

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche  
Scientifique  
Université Ibn Khaldoun – Tiaret  
Faculté des Mathématiques et de l'Informatique  
Département de l'Informatique



## **Mémoire de fin d'études**

Pour l'obtention du Diplôme de Master en Informatique

Spécialité : **Réseaux et Télécommunications**

**Thème :**

*Systeme intégré de gestion d'opticien lunetterie*

Réalisé et présenté par :

- **Elhamoul elfarouk**
- **Rekkab zine elabidine**

Encadreur : Mr **Ghafour Yacine**

*Année universitaire*

*2015/2016*

# Remerciements

*Au moment de clôturer ce mémoire, nous remercions en premier Dieu de nous avoir donné la force, la patience et le courage de terminer ce travail dans des conditions qui n'ont pas été faciles.*

*Nous remercions de tout notre cœur notre encadreur, Monsieur Ghafour Yacine, d'avoir accepté de nous encadrer et de suivre notre travail de près avec sa rigueur scientifique et ses conseils qui nous ont bien orientés dans l'élaboration de ce travail.*

*Nous exprimons nos reconnaissances à tous les enseignants qui ont participé à notre formation depuis le début de nos études universitaires jusqu'au jour d'aujourd'hui.*

*Nous remercions particulièrement les membres du Jury pour les remarques, les critiques et appréciations enrichissantes que vous apportez à ce modeste travail.*

*A toutes les personnes qui ont contribué de près ou de loin, moralement ou matériellement, recevez nos sincères remerciements.*

# *Dédicaces*

*Avec les sentiments de gratitude les plus profondes*

*Je dédie ce modeste travail à :*

*Mes très chers parents.*

*En hommage à tous les sacrifices que vous avez consenti pour moi durant toutes mes années d'études, je vous remercie d'avoir fait de moi ce que je suis maintenant et de m'avoir appris à vivre dans l'honneur et dans la dignité. Aucune dédicace, aucun mot, ne saurait exprimer réellement mon profond amour, mon respect et ma vive gratitude.*

*A mes tendres frères et sœurs, pour tant de confiance, d'amour, de patience et d'abnégation.*

*De près ou de loin, vous m'avez toujours tendu une perche pour sauver mon savoir être, mon savoir vivre et mon savoir faire, en m'orientant toujours vers le droit chemin, celui qui conduit vers la réussite.*

*Il serait ingrat de ne pas songer à mon cher Binôme Rekkab zine elabidine, pour toutes les peines et difficultés partagées ensemble lors de l'élaboration de ce projet.*

*A tous mes amis que j'ai eu l'occasion de connaître au cours de mes études.*

*Elhamoul Elfarouk*

# *Dédicaces*

*Avec les sentiments de gratitude les plus profonds,*

*J'ai un grand plaisir de dédier ce travail A mes très chers parents, eux qui ont gouverné mes premiers pas et qui n'ont pas cessé de me soutenir dès le début de mes études jusqu'à l'heure où je suis aujourd'hui, pour toutes les peines endurées, toutes les privations et sacrifices consentis « Que la réussite de ce projet soit la récompense de leurs efforts ».*

*Veillez trouver dans ce travail le fruit de tous vos sacrifices.*

*A mes tendres frères et sœurs, pour tant de confiance, d'amour, de patience et d'abnégation.*

*A toute ma famille, mes cousins et cousines.*

*Il serait ingrat de ne pas songer à mon cher Binôme Elhamoul Elfarouk, pour toutes les peines et difficultés partagées ensemble lors de l'élaboration de ce projet.*

*A tous mes amis que j'ai eu l'occasion de connaître au cours de mes études.*

*Rekkab Zine elabidine*

## Résumé

Ce projet de fin d'étude propose la création d'une application web pour la gestion d'un cabinet d'opticien qui informatise le système de vente des lunettes médicale pour faveur des assurées affilier en CNAS avec des privilèges et conditions spécifiques. Le projet est constitué des pages "Web" qui inter actent avec la base de données de CNAS. Ces pages sont invoquées à travers un système Servlet /JSP basé sur J2EE on utilise les EJB pour effectuer les opérations d'authentification des utilisateurs et de recherches des assurés et les bénéficiers, et enfin l'enregistrement et l'impression des factures Le projet est un site "internet" dont les utilisateurs sont les opticiens et la CNAS. Les pages de ce site résident dans le serveur "Application" Glassfich, seuls les opticiens identifiées qui sont identifiés par le système y accèdent.

## Abstract

This final project study proposes the creation of a web application for managing an optician firm that the computerized medical eyeglasses sale system for insured CNAS join in with specific privileges and conditions.

The project consists of the "Web" pages that Inter are acting with the base Data CNAS. These pages are invoked through a Servlet / JSP based on J2EE EJB system is used to perform authentication operations of users and provided research and incumbents, and finallyrecording and printing invoices The project is an "internet" site whose users are opticians and CNAS. The pages of this site resides in the server "Application" Glassfich, only identified opticians who are identified by the system accessing it.

**Mot clé** : entreprise java bean(EJB), java server pages (jsp), Conteneur Web(Servlet), java entreprise édition(J2EE) , CNAS

## Table des matières

<i>Résumé</i> .....	2
<i>Abstract</i> .....	2
<i>Mot clé</i> .....	2
<i>Liste des Figures</i> .....	1
<i>Liste des abréviations</i> .....	2
• <i>JPA</i> .....	2
<i>Introduction Générale</i> .....	1
<i>Chapitre1 : L'architecture J2EE</i> .....	2
1. <i>Introduction</i> .....	2
1. <i>Architecture J2EE:</i> .....	2
2.1. <i>Les diffèrent couche de J2EE :</i> .....	3
2.1.1. <i>Couche présentation</i> .....	3
2.1.2. <i>Couche application</i> .....	4
2.1.3. <i>Couche métier :</i> .....	4
2.1.4. <i>La couche persistance :</i> .....	5
2.2. <i>Les API de J2EE :</i> .....	6
2.2.1. <i>Les services d'infrastructures :</i> .....	6
2.2.2 <i>Les services de communication :</i> .....	6
2.3. <i>Les composants de J2EE :</i> .....	7
2.3.1. <i>Les composants web :</i> .....	7
2.3.2. <i>Les composants métier :</i> .....	8
<i>B.5 Les EJB orientés messages</i> .....	12
3. <i>Architecture logiciel</i> .....	12
3.1 <i>MVC (Modèle-vue-contrôleur) :</i> .....	12
3.1.1 <i>La couche Model :</i> .....	12
3.1.2 <i>La couche Vue :</i> .....	13
3.1.3 <i>La couche Controller :</i> .....	13
3.2 <i>Architecture 3 tiers :</i> .....	15
3.2.1. <i>La couche dao :</i> .....	15
3.2.2. <i>La couche métier :</i> .....	15
3.2.3. <i>La couche interface :</i> .....	16
<i>Conclusion :</i> .....	18
<i>Chapitre 2 : Présentation la structure CNAS</i> .....	20

## Liste des matières

<i>Introduction</i> .....	20
<i>1. Définition</i> .....	20
<i>2. Les missions de la CNAS [20]</i> .....	20
<i>3. Présentation de la structure d'accueil [21]</i> .....	21
<i>3.1. La structure d'accueil d'administration générale (Alger)</i> .....	21
a) <i>La direction centrale des prestations</i> .....	21
b) <i>La direction centrale du recouvrement et du contentieux</i> .....	22
d) <i>La direction centrale du contrôle médical</i> .....	22
e) <i>La direction centrale de la prévention des accidents de travail et des maladies professionnelles</i> .....	22
f) <i>La direction centrale des études, des statistiques et de l'organisation</i> .....	23
g) <i>La direction centrale de l'informatique</i> .....	23
h) <i>La direction centrale des opérations financières</i> .....	23
i) <i>La direction centrale des réalisations, des équipements et des moyens généraux</i> .....	24
j) <i>La direction centrale des personnels et de la formation</i> .....	24
<i>3.2 Les Agences wilaya</i> .....	24
a) <i>L'agence de première catégorie</i> .....	25
b) <i>L'agence de la deuxième catégorie</i> .....	25
c) <i>L'agence de la troisième catégorie</i> .....	26
<i>3.2.1. L'agence de la wilaya de Tiaret [24]</i> .....	26
<i>4. Présentation Générale du système CNAS</i> .....	29
<i>4.1. L'assuré Sociale</i> .....	29
4.1.1. <i>Les catégories particulières avec professionnelle</i> .....	29
4.1.2. <i>Les catégories particulières sans activité professionnelle</i> .....	29
4.1.3. <i>Les ayants droit</i> .....	29
<i>4.2. Système CHIFA</i> .....	30
4.2.1. <i>Les composants du système</i> .....	30
4.2.2. <i>Structures intervenantes de la CNAS</i> .....	30
4.2.3. <i>Les procédures d' enrôlement des assurés</i> .....	31
<i>5. La convention liant la CNAS aux opticiens lunettiers :</i> .....	32
5.1. <i>Prestations couvertes :</i> .....	32
5.2. <i>Les frais d'acquisition des montures et des verres correcteurs</i> .....	33
5.3. <i>Les frais de renouvellement des montures et des verres :</i> .....	33
5.3.1. <i>Le renouvellement des verres correcteurs :</i> .....	33

## Liste des matières

5.3.2 Le renouvellement des montures et des verres correcteurs :.....	33
5.4 .Logiciel chifa – optic :.....	33
6. Conclusion.....	34
Chapitre3 : Modélisation et réalisation .....	35
Introduction.....	35
1. présentation de l'UML .....	35
1.1. Définition :.....	35
1.2. Les avantages et les inconvénients d'UML :.....	35
1.2.1. Les points forts d'UML :.....	35
1.2.2. Les points faibles d'UML :.....	36
1.3. Présentation générale des diagrammes d'UML .....	36
1.3.1. Représentation statique du système (structurel) :.....	36
1.3.2. Représentation dynamique du système (comportemental):.....	36
1.4. Diagramme de cas d'utilisation: .....	37
1.4.1. Identification des acteurs : .....	37
1.4.2. Identification des cas d'utilisations:.....	38
1.5. Diagramme de séquence: .....	39
1.6. Diagramme d'activité :.....	40
1.7. Diagramme de classes : .....	41
1.7.1 Son utilisation :.....	41
1.7.2 Identification des classes :.....	42
2. Réalisation de l'application :.....	43
2.1. Langage de Programmation Java :.....	43
2.2. Netbeans.....	43
2.3. La base de donnée java DB :.....	44
2.3.1. Définition : .....	44
2.3.2. Avantages :.....	44
2.4. Présentation de l'application : .....	45
2.4.1. Authentification :.....	45
2.4.2 : page d'accueil :.....	46
.....	46
2.4.3 : Page assuréInfo :.....	46
2.4.4. Page d'ayant-Droit :.....	47
2.4.5. L'impression : .....	48

