:  
Nom, direction, la fréquence, le code qui a été trouvé, le but
- Etude de la forme du document en termes de:  
La qualité du papier, la taille, la couleur, le nombre des copiées
- Description du document:  
L'entête, le corps, la base, le code, la désignation, le type, la longueur, l'observation

On va étudier quelques documents utilisé dans le service de laboratoire.

#### I.3.8 .1 Ensemble de documents utilisés dans le service:

- Le bordereau.
- Technique de résultat microbiologique
- Fiche de lait des collecteurs
- Fiche\_traitement d'eaux

## I.3.8.2.1 Fiche d'étude du document n° 1

Analyse du document				
- <u>Le nom du document</u> : Bordereau D'envoi		- <u>Le source</u> : laboratoire		
- <u>Direction</u> : DCQ		- <u>Support</u> : papier normal		
- <u>Nature</u> : interne		- <u>Fréquence</u> : 1 mois		
Format	Couleur	Nombre d'exemplaires		
A4	Blanc	01		
<b>L'objectif de ce document: fiche d'analyses</b>				
Description du document				
Les parties du document	Spécification	Type	longueur	Observation
Le haut du document	- Laiterie NUMIDIA - La structure - Référence - Date de envoyer le bordereau	A A AN D	02 / 12 10	N/N JJ/MM/AAAA
Corps du document	- Fiches_d'analyse Bactériologiques -Fiches d'analyses Physico- chimiques - Fiches d'analyses des eaux -Fiches d'analyses Physico-chimique Mouillage- Apport en MG et ESD - Nombre - Date d'analyse - Observation -	AN N N N N D AN	/ / / / 10 10	
La base du Document	- date de rédaction - le nom et la signature de chef	D A	8 /	JJ/MM/AAAA

Figure I.7 : Fiche d'étude du document n° 1

I.3.8.2.2 Fiche d'étude du document n° 2

Analyse du document				
- <u>Le nom du document</u> : technique de résultat microbiologique - <u>Le source</u> : laboratoire - <u>Direction</u> : LCQ - <u>Support</u> : papier normal - <u>Nature</u> : interne - <u>Fréquence</u> : 1 mois				
Format	Couleur	Nombre d'exemplaires		
A4	Blanc	01		
<b>L'objectif de ce document: La connaissance du résultat de l'analyse</b>				
Description du document				
Les parties du document	Spécification	Type	Longueur	Observation
Le haut du document	- Laiterie NUMIDIA	A	10	JJ/MM/AAAA  N/N  MM/AAAA
	- Nom de laboratoire	A	30	
	- Structure	D	08	
	- La date ET Lieu	AD	03	
	- Référence Decret	N	30	
	- Date d'Arrete	D	30	
	- Numéro de référence	N	10	
- Titre de page	N	20		
Corps du document	- L.M.P.C	A	08	
	- Les échantillons	AN	20	
	- Les normes	AN	03	
			/	
La base du document	- Définition d'abréviation	D	8	JJ/MM/AAAA
	- date d'analyse	A	/	

Figure I.8 : Fiche d'étude du document n° 2



**I.3.8.2.3 Fiche d'étude du document n° 3**

Analyse du document				
- <u>Le nom du document</u> : fiche de lait des collecteurs - <u>Direction</u> : LCQ - <u>Nature</u> : interne		- <u>Le source</u> : laboratoire - <u>Support</u> : papier normal - <u>Fréquence</u> : 1 mois		
Format	Couleur	Nombre d'exemplaires		
A4	Blanc	01		
<b>L'objectif de ce document: qualité de lait de collecteurs</b>				
Description du document				
Les parties du document	Spécification	Type	Longueur	Observation
Le haut du document	- Laiterie NUMIDIA	A	10	N/N JJ/MM/AAAA MM/AAAA
	- La structure	A	30	
	- Numéro de Référence	N	30	
	- La date et lieu	AD	03	
	- Date d'arrête	D	10	
	- Numéro Décret	N	08	
Corps du document	- Matricule	N	10	
	- Nom et prénom des collecteurs	A	20	
	- quantité	N	/	
	- pourcentage	N	/	
	- densité	N	/	
	- Normes	AN	20	
	- Total	N		
La base du document	- La date de collecte	A	/	
	- observation	A	/	

**Figure I.9** : Fiche d'étude du document n° 3

I.3.8.2.4 Fiche d'étude du document n° 4

Analyse du document				
- <u>Le nom du document</u> : fiche_traitement d'eaux - <u>Direction</u> : LCQ - <u>Nature</u> : externe		- <u>Le source</u> : laboratoire - <u>Support</u> : papier normal - <u>Fréquence</u> : 1 mois		
Format	Couleur	Nombre d'exemplaires		
A4	Blanc	01		
L'objectif de ce document: il consiste sur les résultats physico-chimiques des eaux				
Description du document				
Les parties du document	Spécification	Type	Longueur	Observation
Le haut du document	- Laiterie NUMIDIA - local - nom de laboratoire - Date de réalisation - Titre de la fiche - Référence de la Decret - Référence de la fiche	A A A D A N AN	/ 30 / 08 02	
Corps du document	- Genie Eau - Station permio - effluent 8 - heure - PH - T°C - T°H - TA - TAC - Norme8	A A A N N N N / AN	/ / / / / 10	JJ/MM/AAAA
La base du document	- observation - interprétation - nom responsable des analyses	AN A A	/ /	

Figure I.10 : Fiche d'étude du document n° 4

### I.3.9 Circuits d'information de l'entreprise :

- 1- le client demande une quantité de la production.
- 2- l'unité de vente principal envoi une demande a l'unité de production la quantité demandée à la vente
- 3- l'unité de production se divise en deux : l'unité de production de lait et l'unité d'emballage
- 4- demande à l'unité de stockage le produit de matière première de la préparation du lait et lait de vache naturel des agriculteurs
- 5- après avoir reçu le produit gras du stock commence la production du lait
- 6- à chaque parti de la production serais l'envoi d'une échantillon pour contrôle la qualité et la valeur afin d'avoir l'autorisation de continuer ou d'améliorer ou d'annuler.
- 7- après que le produit serait prêt il faut demander l'emballage nécessaire
- 8- l'unité de production peut avoir des accidents de panne ou d'avertissement au service de maintenance pour résoudre la panne des machines de production
- 9- les appareils de production seront stérilisés et la propreté dans le cas le service d'hygiène et de propreté intervienne pour entretenir les machines

### I.3.10 Flux d'informations :

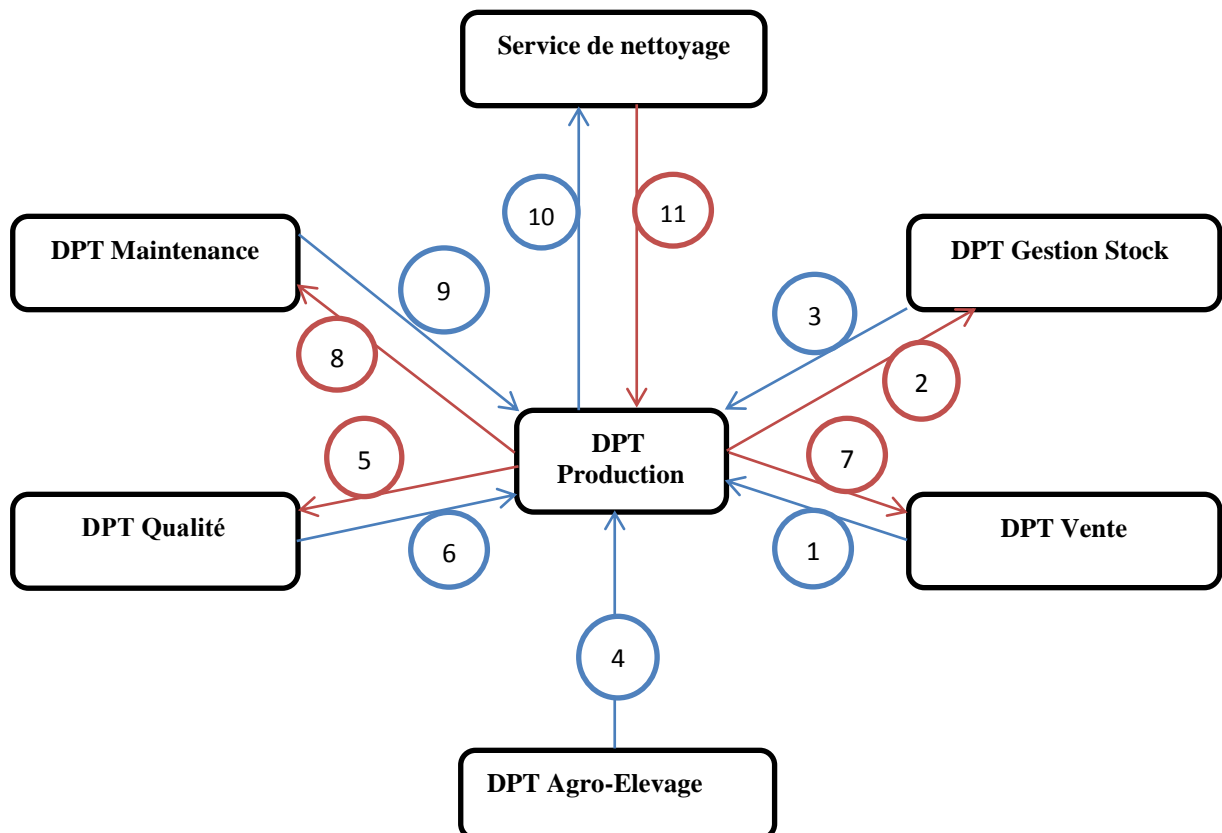


Figure I.11 : Flux d'informations

## Synthèse d'informations :

1.	Demande de La quantité de produit
2.	Demande de La matières premières
3.	Envoyer matières premières
4.	Envoyer le lait de vache
5.	Envoyer un échantillon du produit à chaque étape
6.	Réponse à continuer ou modification
7.	Fournir la quantité nécessaire de produit
8.	Demande de réparation dans le cas d'un dysfonctionnement
9.	Réformes de la demande en cas de dysfonctionnement
10.	Demande de nettoyage
11.	En réponse à la demande pour le nettoyage

**Tableau I.1 :** synthèse d'information

### I.3.11 Les différentes structures de l'entreprise

Laiterie Numidia comprend les structures suivantes :

#### I.3.11.1 Direction administration et finance

La direction administration et finance comprend les départements suivants :

##### I.3.11.1.1 Département finance comptabilité

Ce département comprend les services suivants :

#### a. Service comptabilité générale

Ses missions principales sont :

- Recenser les mouvements d'entrées et de sorties.
- Enregistrer correctement toute les opérations effectuées.
- Participer à l'élaboration du bilan.

#### b. Service comptabilité analytique

Sa mission principale est de traiter toute les informations comptables pour déterminer avec précision les différents coûts de production, de vente et de distribution.

#### c. Service finance

Ses missions principales sont :

- Contrôler l'état de la paie.
- Etablir le rapport d'activité mensuel, semestriel et annuel.
- Vérifier tous les documents comptables.
- Gérer la trésorerie de l'entreprise (caisse, recette et dépenses).

### I.3.11.1.2 Département administration générale

Ce département comprend les services suivants :

#### a. Service personnel

Ses missions principales sont :

- ✚ Recruter les candidats.
- ✚ Gérer la période d'essai.
- ✚ Gérer tous les contrats et les renouveler.
- ✚ Gérer les congés.
- ✚ Evaluer les fichiers des employés qui ont des primes.
- ✚ Régler et contrôler les dossiers de la paie.
- ✚ Gérer et régler l'état social (des allocations familial, remboursement des frais de maladie des employés).
- ✚ Régler et ranger tous les documents et les dossiers de l'entreprise.

#### b. Service paie

Sa mission principale est de régler la paie des employés.

#### c. Service moyens généraux

Sa mission principale est de gérer les dossiers des demandes de tous les moyens utilisés par les employés de l'entreprise et les renvoyer au service d'achat.

### I.3.11.1.3 Département informatique et organisation

Ce département est chargé de la réparation et la maintenance de matériels informatiques et logiciels.

### I.3.11.2 Direction commerciale

La direction commerciale est l'unité de l'entreprise qui s'intéresse par le suivi de la relation ou contact client avec l'entreprise, elle comprend les départements suivants :

#### I.3.11.2.1 Département d'achat

Ce département comprend un seul service qui est le :

##### Service d'achat

Ses missions principales sont :

- Recevoir les demandes matérielles.
- Acheter le matériel nécessaire pour l'entreprise.
- Gérer tous les dossiers des fournisseurs pour le bon déroulement de ses activités.
- Contrôler la quantité de la matière première avant son chargement et sa livraison à l'entreprise.
- Assurer dans les bonnes conditions l'opération de transport de la marchandise.

#### I.3.11.2.2 Département gestion de stock

Ce département comprend un seul service qui est le :

##### **Service gestion de stock**

Ses missions principales sont :

- Veiller à l'organisation des magasins en matière de stockage des produits.
- Coordonner entre les magasins et suivre les mouvements d'entrées et de sorties de la marchandise.
- Alimenter les différents départements des fournitures nécessaires pour la bonne marche de l'entreprise.
- Ranger le stock entré dans les magasins.
- Suivre l'opération d'achat.

#### I.3.11.2.3 Département de vente

Ce département comprend les deux services suivants :

##### **a. Service de recouvrement**

Ses missions principales sont :

- Recevoir le chèque du client et retire le registre de suivi client à terme et le met à jour.
- Remettre le chèque au service finance.
- Transmettre une copie de facture portant les références du chèque à la comptabilité générale.

##### **b. Service de vente**

Ses missions principales sont :

- ❖ Accueillir le client muni d'un dossier administratif dont la composition est diffusée par la direction générale conformément à la réglementation en vigueur.
- ❖ Inscire le client.
- ❖ Recevoir un bon de commercialisation du laboratoire et le classe dans l'emplacement correspondant.

- ❖ Recevoir un bon de livraison du magasin des produits finis.
- ❖ Veiller aux bons stockages des produits finis et propriétés des lieux (quai des distributions, chambre froide).
- ❖ Respecter le temps d'entrée des distributeurs et d'autre client au sein de la laiterie.
- ❖ Veiller au pointage des factures avec la structure des finances.
- ❖ Editer le bon de commande et la facture de client.
- ❖ Editer un bon mensuel des ventes a la fin de chaque mois.
- ❖ Recevoir la structure trésorerie.
- ❖ Editer les contrats.
- ❖ Etablir les commandes.

### I.3.11.3 Département agro-élevage

Ce département est chargé de gérer les dossiers des collecteurs de lait de vache.

### I.3.11.4 Département qualité

Ce département comprend un seul service qui est :

#### **Service physico-chimie :**

Ses missions principales sont :

Contrôler les différents stades de production.

- ✚ Procéder aux analyses, tests et examen de la matière première.
- ✚ Analyser et tester les produits finis.
- ✚ Contrôler les équipements et les diagrammes.
- ✚ Veiller à la régularité de la qualité.

### I.3.11.5 Département maintenance

Ce département comprend deux services qui sont :

#### **a. Service méthode**

Ce service est chargé de la maintenance des machines de la production.

#### **b. Service utilité**

Ce service est chargé des opérations de traitement d'eaux et de chaufferie.

### I.3.11.6 Département production

Ce département comprend un seul service qui est :

#### **Service Cleaning In Place (CIP)**

Ce service est chargé des opérations de nettoyage automatique.

#### I.4 Critiques de l'existant

- Collecter manuellement les informations sur des supports en papier
- Saisir des feuilles de calcul Excel non organisé
- Réseaux local non utilisable
- Manque au niveau des interactions avec le Web.
- L'entreprise est moins connue.
- Limitation au niveau des points de ventes, ce qui génère une limitation au niveau de la production.

Pour faciliter la vente, nous allons proposer un site Web dynamique facilitant la vente en ligne, et offrant un moyen technologique afin d'accroître la productivité de l'entreprise. Il pourrait être utile à un grand nombre de clients et il permet d'accélérer le procédé d'achat (il est devenu plus rapide parce qu'il n'est pas limitatif ni avec le temps ni avec le lieu non plus. Internet est un marché ouvert 24h/24h).

#### I.5 Conclusion

La gestion de production joue un rôle dans la durée de vie des entreprises, et cela en leurs permettant de bien s'organiser, et aussi en leurs donnant les différentes possibilités pour soit faire des bénéfices ou bien d'optimiser un ses critères comme le cout ou la qualité par exemple

A ce moment vient le rôle de l'informatique qui consiste à faire le travail de l'humain mais plus rapidement et efficacement pour sa rapidité de calcul et sa capacité de stockage.

Une gestion de production assistée par ordinateur donnera de meilleurs produits en matière de qualité, quantité et même pour leurs couts et temps de production puisque c'est l'ordinateur qui va calculer à l'avance les différents résultats possibles, et il va choisir le plus optimal au fur et à mesure du travail demandé par l'utilisateur.

Une usine ne peut être gérer par des humains seulement, l'informatisation est obligatoire dans le cas où on veut vraiment développer et optimiser l'entreprise, soit du côté financière ou bien de côté réputation c'est-à-dire rendre le travail plus rapidement.



## II.1 Introduction

Le développement logiciel est en train de changer radicalement vers l'infrastructure informatique distribuée qui est à l'abri des futures tendances pour cette raison plusieurs architectures d'applications distribuées et outils technologiques de constructions ont été développés.

Ce chapitre présente un état de l'art dans le cadre des applications distribuées. Dans le but ici est tout d'abord, de faciliter la compréhension du problème posé par ces applications à travers leur formalisation, leur classement et la difficulté de leur résolution. Nous introduisons pour cela la définition des applications distribuées, Cela nous permet ensuite de présenter en détail des exemples et des exigences des applications distribuées. Et aussi nous allons définir quelques généralités portant sur la méthode et outils mettant en évidence la réalisation de notre projet. Nous allons commencer par présenter le langage de modélisation unifié UML (Unified Modeling Language),

Aussi d'autre partie représente les étapes fondamentales pour le développement et comprendre de façon plus approfondie le fonctionnement de notre système de suivi de la production de l'entreprise « laiterie Numidia ».

Pour l'analyse et la conception de notre application, nous avons choisis de modéliser avec le formalisme UML 2 qui offre une flexibilité marquante qui s'exprime par l'utilisation des diagrammes.

## II.2 Les applications distribuées

### II.2.1 Définition

Une application informatique distribuée est une collection de programmes élémentaires sur des postes qui peuvent être géographiquement séparés connectés à l'aide d'un réseau de communication. Chaque poste exécute des composantes, de telle sorte qu'un utilisateur perçoive le système comme un unique système intégré.

Une propriété importante des applications distribuées est que la distribution est généralement cachée pour l'utilisateur et les programmeurs de l'application. Il préfère l'ensemble comme une seule et unique application et ainsi cacher la complexité de la distribution le plus possible et augmenter la transparence de l'application distribuée. Cela permet de développer le plus possible les applications de la même façon que les applications centralisées.

Une application distribuée est généralement séparable en plusieurs composantes entièrement autonomes. Il n'existe pas de composante maître qui gère les autres et chacune est donc responsable de son propre fonctionnement. Cela permet, entre autres, d'avoir une hétérogénéité dans la technologie utilisée pour chaque composante, ils peuvent être écrits dans différents langages de programmation (java, Cobol, C++, etc.) et s'exécuter sur différents systèmes d'exploitation (Mac OS X, Linux, Windows, etc.). L'autonomie des composantes fait que les applications sont exécutées simultanément (programmation concurrente). De plus, contrairement défaillances (problème de composantes, réseau, trafics, etc.). [5]

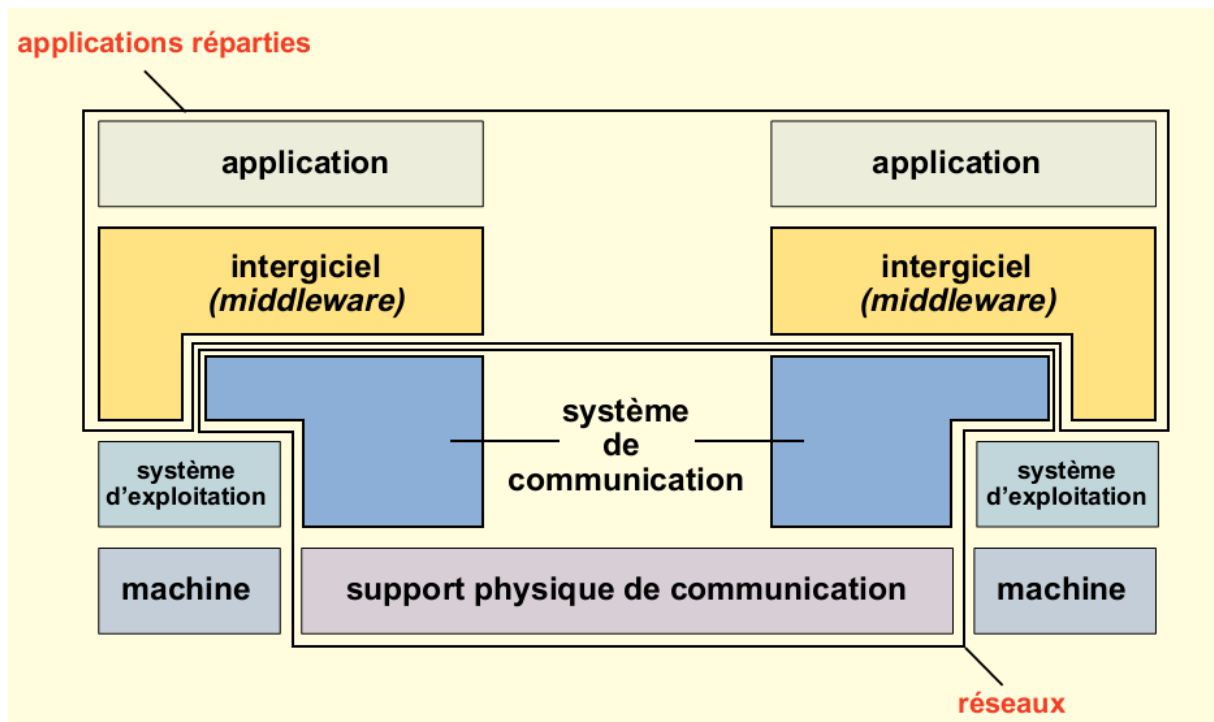


Figure II.1: Présentation de l'application distribuée

## II.2.2 Motivations des applications distribuées

- ✚ **Accès Distance :** Les structures de grandes dimensions telles que la défense, les banques ou les grandes entreprises voient leurs activités s'étendre planétairement et ne peuvent plus centraliser toute l'information sur un système unique. Sa demande de plus en plus de transactions courantes s'effectue via des réseaux informatiques tels que l'internet.
- ✚ **Ressources Distribuées :** Les applications distribuées utilisent des ressources distribuées tels que les fichiers les imprimantes les disques ce qui permet de partager ces ressources entre les utilisateurs ...etc.
- ✚ **Parallélisme :** Avec les applications distribuées le calcul peut être distribué ou réparti, ce qui consiste à répartir un calcul sur plusieurs ordinateurs distincts. Le calcul distribué est une forme de Calcul parallèle. On même répartir les tâches élémentaire pour accomplir un travail par exemple avec une application de gestion on peut avoir un programme qui s'occupe de la collecte des informations sur les clients, un autre sur les fournisseurs et un autre pour faire des calculs des factures...etc. [8]

### II.2.3 Les exigences des applications distribuées

L'utilisation des applications distribuées demande des exigences pour qu'elles puissent jouer leurs rôles :

#### II.2.3.1 Extensibilité « Mise à l'échelle »

- ✓ Les applications distribuées doivent être facilement extensibles pour accomplir les traitements de des nouvelles tâches que les infrastructures auront besoin avec le temps.
- ✓ Les composants des systèmes distribués possèdent des interfaces bien définies ce qui leur permet d'être facilement extensible et modifiable. Les services web sont un exemple de système distribué qui possède une grande ouverture.
- ✓ Les mises à jour des données de base sont moins coûteuses et pénalisantes pour les performances des applications.