*Liste des matières*

*Conclusion* ..... 49

*Bibliographie* ..... 51

## Liste des Figures

Figure 1.1: Architecture générale d'une application en couches (Java EE).....	3
Figure 1.2 : Architecture J2EE.....	8
Figure 1.3 : L'architecture MVC.....	14
Figure 1.4 : Échange d'informations entre les éléments.....	14
Figure 1.5 : La requête du client arrive au contrôleur et celui-ci lui retourne la vue.....	15
Figure 1.6 Architecture 3-tiers et mise en place du MVC .....	16
Figure 1.7 : Architecture EJB simple.....	16
Figure 1.8: Architecture EJB type Business To Business.....	17
Figure 1.9 : Serveur EJB.....	17
Figure 1.10 : Diversité des types de clients pour un EJB.....	18
Figure 2.1 : Organigramme de la CNAS (Agence-Tiaret).....	28
Figure 2.2 : Schéma des relations inter structurent.....	31
Figure 3.1- Diagramme de CAS d'utilisation.....	39
Figure 3.2 - Diagramme de séquence .....	40
Figure 3.3- Diagramme d'activité .....	41
Figure 3.4- Diagramme de classes.....	42
Figure 3.5: logo netbeans .....	44
Figure 3.6 : Notre base de données.....	45
Figure 3.7 : page d'authentification.....	46
Figure 3.8 : page d'accueil.....	46
Figure 3.9 : Page assuréInfo.....	47
Figure 3.10 : Page d'ayant-droit.....	48
Figure 3.11 : Page assuré Info.....	48
Figure 3.12 : la facture.....	49

## Liste des abréviations

- Servlets : Conteneur Web
- JSP : java server pages
- JSF : Java Server Face, Framework Web, extension des JSP
- EJB : entreprise java bean
- JNDI : java naming and directory interface
- JDBC : java database connectivity
- JavaMail : API de gestion des mails
- JMX : Java Management Extension
- JCA : J2EE Connector Architecture
- JAXP : API d'analyse XML
- JAXM : API de communication asynchrone par XML
- JAX-RPC : API de communication synchrone par XML, par exemple à l'aide du protocole SOAP
- JAXB : API de sérialisation par XML
- JAXR : API de gestion des registres XML, permettant d'enregistrer des Web Services en ebXML
- JPA : Java Persistence API
- RMI : API de communication distante entre des objets java
- Java IDL : API de communication entre objets Java et objets non-Java, via le protocole CORBA

## **Introduction Générale**

L'informatique reconnaît une hausse depuis le « soit disant » bug de l'an 2000. Elle est due à l'informatisation de la majeure partie des tâches, à la puissance des processeurs qui ne cesse de grandir et surtout aux nouvelles technologies de l'information et de la communication. Ces changements posent, naturellement, un grand challenge aussi bien pour les décideurs politiques que pour les entreprises. Cette transition amènent les entreprises à repenser leurs stratégies et leurs structures, d'où cet engluement pour les développeurs java J2EE et .NET.

L'Internet est un système de communication qui permet de communiquer et de s'échanger des informations. Cette communication permet donc, de généraliser l'utilisation des outils informatiques (logiciel) plus performants avec des clients légers (navigateur web devenu plus complet et sans besoin d'installer le logiciel sur des machines individuelles). Ceci permet d'accéder aux ressources sans contraintes particulières.

La technologie java J2EE est une technologie qui utilise un ensemble d'API java : **(Servlets, JSP, JSF, EJB, JNDI, JDBC, JavaMail, JMX, JCA, JAXP, JAXM, JAX-RPC, JAXB, JAXR, JPA, RMI, Java IDL)** .

Cette technologie propose ainsi de pouvoir développer des applications qui pourront tourner sous différents navigateurs, tout en assurant la sécurité que procure une application métier java.

Dans ce cadre s'inscrit notre projet de fin d'études qui consiste à réaliser un site Web pour le système intègre pour la gestion des opticiens lunetterie

Pour atteindre notre objectif nous avons partagé le travail comme suit : Le premier chapitre est une prise de connaissance des différentes technologies utilisé pour concevoir les applications web basé sur le langage de programmation Java. Dans le second chapitre, nous allons faire une présentation générale sur la CNAS et ses différentes et la relation entre cette dernière et les opticiens. Le troisième chapitre sera pour la modélisation théorique de l'application avec le langage UML. Le quatrième chapitre sera consacré à la conception de l'application. Il s'agit d'une représentation du et description du sicle de vie de l'application a l'aide des capture faite pour les résultats obtenus.

# **Chapitre 01 :** **L'architecture J2EE**

# Chapitre1 : L'architecture J2EE

## 1. Introduction

Java est un langage de programmation à usage général, évolué et orienté objet dont la syntaxe est proche du C. Ses Caractéristiques ainsi que la richesse de son écosystème et de sa communauté lui ont permis d'être très largement utilisé pour le développement d'applications de types très disparates. Java est notamment largement utilisée pour le Développement d'applications d'entreprises et mobiles.

Dans Ce chapitre Nous allons vous expliquer la plate-forme J2EE. En suite en vous expliquer l'utilité des composants entreprise java beans (EJB) dans la plate-forme java EE, l'architecture trois tiers et le model MVC.

Nous allons ainsi présenter le langage JSP et les servlets qui sont utilisés pour développer des applications web.

### 1. Architecture J2EE:

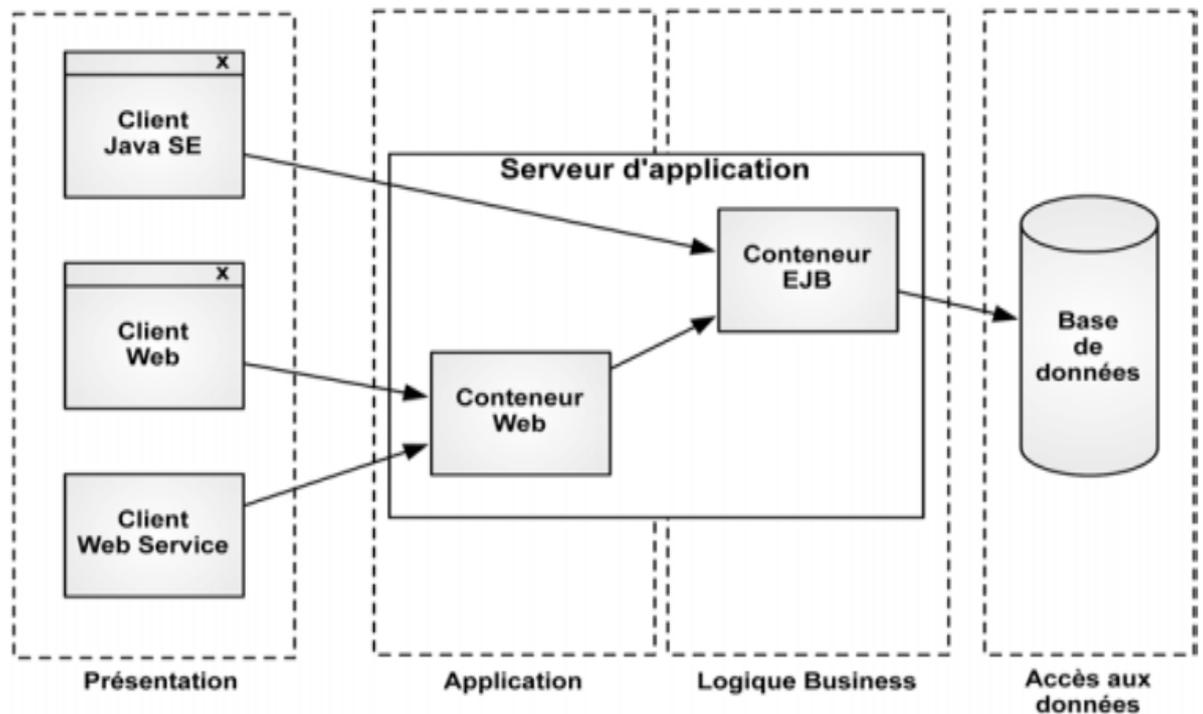
J2EE signifie Java 2 Enterprise Edition. J2EE est une plate-forme fortement orientée serveur pour le développement et l'exécution d'applications distribuées. [9], Elle est composée de deux parties essentielles :

- Un ensemble de spécifications pour une infrastructure dans laquelle s'exécutent les composants écrits en Java :
- Un tel environnement se nomme serveur d'applications.
- Un ensemble d'API qui peut être obtenues et utilisées séparément. Pour être utilisées, certaines nécessitent une implémentation de la part d'un fournisseur tiers. [1]

Les technologies Java utilisées pour créer des applications d'entreprise avec le JDK 1.4 ou antérieur. En même temps que le JDK 1.5 Amenait de nombreuses nouveautés dans le langage Java, Sun introduisait de nouvelles technologies s'appuyant sur ce langage amélioré afin de remédier à des lacunes de ces mêmes technologies dans J2EE. Le terme Java EE 5 a alors été utilisé pour désigner l'ensemble des technologies qui concourent à créer une application d'entreprise avec la plate-forme Java. Au moment de la mise à Jour de ce document, la dernière version de Java EE est Java EE 7. [16]

### 2.1. Les différentes couches de J2EE :

L'ordre de présentation des couches est lié à l'ordre dans lequel elles sont exécutées lors d'une requête de l'application cliente. [6]



*Figure 1.1: Architecture générale d'une application en couches (Java EE).*

#### 2.1.1. Couche présentation

La couche présentation est liée au type de clients utilisés. Si vous souhaitez travailler avec un client riche, vous devrez utiliser l'ensemble des outils et des bibliothèques mis à disposition pour ce type de client (plus particulièrement J2SE). [6]

Dans une architecture Java EE, on utilise principalement des applications web pour l'interface utilisateur. En effet, lorsque votre application n'a aucun besoin d'utiliser des composants graphiques très spécifiques, il est bien plus pratique d'utiliser les possibilités d'un navigateur HTML que de développer spécialement une application graphique. [6]

Dans le cas d'une application web, l'application cliente doit interpréter le HTML envoyé. La génération de celui-ci se situe donc au niveau du serveur d'applications.

En Java EE, les composants utilisés par le développeur sont les JSP (JavaServer

## Chapitre1 : L'architecture J2EE

Pages) et les TagLib. Cette couche est également nommée Vue, dans le concept du MVC (Modèle Vue Contrôleur).

La couche présentation n'est cependant pas toujours synonyme de graphique. La couche présentation d'un service web, par exemple, est la représentation XML des données retournées par celui-ci. [1]

### 2.1.2. Couche application

La couche application sert de médiateur entre la couche présentation et la couche métier, et contrôle l'enchaînement des tâches. Elle est chargée de connaître et de gérer l'état (connecté, en attente, déconnecté...) des sessions des clients connectés, si l'état est de type « conversationnel ». Cette couche représente le contrôleur dans un modèle MVC. [2]

Le composant de base utilisé dans un environnement Java EE est la servlet. Ces composants servent de point d'entrée à l'application. Ils doivent traiter la requête et faire les appels nécessaires afin de récupérer ou d'enregistrer des données. Ils redirigeront, enfin, l'application cliente vers la partie adéquate de la couche présentation. [5]

- ✓ TagLib : bibliothèques de balises JSP qui agissent comme des extensions au HTML ou au XML.
- ✓ MVC (Modèle vue contrôleur) : modèle de conception séparant le modèle de données, l'interface utilisateur et la logique de contrôle

Si vous souhaitez, par exemple, traiter un formulaire, vous utiliserez une servlet plutôt qu'une page JSP. Les pages JSP ne doivent comporter qu'un minimum de code Java afin d'être optimale en terme de maintenance. [5]

Cependant les servlet/JSP étant des composants de bas niveau, elles n'offrent que des fonctionnalités de base et ne permettent pas une interconnexion simple avec les autres couches. Celles-ci peuvent cependant être utilisées au sein de framework fournissant ces fonctionnalités d'interconnexions. Le plus connu est sans doute Struts1 qui met en avant l'ensemble de l'architecture MVC.