#### II.2.3.2 Hétérogénéité

L'application distribuée dans sa globalité doit supporter l'hétérogénéité des composants et les plateformes sur lesquelles se tournent, les composants des applications distribuées peuvent être écrits en différents langages et sur différentes machines. Par exemple, les éléments d'un serveur peuvent être programmés en C++ ou en DELPHI et s'exécuter sous Unix, alors que le client peut être en Java et s'exécuter sous Windows.

#### II.2.3.3 Accès aux ressources partagés

Les applications distribuées fournissent des moyens pour partager les ressources, c'est-à-dire à la fois le matériel, le logiciel et les données.

#### II.2.3.4 Tolérance aux pannes

Les applications distribuées doivent être plus tolérantes aux pannes que les applications centralisées, car la panne d'une composante (programme) ne touche pas tous les services offerts par l'application elle ne doit pas s'arrêter mais elle doit continuer de fonctionner dans sa globalité et on peut même garder des copies des programme sur des postes de réserves. [9]

### II.3 les outils de programmations réparties

Parmi les services nécessaires à la mise en œuvre d'application distribuée, nous distinguons les outils de développement, qui permettent la spécification, la programmation, l'intégration et la mise au point des applications, qui permettent de mettre en place l'application et de l'adapter à l'évolution des besoins.

#### II.3.1 TCP et UDP

TCP est un protocole de transport fiable, en mode connecté et en mode bidirectionnel. Une socket TCP peut être utilisée par plusieurs connexions en même temps. Une connexion est identifiée par le couple adresse IP, port à chaque extrémité de la communication. Un échange TCP est un flot continu d'octets, les données sont reçues dans leur ordre d'émission. Il est possible d'utiliser des tampons pour optimiser les transmissions (buffer).

UDP est un protocole de transport non fiable, sans connexion, en mode bidirectionnel. L'échange de donnée se fait à l'aide de datagrammes. Il n'y a pas de garantie sur l'ordre d'arrivée des données. La communication par sockets est souvent comparée aux communications humaines

On distingue ainsi deux modes de communication :

- **Le mode connecté** (comparable à une communication téléphonique), utilisant le protocole TCP. Dans ce mode de communication, une connexion durable est établie entre les deux processus, de telle façon que l'adresse de destination n'est pas nécessaire à chaque envoi de données.
- **Le mode non connecté** (analogue à une communication par courrier), utilisant le protocole UDP. Ce mode nécessite l'adresse de destination à chaque envoi, et aucun accusé de réception n'est donné. [10]

#### II.3.2 Socket

##### II.3.2.1 Définition :

C'est une norme de communication sur réseau, mis au point à Berkeley (1982), qui permet une application de dialoguer avec un protocole (nous parlerons ici du protocole TCP/IP).

L'objectif des sockets est de fournir des moyens de communication entre processus locaux ou distant (Inter Processus de Communication), en masquant l'implémentation des couches de transport et en fournissant un système programmation (API) proche de la lecture ou l'écriture dans un fichier. On peut traduire la notion de socket par le mot français « prise », c'est un point d'accès pour les services de transport. La socket est caractérisée par un type de protocole utilisé, un ensemble de primitives de service, elle encapsule des données (descriptifs, file d'attente) et elle est identifiée de manière unique par un numéro de port et une adresse IP (dans le cas où le protocole utilisé est TCP/IP). Certains ports sont réservés à des applications connues (53 : DNS, 21 : FTP, 23 : TELNET, 80 : http, ...). Les sockets peuvent être utilisés avec d'autres protocoles (UDP, AppleTalk, Xerox XNS, ...).

### II.3.2.2 Déroulement d'une communication avec sockets :

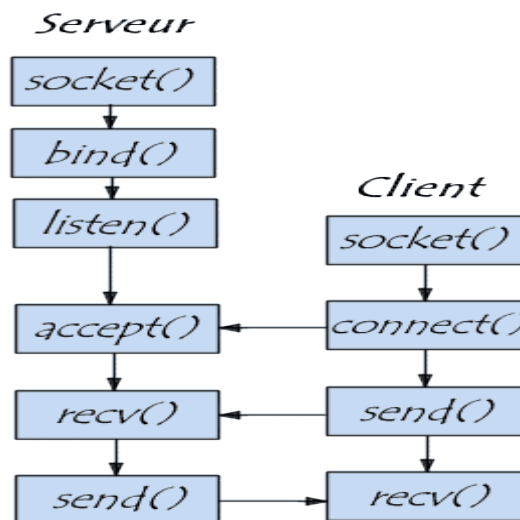
Comme dans le cas de l'ouverture d'un fichier, la communication par socket utilise un descripteur pour désigner la connexion sur laquelle on envoie ou reçoit les données. Ainsi la première opération à effectuer consiste à appeler une fonction créant un socket et retournant un descripteur (un entier) identifiant de manière unique la connexion. Ainsi ce descripteur est passé en paramètres des fonctions permettant d'envoyer ou recevoir des informations à travers le socket.

L'ouverture d'un socket se fait en deux étapes :

- La création d'un socket et de son descripteur par la fonction *socket()*.
- La fonction *bind()* permet de spécifier le type de communication associé au socket (protocole TCP ou UDP)

Un serveur doit être à l'écoute de messages éventuels. Toutefois, l'écoute se fait différemment selon que le socket est en mode connecté (TCP) ou non (UDP).

- **En mode connecté**, le message est reçu d'un seul bloc. Ainsi en mode connecté, la fonction *listen()* permet de placer le socket en mode passif (à l'écoute des messages). En cas de message entrant, la connexion peut être acceptée grâce à la fonction *accept()*. Lorsque la connexion a été acceptée, le serveur reçoit les données grâce à la fonction *recv()*.

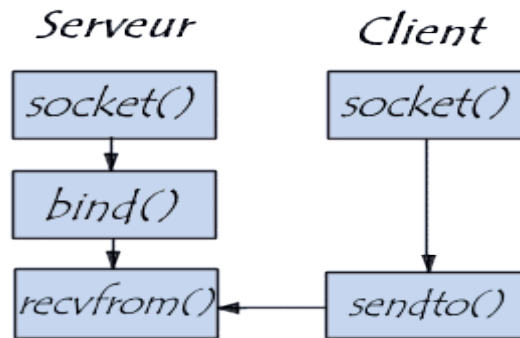


**Figure II.2:** Schéma d'une communication en mode connecté

- **En mode non connecté**, comme dans le cas du courrier, le destinataire reçoit le message petit à petit (la taille du message est indéterminée) et de façon désordonnée.

Le serveur reçoit les données grâce à la fonction *recvfrom()*. Et La fin de la connexion se fait grâce à la fonction *close()*.

Voici le schéma d'une communication en mode connecté:



**Figure II.3:** Schéma d'une communication en mode connecté

### II.3.3 RMI (Remote Method Invocation)

#### II.3.3.1 Définition :

RMI est une API Java permettant de manipuler des objets distants (c'est-à-dire un objet instancié sur une autre machine virtuelle, éventuellement sur une autre machine du réseau) de manière transparente pour l'utilisateur, c'est-à-dire de la même façon que si l'objet était sur la machine virtuelle (JVM) de la machine locale. Ainsi un serveur permet à un client d'invoquer des méthodes à distance sur un objet qu'il instancie. Deux machines virtuelles sont donc nécessaires (une sur le serveur et une sur le client) et l'ensemble des communications se fait en Java.

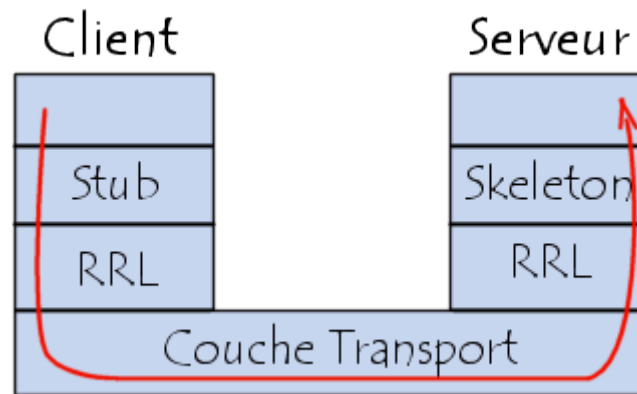
On dit généralement que RMI est une solution "tout Java", contrairement à la norme Corba de l'OMG (*Object Management Group*) permettant de manipuler des objets à distance avec n'importe quel langage. Corba est toutefois beaucoup plus compliqué à mettre en œuvre, c'est la raison pour laquelle de nombreux développeurs se tournent généralement vers RMI.

#### II.3.3.2 Structure des couches RMI :

Les connexions et les transferts de données dans RMI sont effectués par Java sur TCP/IP grâce à un protocole propriétaire (**JRMP**, *Java Remote Method Protocol*) sur le port 1099.

A partir de Java 2 version 1.3, les communications entre client et serveur s'effectuent grâce au protocole RMI-IIOP (*Internet Inter-Orb Protocol*), un protocole normalisé par l'OMG (*Object Management Group*) et utilisé dans l'architecture CORBA.

La transmission de données se fait à travers un système de couches, basées sur le modèle OSI afin de garantir une interopérabilité entre les programmes et les versions de Java.



**Figure II.4:** Structure des couches RMI

Le **stub** (traduisez *souche*) et le **skeleton** (traduisez *squelette*), respectivement sur le client et le serveur, assurent la conversion des communications avec l'objet distant.

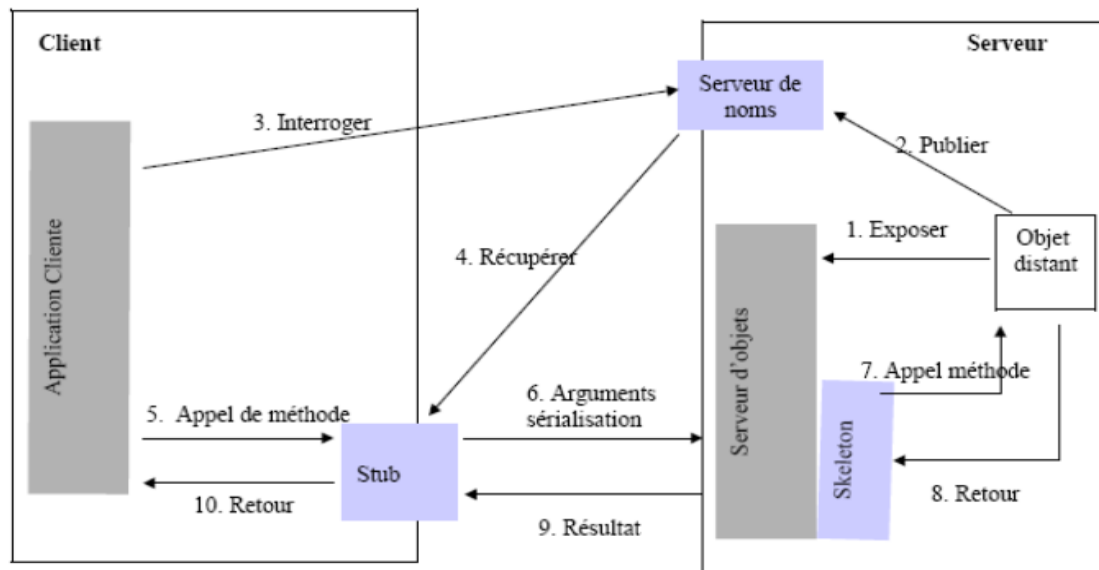
La **couche de référence RRL** (*remote Reference Layer*) est chargée du système de localisation afin de fournir un moyen aux objets d'obtenir une référence à l'objet distant. Elle est assurée par le package *java.rmi.Naming*. On l'appelle généralement **registre RMI** car elle référence les objets.

La **couche de transport** permet d'écouter les appels entrants ainsi que d'établir les connexions et le transport des données sur le réseau par l'intermédiaire du protocole TCP. Les packages *java.net.Socket* et *java.net.SocketServer* assurent implicitement cette fonction.

Ainsi, une application client-serveur basée sur RMI met ainsi en œuvre trois composantes :

- ✓ une application cliente implémentant le stub
- ✓ une application serveur implémentant le skeleton (*squelette*)
- ✓ une application médiatrice (le registre RMI) servie par un processus tiers (*rmiregistry*)

### II.3.3.3 Architecture de RMI : L'architecture de RMI est schématisée ci-dessous :



**Figure II.5:** L'architecture de RMI

Lorsqu'un objet instancié sur une machine cliente désire accéder à des méthodes d'un objet distant, il effectue les opérations suivantes :

- il localise l'objet distant grâce à un service de désignation : le **registre RMI**
- il obtient dynamiquement une image virtuelle de l'objet distant (appelée **stub** ou *souche* en français). Le stub possède exactement la même interface que l'objet distant.
- Le stub transforme l'appel de la méthode distante en une suite d'octets, c'est ce que l'on appelle la sérialisation, puis les transmet au serveur instanciant l'objet sous forme de flot de données. On dit que le stub "*marshalise*" les arguments de la méthode distante.
- Le squelette instancié sur le serveur "désérialise" les données envoyées par le stub (on dit qu'il les "*démarshalise*"), puis appelle la méthode en local
- Le squelette récupère les données renvoyées par la méthode (type de base, objet ou exception) puis les marshalise
- le stub démarshalise les données provenant du squelette et les transmet à l'objet faisant l'appel de méthode à distance

### II.3.3.4 Mise en œuvre de RMI :

Pour créer une application avec RMI il suffit de procéder comme suit :

1. définir la classe distante. Celle-ci doit dériver de `java.rmi.server.UnicastRemoteObject` (utilisant elle-même les classes `Socket` et `SocketServer`, permettant la communication par protocole TCP)



2. définir l'interface pour la classe distante. Celle-ci doit implémenter l'interface *java.rmi.Remote* et déclarer les méthodes publiques globales de l'objet, c'est-à-dire les méthodes partageables. De plus ces méthodes doivent pouvoir lancer une exception de type *java.rmi.RemoteException*.
3. créer les classes pour le stub et le squelette grâce à la commande *rmic*
4. Lancer le registre RMI et lancer l'application serveur, c'est-à-dire instancier l'objet distant. Celui-ci lors de l'instanciation créera un lien avec le registre
5. Créer un programme client capable d'accéder aux méthodes d'un objet sur le serveur grâce à la méthode *Naming.lookup()*
6. Compiler l'application cliente
7. Instancier le client. **[12]**

## II.4 Etude préliminaire

C'est une étape primordiale et cruciale au début de chaque démarche de développement. Son but est de veiller à développer un logiciel adéquat, sa finalité est la description générale des besoins du système (les besoins peuvent être fonctionnels ou opérationnels) en utilisant principalement le texte. Cette étape prépare l'activité plus formelle de description des besoins par les cas d'utilisation.

### II.4.1 Présentation du projet a réalisé

Le but de ce projet de fin d'étude, est de faire une conception, et une implémentation d'un système informatique répartie pour la gestion de production de l'entreprise « laiterie Numidia» depuis l'acquisition des matières premières jusqu'à livraison du produit fini.

### II.4.2 Définition des grands choix technique

Cette phase consiste à fixer les techniques clés qu'on va utiliser pour le développement parmi les techniques suivies de notre projet est :

- ✚ **L'outil de modélisation** : langage UML 2 pour les applications Java.
- ✚ **L'architecture de logiciel** : Client /serveur (deux tiers).

### II.4.3 Présentation générale du problème

Afin de faciliter et d'accélérer l'opération de gestion production de l'entreprise « laiterie Numedia », Nous vous proposons la création d'un système informatique pour la gestion de production, ce système est soumis aux règles suivantes :

- Le service commercial transmet une commande indiquant la quantité de chaque produit a préparé vers le service de fabrication.
- Le chef de service de fabrication fait l'objet d'une demande d'approvisionnement vers le service de stockage, ce dernier délivre les matières exigées accompagner d'un bon de transfère (contenant les matières de préparation et de conditionnement : M.G.L.A, poudre de lait, pochettes) ou un bon de sortie (produit de nettoyage : Acide nitrique, Soude caustique...) ces bons porterons un numéro unique, la date, désignation des matières avec leur quantité et la signature des deux services.
- Le chef de service de fabrication accède à la préparation des produits, cette préparation est suivie par le service de la qualité qui contrôle le produit sur des normes bien définis.
- Si les résultats de contrôle sont valides, le produit préparé sera transféré vers l'atelier de conditionnement, qui conditionnera le produit dans des pochettes cette opération nous donnera un produit fini.
- Si une des machines tombe en panne le chef de service de fabrication avise le service de maintenance pour activer la réparation.

- Après chaque fin de travail le service de fabrication consulte le service de nettoyage (CIP : Cleaning In Place) pour procéder au nettoyage des machines.
- Le service conditionnement livre le produit fini au service commercial accompagné d'un bon de session comportant un numéro identique, la date, la désignation des produits avec leur quantité et la signature des deux services.
- En cas ou une défaillance de nos produit est signalé par le service commercial cela nécessite la délivrance d'un bon de retour libellé (date, désignation, quantité, heure et la signature des services concernés) vers le service conditionnement. Le même scénario se produira entre le service conditionnement et celui de fabrication qui fera la remise d'un bon de recyclage (date, heure, quantité, nombre de cuve et la signature des deux services).
- Le service fabrication et le chef département de la production s'intéressent au suivi des consommations des matières premières, et la réalisation des produits finis chaque jour afin de créer un bulletin de renseignement quotidien (BRQ) qui sera transmis vers Alger pour information et estimation de la production. Et assurer la gestion des personnels pour un bon fonctionnement de l'entreprise.
- Le chef département production calcule le cout estimé de chaque produit.
- Le service maintenance doit suivre attentivement toutes pannes nuisibles à l'amélioration de la production.

#### II.4.4 Recueil des besoins fonctionnels

Permet de décrire textuellement les grandes fonctionnalités attendues du système à construire :

- ❖ Suivi des consommations des matières premières par produit et par équipe.
- ❖ suivi de la répartition de consommation de lait de vaches par les produits.
- ❖ Contrôle de la qualité des produits selon des normes.
- ❖ Gestion des réalisations de produit fini par produit, par équipe et par jour.
- ❖ Répartition des charges (dépenses) sur les produits.
- ❖ Suivi des contraintes à la production par atelier et par équipement.
- ❖ Edition des états statistiques sur la production.
- ❖ Gestion du personnel de la production (ressources humaines).
- ❖ Gestion de vente du produit fini

#### II.4.5 Recueil des besoins opérationnels

C'est-à-dire les besoins liés à l'exploitation du système (pour garantir la sécurité).

- ✓ Chaque acteur doit être authentifié avant d'accéder au système (Nom d'utilisateur et mot de passe).

#### II.4.6 Description du contexte du système

Consiste les activités successives suivante :

### II.4.6.1 L'identification des acteurs

Nous allons maintenant énumérer les acteurs susceptibles d'interagir avec le système, mais d'abord nous donnons une définition de l'acteur.

Un acteur représente un rôle joué par une personne ou une chose (dispositif matériel ou autre système) qui interagit directement avec le Système étudié. Les acteurs existant dans notre conception sont :

1. Chef département de production.
2. Chef service de fabrication.
3. Magasinier.
4. Agent commercial.
5. Membre de laboratoire.
6. Agent de nettoyage CIP (Cleaning In Place).
7. Agent de maintenance.
8. Administrateur.

### II.4.6.2 L'identification des cas d'utilisations

Le cas d'utilisation est une description des interactions qui vont permettre à l'acteur d'atteindre son objectif en utilisant le système.

#### II.4.6.2.1 Suivi des consommations des matières premières

Permet de gérer la consommation des matières premières de chaque produit dans les différentes étapes de production (préparation, pasteurisation et conditionnement), gérer le stock initial, la réception de nouveau stock, quantité consommée et le résultat de stock final de chaque journée pour chaque équipe de travail (équipe de soir ou équipe de matin) qui s'organise en suivant un emploi du temps spécifié.

#### II.4.6.2.2 Suivi de la répartition de consommation de lait de vaches sur les produits

Ce cas d'utilisation présente l'affectation de lait de vache sur les différents produits.

#### II.4.6.2.3 Contrôle de la qualité des produits

L'équipe du laboratoire doit suivre la qualité du produit dans chaque phase de cycle de vie en assumant des normes bien définies pour assurer un produit de haute qualité.

#### II.4.6.2.4 Gestion des réalisations du produit fini

Permet de calculer la quantité exacte des produits finis réalisés par les équipes de travail de chaque journée et estimer le taux de production réalisés en fonction de l'objectif fixé à atteindre.

#### II.4.6.2.5 Répartition des charges sur les produits

Estimation du cout de chaque produit, en prenant en considération toutes les Dépenses directe ou indirecte impliqué dans la procédure de production: main d'œuvre, matières premières, eau, Électricité et gaz ..... ect.

#### II.4.6.2.6 Suivi des arrêts de la production

Permet de suivre toutes les contraintes et les handicapes qui résulte la diminution de la production comme les pannes des machines, la coupure d'électricité, retard de travailleurs..ect

#### II.4.6.2.7 Gestion du personnel de la production (ressources humaines)

Son fonctionnement ne permettra la programmation et l'organisation des plannings des équipes de travail, matin et soir en spécifiant les 6 jours de travail et les 2 jours de congé pour chaque ouvrier. La planification ne permettra aussi de gérer tous les retards et les absences du personnel de travail et organiser les congés pour assurer une bonne marche de l'entreprise.

#### II.4.6.2.8 Edition des états et statistiques sur la production

Permet de présenter le traitement des données collectées, l'interprétation, et la présentation, afin de rendre les données compréhensibles par tous et sous forme de graphe.