### 2.1.3. Couche métier :

La couche métier est la couche principale de toute application. Elle doit s'occuper aussi bien des accès aux différentes données qu'à leurs traitements, suivant les processus définis par l'entreprise. On parle généralement de traitement métier. Cette expression regroupe :

- la vérification de la cohésion entre les données.
- l'implémentation du logique métier de l'entreprise au niveau de l'application.

## Chapitre1 : L'architecture J2EE

- la gestion du workflow2. [1]

Dans le domaine de l'assurance, par exemple, un traitement métier pourrait s'apparenter à une méthode de calcul d'une prime d'assurance. [1]

Il est cependant plus propre de séparer toute la partie accès aux données de la partie traitement du logique métier. Cela offre plusieurs avantages. Tout d'abord, les développeurs ne se perdent pas entre le code métier, qui peut parfois être complexe, et le code d'accès aux données, plutôt élémentaire mais conséquent. Cela permet aussi d'ajouter un niveau d'abstraction sur l'accès aux données et donc d'être plus modulable en cas de changements de type de stockage. Il est alors beaucoup plus facile de se répartir les différentes parties au sein d'une équipe de développement. [6]

Dans le cas d'applications Java EE de grande envergure, cette couche est représentée par les EJB (Entreprise JavaBeans).

### 2.1.4. La couche persistance :

La partie du code responsable de l'accès aux données dans une application multi niveau doit être encapsulée dans une couche dédiée aux interactions avec la base de données de l'architecture généralement appelée couche de persistance. [4]

Celle-ci permet notamment :

- D'ajouter un niveau d'abstraction entre la base de données et l'utilisation qui en est faite.
- De simplifier la couche métier qui utilise les traitements de cette couche.
- De masquer les traitements réalisés pour mapper les objets dans la base de données et vice versa. [4]
- De faciliter le remplacement de la base de données utilisée.

La couche métier qui va utiliser la couche de persistance reste indépendante du code dédié à l'accès à la base de données. [8]

Ainsi la couche métier ne contient aucune requête SQL, ni code de connexion ou d'accès à la base de données. La couche métier utilise les classes de la couche persistance qui encapsulent ces traitements. Ainsi la couche métier manipule uniquement des objets pour les accès à la base de données. [7]

Le choix des API ou des outils dépend du contexte : certaines solutions ne sont utilisables qu'avec la plate-forme Enterprise Edition (exemple : les EJB) ou sont utilisables indifféremment avec les plates-formes Standard et Enterprise Edition. [7]

## Chapitre1 : L'architecture J2EE

L'utilisation d'une API standard permet de garantir la pérennité et de choisir l'implémentation à mettre en œuvre. [4]

Les solutions open source et commerciales ont les avantages et inconvénients inhérents à leur typologie respective. [4]

### 2.2. Les API de J2EE :

Les API de J2EE peuvent se répartir en trois grandes catégories qui sont représenté par les services

#### 2.2.1. Les services d'infrastructures :

Il en existe un grand nombre, définis ci-dessous :

- JDBC (Java DataBaseConnectivity) est une API d'accès aux bases de données relationnelles.
- JNDI (Java Naming and Directory Interface) : est une API d'accès aux services de nommage et aux annuaires d'entreprises tels que DNS, NIS, LDAP, etc.
- JTA/JTS (Java Transaction API/Java Transaction Services) est un API définissant des interfaces standard avec un gestionnaire de transactions.
- JCA (J2EE Connector Architecture) est une API de connexion au système d'information de l'entreprise, notamment aux systèmes dits «Legacy» tels que les ERP (*enterprise resource planning*).
- JMX (Java Management Extension) fournit des extensions permettant de développer des applications web de supervision d'applications. [5]

#### 2.2.2 Les services de communication :

- JAAS (Java Authentication and Authorization Service) est une API de gestion de l'authentification et des droits d'accès.
- JavaMail est une API permettant l'envoi de courrier électronique.
- JMS (Java Message Service) fournit des fonctionnalités de communication asynchrone (appelées MOM pour Middleware Object Message) entre applications.
- RMI-IIOP est une API permettant la communication synchrone entre objets.

L'architecture J2EE permet ainsi de séparer la couche présentation, correspondant à l'interface homme-machine (IHM), la couche métier contenant l'essentiel des traitements de données en se basant dans la mesure du possible sur des API existantes, et enfin la couche de données correspondant aux informations de l'entreprise stockées dans des fichiers, dans des

## Chapitre1 : L'architecture J2EE

bases de données relationnelles ou XML, dans des annuaires d'entreprise ou encore dans des systèmes d'information complexes .

### 2.3. Les composants de J2EE :

On distingue habituellement deux familles de composants : [2] [4]

#### 2.3.1. Les composants web :

Servlets et JSP (Java Server Pages). Il s'agit de la partie chargée de l'interface avec l'utilisateur (on parle de logique de présentation) :

##### 2.3.1.1. Les servlets :

Une servlet est un programme qui s'exécute côté serveur en tant qu'extension du serveur. Elle reçoit une requête du client, elle effectue des traitements et renvoie le résultat. La liaison entre la servlet et le client peut être directe ou passer par un intermédiaire comme par exemple un serveur http.

Même si pour le moment la principale utilisation des servlets est la génération de pages html dynamiques utilisant le protocole http et donc un serveur web, n'importe quel protocole reposant sur le principe de requête/réponse peut faire usage d'une servlet.

Ecrit en Java, une servlet en retire ses avantages : la portabilité, l'accès à toutes les API de Java dont JDBC pour l'accès aux bases de données, ...

Une servlet peut être invoquée plusieurs fois en même temps pour répondre à plusieurs requêtes simultanées.

Dans une architecture Client/Serveur trois tiers, la servlet se positionne dans le tiers du milieu entre le client léger chargé de l'affichage et la source de données.

##### 2.3.1.2. Les JSP : [2] [7]

Les JSP permettent d'introduire du code Java dans des tags prédéfinis à l'intérieur d'une page HTML. La technologie JSP mélange la puissance de Java côté serveur et la facilité de mise en page d'HTML côté client, Les fichiers JSP possèdent par convention l'extension .jsp.

Une JSP est habituellement constituée :

- de données et de tags HTML.
- de tags JSP.
- de scriptlets (code Java intégré à la JSP).

## Chapitre 1 : L'architecture J2EE

Concrètement, les JSP sont basées sur les servlets. Au premier appel de la page JSP, le moteur de JSP génère et compile automatiquement une servlet qui permet la génération de la page web. Le code HTML est repris intégralement dans la servlet. Le code Java est inséré dans la servlet.

La servlet générée est compilée et sauvegardée puis elle est exécutée. Les appels suivants de la JSP sont beaucoup plus rapides car la servlet, conservée par le serveur, est directement exécutée.

### 2.3.2. Les composants métier :

#### 2.3.2.1. EJB (Enterprise JavaBean) :

Les entreprises JavaBeans sont tout simplement un modèle de composants qui permet de développer des objets métier pouvant migrer d'une marque de conteneur d'EJB (serveur EJB) à un autre. [3]

Un Bean métier présente un modèle de programmation simple permettant au développeur de se concentrer sur l'aspect métier. [3]

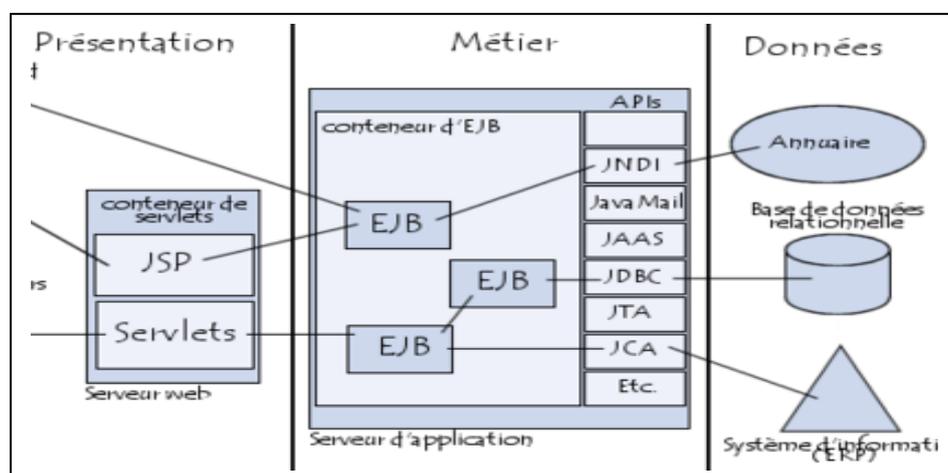


Figure 1.2 : Architecture J2EE

#### a. L'historique des EJB :

- ✓ **EJB 1.1** : publié en décembre 1999, intégré dans J2EE 1.2, se compose par :
  - ✓ Session beans (stateless/stateful).
  - ✓ Entity Beans .
  - ✓ Interface Remote uniquement.

## Chapitre 1 : L'architecture J2EE

- ✓ **EJB 2.0** : publié en septembre 2001, intégré à J2EE 1.3, se compose par :
  - ✓ Message-Driven Beans.
  - ✓ Entity 2.x reposant sur EJB.
  - ✓ Interface Local pour améliorer les performances des appels dans la même JVM.
- ✓ **EJB 2.1** : publié en novembre 2003, intégré à J2EE 1.4 se compose
  - ✓ EJB Timer Service.
  - ✓ EJB Web Service Endpoints via JAX-RPC.
  - ✓ Amélioration du langage EJB.
- ✓ **EJB 3.0** : intégré à Java EE 5, il se remarque par :
  - ✓ Utilisation de POJO et POJI, plus d'interface Home
  - ✓ Utilisation des annotations, le descripteur de déploiement est optionnel.
  - ✓ Utilisation de JPA pour les beans de type entity.
- ✓ **EJB 3.1** : intégré à Java EE 6. [2]

### B. Types des EJB :

#### B.1. Les EJB de type Session :

Les EJB Session sont généralement utilisés comme façade pour proposer des fonctionnalités qui peuvent faire appel à d'autres composants ou entités tels que des EJB session, des EJB Entity, des POJO. L'acronyme POJO, pour Plain Old Java Object (traduisible par "Java Objet, tout simplement"), est couramment utilisé aujourd'hui pour parler du mouvement visant à simplifier le plus possible la conception d'une application Java [8].

La version 3.0 des EJB rend inutile l'implémentation d'une interface spécifique à l'API EJB. Mais même si cela n'est pas obligatoire, il est fortement recommandé (dans la mesure du possible) de définir une interface dédiée à l'EJB qui va notamment préciser son mode d'accès et les méthodes utilisables.