#### II.4.6.2.9 Vente des produits fini

Le service conditionnement avis le service commercial de la disponibilité des produits, un bon de session sera établit comportant la quantité et la désignation des produits commandés, cette marchandise sera acheminés vers le point de vente commercial.

## II.5 Modélisation

### II.5.1 Langage de modélisation UML

#### II.5.1.1 Introduction

Merise et UML sont deux grands principes de « traduction » ou modélisation d'un système d'information. Néanmoins, ils ne sont pas aussi proches qu'on pourrait le penser. L'une ou l'autre présente des avantages et des inconvénients. Il est réservé au concepteur de choisir la méthode la mieux adaptée pour son cas, La conception de notre application mérite bien une grande précision et une exploitabilité maximale. C'est la raison pour laquelle on va retenir UML, et aussi UML (en anglais **Unified Modeling Language**, que l'on peut traduire par « langage de modélisation unifié ») est une notation unique et standard pour faciliter la conception de programmes. Ce langage est l'accomplissement de la fusion de précédents langages modélisation objet. Parmi ces nombreux avantages on peut citer :

- UML est un langage formel et normaliser



- Gain de précision.
  - Gage de stabilité.
  - Encourage l'utilisation d'outils.
- UML est un support de communication performant
    - Il cadre l'analyse.
    - Il facilite la compréhension de représentations abstraites complexes.
    - Son caractère polyvalent et sa souplesse en font un langage universel.
  - UML « unifie » des méthodes de conception logicielle orientées objet.

### II.5.1.2 Définition d'UML et les caractéristiques

UML est une norme du langage de modélisation objet qui a été publiée, dans sa première version, en novembre 1997 par l'OMG (Object Management Group), instance de normalisation internationale du domaine de l'objet, UML est une représentation standardisée d'un système orienté objet.

UML ne propose pas de méthode de réalisation. UML est totalement indépendant des langages objet de développement. UML est une notation graphique formelle normalisée, il est fondé sur un méta-modèle qui définit les éléments de modélisation et la sémantique de ces éléments.

### II.5.1.3 La modélisation UML

Combinés, les différents types de diagrammes UML offrent une vue complète des aspects statiques et dynamiques d'un système, en UML 2 il existe 13 diagramme :

#### II.5.1.3.1 Les vues statiques

**Diagrammes de cas d'utilisation** : décrit les interactions entre le futur système et ses utilisateurs.

**Diagrammes de classes** : est généralement considéré comme le plus important dans un développement orienté objet exprime la structure statique du système en termes de classes et de relations entre ces classes

**Diagrammes d'objets** : représente des instances d'objets qui existent à un moment donné dans le système.

**Diagramme de packages** : Il montre l'organisation logique du modèle et les relations entre packages

**Diagramme de structure composite** : Il montre l'organisation interne d'un élément statique complexe.

**Diagrammes de composants** : décrivent les composants et leurs dépendances dans l'environnement de réalisation

**diagrammes de déploiement** : montrent la disposition physique des différents matériels (les nœuds) sur lesquels le logiciel sera déployé.

### II.5.1.3.2 Les vues dynamiques

**Diagrammes de collaboration** : permet de mettre en évidence les interactions entre les différents objets du système.

**Diagramme de vue d'ensemble des interactions** : Il fusionne les diagrammes d'activité et de séquence pour combiner des fragments d'interaction avec des décisions et des flots.

**Diagramme de temps** : Il fusionne les diagrammes d'états et de séquence pour montrer l'évolution de l'état d'un objet au cours du temps.

**Diagrammes de séquence** : montre un enchaînement d'opérations entre des objets du système

**diagrammes d'états-transitions** : Ils ont pour rôle de représenter les traitements (opérations) qui vont gérer le domaine étudié. Ils définissent l'enchaînement des états de classe et font donc apparaître l'ordonnancement des travaux.

**Diagrammes d'activités** : il montre l'enchaînement des activités qui concourent au processus.

II.6 Partie d'analyse détaillée

II.6.1 Présentation diagramme de cas d'utilisation

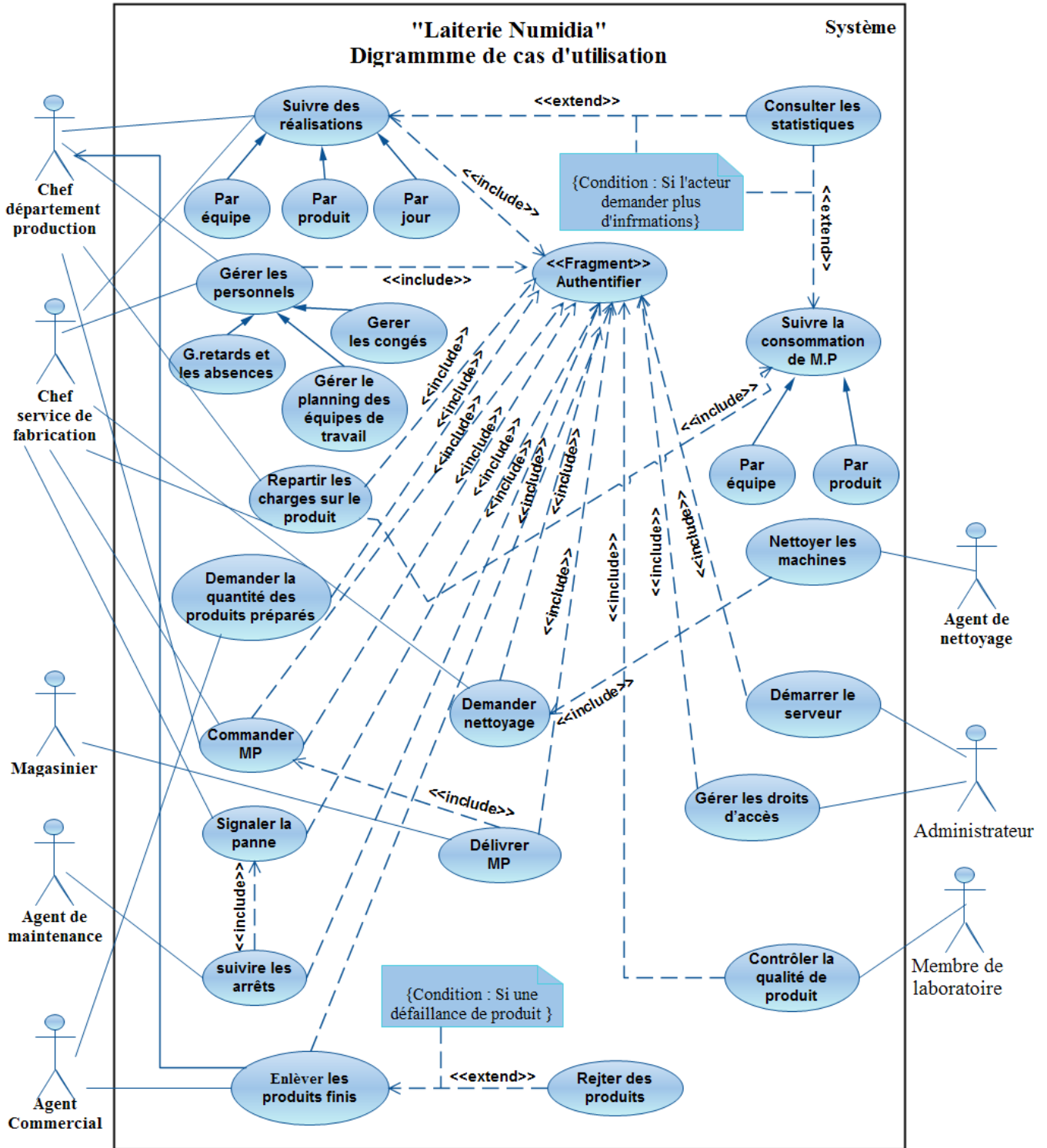


Figure II.6 : Diagramme cas d'utilisation



Dans ce qui suit, on va établir pour chaque cas d'utilisation une fiche descriptive permettant de classifier le fonctionnement du futur système. Gérer les droits d'accès.

## II.6.2. Les fiches descriptives

### II.6.2.1 Cas d'utilisation « Authentifier »

Pour le cas d'utilisation 'Authentifier' la fiche descriptive est la suivante :

<p><b>Nom :</b> Authentifier</p> <p><b>Objectif :</b> Permet l'accès au système (application) par les agents concernés au système et l'agent administratif, et pour donner plus de sécurité à notre application.</p> <p><b>Acteurs principaux</b> Chef département de production, Chef service de fabrication, Magasinier, Agent commercial, Membre de laboratoire, Agent de nettoyage, Agent de maintenance, Administrateur.</p> <p><b>Acteurs secondaires :</b> Néant</p>
<p><b>Les Préconditions :</b></p> <ul style="list-style-type: none"> <li>• possède l'application</li> <li>• possède un mot de passe</li> </ul> <p><b>Des scénarii :</b></p> <p><b>Scénario nominal :</b></p> <ol style="list-style-type: none"> <li>1- l'acteur de système lance l'application.</li> <li>2- Le Système renvoi un formulaire d'identification.</li> <li>3 l'agent de système rempli le formulaire d'identification.</li> <li>4- Le System affiche l'espace de l'acteur.</li> </ol>
<p><b>Enchaînements alternatifs :</b></p> <p><b>3.0 :</b> L'authentification est incorrecte (Le système ne reconnaît pas le mot de passe) L'enchaînement 3.0 démarre au point 3 du scénario nominal.</p> <p>3.1- Le Système réaffiche le formulaire d'identification avec un message d'erreur.</p> <p><b>Enchaînements d'erreur :</b></p> <p><b>2.0 :</b> Annulation de l'authentification L'enchaînement 2.0 démarre au point 2 du scénario nominal.</p> <p>2.1 L'acteur annule l'authentification</p> <p>2.2 Le Système se ferme automatiquement, et le cas d'utilisation se termine en échec.</p> <p><b>Les Postconditions :</b> Affiche l'espace (l'agent de système concerné).</p>

**Tableau II.1** La fiche descriptive de cas « Authentifier »

### II.6.2.2 Cas d'utilisation « Suivre la consommation des matières premières »

Pour le cas d'utilisation 'Suivre la consommation des MP' la fiche descriptive est la suivante :

<p><b>Nom :</b> Suivre la consommation des MP</p> <p><b>Objectif :</b> Permet de gérer la consommation des matières premières de chaque produit.</p> <p><b>Acteurs principaux :</b> Chef service de fabrication, Chef département de production</p> <p><b>Acteurs secondaires :</b> Néant</p>
<p><b>Les Préconditions :</b></p> <ul style="list-style-type: none"> <li>• Authentification des acteurs concernés.</li> <li>• Le planning des équipes de travail est déjà créer.</li> </ul> <p><b>Des scénarii :</b></p> <p><b>Scénario nominal :</b></p> <ol style="list-style-type: none"> <li>1- L'acteur sélectionne le suivi de consommation MP.</li> <li>2- Le Système affiche le formulaire de suivi de consommation MP.</li> <li>3- L'acteur choisit la matière première et l'équipe de travail et la date.</li> <li>4- L'acteur remplit les formulaires de suivi MP avec l'ensemble des informations Nécessaires à la comptabilité matières.</li> <li>5- L'acteur demande l'enregistrement de fiche suivi MP.</li> <li>6- Le système enregistre la fiche dans le serveur avec un message de confirmation de l'envoi.</li> <li>7- Le système réaffiche le récapitulatif de suivi de MP de mois.</li> </ol> <p><b>Enchaînements alternatifs :</b></p> <p><b>5.0 : la fiche de suivi MP n'est pas enregistrée (si sélectionne même jour, équipe et même matière première).</b></p> <p>L'enchaînement 5.0 démarre au point 5 du scénario nominal.</p> <ol style="list-style-type: none"> <li>5.1- Le Système affiche le message d'erreur.</li> <li>5.2 - Le scénario nominal continue au point 3.</li> </ol> <p><b>Enchaînements d'erreur :</b></p> <p>2.0 : choisit une autre option</p> <p>L'enchaînement 2.0 démarre au point 2 du scénario nominal.</p> <ol style="list-style-type: none"> <li>2.1-L'acteur demande une autre option</li> <li>2.2 le Système affiche les informations concerné.</li> </ol> <p><b>Les Postconditions :</b></p> <p>La fiche de suivi de matière première a été enregistrée.</p>

**Tableau II.2 :** La fiche descriptive de cas « Suivre de consommation matière premier »

### II.6.2.3 Cas d'utilisation « Suivre les arrêts »

Pour le cas d'utilisation ‘*Suivre les arrêts*’ la fiche descriptive est la suivante :

<p><b>Nom :</b> Suivre les arrêts.</p> <p><b>Objectif :</b> Permet de suivi toutes les contraintes qui entravent la production.</p> <p><b>Acteurs principaux :</b> Agent de maintenance</p> <p><b>Acteurs secondaires :</b> Néant</p>
<p><b>Les Préconditions :</b></p> <ul style="list-style-type: none"> <li>• Authentification d’agent de maintenance.</li> </ul> <p><b>Des scénarii :</b></p> <p><b>Scénario nominal :</b></p> <ol style="list-style-type: none"> <li>1- L’acteur sélectionne la gestion des arrêts.</li> <li>2- Le Système renvoi la fenêtre de gestion des arrêts.</li> <li>3- L’acteur choisit l’équipe et l’atelier et le jour.</li> <li>4- L’acteur remplit les formulaires de gestion des contraintes.</li> <li>5- L’acteur demande l’enregistrement de fiche gestion des contraintes.</li> <li>6- Le système enregistre la fiche.</li> <li>7- Le Système renvoi un message de confirmation de sauvegarde.</li> </ol> <p><b>Enchaînements alternatifs :</b> Néant</p> <p><b>Enchaînements d’erreur :</b></p> <p>2.0 : L’acteur choisit option de consulter les contraintes L’enchaînement 2.0 démarre au point 2 du scénario nominal.</p> <p>2.1-L’acteur demande cette option.</p> <p>2.2 le Système affiche les détails des contraintes.</p> <p><b>Les Postconditions :</b> La fiche de gestion des contraintes a été enregistrée.</p>

**Tableau II.3:** La fiche descriptive de cas « Suivre des arrêts de la production »

### II.6.2.4 Cas d'utilisation « Gérer les personnels »

Pour le cas d'utilisation 'Gérer les personnels' la fiche descriptive est la suivante :

<p><b>Nom :</b> Gérer les personnels.</p> <p><b>Objectif :</b> Permet de de gérer les plannings des équipes de travail, le retard et les congés des ouvrier.</p> <p><b>Acteurs principaux :</b> Chef service de fabrication.</p> <p><b>Acteurs secondaires :</b> Néant</p>
<p><b>Les Préconditions :</b></p> <ul style="list-style-type: none"> <li>• Authentification des acteurs concernés.</li> <li>• La fiche de planning de travail précédente</li> </ul> <p><b>Des scénarii :</b></p> <p><b>Scénario nominal :</b></p> <ol style="list-style-type: none"> <li>1- L'acteur sélectionne la gestion du personnel.</li> <li>2- Le Système renvoie la fenêtre de gestion du personnel.</li> <li>3- L'acteur choisit l'option concerné (planning des équipes, retard, congé).</li> <li>4- Le système affiche le tableau.</li> <li>5- L'acteur remplit le tableau.</li> <li>6- L'acteur demande d'enregistrement de tableau.</li> <li>7- Le système enregistre le tableau.</li> <li>8- Le Système renvoie un message de confirmation de sauvegarde.</li> </ol> <p><b>Enchaînements alternatifs :</b></p> <p><b>6.0 le saisi de tableau erroné (manque d'information)</b> L'enchaînement 6.0 démarre au point 6 du scénario nominal.</p> <p>6.1 Le système envoie un message d'information.</p> <p><b>Enchaînements d'erreur :</b></p> <p><b>4.0 :</b> L'acteur choisit une autre option (tableau de planning de mois passé.....) L'enchaînement 4.0 démarre au point 4 du scénario nominal.</p> <p>4.1-L'acteur demande cette option.</p> <p>4.2 le Système affiche le détail existe.</p> <p><b>Les Postconditions :</b> La création de table planning des équipes de mois, le congé ou retard a été enregistrée.</p>

**Tableau II.4:** La fiche descriptive de cas «Gestion du personnel»

### II.6.2.5 Cas d'utilisation « Démarrer le serveur »

Pour le cas d'utilisation '*Démarrer le serveur*' la fiche descriptive est la suivante :

<p><b>Nom :</b> Démarrer le Serveur</p> <p><b>Objectif :</b> L'administrateur lance le serveur pour traiter les requêtes des clients.</p> <p><b>Acteurs principaux :</b> Administrateur</p> <p><b>Acteurs secondaires :</b> Néant</p>
<p><b>Les Préconditions :</b></p> <ul style="list-style-type: none"> <li>• authentication d'administrateur.</li> </ul> <p><b>Des scénarii :</b></p> <p><b>Scénario nominal :</b></p> <ol style="list-style-type: none"> <li>1- L'administrateur demande le démarrage de serveur.</li> <li>2- Le système lance le serveur et affiche un message de fin d'opération avec succès.</li> </ol> <p><b>Enchaînements d'erreur :</b></p> <p><b>1.0 :</b> le système ne répond pas à la demande de lancement (le port de serveur déjà réservé par un autre programme ou une erreur inconnu). L'enchaînement 1.0 démarre au point 1 du scénario nominal.</p> <ol style="list-style-type: none"> <li>1.1- le Système renvoi un message d'erreur.</li> </ol> <p><b>Les Postconditions :</b> Le serveur fonctionne correctement avec tous les services.</p>

**Tableau II.5:** La fiche descriptive de cas «Démarrer le serveur »

### II.6.2.6 Cas d'utilisation « Contrôler la qualité de produit »

Pour le cas d'utilisation ‘ *Contrôle de la qualité de produit* ’ la fiche descriptive est la suivante :

<p><b>Nom :</b> Contrôle la qualité de produit.</p> <p><b>Objectif :</b> Ce service permet aux membres du laboratoire d’assurer la qualité selon les normes de fabrication et partager les résultats d’analyse avec le département production (chef département, chef service de fabrication).</p> <p><b>Acteurs principaux :</b> Membre du laboratoire</p> <p><b>Acteurs secondaires :</b> Néant.</p>
<p><b>Les Préconditions :</b></p> <ul style="list-style-type: none"> <li>• authentifier Membre.</li> <li>• fait des analyses chimique.</li> </ul> <p><b>Les scénarii :</b></p> <p><b>Scénario nominal :</b></p> <ol style="list-style-type: none"> <li>1-Le membre demande l’ajout d’un nouveau document d’analyse.</li> <li>2-Le système affiche la zone pour choisissez un type de document.</li> <li>3 Le membre choisit le type de document.</li> <li>4-Le système affiche le formulaire avec les informations correspond le type choisie.</li> <li>5-Le membre remplit le document par les résultats d’analyse chimique.</li> <li>6-Le membre demande l’enregistrement du document créé.</li> <li>7-Le Système enregistre le document et affiche un message de succès.</li> <li>8-Le membre demande l’impression de document.</li> <li>9-Le Système imprime le document.</li> </ol> <p><b>Enchaînements alternatifs :</b></p> <p><b>6.0 :</b> manque des informations nécessaires pour le document. L’enchaînement 6.0 démarre au point 6 du scénario nominal.</p> <ol style="list-style-type: none"> <li>6.1-Le système indique au membre que les informations de l’analyse n’est pas complet.</li> <li>6.2-Le scénario nominal continue au point 5.</li> </ol> <p><b>Les Postconditions</b> Enregistrer le fichier d’analyse.</p>

**Tableau II.6:** La fiche descriptive de cas «Contrôler la qualité de produit»

**II.6.2.7 Cas d'utilisation « Suivre des réalisations »**

Pour le cas d'utilisation ‘ *Suivre des réalisations* ’ la fiche descriptive est la suivante :

<p><b>Nom :</b> Suivre des réalisations</p> <p><b>Objectif :</b> Ce service assuré ou fournit une manière efficace pour suivi la quantité de fabrication pour chaque produit par jours et par équipe.</p> <p><b>Acteurs principaux :</b> Chef département de production, Chef service de production.</p> <p><b>Acteurs secondaires :</b> Néant.</p>
<p><b>Les Préconditions :</b></p> <ul style="list-style-type: none"> <li>• authentifier les acteurs.</li> </ul> <p><b>Les scénarii :</b></p> <p><b>Scénario nominal :</b></p> <ol style="list-style-type: none"> <li>1-L’acteur accède a l’interface concerne le suivi des réalisations.</li> <li>2-Le système affiche la fenêtre requis.</li> <li>3-L’acteur demande l’ajout des informations sur un nouveau document.</li> <li>4-Le système fournit à l’acteur plusieurs outils pour facilite l’ajout des informations.</li> <li>5-L’acteur remplit le document.</li> <li>6-L’acteur demande l’enregistrement de document.</li> <li>7-Le Système enregistre le document et affiche un message de succès.</li> </ol> <p><b>Enchaînements alternatifs :</b></p> <p><b>5.0 :</b> L’acteur demande une modification des informations et des documents récents. L’enchaînement 5.0 démarre au point 5 du scénario nominal.</p> <ol style="list-style-type: none"> <li>6.1-Le système fournit à l’utilisateur un outil de recherche et modifier et permet aussi d’importer un document récent.</li> <li>6.2-Le scénario nominal continue au point 5.</li> </ol> <p><b>Les Postconditions :</b> Enregistrer le fichier correctement.</p>

**Tableau II.7 :** La fiche descriptive de cas «Suivre des réalisations»

**II.6.2.8 Cas d'utilisation « Commander matière première »**

Pour le cas d'utilisation ' **Commander MP** ' la fiche descriptive est la suivante :

<p><b>Nom :</b> Commander matière première</p> <p><b>Objectif :</b> Ce service permet au chef département et chef service l'écriture d'une demande (bon de commande) pour ... les matières première et envoyer ce demande au magasinier.</p> <p><b>Acteurs principaux :</b> Chef département de production, Chef service de fabrication.</p> <p><b>Acteurs secondaires :</b> Néant.</p>
<p><b>Les Préconditions :</b></p> <ul style="list-style-type: none"> <li>• authentifier les acteurs.</li> </ul> <p><b>Les scénarii :</b></p> <p><b>Scénario nominal :</b></p> <ol style="list-style-type: none"> <li>1-L'acteur clique sur l'espace commander matière première.</li> <li>2-Le système affiche l'espace concernée.</li> <li>3-L'acteur écrit une nouveau bon de commande.</li> <li>4-L'acteur envoie le bon de commande au magasinier.</li> <li>5- Le système enregistre une copie de bon de commande.</li> <li>6- Le système effectue l'envoi de bon sur le réseau.</li> <li>7-Le Système affiche un message de succès.</li> <li>8-L'acteur demande l'impression de document.</li> <li>9-Le Système imprime le document.</li> </ol> <p><b>Enchaînements alternatifs :</b></p> <p>L'enchaînement 4.0 démarre au point 4 du scénario nominal.</p> <ol style="list-style-type: none"> <li>5.1-Le système affiche un message d'erreur (manque des informations).</li> <li>5.2-Le scénario nominal continue au point 3.</li> </ol> <p><b>Enchaînements d'erreurs :</b></p> <p><b>6.</b> Le système ne peut pas envoyer le bon sur le réseau. L'enchaînement 6.0 démarre au point 6 du scénario nominal.</p> <ol style="list-style-type: none"> <li>7.1-Le système affiche un message d'échec.</li> </ol> <p><b>Les Postconditions :</b> Enregistrer le fichier correctement.</p>

**Tableau II.8:** La fiche descriptive de cas «Commander matière première»



### II.6.3 Les Diagrammes de séquence système

Nous complétons cette description textuelle par une représentation graphique UML très utile : le diagramme séquence système

Il faut savoir qu'un scénario décrit une extension particulière d'un cas d'utilisation du début à la fin. Il correspond à un enchainement du cas d'utilisation se terminant par une fin normale ou erreur.

#### II.6.3.1 Diagramme de séquence système de « Authentifier »

Ce diagramme de séquence système pour le cas d'authentifier

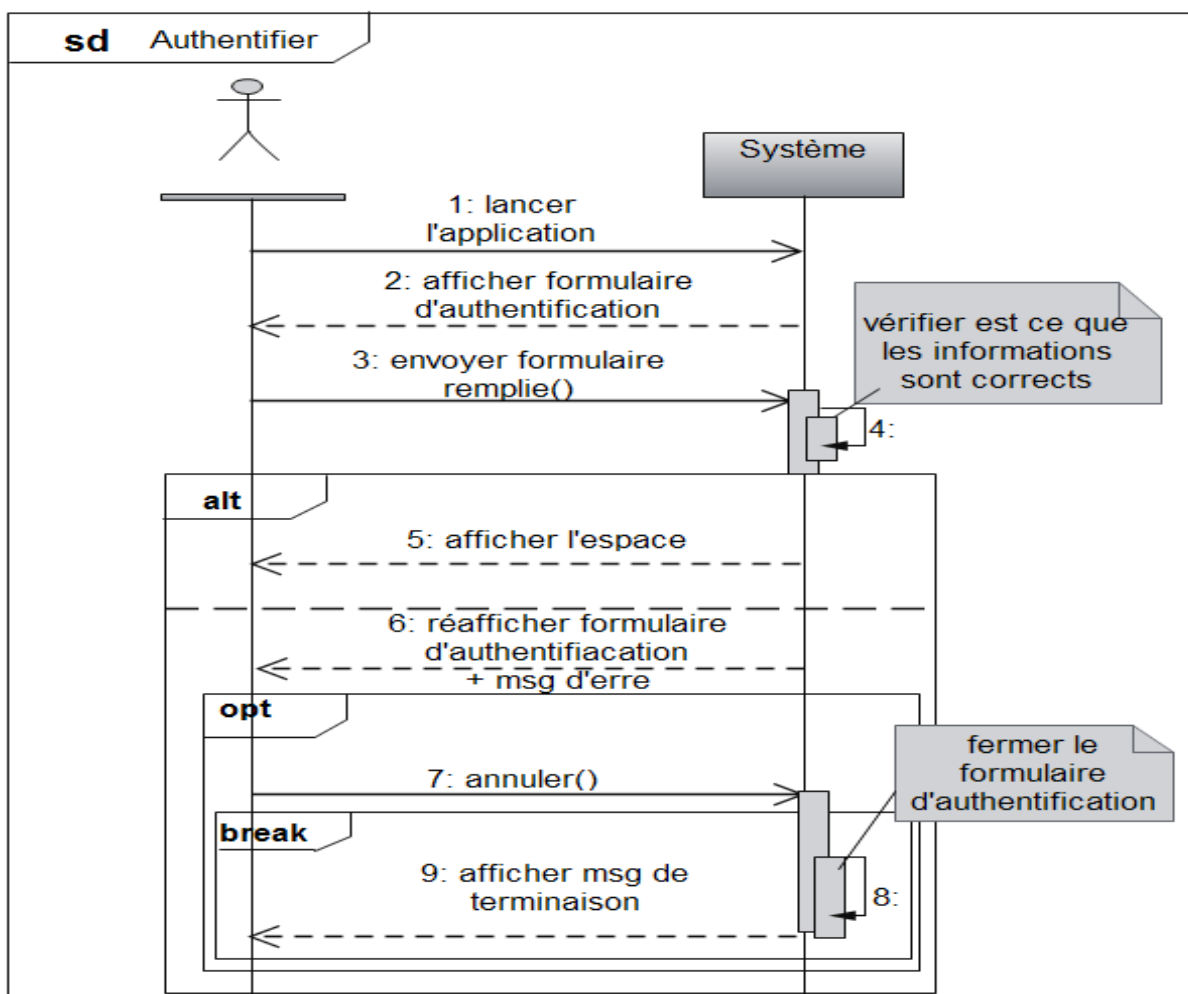
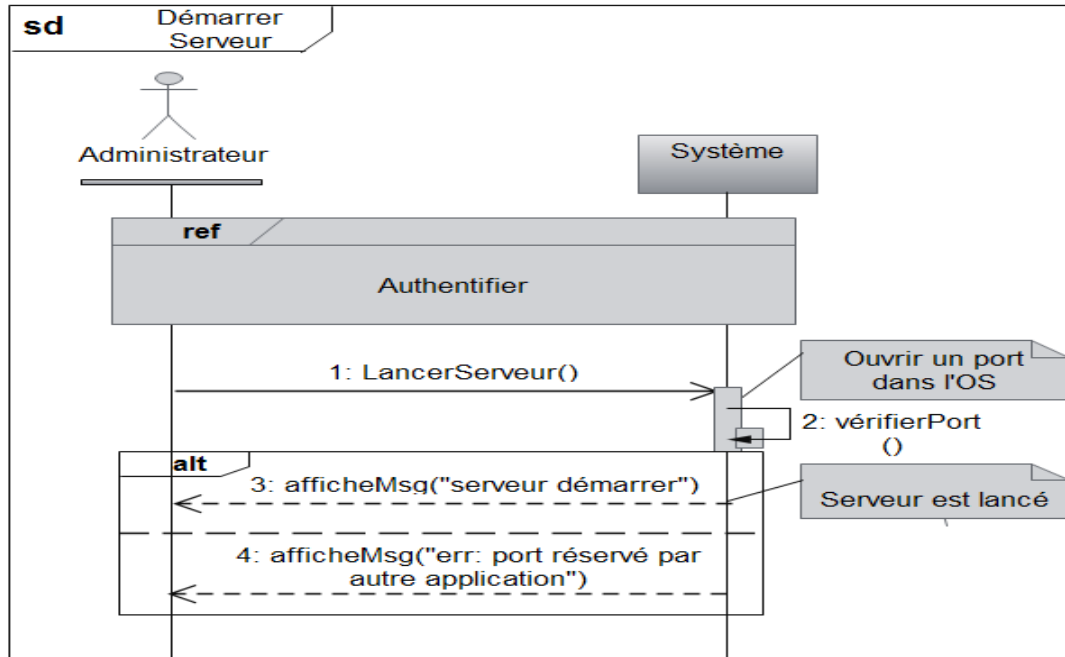


Figure II.7 : DSS d'authentifier

**II.6.3.2 Diagramme de séquence système de « Démarrer le serveur »**

Ce diagramme de séquence système pour le cas démarrer le serveur :



**Figure II.8** : DSS démarré le serveur

II.6.3.3 Diagramme de séquence système de « Suivre consommation MP »

Ce diagramme de séquence système pour le cas suivre consommation MP :

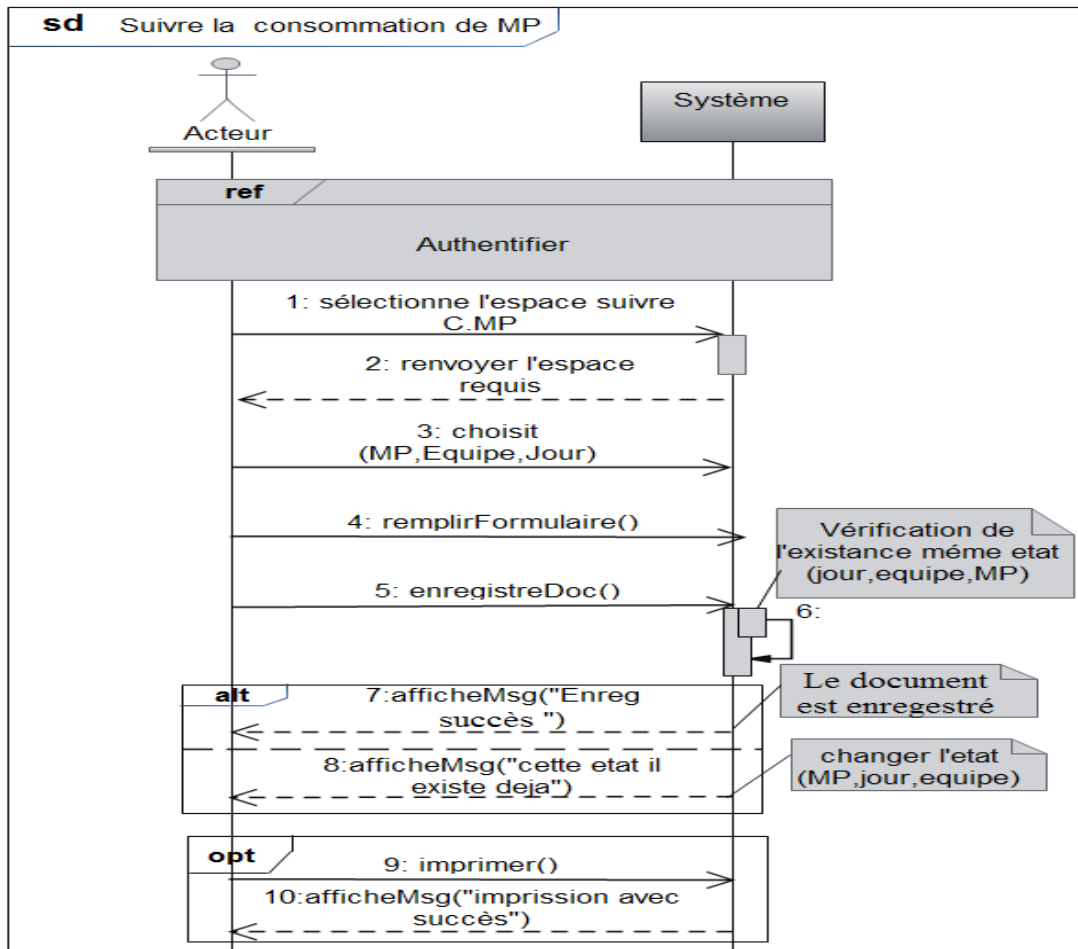
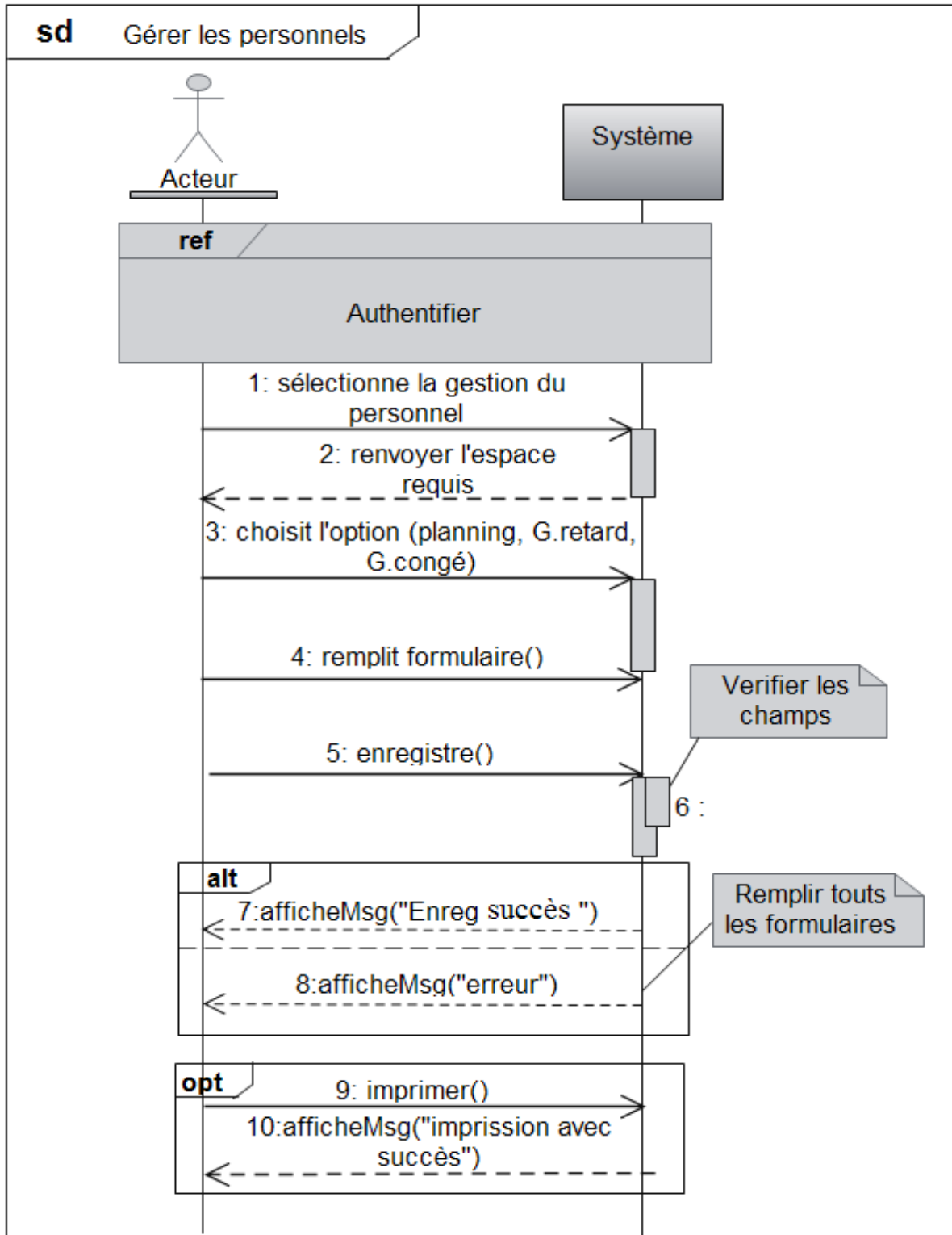


Figure II.9 : DSS « Suivre consommation MP »

II.6.3.4 Diagramme de séquence système de « Gérer le personnel »

Ce diagramme de séquence système pour le cas gérer les personnels :



FigureII.10 : DSS gérer le personnel

**II.6.3.5 Diagramme de séquence système de « Suivre des réalisations »**

Ce diagramme de séquence système pour le cas suivre des réalisations

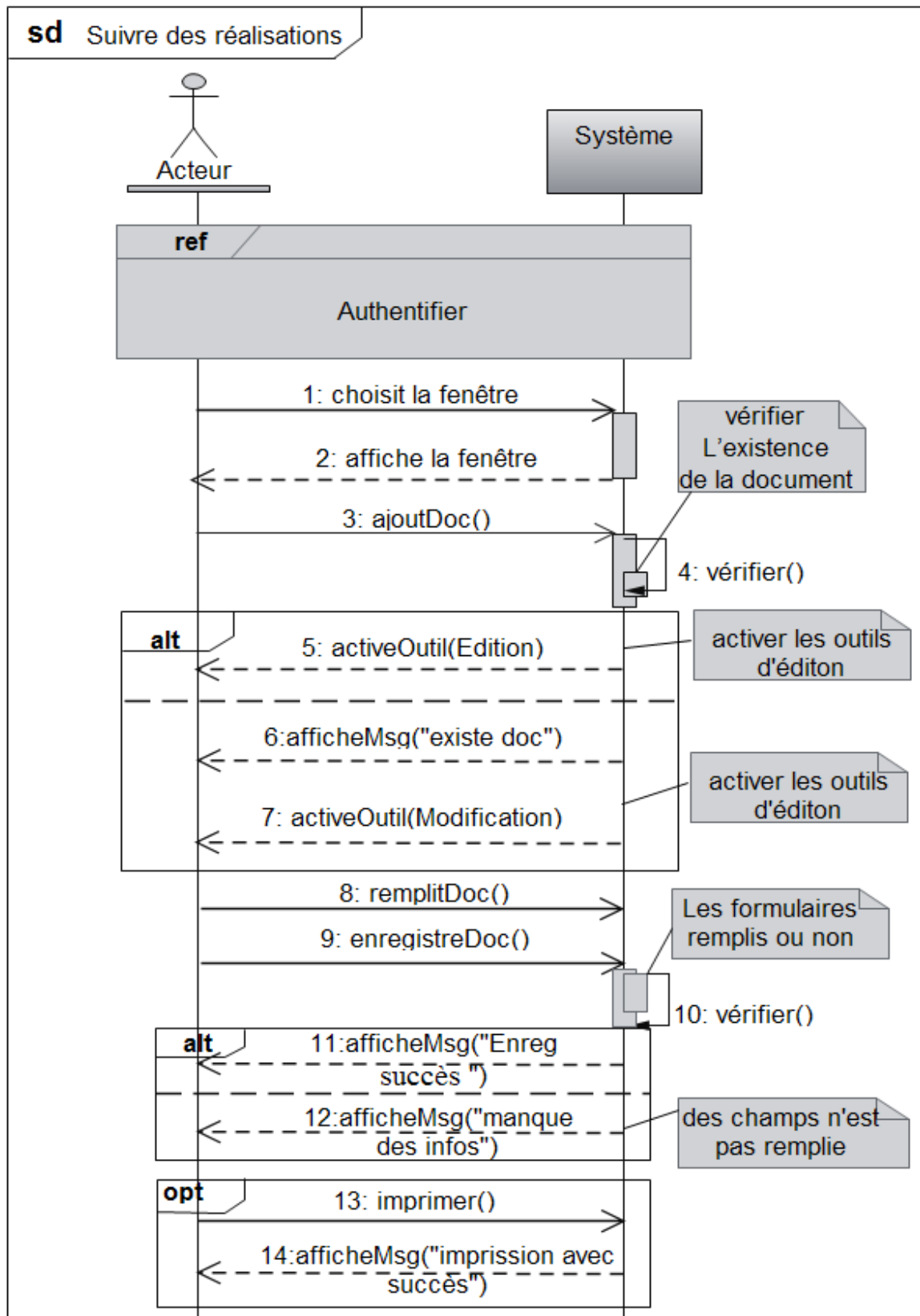
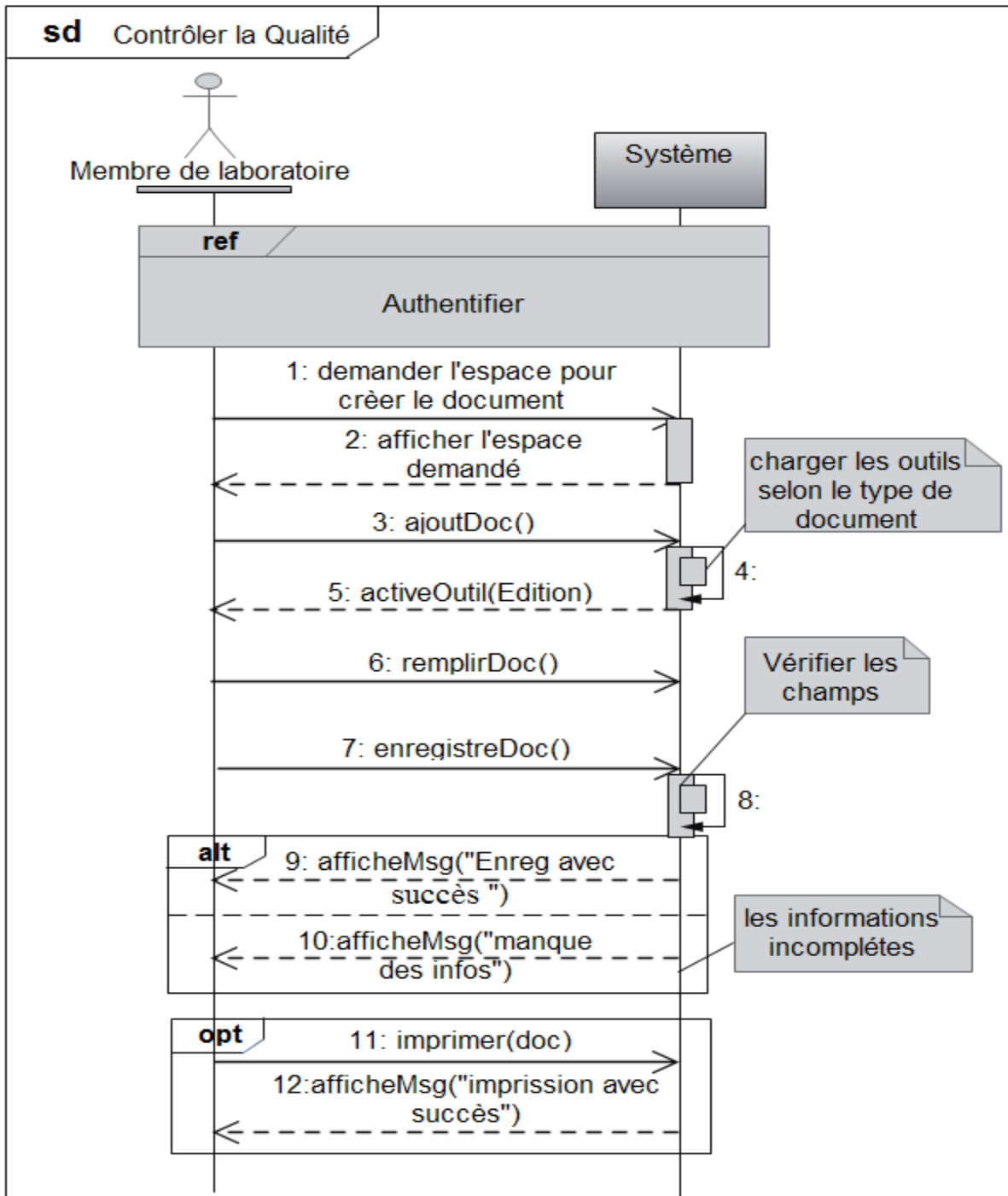


Figure II.11 DSS Suivre des réalisations

**II.6.3.6 Diagramme de séquence système de « Contrôler la qualité de produit »**

Ce diagramme de séquence système pour le cas contrôler la quantité de produit :



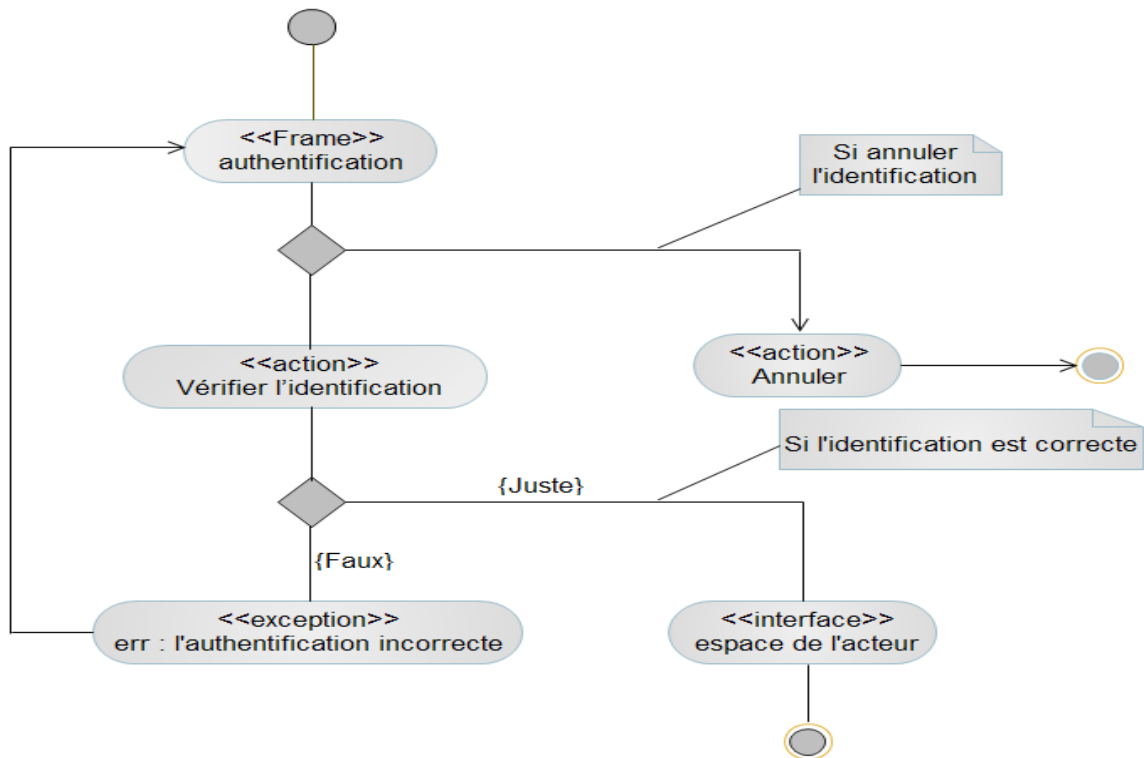
**Figure II.12:** DSS de contrôler la qualité de produit

**II.6.4 Les Diagrammes d'activité**

Dans cette section et pour chaque cas d'utilisation, nous allons présenter un diagramme d'activité qui permet de représenter précisément la navigation ou le déroulement d'un cas d'utilisation.