#### B.2. Les beans de type Stateless

Les Beans de type stateless sont les plus simples car le conteneur gère un pool d'instances qui sont utilisées au besoin, ce qui évite des opérations d'instanciation et de destruction à chaque utilisation. Ceci permet une meilleure montée en charge de l'application. [3]. Un Stateless Session Bean est une collection de services dont chacun est représenté par une méthode. Stateless signifie que le service est autonome dans son exécution et donc qu'il ne dépend pas d'un contexte particulier ou d'un autre service. Le point important réside dans le fait qu'aucun état n'est conservé entre deux invocations de méthodes. Lorsqu'une

## Chapitre1 : L'architecture J2EE

application cliente appelle une méthode d'un Session Bean, celui-ci exécute la méthode et retourne le résultat. L'exécution ne se préoccupe pas de ce qui a pu être fait avant ou ce qui pourra être fait après.

### *Exemple :*

```
Importjavax.ejb.Local;  
@Local  
Public interface CalculLocal {  
Public long additionné (int valeur1, int valeur2);  
}
```

#### 3.2.1.2. Les beans de type Stateful

Les beans de type Stateful sont capables de conserver leur état durant toute leur utilisation par le client. Cet état n'est cependant pas persistant : les données sont perdues à la fin de son utilisation ou à l'arrêt du serveur. Un Stateful Session Bean est une extension de l'application cliente. Il introduit le concept de session entre le client et le serveur. On parle précisément d'état conversationnel pour qualifier ce type de communication. De ce fait, une méthode appelée sur l'EJB peut lire ou modifier les informations sur l'état conversationnel.

### **B.3. Les EJB de type Entité :**

Dans les versions antérieures des EJB, les EJB de type Entity avaient la charge de la persistance des données. Les EJB de type Entity CMP (Container Managed Persistence) nécessite simplement un fichier de description. [2]

Les EJB 3.0 proposent d'utiliser l'API Java Persistence pour assurer la persistance des données dans les EJB : ils utilisent un modèle de persistance léger standard en remplacement des entity beans de type CMP. [3]

JPA repose sur des beans entity qui sont de simples POJO enrichis d'annotations qui permettent de mettre en œuvre les concepts de POO tels que l'héritage ou le polymorphisme.

Jusqu'à la version 3.0 des EJB, les Entitybeans sont des composants qui dépendent pleinement du conteneur d'EJB du serveur d'applications dans lequel ils s'exécutent. L'utilisation de POJO avec l'API Java Persistence permet de rendre les beansentity indépendants du conteneur. Ceci possède plusieurs avantages dont celui de pouvoir facilement tester les beans puisqu'ils ne requièrent plus de conteneur pour leur exécution. [8]

- La création d'un bean Entity :

## Chapitre1 : L'architecture J2EE

Les beans de type Entity sont dans la version 3.0 des spécifications de simple POJO utilisant les annotations de l'API Java Persistence (JPA) pour définir le mapping. [2]

Les informations de mapping entre une table et un objet peuvent être définies grâce aux annotations mais aussi par un fichier de mapping qui permet d'externaliser les informations du POJO. Il est possible de mixer les deux (annotations et fichiers de mapping) mais les données incluses dans le fichier sont prioritaires par rapport sur les annotations.[1]

Il peut être pratique pour un bean de type entity d'implémenter l'interface Serializable : le bean pourra être utilisé dans les paramètres et la valeur de retour des méthodes métiers d'un EJB. Le bean peut ainsi être utilisé pour la persistance et le transfert de données.[3]

- b. La persistance des entités :

La version 3.0 propose une refonte complète des EJB entités afin de simplifier leur développement. Cette simplification est assurée en grande partie par la mise en œuvre de JPA qui permet :

- La standardisation du mapping O/R
- L'utilisation de POJO annotés avec support de l'héritage et du polymorphisme

La possibilité d'utiliser les EJB entités en dehors du conteneur d'EJB qui permet notamment la mise en œuvre de tests unitaires automatisés.

La persistance d'objets avec JPA repose sur plusieurs fonctionnalités :

- Un ensemble d'entités annotées qui représente le modèle objet du domaine
- Une API contenue dans le package javax.persistence
- Un cycle de vie pour les entités La classe EntityManager est responsable de la gestion

des opérations sur une entité notamment grâce à plusieurs méthodes :

- persist ()
- remove ()
- merge ()
- flush ()
- find ()
- refresh ()

- . La création d'un EJB Session pour manipuler le beanEntity

Le bean entity n'est utilisé que pour le mapping. Pour réaliser des opérations avec le bean entity, il faut développer un EJB Session qui va encapsuler la logique des traitements à réaliser avec le bean entity.

Il faut définir l'interface Local et/ou Remote des méthodes métiers de l'EJB.

Il faut ensuite définir l'EJB qui va utiliser l'API Java Persistence et le bean entity.

## Chapitre1 : L'architecture J2EE

L'injection de dépendances est utilisée pour obtenir une instance de l'EntityManager par le conteneur.

Exemple :

```
@PersistenceContext  
privateEntityManagerem;
```

L'annotation @PersistenceContext demande au conteneur d'injecter une instance de la classe EntityManager.

Le conteneur retrouve l'EntityManager grâce au nom de l'unité de persistance fourni comme valeur à la propriété unit Name de l'annotation si plusieurs unités de persistance sont définies.

L'instance de type Entity Manager peut être utilisée dans les méthodes métiers pour réaliser des traitements sur le bean Entity . [3]

### B.5 Les EJB orientés messages

Les EJB de type MessageDriven permettent de réaliser des traitements asynchrones exécutés à la réception d'un message dans une queue JMS.

JMS est l'API utilisée pour l'accès à un système de messagerie d'entreprise. C'est la solution Java EE au concept du MOM (Message Oriented Middleware).

## 3. Architecture logiciel

### 3.1 MVC (Modèle-vue-contrôleur) :

Pour suivre le motif de conception logicielle MVC. Programmer en utilisant MVC sépare votre application en 3 couches principales : [1]

#### 3.1.1 La couche Model :

La couche Model représente la partie de l'application qui exécute le logique métier. Cela signifie qu'elle est responsable de récupérer les données, de les convertir selon des concepts chargés de sens pour votre application, tels que le traitement, la validation, l'association et beaucoup d'autres tâches concernant la manipulation des données. [5]

A première vue, l'objet Model peut être vu comme la première couche d'interaction avec n'importe quelle base de données que vous pourriez utiliser pour votre application. Mais

## Chapitre1 : L'architecture J2EE

plus globalement, il fait partie des concepts majeurs autour desquels vous allez exécuter votre application. [8]

Dans le cas d'un réseau social, la couche Model s'occupe des tâches comme de sauvegarder des données, de sauvegarder des associations d'amis, d'enregistrer et de récupérer les photos des utilisateurs, de trouver des suggestions de nouveaux amis, etc ... Tandis que les objets Model seront "Ami", "User", "Commentaire", "Photo". [2]

### 3.1.2 La couche Vue :

La Vue retourne une présentation des données venant du model. Etant séparée par les Objets Model, elle est responsable de l'utilisation des informations dont elle dispose pour produire une interface de présentation de votre application.

Par exemple, de la même manière que la couche Model retourne un ensemble de données, la Vue utilise ces données pour fournir une page HTML les contenant. Ou un résultat XML formaté pour que d'autres l'utilisent. [1]

La couche Vue n'est pas seulement limitée au HTML ou à la représentation en texte de données. Elle peut aussi être utilisée pour offrir une grande variété de formats en fonction de vos besoins, comme les vidéos, la musique, les documents et tout autre format auquel vous pouvez penser. [1]

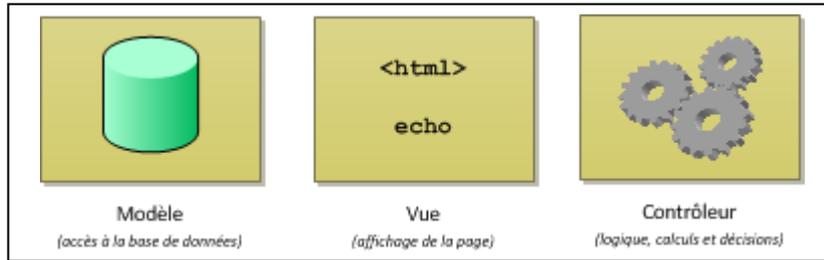
### 3.1.3 La couche Controller :

La couche Controller gère les requêtes des utilisateurs. Elle est responsable de retourner une réponse avec l'aide mutuelle des couches Model et Vue. [5]

Les Contrôleurs peuvent être imaginés comme des managers qui ont pour mission que toutes les ressources souhaitées pour accomplir une tâche soient déléguées aux travailleurs corrects. Il attend des requêtes des clients, vérifie leur validité selon l'authentification et les règles d'autorisation, délègue les données récupérées et traitées par le Model, et sélectionne les types de présentation correctes que le client accepte, pour finalement déléguer le processus d'affichage à la couche Vue. [5]

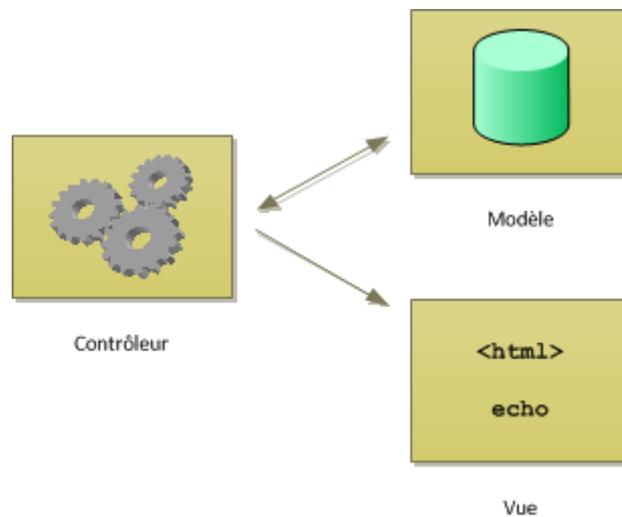
. La figure suivante schématise le rôle de chacun de ces éléments :

## Chapitre 1 : L'architecture J2EE



*Figure 1.3 : L'architecture MVC*

Il est important de bien comprendre comment ces éléments s'agencent et communiquent entre eux. Regardez bien la figure 1.4 :



*Figure 1.4 : Échange d'informations entre les éléments*

Il faut tout d'abord retenir que le contrôleur est le chef d'orchestre : c'est lui qui reçoit la requête du visiteur et qui contacte d'autres fichiers (le modèle et la vue) pour échanger des informations avec eux. [2]

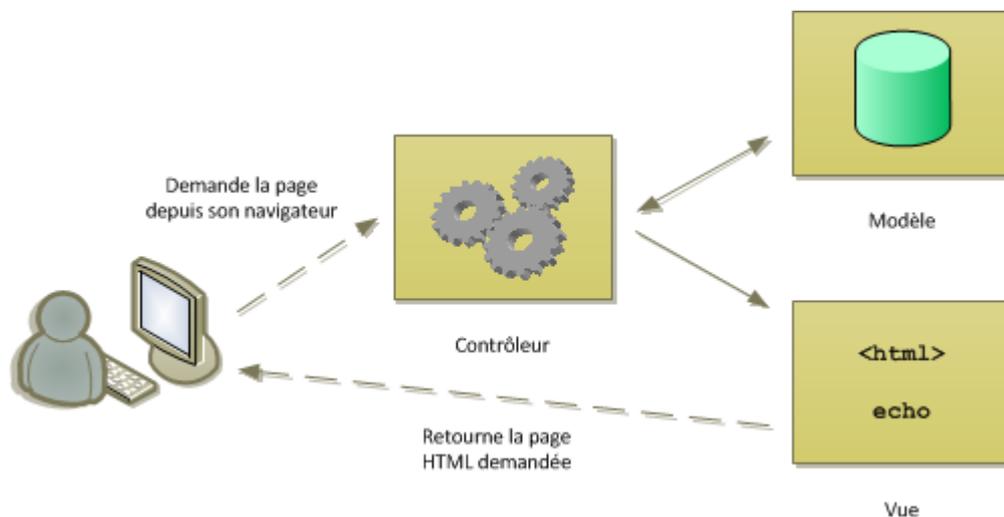
Le fichier du contrôleur demande les données au modèle sans se soucier de la façon dont celui-ci va les récupérer. Par exemple : « Donne-moi la liste des 30 derniers messages du forum no 5 ». Le modèle traduit cette demande en une requête SQL, récupère les informations et les renvoie au contrôleur.