**II.6.4.1 Diagramme d'activité « Authentifier »**

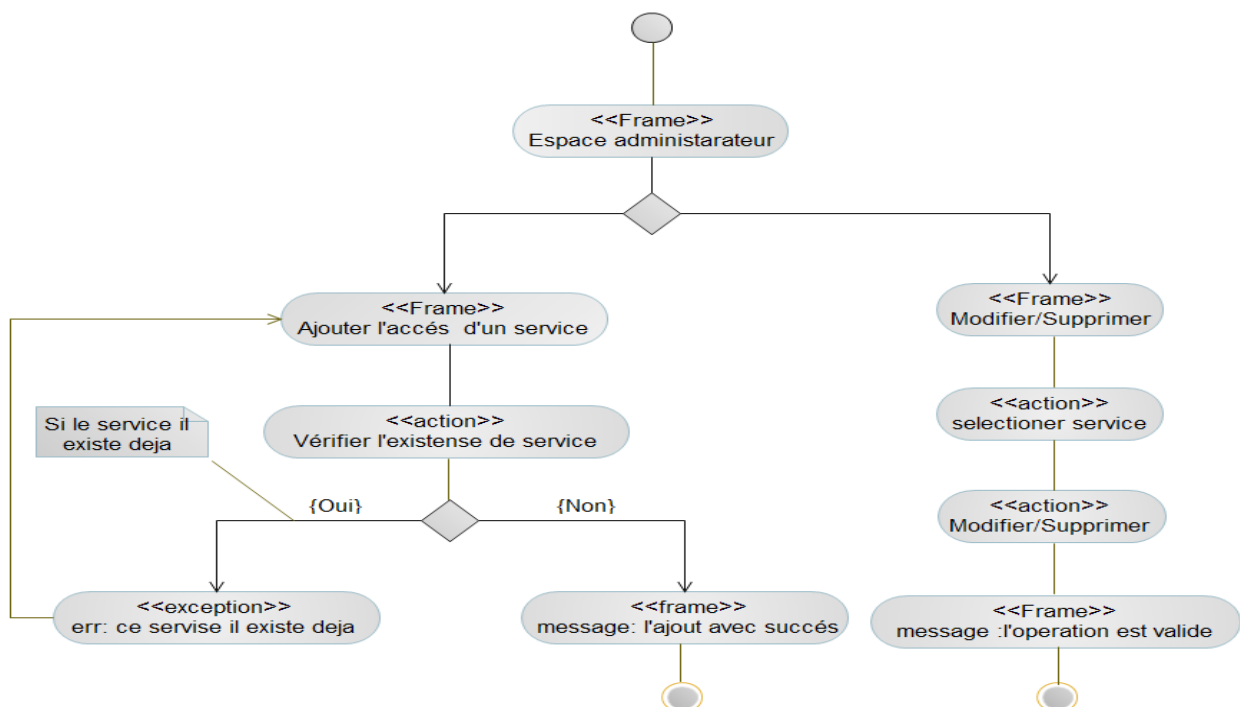
Le diagramme d'activité de cas 'Authentifier' est le suivant :



**Figure II.13** :Diagramme d'activité d'authentifier

**II.6.4.2 Diagramme d'activité « Gérer les droit d'accès »**

Le diagramme d'activité de cas 'Gérer les droit d'accès' est le suivant :



**Figure II.14** Diagramme d'activité gérer les droit d'accès

II.6.4.3 Diagramme d'activité « Commander matière première »

Le diagramme d'activité de cas 'Commander MP' est le suivant :

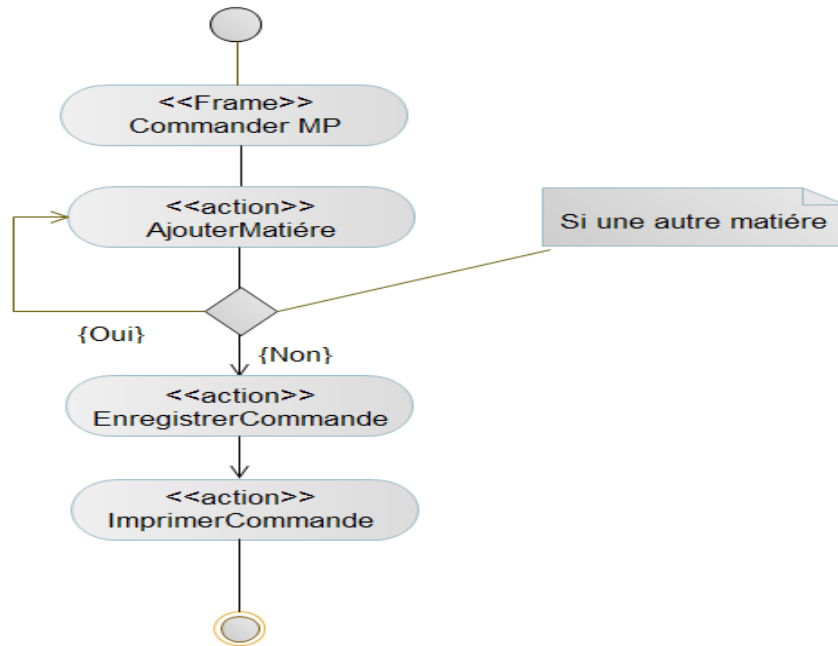


Figure II.15 Diagramme d'activité commandé matière première

II.6.4.4 Diagramme d'activité « demander et rejeter des produits »

Le diagramme d'activité de cas 'demander et rejeter des produits' est le suivant :

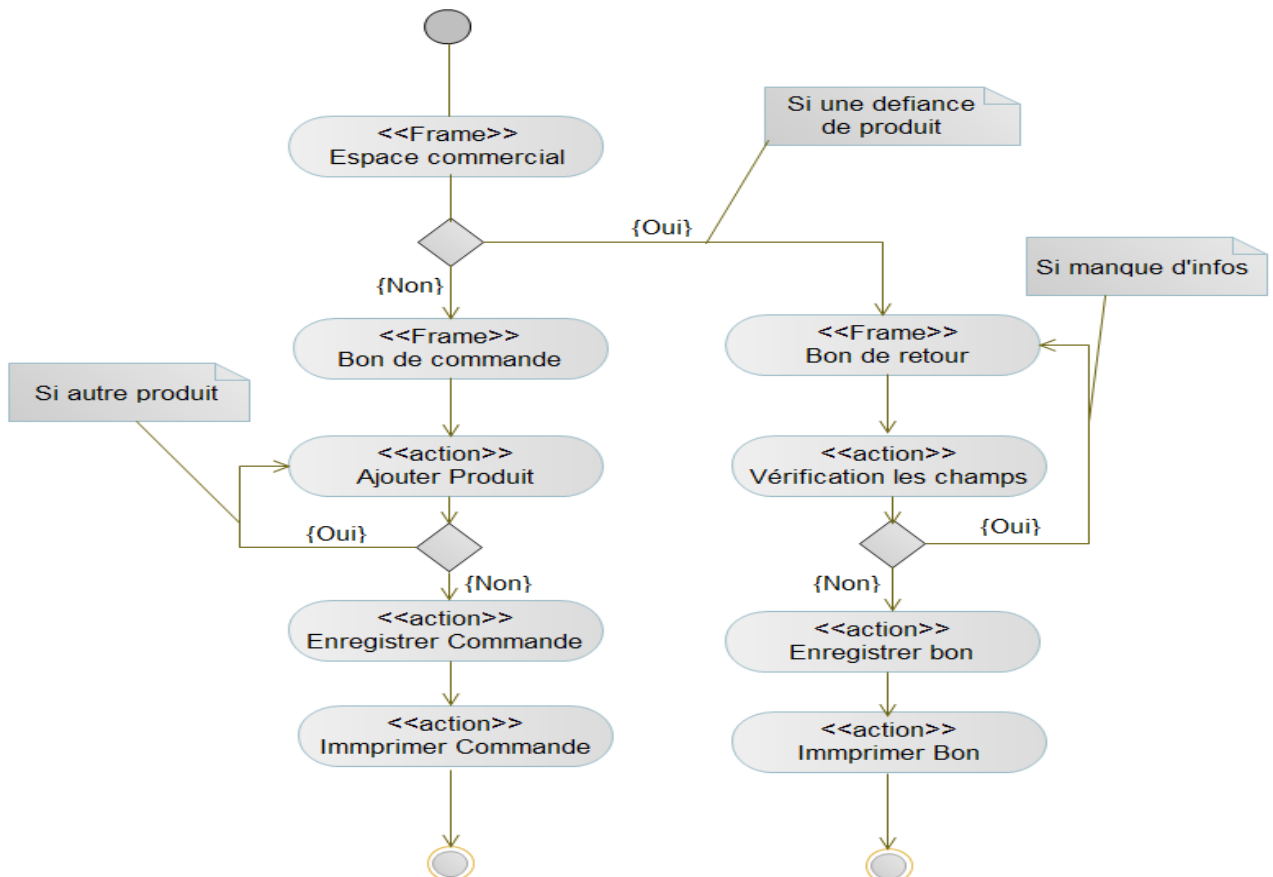


Figure II.16 : Diagramme d'activité demander et rejeter des produits



II.6.5 Diagramme de classe

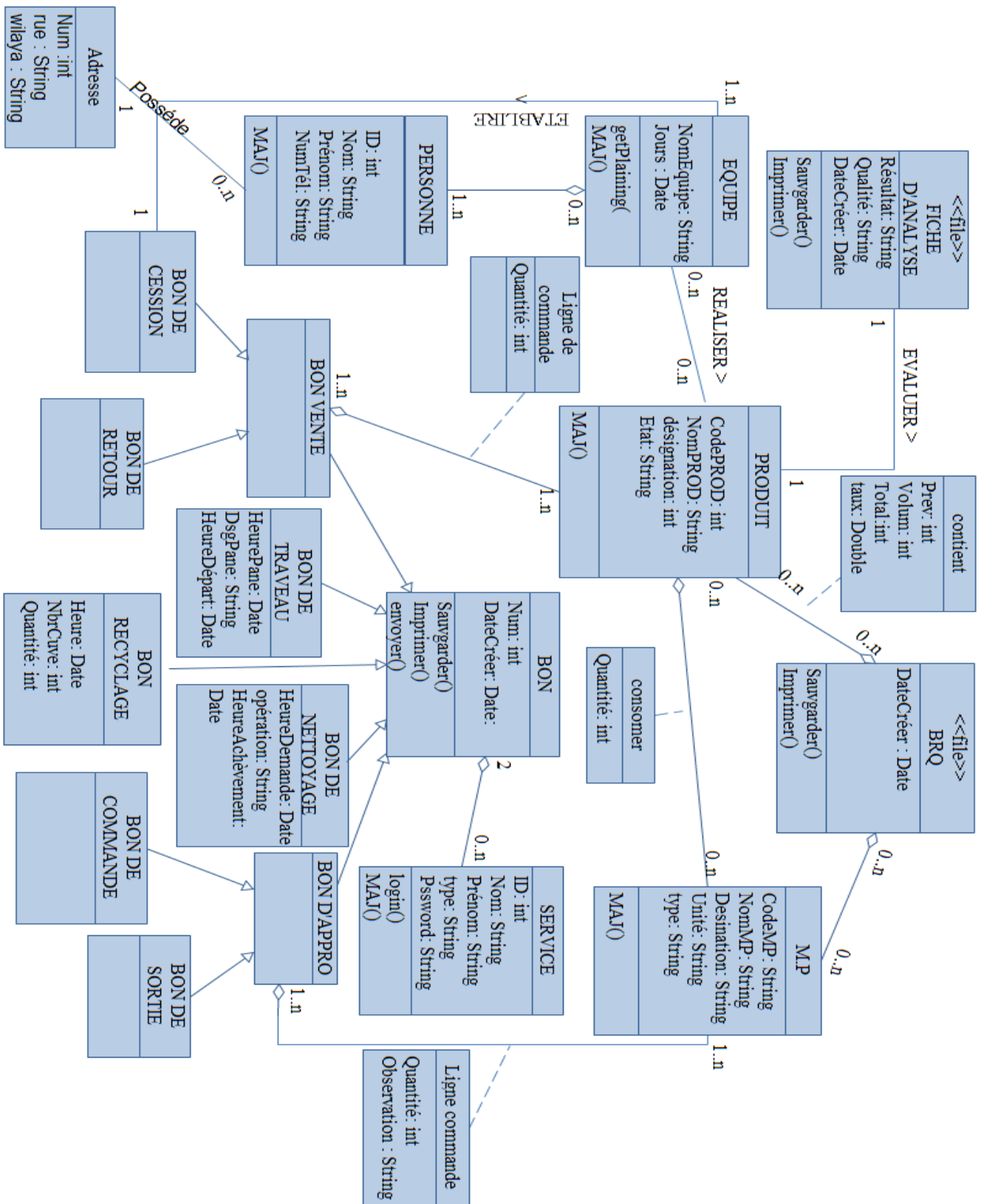
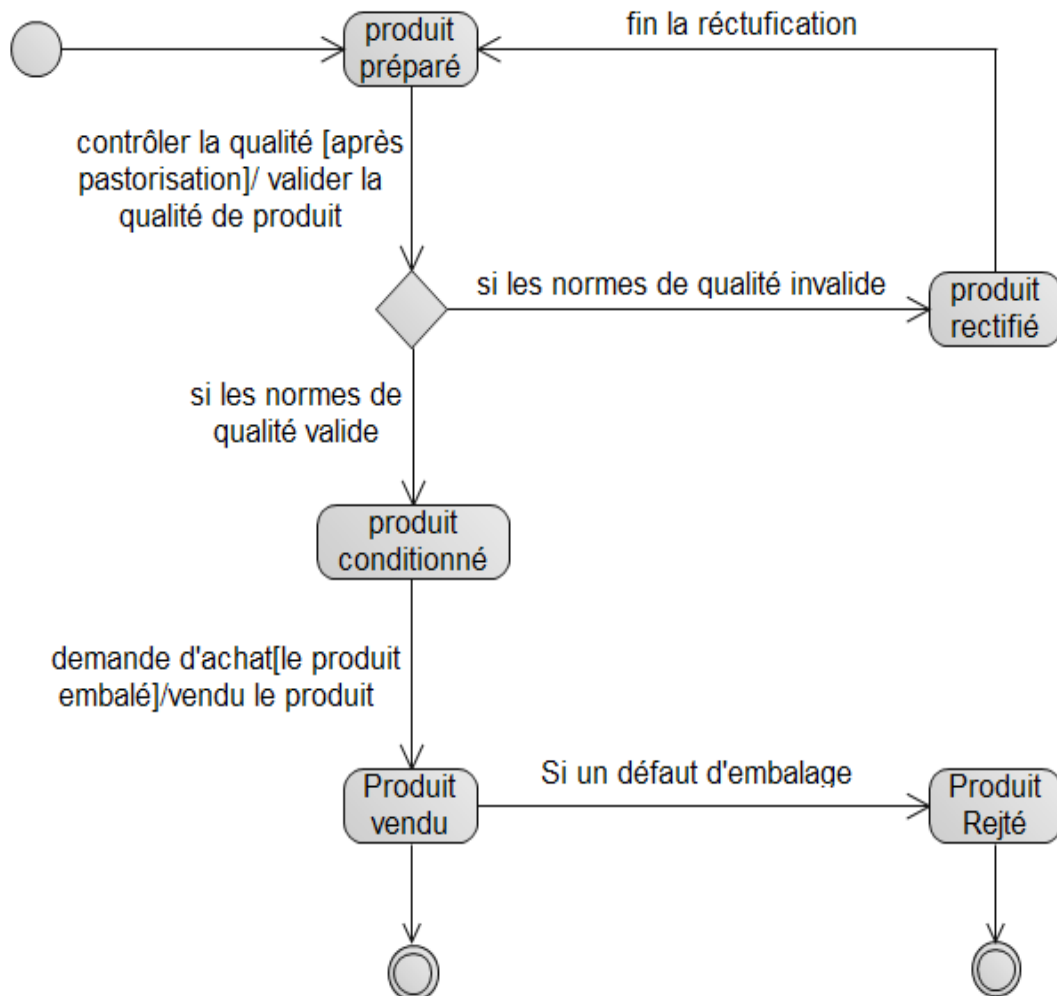


Figure II.17: Diagramme de classe

### II.6.6 Diagramme état transition

Pour décrire une classe par rapport à ses états possibles et aux transitions qui la font évoluer on a choisi un seul exemple de diagramme d'état transition sur classe produit.



**Figure II.18** Diagramme d'état transition de la classe produit

II.6.7 Diagramme de déploiement

La disposition et la configuration physique des différents matériels sur lesquels le logiciel sera déployé, cette configuration est schématisée par le diagramme de déploiement suivant :

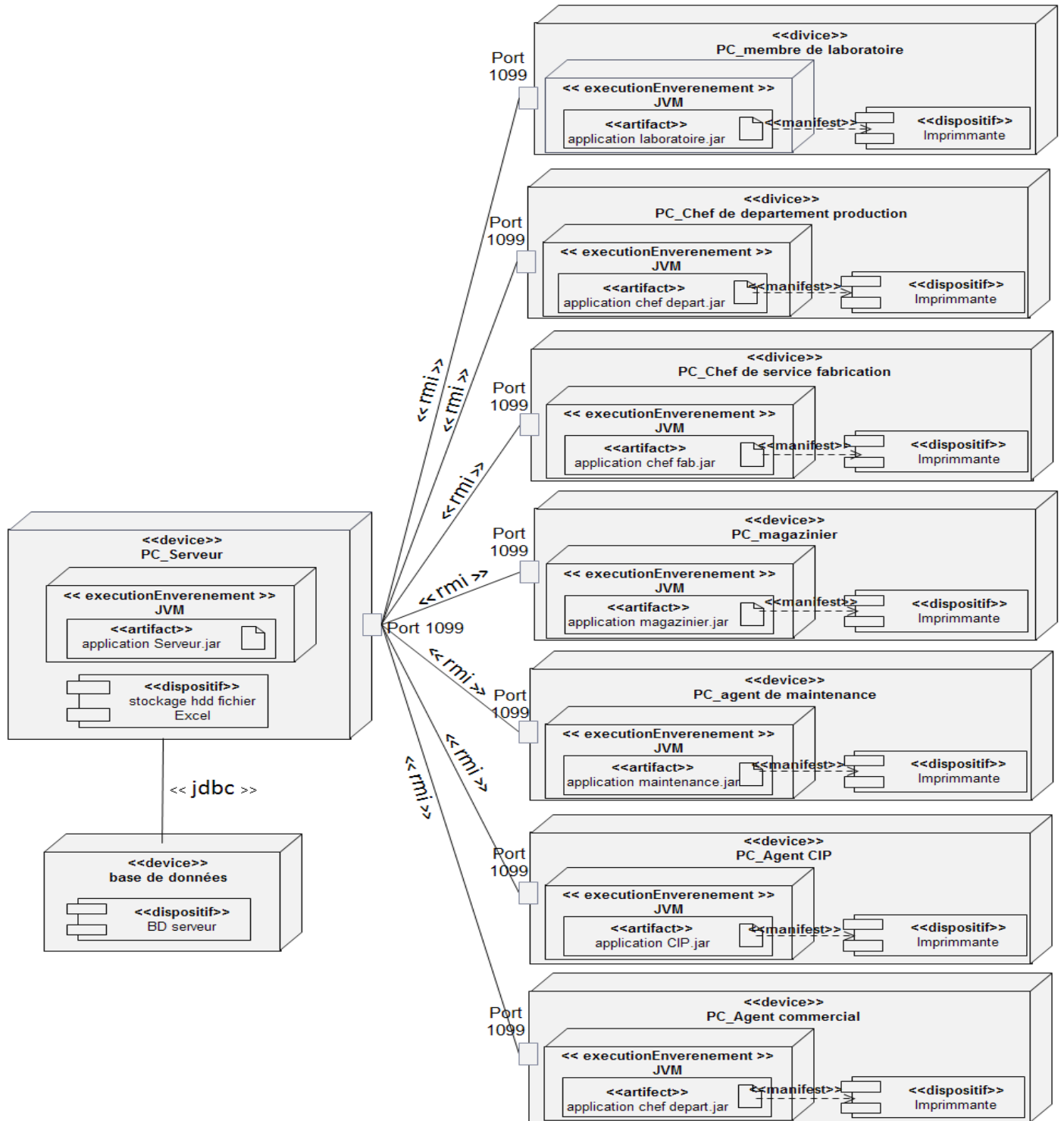


Figure II.19 Diagramme de déploiement

## II.7 L'architecture Client/serveur

Chaque jour, des millions de transactions opérationnelles démarrent sur des programmes client tournant sur des PC pour s'exécuter ensuite sur des systèmes serveur distribués. Dans ce qui suit, nous allons présenter l'architecture d'un système client/serveur, ces types de serveur, les avantages et inconvénient, et le fonctionnement de cette architecture, les type d'architecture client-serveur existe. L'architecture client/serveur désigne un mode de communication entre plusieurs ordinateurs d'un réseau qui distingue un ou plusieurs postes clients du serveur : chaque logiciel client peut envoyer des requêtes à un serveur. Un serveur peut être spécialisé en serveur d'applications, de fichiers, de terminaux, ou encore de messagerie électronique.

### II.7.1 Présentation d'architecture client serveur

#### II.7.1.1 Qu'est-ce qu'un serveur ?

On appelle logiciel serveur un programme qui offre un service sur le réseau. Le serveur accepte des requêtes, la traite et renvoie le résultat au demandeur. Le terme serveur s'applique à la machine sur laquelle s'exécute le logiciel serveur.

Le serveur lui envoie ensuite les informations grâce à l'adresse IP de la machine cliente.

#### II.7.1.2 Qu'est-ce qu'un client ?

On appelle logiciel client un programme (client FTP, client mail, ..) qui utilise le service offert par un serveur. Le client envoie une requête et reçoit la réponse. Le client peut être raccordé par une liaison temporaire.

Le client pour recevoir des informations du serveur lui émet une requête passant par un port du PC (exemple : port 25 pour les mails, port 80 pour le HTTP et 21 pour le FTP).

#### II.7.1.3 Qu'est-ce qu'une Requête ?

Message transmis par un client à un serveur décrivant l'opération à exécuter pour le compte du client.

#### II.7.1.4 Qu'est-ce qu'une Réponse ?

Message transmis par un serveur à un client suite à l'exécution d'une opération, contenant le résultat de l'opération

### II.7.2 Caractéristiques d'Architecture Client/ Serveur

#### II.7.2.1 Service

Le serveur est fournisseur de services. Le client est consommateur de services.

#### II.7.2.2 Protocole

C'est toujours le client qui déclenche la demande de service. Le serveur attend passivement les requêtes des clients.

### II.7.2.3 Partage des ressources

Un serveur traite plusieurs clients en même temps et contrôle leurs accès aux ressources.

### II.7.2.4 Localisation

Le logiciel client/serveur masque aux clients la localisation du serveur.

### II.7.2.5 Hétérogénéité

Le logiciel client/serveur est indépendant des plates-formes matérielles et logicielles.

### II.7.2.6 Redimensionnement

Il est possible d'ajouter et de retirer des stations clientes. Il est possible de faire évoluer les serveurs.

### II.7.2.7 Souplesse et Adaptabilité

On peut modifier le module serveur sans toucher au module client. La réciproque est vraie. Si une station est remplacée par un modèle plus récent, on modifie le module client (en améliorant l'interface, par exemple) sans modifier le module serveur.

## II.7.3 Les Avantages et les inconvénients de cette architecture

### II.7.3.1 Les avantages

- **Unicité de l'information** : pour un site web dynamique par exemple (comme vulgarisation-informatique.com), certains articles du site sont stockés dans une base de données sur le serveur. De cette manière, les informations restent identiques. Chaque utilisateur accède aux mêmes informations.
- **Meilleure sécurité** : Lors de la connexion un PC client ne voit que le serveur, et non les autres PC clients. De même, les serveurs sont en général très sécurisés contre les attaques de pirates.
- **Meilleure fiabilité** : En cas de panne, seul le serveur fait l'objet d'une réparation, et non le PC client.
- **Facilité d'évolution** : Une architecture client/serveur est évolutive car il est très facile de rajouter ou d'enlever des clients, et même des serveurs.

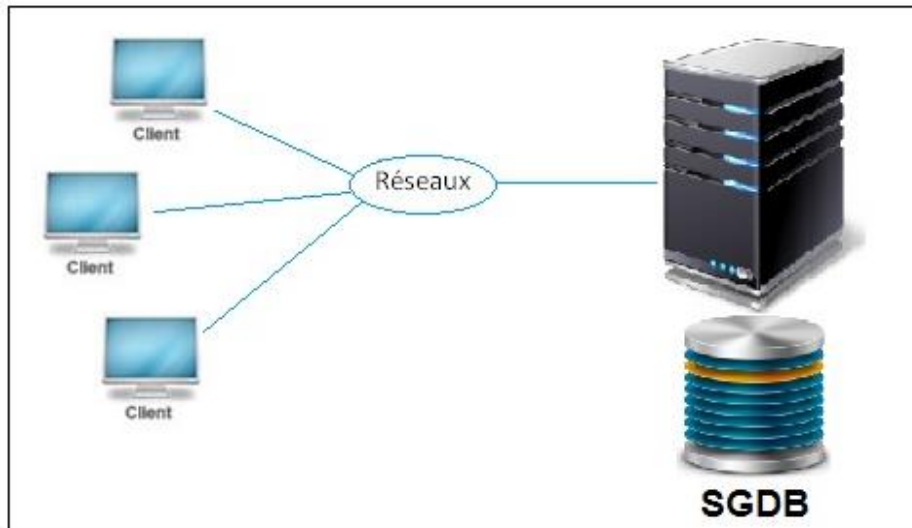
### II.7.3.2 Les inconvénients

- Un coût d'exploitation élevé (bande passante, câbles, ordinateurs surpuissants)

## II.7.4 les types des Architectures Client/serveur

### II.7.4.1 Architecture à 2 niveaux

L'architecture à deux niveaux (aussi appelée 2tier en anglais) caractérise le système Client/serveur dans lequel le client demande une ressource au serveur qui la lui fournit directement (le serveur ne fait appel à aucune application pour fournir ce service).



**Figure II.20:** Architecture à 2 niveaux

### II.7.4.2 Architecture à 3 niveaux

Dans l'architecture à 3 niveaux il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre :

1. **Le Client** : le demandeur de ressources
2. **Le Serveur d'Application** (appelé aussi **middleware**) : le serveur chargé de fournir la ressource mais faisant appel à un autre serveur
3. **Le Serveur Secondaire** : (généralement un serveur de base de données), fournissant un service au premier serveur. [23]

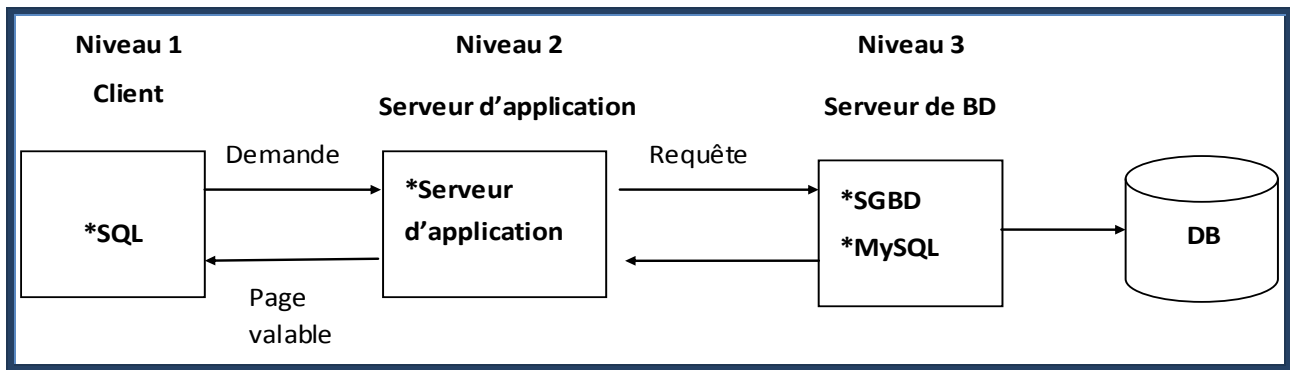


Figure II-21 Architecture Client/serveur à 3 niveaux

### II.7.4.3 Comparaison entre les deux types

	Deux niveaux	Trois niveaux
<b>Administration du système</b>	Complexe (plus de logique à gérer coté client)	Moins complexe
<b>Sécurité</b>	Faible (sécurité au niveau des données)	Importante
<b>Encapsulation des données</b>	Faible (les tables de données sont accessibles)	Importante (le client invoque des services ou des méthodes)
<b>Performance</b>	Faible	Bonne
<b>Facilité de développement</b>	Importante	Envoi d'amélioration (des outils standard peuvent être utilisés pour créer les clients et des outils capables de développer les cotés serveur et client de l'application sont apparus)
<b>Souplesse de l'architecture matérielle</b>	Limitée	Excellente (les trois niveaux peuvent résider sur des ordinateurs différents)
<b>Disponibilité</b>	Faible	Excellente (relance possible d'intermédiaire sur d'autres serveurs)
<b>Variété des choix de communication</b>	Nom (uniquement appels synchrones orientés connexion)	OUI (support d'appels RPC mais aussi échange de message sans connexion files Publication/abonnement et diffusion)

Tableau II-9 : Comparaison entre l'architecture 2 niveaux et 3 niveaux

### II.7.5 Bases de données en client-serveur

Dans une architecture client-serveur, un applicatif est constitué de trois parties : l'interface utilisateur, la logique des traitements et la gestion des données. Le client n'exécute que

l'interface utilisateur et la logique des traitements, laissant au serveur de bases de données la gestion complète des manipulations de données.

- Le serveur de bases de données fournit des services aux processus clients. Les tâches qu'il doit prendre en compte sont : la gestion d'une mémoire cache, l'exécution de requêtes exprimées en SQL, exécuter des requêtes mémorisées, la gestion des transactions, la sécurité des données.
- Le client doit ouvrir une connexion pour pouvoir profiter des services du serveur. Il peut ouvrir plusieurs connexions simultanées sur plusieurs serveurs. Le client peut soumettre plusieurs requêtes simultanément au serveur et traiter les résultats de façon asynchrone.

Dans le cas de serveur de base de donnée, le client émet des requête SQL sous forme de message en direct du serveur, le résultat de chaque requête SQL set envoyé sur le réseau, le code qui traite la requête ainsi que les donnée résident sur la même machine. Le serveur utiliser sa propre capacité de traitement pour rechercher les donnée demandées au lieu de transmettre tous les articles au client et de laisser ce dernier en faire la sélection, comme c'était le cas de serveur de fichier. La puissance ainsi répartie est utilisée de façon beaucoup plus efficace.

Dans cette approche, le code du serveur est produit commerciale de l'éditeur de logiciel. Mais c'est à l'utilisateur de créer les tables SQL et les garnir de données. Le code de l'application réside chez le client. Il est nécessaire d'écrire du code pour l'application cliente, soit acheter un outil d'interrogation tout écrit. Les serveurs de bases de données forment la base de systèmes d'aide à la décision qui réclament des interrogations variées et une grande souplesse dans la mise en forme des rapports. Ils jouent également un rôle important dans le traitement des entrepôts de données. [24]



## II.8 Conclusion

Les outils de modélisation que n'a présenter dans ce chapitre son des outils plus utilisable et le plus puissant pour la conception des applications informatique, l'architecture client /serveur une étape importante dans l'évolution des systèmes informatiques. Le client/serveur est sorti ses frontières départementales et nous déployons maintenant des applications client/serveur desservant des milliers de clients dans l'entreprise. Ces applications tournent souvent sur de nombreux serveurs et consistent en des certaines des composants

A l'issue de cette étape nous a permis de bien représenter le système à réaliser ainsi qu'elle a facilité la tâche de la programmation permettant de représenter la manière dont notre système est réalisé. Pour la dernier étape qui celle l'implémentation du l'application, on va utiliser des outils technique pour la programmation.

### III.1 Introduction

Après la phase de conception du logiciel, on va passer à l'étape finale dans ce projet, cette étape consiste à représenter brièvement la structure, le langage de programmation et les outils utilisés pour la réalisation de notre application.

### III.2 Environnement de réalisation de notre application

#### III.2.1 Choix du langage de programmation

Java est un langage de programmation orienté objet et un environnement d'exécution, développé par Sun Microsystems. Il fut présenté officiellement en 1995. Le Java était à la base un langage pour Internet, pour pouvoir rendre plus dynamiques les pages (tout comme le JavaScript aujourd'hui).

Java est pourvu d'une grande sécurité, la richesse de ses bibliothèques, son adaptation à plusieurs plateformes, la qualité présentée par ses composantes graphiques (Swing), et sa facilité de déploiement en réseau (RMI), en font de lui un langage redoutable puissant et performant.

Avec ses multiples avantages, nous permettent de choisir Java comme langage d'implémentation de notre application, L'environnement minimal pour développer l'application se compose d'un **JDK** et d'un éditeur de texte open source **NetBeans**. Il est important d'avoir l'outil NetBeans pour vous permettre d'accélérer nos développements (IDE ou Integrated Development Environment).

##### III.2.1.1 Le JDK

Indispensable pour le développement et l'exécution de notre application

Le Java Development Kit, communément appelé "**JDK**", est le kit de développement proposé gratuitement par Sun. Il comprend plusieurs outils, Parmi lesquels:

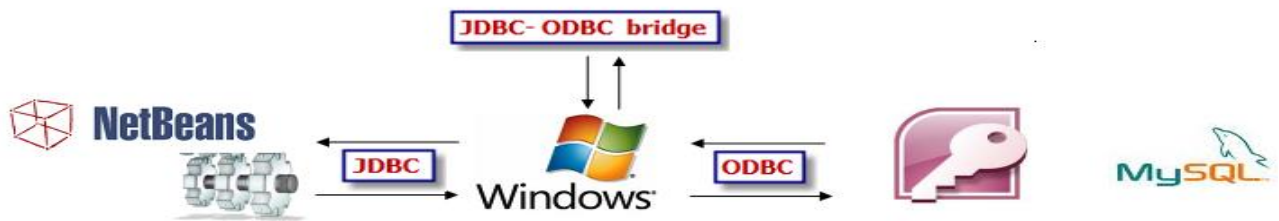
- **javac** : le compilateur Java;
- **java** : un interpréteur d'applications (machine virtuelle);
- **javadoc** : un générateur de documentation;
- **jar** : un outil de compression de classes Java.

Le JDK nous permettra de compiler et d'exécuter l'application ainsi que d'autres outils.

##### III.2.1.2 NetBeans

Le développement d'application et son fonctionnement en tant que application Java nous a amené à choisir un outil qu'est le « *NetBeans* ».

*NetBeans (IDE java)* est un outil très puissant pour la réalisation d'application java, elle offre un gain de temps en matière de productivité, reste à se familiariser avec ses outils rapidement.



**Figure III-1:** Communication entre NetBeans et les bases de données

### III.2.1.3 JDBC

JDBC (Java Data Base Connectivity) est un API, et c'est la couche logicielle standard, il Offerte aux développeurs pour accéder à des bases de données relationnelles.

Elle se charge de trois étapes indispensables à l'accès des données:

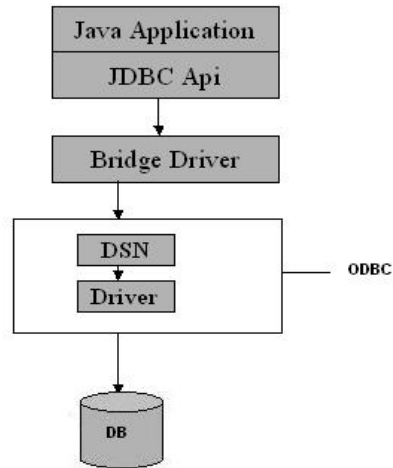
- ❖ la création d'une connexion à la base;
- ❖ l'envoi d'instructions SQL;
- ❖ l'exploitation des résultats provenant de la base.

Cette API fait partie intégrante de la plate-forme Java depuis la version 1.1 du JDK. Elle est représentée par le paquetage `java.sql`.

Qu'encore largement utilisée, elle a tendance à disparaître au profit d'outils de mapping objet-relationnel. En effet, développer une couche d'accès aux données avec JDBC de manière non transparente.

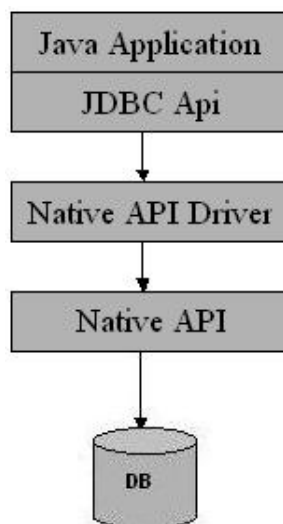
Il y a 4 catégories de pilote JDBC :

- **Pilotes de type 1:** Pilotes accédant à une base de données par l'intermédiaire d'une autre technologie (on parle de *passerelle*). Les passerelles JDBC-ODBC, permettant une connexion via un pilote ODBC en sont l'exemple le plus courant. Le pilote convertit les appels de données Java en appel ODBC valide, et les exécute ensuite à l'aide du pilote ODBC



**Figure III.2:** Pilotes JDBC de type 1

- **Pilotes de type 2:** Pilotes d'API natifs. Il s'agit d'un mélange de pilotes natifs et de pilotes Java. Les appels JDBC sont convertis en appels natifs pour le serveur de bases de données (Oracle, Sybase, ou autres).



**Figure III.3:** Pilotes JDBC de type 2

- **Pilotes de type 3:** Pilotes convertissant les appels JDBC en un protocole indépendant du SGBD. Un serveur convertit ensuite ceux-ci dans le protocole SGBD requis (modèle à 3 couches)

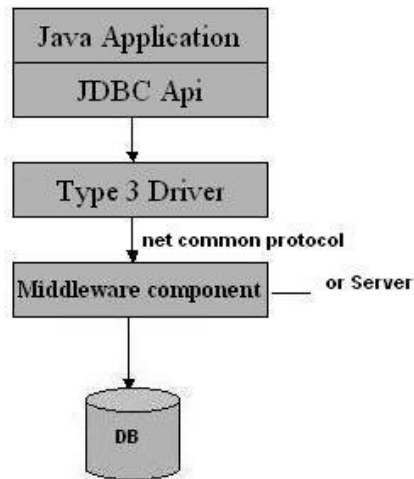


Figure III.4: Pilotes JDBC de type 3

- Pilotes de type 4:** Pilotes convertissant les appels JDBC directement en un protocole réseau exploité par le SGBD. Ces pilotes encapsulent directement l'interface cliente du SGBD et sont fournis par les éditeurs de base de données. Cette solution est à préconiser dans le cadre d'un intranet.

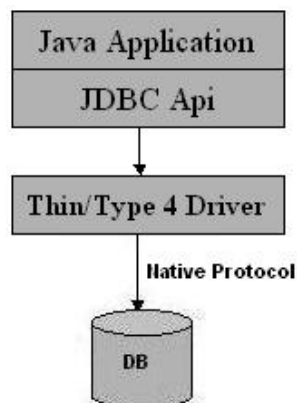
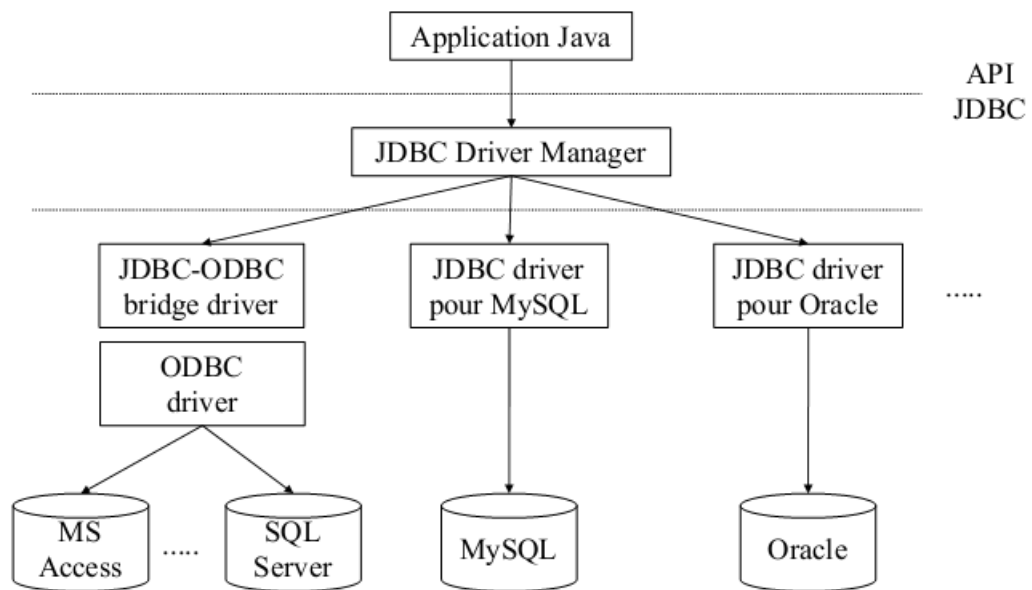


Figure III.5: Pilotes JDBC de type 4



**Figure III.6** Architecture JDBC

Dans notre application nous utilisons le pilote 2 (JDBC natif) qui est compatible avec notre choix de SGBD. [23]

### III.2.1.4 Apache POI

POI est l'acronyme de **Poor Obfuscation Implementation**. C'est un projet open source du groupe Apache, sous licence Apache V2, dont le but est de permettre la manipulation de fichiers de la suite bureautique Office de Microsoft, dans des applications Java mais sans utiliser Office, L'implémentation de POI est intégralement réalisée en pur Java.

La manipulation ne peut se faire que sur des documents reposant sur le format Microsoft OLE2 (Object Linking and Embedding) Compound Document ce qui inclus les documents de la suite Office mais aussi les applications qui utilisent les ensembles de propriétés MFC pour sérialiser leurs documents, La version 3.8 a été diffusée 26 March 2012, apporter le support des formats Office Open XML proposés depuis la version 2010 d'Office,

Le site officiel du projet est à l'url <http://poi.apache.org/>, La version utilisée dans notre application est la 3.8, Le téléchargement de l'archive contenant la version binaire de POI se fait à l'url :

(<http://www.apache.org/dyn/closer.cgi/poi/release/bin/poi-bin-3.8-20120326.zip>) Il faut ensuite décompresser l'archive poi-bin-3.1-FINAL-3.8-20120326.zip obtenue dans un répertoire du système, et pour utiliser POI il suffit d'ajouter le fichier poi-3.8-FINAL-20120326.jar au Classpath de l'application.