Une fois les données récupérées, le contrôleur les transmet à la vue qui se chargera d'afficher la liste des messages. [8]

Dans les cas les plus simples, ce sera probablement le cas. Mais comme je vous le disais, le rôle du contrôleur ne se limite pas à cela : s'il y a des calculs ou des vérifications d'autorisations à faire, des images à miniaturiser, c'est lui qui s'en chargera. [5]

## Chapitre 1 : L'architecture J2EE

Concrètement, le visiteur demandera la page au contrôleur et c'est la vue qui lui sera retournée, comme schématisé sur la figure suivante. Bien entendu, tout cela est transparent pour lui, il ne voit pas tout ce qui se passe sur le serveur. C'est un schéma plus complexe que ce à quoi vous avez été habitués, bien évidemment : c'est pourtant sur ce type d'architecture que repose un très grand nombre de sites professionnels ! [5]



*Figure 1.5 : La requête du client arrive au contrôleur et celui-ci lui retourne la vue*

### 3.2 Architecture 3 tiers :

Une application Web possède souvent une architecture 3-tiers : [1]

#### 3.2.1. La couche dao :

S'occupe de l'accès aux données, le plus souvent des données persistantes au sein d'un SGBD. [1]

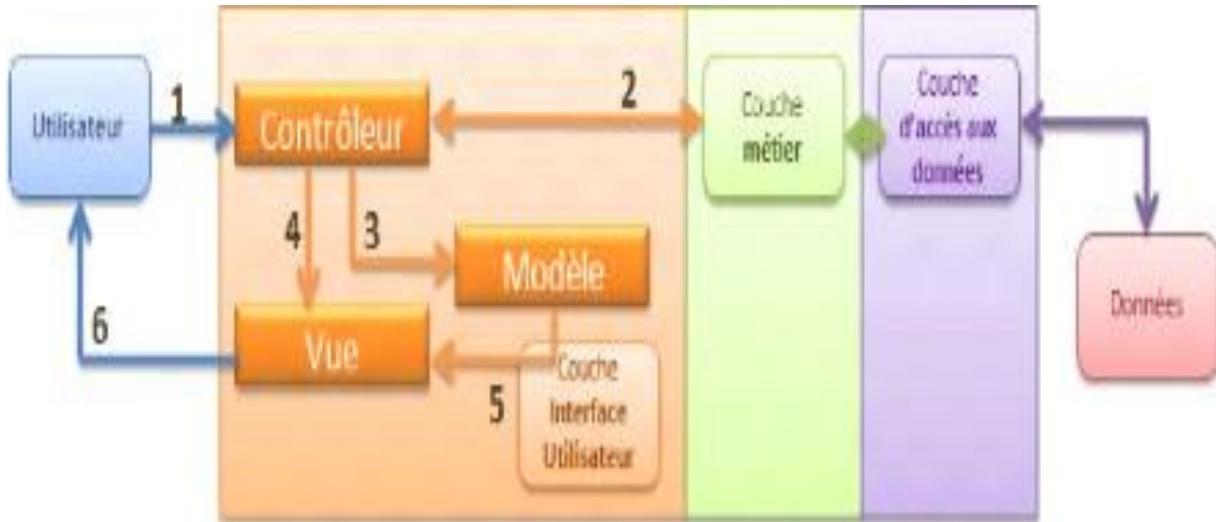
#### 3.2.2. La couche métier :

Implémente les algorithmes " métier " de l'application. Cette couche est indépendante de toute forme d'interface avec l'utilisateur. Ainsi elle doit être utilisable aussi bien avec une interface console, une interface web, une interface de client riche. Elle doit ainsi pouvoir être testée en-dehors de l'interface web et notamment avec une interface console. C'est généralement la couche la plus stable de l'architecture. Elle ne change pas si on change l'interface utilisateur ou la façon d'accéder aux données nécessaires au fonctionnement de l'application. [2]

## Chapitre 1 : L'architecture J2EE

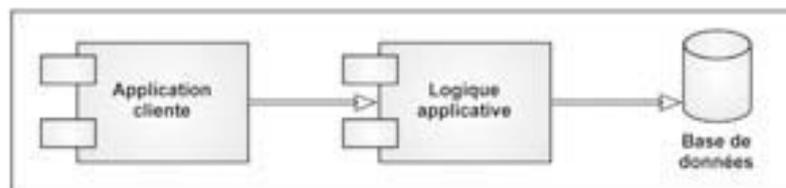
### 3.2.3. La couche interface :

Utilisateur qui est l'interface (graphique souvent) qui permet à l'utilisateur de piloter l'application et d'en recevoir des informations. [2]



*Figure 1.6 Architecture 3-tiers et mise en place du MVC.*

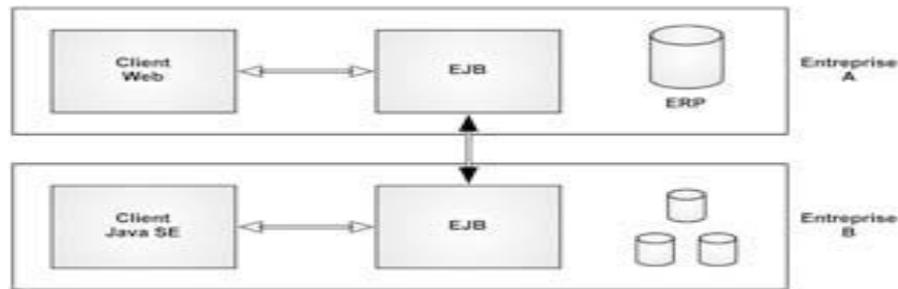
Une architecture EJB est composée d'au moins 3-tiers : une machine cliente, une machine contenant la logique applicative et une base de données (figure 1.7) :



*Figure 1.7 : Architecture EJB simple*

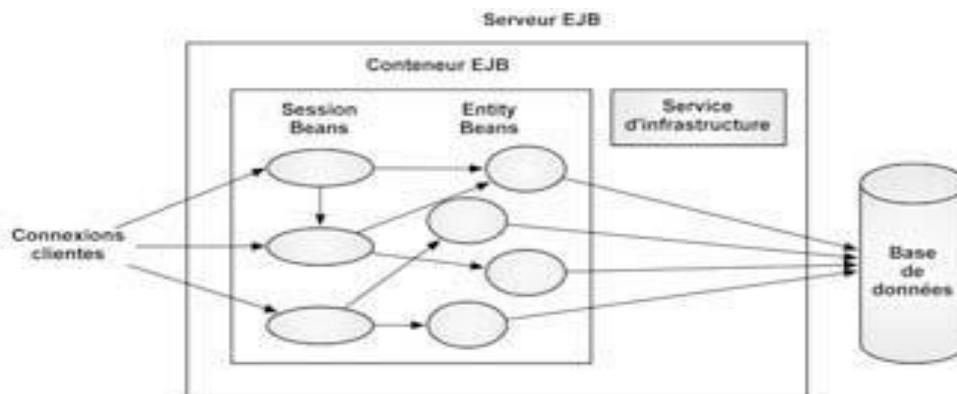
Dans ce type d'architecture, les EJB se situent dans le tiers « logique applicative ». Les EJB implémentent le logique métier de l'application et représentent une abstraction de l'accès à la base de données. Cependant, l'architecture EJB est avant tout une architecture évolutive. Il est alors possible de faire évoluer une application EJB simple vers une application orientée Business To Business connectant vers d'autres systèmes d'entreprise (fig. 1.8). [6]

## Chapitre 1 : L'architecture J2EE



**Figure 1.8: Architecture EJB type Business To Business**

Les EJB sont des applications exécutées côté serveur mais également englobées dans un conteneur. Chaque partie a ses propres obligations et règles, ce qui permet de faire évoluer une partie du serveur sans avoir à tout faire évoluer. [6]



**Figure 1.9 : Serveur EJB**

La figure 1.9 présente la structure d'un serveur EJB. Même si le conteneur a le rôle le plus important, c'est le serveur qui aiguille l'ensemble des requêtes et gère l'ensemble des conteneurs et services. Le serveur se doit de gérer, de plus, un ensemble de services d'infrastructure communs à l'ensemble des conteneurs ou des services. [6]

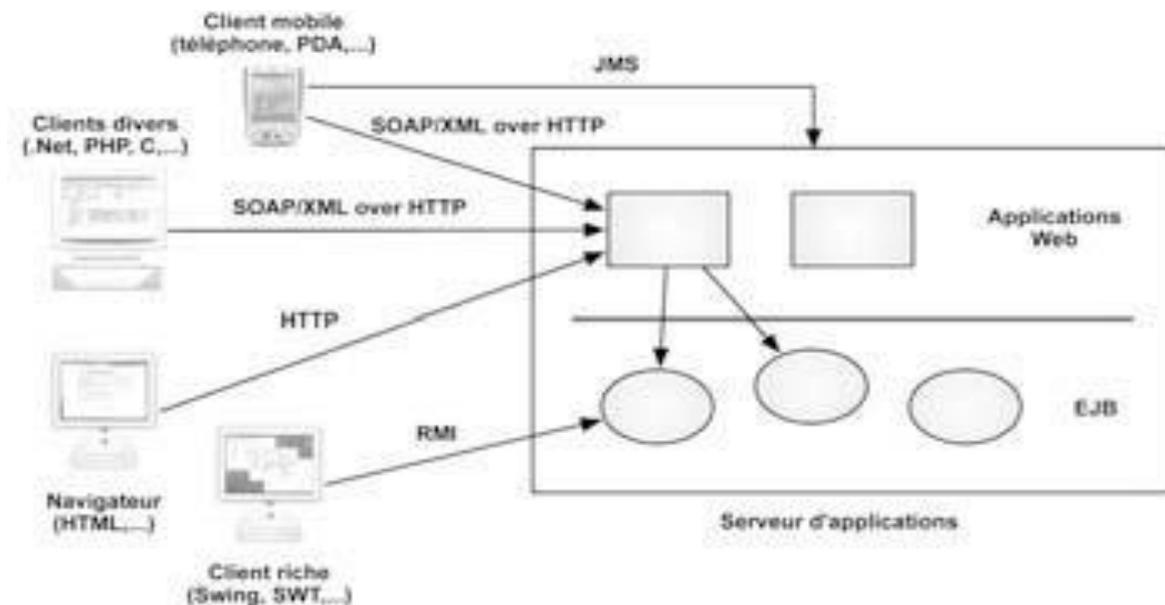
Le tiers client est représenté par les applications se connectant aux EJB. Ces applications sont généralement écrites en Java, toutefois, il est également possible de se connecter à un EJB avec un client écrit dans un autre langage via un accès par service web (figure.1.10).

Les clients Java utilisent généralement JNDI et RMI pour se connecter et pour appeler les méthodes des EJB. De ce fait, ils peuvent aussi bien être des applications graphiques fenêtrées (appelées clients riches), d'autres EJB, des applications internes, et, bien évidemment, des clients légers (navigateur web). Il est également possible de faire communiquer les systèmes extérieurs avec une messagerie inter applications, comme JMS. [6]

## Chapitre 1 : L'architecture J2EE

La spécification Java EE oblige le serveur à offrir un service d'annuaire JNDI et un service de transaction. Bien entendu, les serveurs d'applications fournissent généralement d'autres services facilitant le développement des applications. [2]

De façon imagée, on peut considérer le serveur EJB comme un orchestre. Le rôle de ce serveur est celui du chef d'orchestre. C'est lui qui dirige l'ensemble des services et leur cycle de vie (démarrage, arrêt, pause...). Chaque partie de l'orchestre correspond à un service (EJB, transaction, base de données...). Ils sont tous indépendants, mais ils travaillent ensemble pour produire un résultat commun. [2]



*Figure 1.10 : Diversité des types de clients pour un EJB*

De façon imagée, le client est le public qui écoute le concert. Le client a fait la demande pour rentrer dans la salle et a payé son ticket à la caisse (RMI), par un revendeur (HTTP, services web...) ou par petites annonces (JMS). [6]

## Conclusion :

Les applications J2EE tendent à supplanter les applications de gestion classique grâce aux nombreux avantages qu'elles apportent :

- Aucune installation de logiciel n'est nécessaire sur le poste client, un navigateur Web suffit.
- Evolutivité aisée des applications, vu qu'il n'est pas nécessaire de les déployer sur les postes clients.
- Possibilité d'accès à distance.

## **Chapitre1 : L'architecture J2EE**

· Compatibilité avec différents systèmes d'exploitation : Windows, Mac OS, Linux, Pocket PC ou téléphone mobile.

Pour saisir ces avantages, notre projet de fin d'études a été abordé dans le but de réaliser une application Web en J2EE pour la gestion des opticiens lunetterie.

Ce projet de fin d'études nous a été bénéfique car il nous a permis de bien nous familiariser à programmer en J2EE.

**Chapitre02:**  
**Présentation de la structure CNA**

## **Chapitre 2 : Présentation la structure CNAS**

### **Introduction**

Dans ce chapitre, nous allons voir les différents fonctionnements de la CNAS, et comment gérer leur travail

### **1. Définition**

CNAS « Caisse Nationale des Assurances Sociales » est un établissement

Conçu en conformité avec arrêté interministériel du 24 janvier 1987, portant organisation interne de la caisse nationale des assurances sociales, des accidents de travail et des maladies professionnelles.

### **2. Les missions de la CNAS**

La CNAS a pour mission, dans le cadre des lois et réglementent vigueur :

- ✓ De gérer les prestations nature et en espèces des assurances sociales, des accidents de travail et des maladies professionnelles.
- ✓ De gérer les prestations familiales.
- ✓ D'assurer le recouvrement des cotisations destinées au financement des prestations prévues par la CNAS.
- ✓ De contribué à promouvoir la politique de prévention des accidents de travail et des maladies professionnelles et de gérer le fond de cette prévention.
- ✓ De gérer les prestations dues aux personnes bénéficiaires des conventions et des accords internationaux de sécurité sociale.
- ✓ D'organiser, de coordonner et d'exercer le contrôle médical.
- ✓ D'entreprendre des actions sous forme de réalisation à caractère sanitaire et sociale.
- ✓ D'entreprendre des actions de prévention, d'éducation et d'information sanitaire après proposition du conseil d'administration de la caisse.
- ✓ De gérer le fond d'aide et de secours.
  - ✓ De rembourser les dépenses occasionnées par le fonctionnement des diverses commissions ou juridictions appelées à trancher suite à des litiges nés des décisions rendues par la caisse .

### **3. Présentation de la structure d'accueil**

#### **3.1. La structure d'accueil d'administration générale (Alger)**

Dans le cadre d'un stage pratique au sein de la caisse nationale des assurances sociales agence de Tiaret au niveau de la sous-direction des réalisations, des équipements et des moyens généraux et d'administration générale, je présente l'établissement d'accueil comme suit : [27]

Sous l'autorité du directeur générale, assisté du secrétaire général, le siège de la CNAS (Alger) comprend :

- ✓ La direction centrale des prestations.
- ✓ La direction centrale du recouvrement et du contentieux.
- ✓ La direction centrale du contrôle.
- ✓ La direction centrale du contrôle médical.
- ✓ La direction centrale de la prévention des accidents de travail et des maladies professionnelles.
- ✓ La direction centrale des études, des statistiques et de l'organisation.
- ✓ La direction centrale de l'informatique.
- ✓ La direction centrale des opérations financières.
- ✓ La direction centrale des réalisations, des équipements et des moyens généraux.
- ✓ La direction centrale des personnels.

##### **a) La direction centrale des prestations**

Est chargée :

- ✓ D'organiser et de suivre la gestion des prestations des assurés et les prestations dues au titre des congés payés.
- ✓ D'assurer le fonctionnement de la commission d'aide et de secours.
- ✓ De veiller à l'application des dispositions prévues par les accords bilatéraux de sécurité sociale et d'effectuer les apurements des comptes nés de l'application de ces accords.
- ✓ De suivre l'application des mesures arrêtées en matière de transfert pour soins à l'étranger, de centraliser les factures en ce domaine, d'effectuer les opérations de paiement au profit des établissements de soins et d'en établir des bilans périodiques.

## Chapitre 2 : Présentation de la structure CNAS

### b) La direction centrale du recouvrement et du contentieux

Est chargée :

- ✓ De suivre le recouvrement des cotisations.
- ✓ De veiller à ce que les assujettis respectent les obligations mises à leur charge.
- ✓ D'attribuer un numéro d'immatriculation national à chaque assuré social.
- ✓ De suivre les questions relatives aux contentieux général, technique et médical.
- ✓ D'assurer le fonctionnement de la commission de recours préalable.
- ✓ *c) La direction centrale du contrôle*

Est chargée d'effectuer des missions de contrôle :

- ✓ Sur l'état d'application de la législation et de la réglementation dans les domaines liés aux missions de la caisse.
- ✓ Sur les prestations payées.
- ✓ Sur le fonctionnement financier et comptable des agences de wilaya et autres annexes. et sur l'organisation et le fonctionnement des agences de wilaya et autres annexes.

### d) La direction centrale du contrôle médical

Est chargée :

- ✓ D'assurer le rôle de conseil médical auprès du siège.
- ✓ D'organiser et d'uniformiser le contrôle médical et d'en coordonner les activités.
- ✓ De participer, dans le cadre des lois et règlements en vigueur, à la commission technique à caractère médical.
- ✓ D'effectuer toute étude relative ; au barème des incapacités de travail, à la nomenclature générale des actes professionnels et à la liste des produits pharmaceutiques remboursables, aux appareillages et prothèses, aux tableaux des maladies professionnelles.

### e) La direction centrale de la prévention des accidents de travail et des maladies professionnelles

Est chargée :

- ✓ De contribuer à mettre en œuvre les mesures arrêtées en matière de prévention des accidents de travail et des maladies professionnelles.

## **Chapitre 2 : Présentation de la structure CNAS**

- ✓ D'élaborer et de proposer le programme d'action de la caisse en matière de prévention des accidents de travail et des maladies professionnelles.
- ✓ De gérer le fonds de prévention des accidents de travail et des maladies professionnelles.
- ✓ De centraliser et d'exploiter les enquêtes effectuées auprès des entreprises.
- ✓ D'organiser des séminaires de sensibilisation.

### **f) La direction centrale des études, des statistiques et de l'organisation**

Est chargée :

- ✓ D'effectuer des études et de faire des propositions en matière d'investissements.
- ✓ D'étudier, d'élaborer et de proposer des ratio-types de gestion.
- ✓ D'effectuer des études actuarielles.
- ✓ De collecter, de centraliser et de traiter les données et les informations statistiques.
- ✓ D'élaborer et de mettre en œuvre des programmes d'information en direction des assurés sociaux et des employeurs.
- ✓ De mettre en place des procédures d'information en direction des travailleurs de la caisse.
- ✓ De définir des méthodes d'organisation en vue d'unifier et d'homogénéiser les procédures et les documents, et de mettre en œuvre les méthodes arrêtées.
- ✓ De constituer et de gérer une documentation technique.

### **g) La direction centrale de l'informatique**

Est chargée :

- ✓ D'élaborer le plan informatique de la caisse et de mettre en œuvre le plan adopté.
- ✓ De mener les études informatiques et la réalisation des applications informatiques.
- ✓ De gérer le centre de calcul ainsi que l'ensemble des moyens informatiques.

### **h) La direction centrale des opérations financières**

Est chargée :

## **Chapitre 2 : Présentation de la structure CNAS**

✓ De préparer, en liaison avec les structures concernées, le projet de budget de la caisse.

✓ De tenir la comptabilité du siège et de centraliser celle des agences de wilaya et, le cas échéant, des antennes d'entreprise et des antennes d'administration.

✓ De veiller à la bonne exécution des opérations financières ; d'assurer la coordination financière.

### **i) La direction centrale des réalisations, des équipements et des moyens généraux**

Est chargée :

✓ De coordonner et de suivre la réalisation des investissements planifiés, et de suivre la gestion des investissements réalisés.

✓ D'arrêter les besoins en équipement pour l'ensemble des structures de la caisse et d'assurer l'acquisition et la gestion.

✓ De réaliser les opérations d'approvisionnement du siège en matière de fournitures, mobiliers et matériels de fonctionnement.

✓ De dresser les inventaires des biens meubles et immeubles de la caisse.

✓ De gérer les archives.

### **j) La direction centrale des personnels et de la formation**

Est chargée :

✓ D'assurer la gestion des personnels dans le cadre des dispositions législatives et réglementaires en vigueur.

✓ D'établir, en liaison avec les structures concernées, le plan de formation des personnels et d'organiser les actions de perfectionnement et de recyclage.

✓ D'établir les programmes de généralisation de l'utilisation de la langue nationale.

✓ D'étudier et de proposer les mesures nécessaires à l'amélioration des conditions de travail des personnels de la caisse.

✓ De suivre la gestion des œuvres sociales de la caisse.

## **3.2 Les Agences wilaya**

Sont chargées :

✓ D'organiser, de coordonner et de contrôler les activités des centres de communes et des antennes d'entreprise et, le cas échéant, d'antennes d'administration ;

## Chapitre 2 : Présentation de la structure CNAS

✓ D'assurer :

-Les services des prestations dues au titre des assurances sociales, des accidents de travail et des maladies professionnelles.