#### III.2.1.4.1 Les composantes de POI

Ce projet contient plusieurs composants :

- POIFS (Poor Obfuscation Implementation File System) : manipulation de fichiers utilisant le format Microsoft OLE 2 Compound Document
- HSSF (Horrible SpreadSheet Format) : manipulation des fichiers Excel (XLS) en lecture et écriture.
- HWPF (Horrible Word Processor Format) : manipulation de fichiers Word en lecture et certaines fonctionnalités en écriture.
- HPSF (Horrible Slide Layout Format) : manipulation de fichiers PowerPoint en lecture et écriture pour certaines fonctionnalités mais pas toutes
- HDGF : lecture et uniquement extraction de texte de fichiers Visio
- HPSF : API pour manipuler les propriétés d'un fichier au format OLE 2 en lecture et en écriture

#### III.2.1.4.2 L'API POI-HSSF

HSSF permet la manipulation de document Excel de la version 97 à la version 2010, pour les deux formats : le format OLE2 (fichier avec l'extension .xls) et le format OOXML d'Excel 2007&2010 est supporté (fichier avec l'extension .xlsx).

HSSF est une solution riche en fonctionnalités et fiable pour la manipulation de documents Excel en Java.

Un document Excel est composé de plusieurs éléments : un Dossier (Workbook) qui contient un ou plusieurs Feuilles (Worksheets) étant elle-même constituées de Lignes (Rows) comportant des cellules (Cells).

HSSF propose deux API pour manipuler un document Excel :

- **user API** : API la plus riche qui permet la lecture et l'écriture mais qui consomme beaucoup de ressources car le document est intégralement représenté dans un graphe

D'objets (le pendant pour le traitement de documents XML pourrait être DOM). Les classes de cette API sont regroupées dans le package `org.apache.poi.hssf.usermodel`

- **event API** : API pour la lecture uniquement qui consomme peu de ressources (le pendant pour le traitement de documents XML pourrait être SAX). Les classes de cette API sont regroupées dans le package `org.apache.poi.hssf.eventmodel` et `org.apache.poi.hssf.eventusermodel`.

### III.2.1.4.3 La liste des packages de HSSF

Package	Rôle
org.apache.poi.hssf.eventm	Classes pour gérer les événements émis lors de la lecture
org.apache.poi.hssf.eventus	Classes pour lire un document
org.apache.poi.hssf.extracto	Classes pour extraire le texte d'un document
org.apache.poi.hssf.record.f	Classes pour le support des formules dans les cellules
org.apache.poi.hssf.usermo	Classes pour la manipulation de documents
org.apache.poi.hssf.util	Utilitaires pour faciliter la mise en oeuvre de certaines

**Tableau III.1** Les packages de l'API 'HSSF'

### III.2.1.4.4 Les fonctionnalités de l'API « HSSF »

Parmi les nombreuses fonctionnalités proposées par cette API, il y a :

- ❖ lecture et écriture de document
- ❖ création et modification des différentes entités qui composent un document (document, feuille, ligne, cellule, ...)
- ❖ support de fonctionnalités avancées sur la feuille : sélection, zoom, support des panneaux, ...
- ❖ support des types de données d'une cellule (numérique et date, chaîne de caractère, formule)
- ❖ formatage des cellules (alignement, police, couleur, bordures, formats de données proposés en standard ou personnalisés,
- ❖ fonctionnalités avancées sur les cellules : taille, taille optimale, fusion, commentaires,
- ❖ paramètre d'impression d'une page (sélection de la zone d'impression, faire tenir sur une page, bas de page, ...)
- ❖ support graphique : dessin de primitives, d'images, ... [26]



### III.2.1.4.5 RMI

RMI est une API de communication entre objets distants de java présente depuis longtemps dans la version standard de java du JRE, contrairement à CORBA, RMI ne permet que la communication entre programme java à l'exclusion de tout autre langage par contre, il assure des échanges entre systèmes d'exploitation différents possédant une machine virtuelle java. Il peut être utilisé également pour la communication entre deux machines virtuelle java sur la même machine physique.

RMI est souvent vu comme une version légère de CORBA, car il présente la même modèle de programmation, mais le fait d'utiliser uniquement Java simplifier le processus de développement et de plus RMI ne possède pas les couches de service complexe qui ont été spécifiées avec CORBA : découverte des interfaces persistances, etc. RMI a toutefois rajouté dans la version récente des mécanismes d'activation de serveurs et de persistance. Il possède enfin ses propres mécanismes de sécurité.

Les appels de méthodes distantes en RMI nécessitent toutefois une réécriture de certaines interfaces, et il est donc nécessaire d'avoir la maîtrise de code pour déployer une application de manière répartie en RMI. [12]

## III.2.2 Choix du SGBD

### III.2.2.1 SGBD



MySQL est un système de gestion de base de données (SGBD). Selon le type d'application, sa licence est libre ou propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle et Microsoft SQL Server.

MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multi-thread et multi-utilisateur.

### III.2.2.2 SQL

Le langage SQL (structured query language) c'est-à-dire Langage d'interrogation structuré peut être considéré comme un langage complet d'accès normalisé et de gestion aux bases de données relationnelles, il est aujourd'hui supporté par la plupart des produits commerciaux que ce soit par les système de gestion de bases de données micro tel que MYSQL ou par les produits plus professionnels tels qu'Oracle ou Sysbase. Il a fait l'objet de plusieurs normes ANSI/ISO dont la plus rependue aujourd'hui est la norme SQL2 qui a été définie en 1992. C'est à la fois :

Un langage d'interrogation de la base, un langage de manipulation des données, un langage de définition des données, un langage de contrôle de l'accès aux données.

### III.2.3 Environnement matériel

#### III.2.3.1 Caractéristiques techniques du serveur

L'ordinateur doit se présenter sous la configuration suivante :

Unité	Caractéristique
Processeur	Intel® Pentium® DUAL CPU
Mémoire RAM	2048 Mo (2 Go)
Disque dur	160 Go
Ecran	15 ''

**Tableau III.2** Les caractéristiques techniques de serveur

#### III.2.3.2 Caractéristiques techniques du Client

L'ordinateur doit se présenter sous la configuration suivante :

Unité	Caractéristique
Processeur	Intel® Pentium® 4
Mémoire RAM	512 Mo
Disque dur	80 Go
Ecran	15 ''

**Tableau III.3** les caractéristiques techniques de client

## III.3 Mapping objet-relationnel

### III.3.1 PowerAMC

Pour vous permettre de faire le Mapping il existe plusieurs API et frameworks, Chaque outil est évalué selon la qualité dont il dispose à implémenter différents critères de UML 2 (associations binaires, n-aires, classes-associations, agrégations, contraintes inter-associations, héritage multiple avec contrainte et rétroconception d'une base de données).

**PowerAMC** est considéré parmi les meilleures logiciel pour transférer le passage de l'orienté objet au relationnel. Celui-ci (anciennement AMC\*Designor) est la version française de l'outil de modélisation PowerDesigner de Sybase. Concernant les bases de données, l'outil prend en charge trois types de modèles qu'on peut transformer entre eux (MCD, MPD, MOO). [28]

Le teste de ce logiciel est d'une grand fiabilité prouver par des études effectuer manuellement et qui on donner le même résultat.

### III.3.2 Les règles de passage de l'orienté objet au relationnel

L'utilisation d'un SGBDR impose un enchainement de représentation entre la structure des classes et la structure des données relationnelles, les deux structures ayant des analogies, les équivalences exprimées au tableau sont utilisées pour en réaliser le rapprochement.

Une classe définie une structure de données à laquelle souscrivent des instances, elle correspond donc à une table du model relationnel.

Chaque attribut donne lieu à une colonne, chaque instance stocke les données dans une ligne (T-Uplet) et son ID sert de clé primaire.

Certain attributs de type complexe ne correspondent à aucun des type de SQL ; on rencontre fréquemment ce cas pour les attributs représentant une structure de données. Un type complexe peut être conçu :

- Soit avec plusieurs colonnes, chacune correspondent à un champ de la structure.
- Soit avec table spécifique d'une clé étrangère pour relier les instances aux valeurs de leurs attributs complexe.

L'ensemble de règles utilisée pour le passage de l'orienté objet vers le relationnel est décrit ci-Dessous :

Model d'objet	Model relationnel
Classe	Table
Attribut de type simple	Colonne
Attribut de type composé	Colonne ou clé étrangère
Instance	T-uplet
ID	Clé primaire
Association	Clé étrangère ou table de lien
Héritage	Clé primaire identique sur plusieurs tables

**Tableau III.4** les règles de passage de l'orienté objet au relationnel

### III.3.3 Equivalence entre les concepts objets et relationnels

#### La réalisation d'association

A Max B	Max	1	>1
1		Si la clé de A = la clé de B ne rien faire Sinon choisir une des autres solutions	Ajouter la clé B dans la relation de A comme attribut
>1		Ajouter la clé de A dans la relation de B comme attribut	Créer une relation R ayant comme attribut la clé de A et la clé de B

**Tableau III.5** la réalisation d'association

**Remarque :**

La relation de généralisation / spécification peut avoir deux solution :  
 La première solution : proposé ci-dessus --> le tout dans la mémé table avec la gestion des nul (fragmentation interne).  
 La deuxième solution : chaque classe est une relation avec la mémé clé.

### III.3.4 Le modèle relationnel

**MP** (CodeMP, NumMP, Désignation, Unité, Type).

**Consommé** (NumProd, CodeMP, Qté).

**Produit** (NumProd, nomProd, Désignation, etat).

**Réalise** (NumProd, nomequipe).

**Equipe** (nomequipe, jour).

**Equipe\_contient\_pers** (nomequipe, id).

**Personne** (id, nom, prénom, tel, adresse).

**Service** (typeSERV, nom, prénom, passward).

**Bon\_contient\_serv** (num, typeSERV).

**Ligne\_commande** (num, codeMP, Quntité, observation).

**LigneCommandeVente** (num, numProd, Quantité).

**Bon\_vente** (num, datecréer, type).

**Bondenetoyage** (num, datecréer, HeureDemande, opération, HeureAchévenemnt).

**BonTraveau** (num, datecréer, HeurePane, DisgPane, HeureDepar).

**Bon\_recyclage** (num, datecéer, Heure, nbreCuve, Quantité).

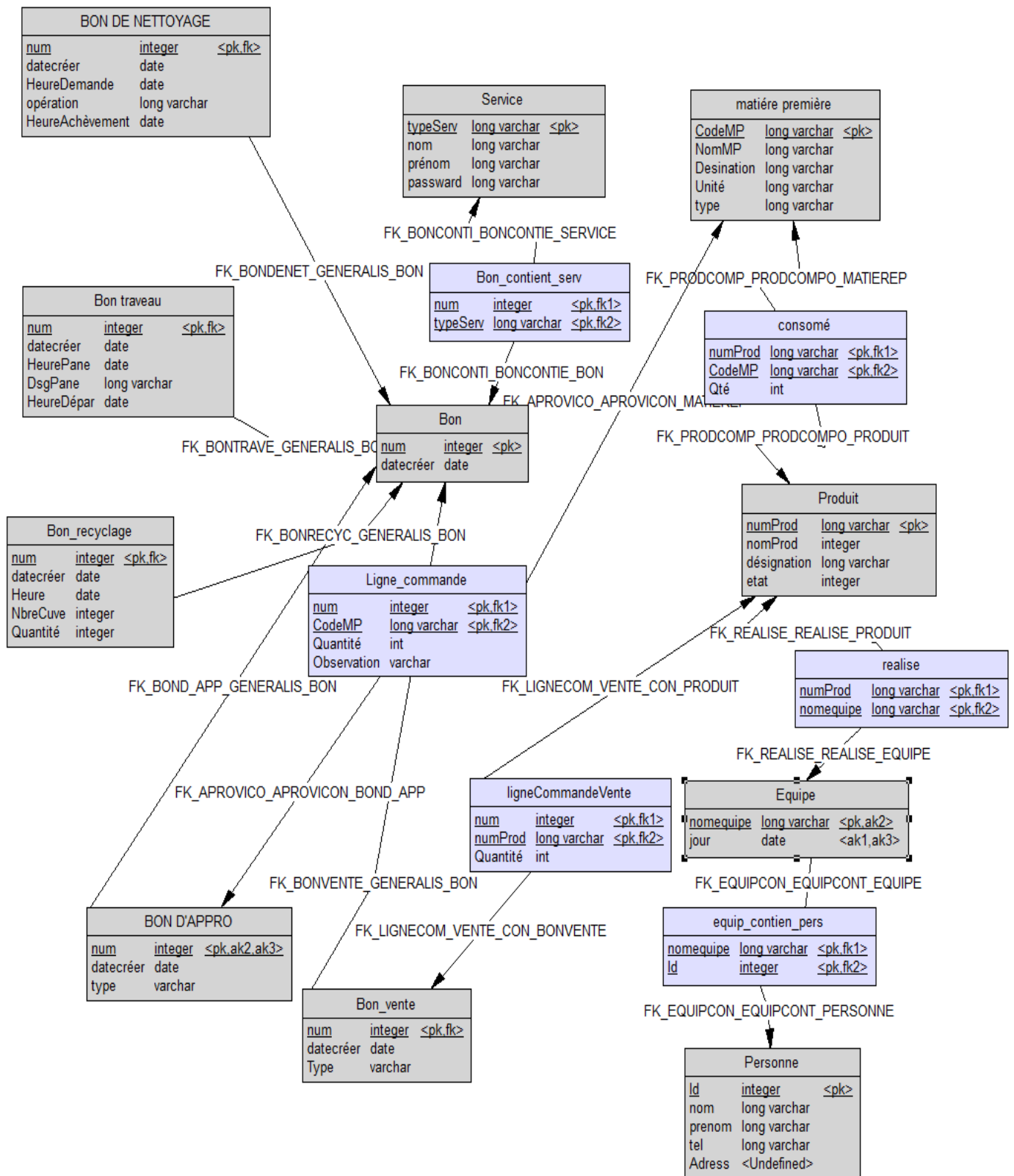


Figure III.7 Le modèle relationnel

**Remarque :**

Pour L'héritage on a utilisé le fragment interne

**PK** : Clé primaire.

**FK** : Clé étrangère.

### III.4 Architecture de réseaux de notre système distribué

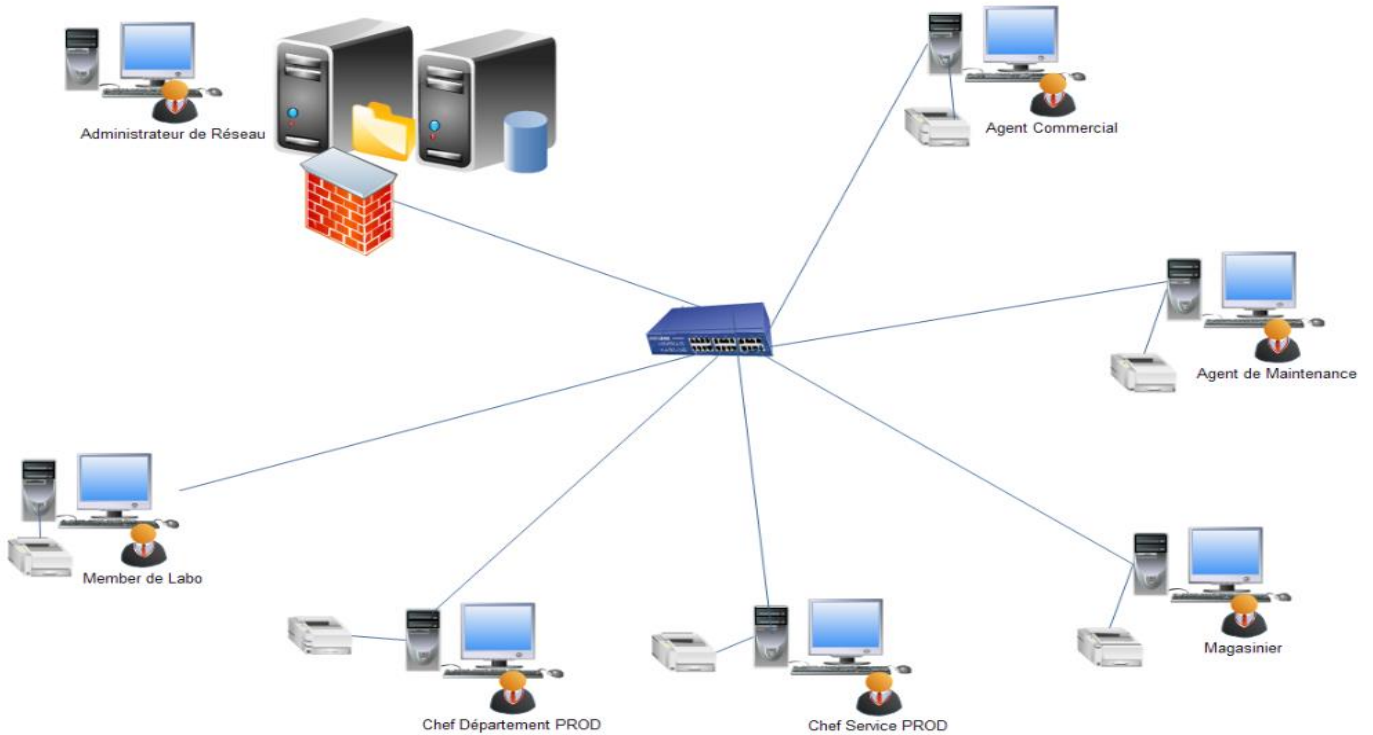


Figure III.8 Architecture de réseaux de notre système

### III.5 Structure de logiciel

Afin d’assurer une bonne présentation du logiciel on a créé pour chaque acteur un espace personnelle, et plus pour augmenter la sécurité des données, chaque conventionné possède une authentification pour accéder le système Et voici des exemples sur les interfaces principales de notre application.

#### III.5.1 La page d’authentification de l’administrateur



Figure III.9 Fenêtre d’authentification

### III.5.2 La fenêtre de l'espace administrateur

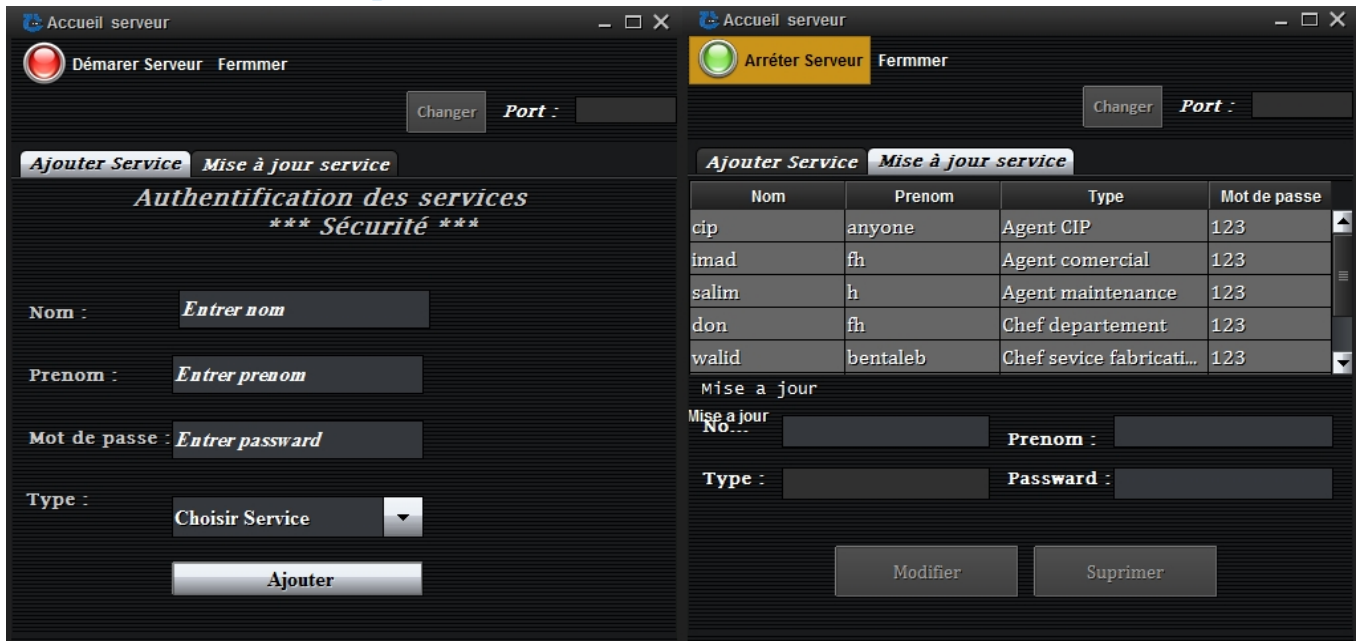


Figure III.10 Fenêtre de l'espace administrateur

Cette fenêtre présente les Fonctionnalité suivantes : démarrer serveur, changer le port RMI par défaut (1099) en cas réservé, et aussi les mises à jour des différents services.

### III.5.3 La fenêtre d'authentification des services



Figure III.11 Fenêtre d'authentification service

### III.5.4 La fenêtre des mises à jour de matière première (Espace magasinier)

Permet d'effectuer l'ajout, modification, suppression, et la recherche par code ou par désignation d'une matière première.



Figure III.12 Onglet de MAJ matière

### III.5.5 La fenêtre de création un bon de livraison (Espace magasinier)

Cette fenêtre consiste à créer un bon de livraison de matière première vers l'atelier de fabrication, il existe 2 type de bon (transfert et sortie).