-Le service des prestations dues au titre des congés payés.

-Les actions mises à leur charge en matière de prévention des accidents de travail et des maladies professionnelles ;

✓ D'assurer le recouvrement des cotisations, le contrôle des obligations des assujettis et d'effectuer les opérations du contentieux du recouvrement des cotisations.

✓ D'exercer le contrôle médical.

✓ De tenir la comptabilité d'assurer l'exécution des opérations financières et leur coordination.

✓ D'assurer la gestion courante des moyens matériels et humains de l'agence.

Les agences de wilaya sont classées en trois catégories :

-1<sup>ère</sup> catégorie : agences gérant au moins 200.000 assurés sociaux.

-2<sup>ème</sup> catégorie : agences gérant moins de 200.000 et au moins 100.000 assurés sociaux.

-3<sup>ème</sup> catégorie : agences gérant moins de 100.000 assurés sociaux.

### **a) L'agence de première catégorie**

Comprenant cinq sous-structures chargées, respectivement :

-Des prestations, dont les tâches sont réparties entre deux à quatre responsables de gestion.

-Du recouvrement et du contentieux, dont les tâches sont réparties entre trois responsables de gestion.

-Des opérations financières, dont les tâches sont réparties entre deux responsables de gestion.

-De l'administration des moyens et des réalisations à caractère sanitaire et social, dont les tâches sont réparties entre deux ou trois responsables de gestion.

-Du contrôle médical dirigé par un médecin.

### **b) L'agence de la deuxième catégorie**

Comprenant quatre sous-structures chargées, respectivement :

-Des prestations, dont les tâches sont réparties entre deux, trois ou quatre responsables de gestion.

-Du contrôle médical dirigé par un médecin.

## Chapitre 2 : Présentation de la structure CNAS

### c) L'agence de la troisième catégorie

Comprenant quatre sous-structures chargées, respectivement :

- ✓ Des prestations.
- ✓ Des opérations financières, du recouvrement et du contentieux.
- ✓ De l'administration des moyens et des réalisations à caractère sanitaire et social.
- ✓ Du contrôle médical dirigé par un médecin.

#### 3.2.1. L'agence de la wilaya de Tiaret [24]

On peut la classer dans la deuxième catégorie.

La structure d'accueil de l'agence de la wilaya de Tiaret

Le siège de la caisse nationale des assurances sociales de la wilaya de Tiaret contient :

9 centres de paiement (gérant chacun plus de 10000 assurés) et 02 antennes (gérant chacune moins de 10000 assurés) classés comme suit :

Au niveau de la wilaya il y a trois centres (le siège de la CNAS et le centre d'AMMOR AEK, azzedine el habib).

- Un centre à Sougueur.
- Un centre à Frenda.
- Un centre à Mahdia.
- Un centre à Rahouia.
- Un centre à Ain dheb.
- Une antenne à Ksar echalalla.
- Une antenne à Ain kermes.

Tous ces centres sont gérés à partir du siège (Tiaret), ce dernier est constitué de quatre sous-directions et un centre de calcul.

##### a) La sous-direction chargée de la gestion du centre de calcul

Techniquement, le centre de calcul contient deux serveurs, le premier, appelé SECU, est chargé de gérer les informations sur :

- L'immatriculation assurée, l'immatriculation employeurs, l'affiliation et les cotisants.
- Le traitement des AF (allocation familiale) et des AT (accident de travail).

Le deuxième serveur est chargé de gérer la base de données pour les prestations nature et en espèce.

Chaque centre de paiement est relié avec le centre de calcul par un réseau intranet.

## **Chapitre 2 : Présentation de la structure CNAS**

### **b) La sous-direction du contrôle médical**

Sous l'autorité d'un médecin chef, la sous-direction du contrôle médicale gère dix services (un service dans chaque centre de paiement). Chaque service contient au moins deux personnes qui sont le médecin conseil et le secrétaire pré-médical

Trois types d'information circulent dans cette sous-direction :

-Les informations médicalisées gérées par les médecins conseil.  
-Les informations administratives gérées par les secrétaires pré-médicaux et les informations médico-administratives gérées par les médecins conseil.

-La sous-direction de contrôle médical est chargée de gérer :

Les assurances maladie (congé maladie, les dossiers de soin « les ordonnances et les actes médicaux »).

Les assurances accident de travail et les maladies professionnelles.

### **c) La sous-direction des prestations**

Sous l'autorité d'un sous-directeur la sous-direction des prestations est constituée comme suit :

- Le secrétariat.
- Service d'invalidité.
- Service d'AT.
- 10 centres de paiement.
- Service statistique et le service de contractualisation hôpitaux cliniques.
- Service CHIFA.

Le rôle de cette sous-direction est de rembourser les prestations en espèce et en nature.

### **d) La sous-direction des réalisations, des équipements et des moyens généraux**

Sous l'autorité d'un sous-directeur, la sous-direction des réalisations, des équipements et des moyens généraux est constituée de trois services (le secrétariat, le personnel et le service des moyennes généraux, ce dernier contient le service de comptabilité, le magasin et le parc automobile). Cette sous-direction est chargée :

- ✓ D'assurer la gestion des personnels et de suivre la gestion des œuvres sociales de la CNAS.
- ✓ De réaliser et de suivre les investissements planifiés.
- ✓ D'arrêter les besoins en équipement pour l'ensemble des structures de la caisse et d'assurer l'acquisition et la gestion.
- ✓ De dresser les inventaires des biens meubles et immeubles de la caisse.
- ✓ De gérer les archives.

## Chapitre 2 : Présentation de la structure CNAS

### e) La sous-direction de finance et recouvrement

Sous l'autorité d'un sous-directeur, la sous-direction de finance et recouvrement est constituée de quatre services (CRP, service contentieux, service cotisant et service contrôle employeurs). La sous-direction de finance et recouvrement est chargée :

- ✓ De suivre le recouvrement des cotisations.
- ✓ D'attribuer un numéro d'immatriculation à chaque assuré social.
- ✓ De suivre les questions relatives aux contentieux général, technique et médical.
- ✓ D'étudier les recours.

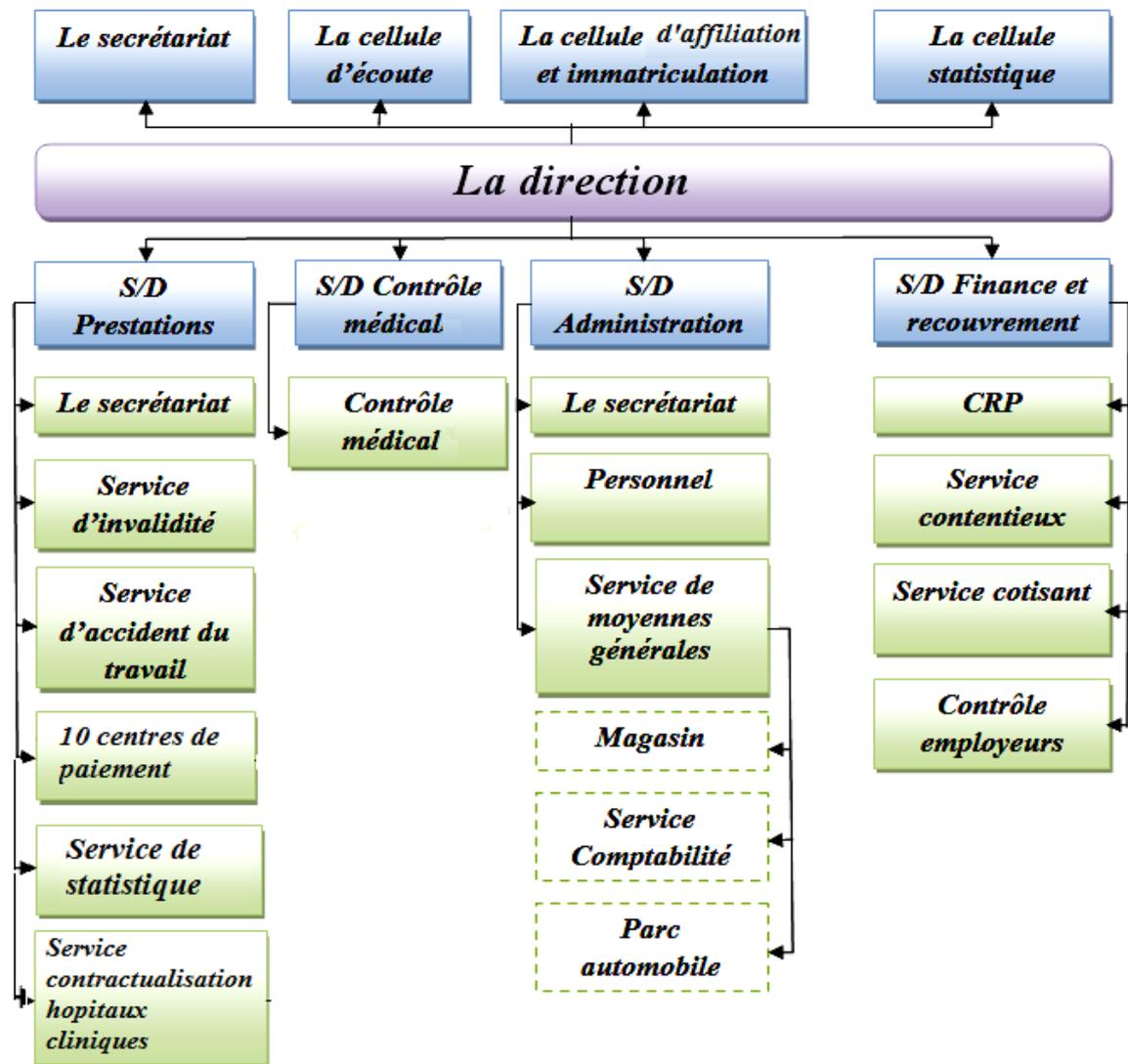


Figure 2.1 : Organigramme de la CNAS (Agence-Tiaret).

### 4. Présentation Générale du système CNAS

CNAS est un système Offrant aux différentes personnes le droit d'assurance dans le cas des accidents de travail ou de maladie professionnelle.

#### 4.1. L'assuré Sociale

Tout personne appartiennent à l'un des catégories suivante est un assuré social.

##### 4.1.1. Les catégories particulières avec professionnelle

- Les travailleurs salariés quel que soit le secteur d'activité auquel ils appartiennent

- Les travailleurs assimilés à des salariés.

- Les travailleurs à domicile.

- Les personnes employées par des particuliers.

- Les artistes, comédiens et figurants de théâtre, de cinéma et autres établissements de spectacles.

- Les marins pêcheurs à la part, embarqués.

- Les patrons pêcheurs à la part, embarqués.

Cette catégorie a le droit de : maternité ; congé de maladie ; Pensions d'invalidité et capital décès.

##### 4.1.2. Les catégories particulières sans activité professionnelle

- Les étudiants.

- Les moudjahidines et victimes de la guerre de libération nationale.

- Les handicapés.

- Les porteurs de bagages occupés dans les gares autorisées.

- Les gardiens de parkings non-payants, autorisés.