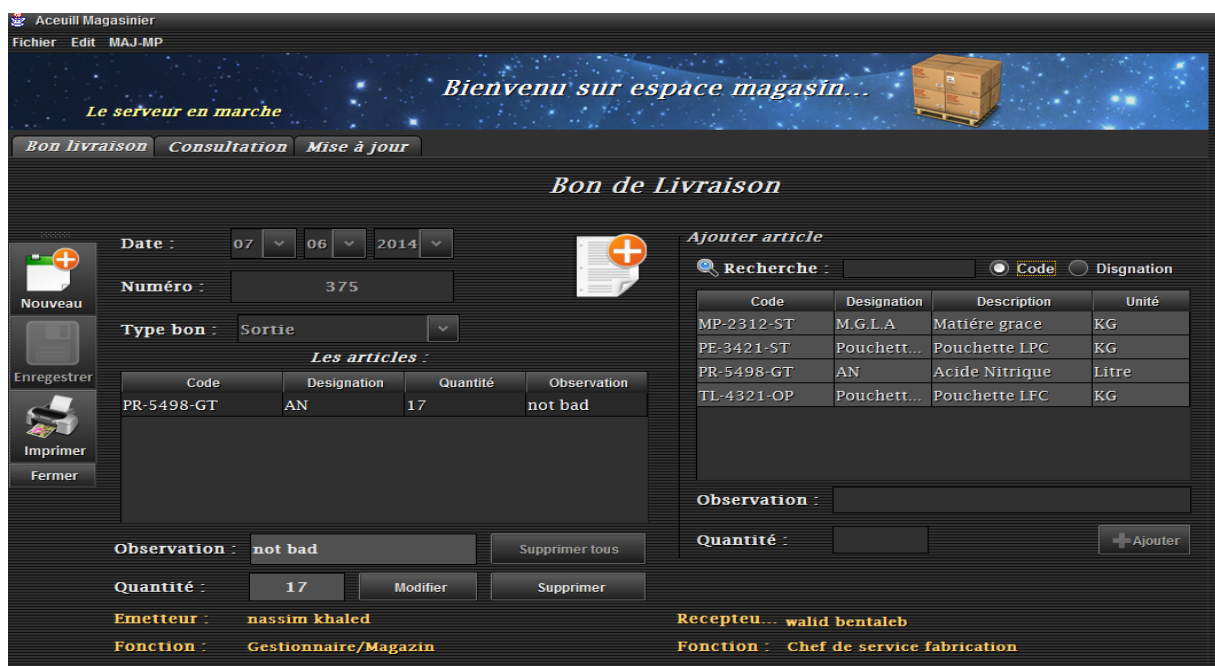


Figure III.13 Onglet de création un bon de livraison



### III.5.6 La fenêtre de suivi la consommation de matière première par jour

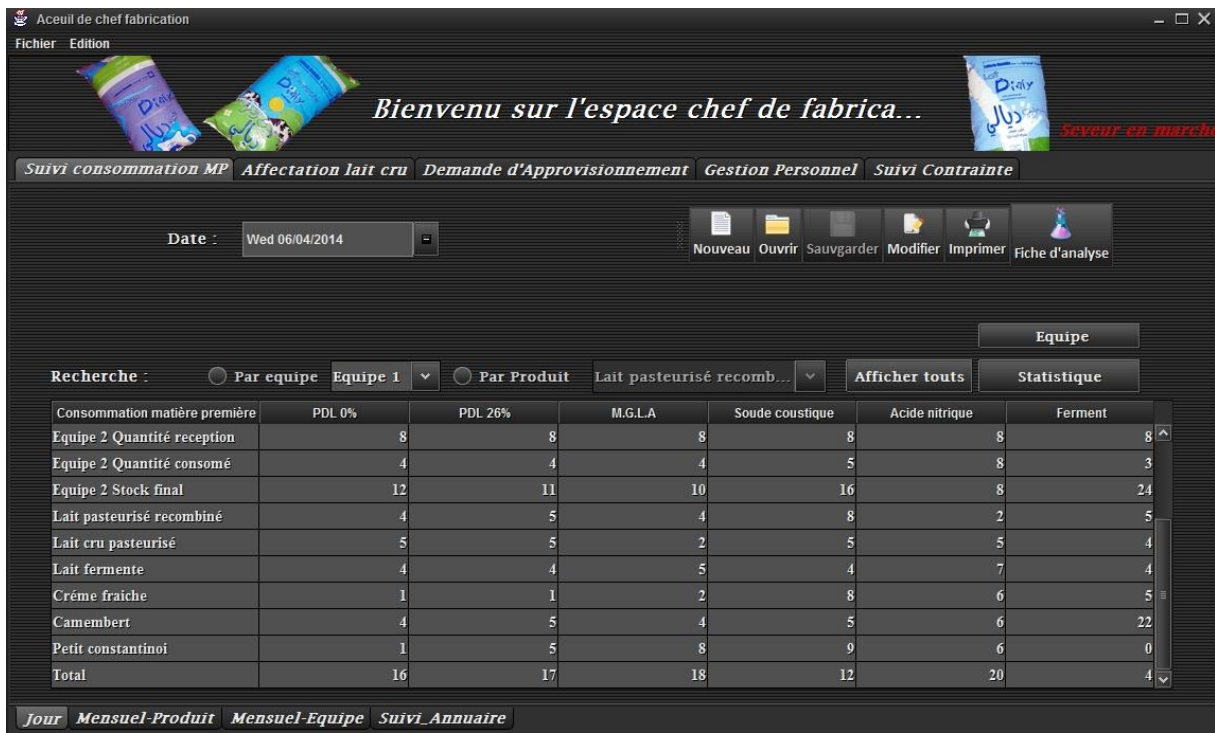


Figure III.14 Onglet de suivi la consommation matière première par jour

### III.5.7 La fenêtre de suivi mensuel de la consommation de matière première par équipe



Figure III.15 Onglet de suivi mensuel de la consommation matière par équipe

### III.5.8 La fenêtre de l'affectation de lait cru sur déférent produits

Permet de suivi l'affectation de lait cru sur déférent produit et ces détails de réception

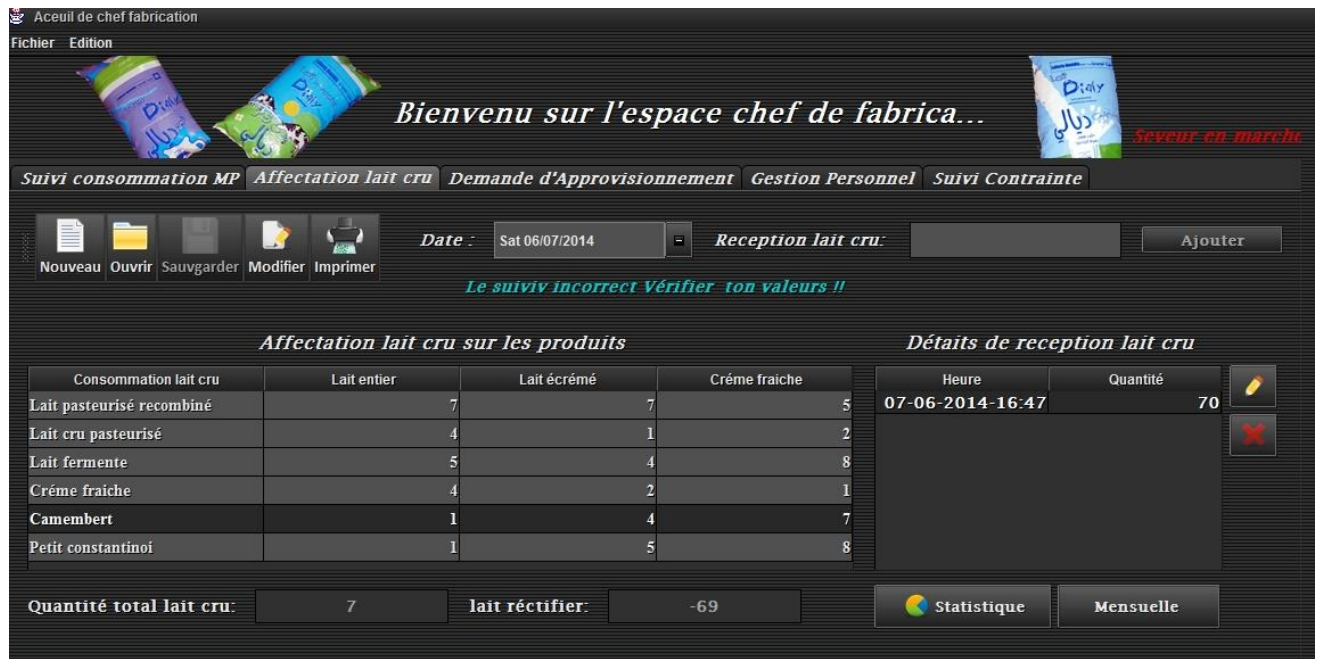


Figure III.16 Onglet de l'affectation de lait cru

### III.5.9 La fenêtre des statistiques de consommation matière première par jour

Cette fenêtre permet de consulter les statistiques de consommation matière première par équipe et par produit de chaque jour

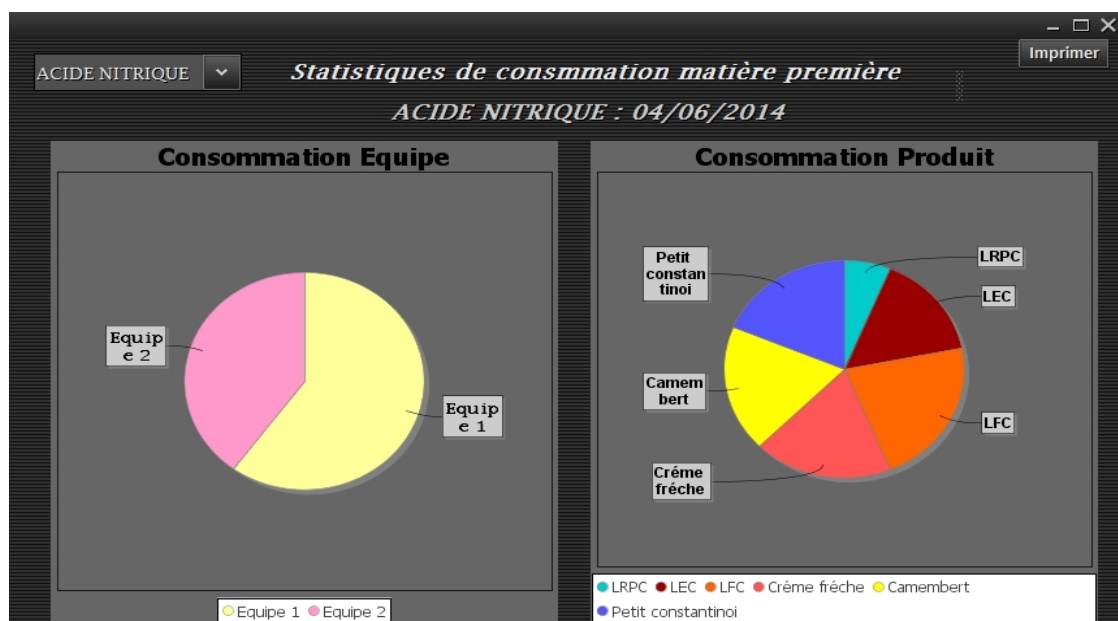


Figure III.17 Fenêtre de Statistiques de consommation matière première

### III.5.10 La fenêtre de planning des équipes de travail

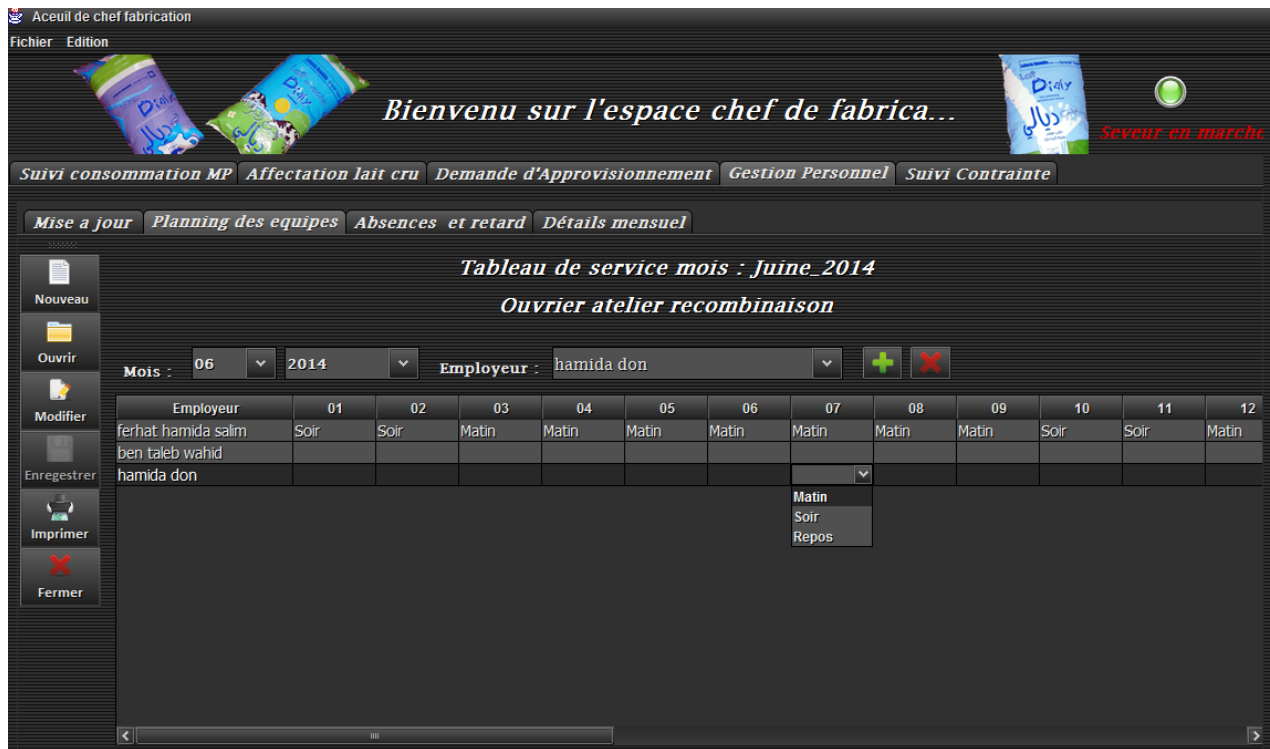


Figure III.18 Fenêtre de planning des équipes de travail

### III.5.11 La fenêtre de suivi matières premières au niveau de chef département

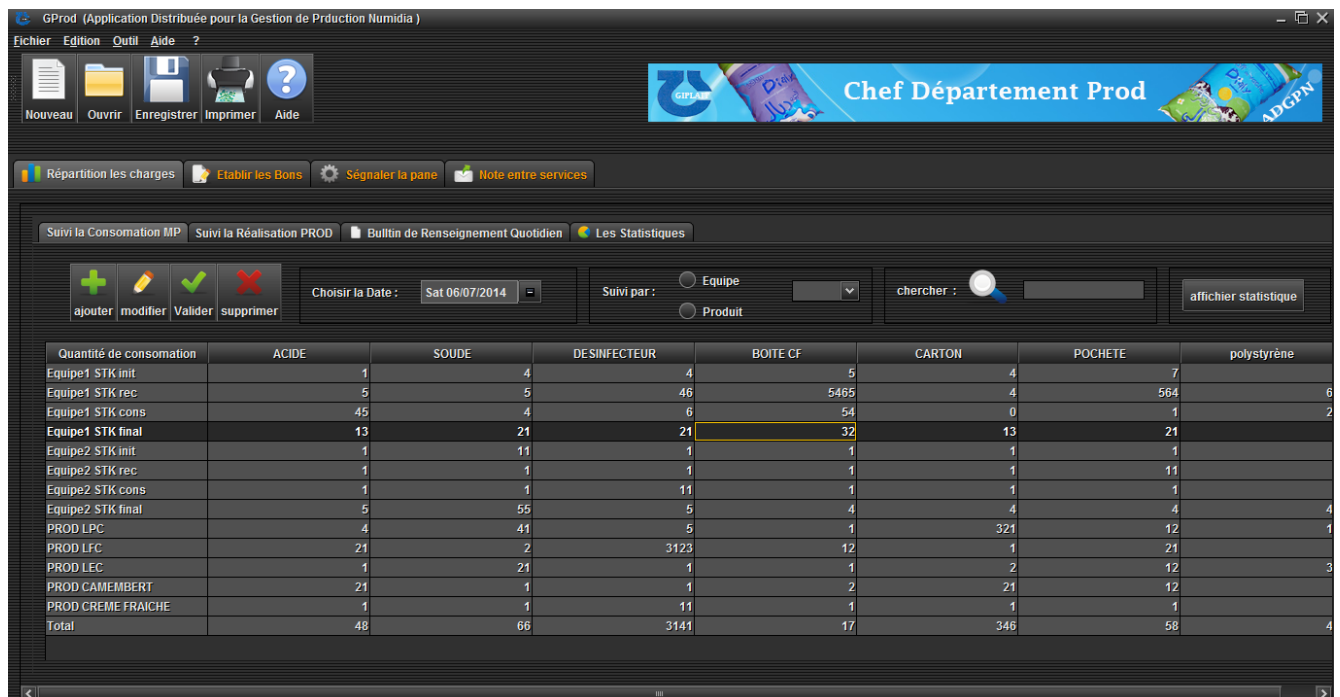


Figure III.19 Fenêtre de suivi matière première dans le chef département

### III.5.12 La fenêtre de l'espace membre de laboratoire

Quantité de consommation	ACIDE	SOUDE	DESINFECTEUR	BOITE CF	CARTON	POCHETE	polystyrène
Equipe1 STK init	1	4	4	5	4	7	
Equipe1 STK rec	5	5	46	5465	4	564	6
Equipe1 STK cons	45	4	6	54	0	1	2
Equipe1 STK final	13	21	21	32	13	21	
Equipe2 STK init	1	11	1	1	1	1	
Equipe2 STK rec	1	1	1	1	1	11	
Equipe2 STK cons	1	1	11	1	1	1	
Equipe2 STK final	5	55	5	4	4	4	4
PROD LPC	4	41	5	1	321	12	1
PROD LFC	21	2	3123	12	1	21	
PROD LEC	1	21	1	1	2	12	3
PROD CAMEMBERT	21	1	1	2	21	12	
PROD CREME FRAICHE	1	1	11	1	1	1	
Total	48	66	3141	17	346	58	4

Figure III.20 Fenêtre de l'espace de membre de laboratoire

### III.6 Conclusion

Dans ce chapitre, nous avons décrit brièvement le processus de réalisation de notre application ADGPNumidia (Application Distribuée pour la Gestion de Production Numidia) en spécifiant l'environnement et les outils de développement, l'implémentation de la base des données et les fichiers Excel.

En effet, nous avons achevé l'implémentation de toutes les fonctionnalités de cahier de charge, tout en respectant la conception élaborée.

## Conclusion :

Ce projet a été avant tout une synthèse de notions acquises durant les cinq années précédentes et il nous permet ainsi d'enrichir nos connaissances dans des domaines très variés comme : L'Orienté Objet, UML, le langage JAVA.....ect.

Pour réaliser effectivement ce travail, on s'est référé à divers outils de modélisation et de programmation pour les applications.

Pour l'analyse et la conception le fait d'aboutir à un travail parfait, nous avons suivi une démarche, basée sur le langage de modélisation UML. Le résultat de l'application nous a permis d'élaborer un système qui facilite l'interaction entre les services de l'entreprise en cours de la gestion de la production.

Le langage est le plus important dans le monde d'implémentation des applications parmi ces langages il existe Java, ce dernier réalise la possibilité de communication avec la base de données prête à être améliorée à tout moment, et aussi la manipulation des fichiers Excel. Cette application a été testée et à donner des résultats satisfaisants.

Pour conclure, on peut dire que tous les objectifs visés ont été réalisés, dans la limite du temps qui nous a été accordé, notre objectif principal est la réalisation de l'application pour le suivi de la production. En utilisant les nouvelles technologies.

L'application restera toujours ouverte aux critiques et suggestions pouvant la réajuster, surtout aux améliorations pouvant lui être amenées par des études ultérieures. Nous souhaitons que ce modèle sera suivi par les gens intéressés par cette démarche.

# Bibliographie

## Les livres électroniques

- [2] : « **Gestion de Production** », Auteur : Pierre Lopez, LAAS-CNRS Laboratoire d'Analyse et d'Architecture des Systèmes Centre National de la Recherche Scientifique Toulouse.
- [3] : « **Gestion de stock et de la production** », Auteur : P. GASPART, Université libre de Bruxelles, Faculté des sciences appliqués, Systèmes flexibles de production ;
- [4] : « **Fonction ordonnancement au d'un système de gestion de production** », Auteur : Mohsen Akrouf et Faouzi Masmoudi, Date : Octobre 2007, Ecole Nationale d'Ingénieurs de Sfax, B.P. W, 3038 Sfax, Tunisie.
- [5] : « **Introduction aux Systèmes Distribués** », Université de Nancy France.
- [6] : « **Un modèle d'événements pour le support de la coordination dans un système à objets répartis** », Auteur : F. Boyer - E.Lenormand - V. Marangozov, Ecole d'ingénieur ISNA Rouen
- [7] : « **une plate-forme générique pour la construction de collecticiels synchrones** » Acteur : R. Balter - S. Ben Atallah - R. Kanawati- M. Riveill, *Rapport Technique Sirac* 11-96, IMAG et INRIA Rhône-Alpes.
- [8] : M. N. Nelson, M. Linton, S. Owicki. A Highly Available ITV System, *Proc. 15th ACM Symposium on Operating Systems Principles*, Copper Mountain, pp. 54-67.
- [9] : « **Introduction aux Applications Réparties** », Acteur : Sacha Krakowiak Université Joseph Fourier & INRIA.
- [10] : « **Systèmes et Applications Réparties** », Acteur : Michel Daydé, Date : Janvier 26, 2005, ENSEEIHT-IRIT 2 rue Camichel 31071 Toulouse France.
- [14] : « **Systèmes et Applications Réparties** », Acteur : Claude Duvallet, Université du Havre UFR Sciences et Techniques 25 rue Philippe Lebon - BP 540 76058 LE.
- [16] : « **des concepts à la pratique** », Auteur : Jean-Marc Geib - Christophe Gransart - Philippe Merle, Université des Sciences et Technologies de Lille.
- [17] : « **presentation de l'archetecture com \ dcom** », Auteur : Yves laloum, Conseil audit desysteme d'info CISA.

## Les livres

[11] : «**Les réseaux avec Cisco** » Auteur : André VAUCMPS, Editeur : ENI, ISBN : 978-2-7460-4944-4, Auteur : Juin 2009.

[12] : «**Architectures Réparties en Java** » : RMI, CORBA, JMS, sockets, SOAP, services web, **Auteur** : Annick FRON, ISBN : 978-2-10-051141-9, Éditions : Dunod

[13] : «**Distributed Systems** » Auteur : Joel M. Crichlow, ISBN : 978-81-203-3810-4.

[21] : «**UML 2**» Auteur : Pascal Roques Editeur : EYROLLES 4<sup>e</sup> édition.

[24] : «**Client serveur** » Auteur : Robert Orfali Editeur : Vuibert 3<sup>e</sup> édition

## Les mémoires

[1] : «**Gestion de production auto organisateur à base d'agent** », date : juillet 2010  
N° : LMD 179.

[23] : «**Conception et réalisation d'une application de gestion d'assurance automobile au niveau de la saa** », Timousserh Fatima zohra, Labiod Ghania, Ghdbane Hanane, Université mentouri constantine N° ing 333.

[25] : «**Modélisation et réalisation d'un site web dynamique d'un agence commerciale** » avec le langage de modélisation UML et le processus de développement 2TUP ; université mentouri Constantine ; Session 2007

## Les sites web

[19] : «**UML 2** », Auteur : Laurent Audiber,  
<http://laurentaudibert.developpez.com/Cours-UML/html/Cours-UML.html#htoc20>

[20] : <http://www.irisa.fr/manifestations/seminaires2000/10nov00/Bezivin/sld008.htm>