##### 4.1.3. Les ayants droit

Les ayants-droit de l'assuré social bénéficient des prestations en nature :

- Des assurances maladie et maternité pour le conjoint.

- De l'assurance maladie pour les enfants et les ascendants à charge.

- Le conjoint de l'assuré, s'il n'est pas lui-même assuré au titre de sa propre activité professionnelle.

- Les enfants à charge, au sens de la réglementation de la sécurité sociale, âgés de moins de 18 ans.

## **Chapitre 2 : Présentation de la structure CNAS**

- Une rémunération inférieure à la moitié du salaire national minimum garanti.
- Les enfants de moins de 21 ans qui poursuivent leurs études.
- Les enfants à charge et les collatéraux au troisième degré à charge, de sexe féminin, sans revenu, quel que soit leur âge.
- Les enfants quel que soit leur âge, qui sont par suite d'infirmité ou de maladie chronique.
- Les ascendants de l'assuré ou de son conjoint.

### **4.2. Système CHIFA**

C'est un projet novateur car reposant sur le recours à une réelle technologie, la carte CHIFA a été conçue et introduite dans le système des assurances sociales dans le cadre de l'amélioration et de la modernisation de la gestion de la CNAS.

La carte CHIFA est pour l'objet d'améliorer la qualité des prestations en direction des assurés par la simplification des procédures et par le remboursement systématique dans des délais acceptables, d'améliorer les relations avec les prestataires et de maîtriser la gestion.

#### **4.2.1. Les composants du système**

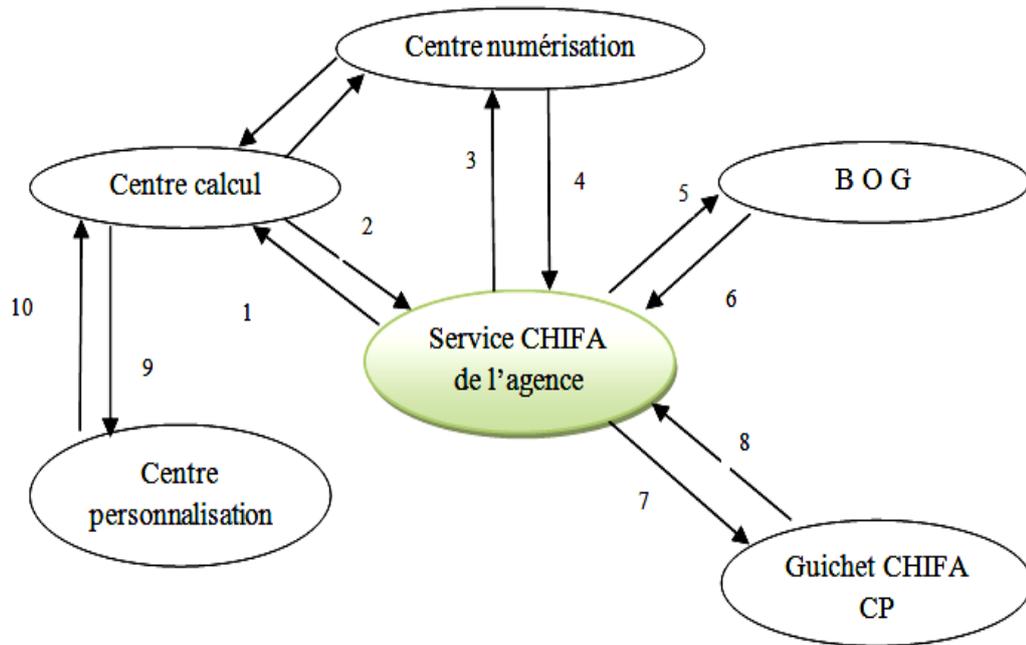
Système CHIFA consiste en la mise en relation via :

- ✓ Des procédures organisationnelles.
- ✓ Un réseau de télétraitement.
- ✓ Des logiciels adaptés.
- ✓ Des équipements.
- ✓ Des acteurs et partenaires du système de sécurité sociale qui sont :
  - Les assurés sociaux.
  - Les professionnels de la santé.
  - Les organismes de sécurité sociale.

#### **4.2.2. Structures intervenantes de la CNAS**

- ✓ Le centre de calcul.
- ✓ Le service CHIFA de l'agence.
- ✓ Le centre de numérisation et de personnalisation.
- ✓ La structure CHIFA de centre payeur.

## Chapitre 2 : Présentation de la structure CNAS



*Figure 2.2 : Schéma des relations inter structurent.*

### 4.2.3. Les procédures d'enrôlement des assurés

Elle s'effectue en trois étapes :

#### Phase 1

- ✓ La sélection et l'extraction des fichiers des assurés sociaux.
- ✓ L'édition des conventions et leur envoi aux assurés sociaux.
- ✓ La réception des assurés par la structure CHIFA et le traitement des dossiers.
- ✓ Le transfert des lots de dossiers de la cellule CHIFA du centre de paiement vers les services CHIFA de l'agence.

#### Phase 2

- ✓ La réception des lots par le service CHIFA de l'agence.
- ✓ La saisie des données de l'assuré social.
- ✓ Le transfert des lots physiques et fichiers électronique au centre de numérisation et personnalisation.
- ✓ Le traitement des demandes par le centre de personnalisation.

#### Phase 3

- ✓ La réception des cartes par le service CHIFA.

## **Chapitre 2 : Présentation de la structure CNAS**

- ✓ L'édition des avis de mise à disposition des assurés sociaux des cartes CHIFA.
- ✓ L'envoi des avis de mise à disposition des cartes CHIFA aux assurés sociaux.
- ✓ L'envoi des cartes CHIFA aux structures CHIFA des centres payeurs.
- ✓ Le retrait des cartes CHIFA par les assurés sociaux auprès des structures CHIFA.

### **5. La convention liant la CNAS aux opticiens lunettiers :**

Dans le cadre de la mise en œuvre du système du tiers payant –opticien lunettiers prévu par le décret exécutif n° 12-159 du 01 avril 2012, portant passation d'une convention liant la CNAS aux opticiens lunettiers, destinée à la prise en charge des frais d'acquisition et de renouvellement des montures de lunettes et les verres correcteurs de la vue au profit des enfants d'âge scolaire et prés scolaire . A cet effet, les enfants bénéficiaires doivent répondre aux conditions suivantes :

#### **a. Conditions d'âge :**

- ✓ Enfant d'âge préscolaire (moins de six ans).
- ✓ Enfant d'âge scolaire (six ans et plus).

#### **b. Condition médicale : il s'agit des enfants souffrant :**

- ✓ D'amblyopie,
- ✓ De vices de réfraction
- ✓ De strabisme.

#### **c. Condition de revenu de l'assuré social :**

Le revenu de l'assuré social concerné ne doit pas excéder vingt mille (20.000) dinars par mois.

### **5.1. Prestations couvertes :**

Les prestations couvrent les frais d'acquisition et de renouvellement des montures et des verres correcteurs de la vue.

## **Chapitre 2 : Présentation de la structure CNAS**

### **5.2. Les frais d'acquisition des montures et des verres correcteurs**

L'acquisition des montures et des verres n'est soumise à aucune formalité particulière, si ce n'est celle de fournir la prescription médicale établie par le médecin ophtalmologue, et ce, quel que soit l'âge de l'enfant bénéficiaire (moins de six ans, six ans et plus), aucune distinction, n'est à cet égard établie.

### **5.3. Les frais de renouvellement des montures et des verres :**

Une distinction doit être faite selon que l'enfant bénéficiaire est âgé de moins de six (6) ans ou de six (6) ans et plus.

- Enfant âgé de moins de six (6) ans :

Le renouvellement des montures et des verres correcteurs de la vue s'effectue sur simple présentation d'une prescription médicale délivrée par un médecin ophtalmologue.

- Enfant âgé de six (6) ans et plus :

Dans ce cas de figure, la solution diffère selon que ce renouvellement porte sur les verres correcteurs uniquement ou sur les montures et les verres correcteurs à la fois.

#### **5.3.1. Le renouvellement des verres correcteurs :**

Il se fait sur la base de la prescription médicale du médecin ophtalmologue.

#### **5.3.2 Le renouvellement des montures et des verres correcteurs :**

Il est subordonné à trois conditions supplémentaires, à savoir :

- ✓ Si la monture et les verres correcteurs sont devenus inadaptés à l'enfant
- ✓ Si la formalité de l'entente préalable a été respectée ;
- ✓ Si le même enfant n'a pas déjà bénéficié au cours de la même année d'un précédent renouvellement (celui-ci étant limité à une fois par an).

### **5.4 .Logiciel chifa – optic :**

La CNAS met à la disposition de l'opticien lunettier une clé électronique de professionnel de la santé et le logiciel permettant l'utilisation de la carte Chifa. Elle doit intégrer dans ce logiciel les éléments d'information suivants :

- La liste des montures et des verres correcteurs de la vue pris en charge par la CNAS ;
- Leurs montants ;

## **Chapitre 2 : Présentation de la structure CNAS**

- La liste des assurés sociaux bénéficiaires du dispositif du tiers payant en matière de lunettes de vue.

Par ailleurs, elle doit également :

Mentionner dans le contenu de la carte électronique Chifa le taux de prise en charge en matière de sécurité sociale de chaque enfant bénéficiaire.

Assurer la mise à jour régulière du logiciel.

Assurer la maintenance du système Chifa en permanence.

## **6. Conclusion**

Dans ce chapitre, nous avons essayé de décrire la situation actuelle de la CNAS afin de comprendre son fonctionnement général, et on a parlé sur La convention liant la CNAS aux opticiens lunettiers

# Chapitre3 : Modélisation et réalisation

## Introduction

La réalisation d'une application web doit être impérativement précédée d'une méthodologie d'analyse et de conception qui a pour objectif de permettre de formaliser les étapes préliminaires du développement d'un site afin de rendre ce développement plus fidèle aux besoins du client. La phase d'analyse permet de lister les résultats attendus, en termes de fonctionnalités et La phase de conception permet de décrire de manière non ambiguë, le plus souvent en utilisant un langage de modélisation, le fonctionnement futur du système, afin d'en faciliter la réalisation. . [9]

Dans ce chapitre, nous allons modéliser notre application web en utilisant un langage de modélisation objet qui est UML. La conception de notre application se base sur les diagrammes des cas d'utilisation(les acteurs, procédures d'administrateur), les diagrammes de classes, les diagrammes de séquences et les diagrammes d'activité. [10]

## 1. présentation de l'UML

### 1.1. Définition :

UML (Unified Modeling Language), se définit comme un langage de modélisation graphique et textuel destiné à comprendre et à définir des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. UML modélise l'ensemble des données et des traitements en élaborant des différents diagrammes. En clair, il ne faut pas designer UML en tant que méthode mais plutôt comme une boîte d'outils qui sert à améliorer les méthodes de travail. [12]

### 1.2. Les avantages et les inconvénients d'UML :

#### 1.2.1. Les points forts d'UML :

- UML est un langage formel et normalisé :

Il permet le gain de précision, encourage l'utilisation d'outils et constitue à cet effet un gage de stabilité. [9]

## **Chapitre3 : Modélisation et réalisation**

- UML est un support de communication performant :

Il cadre l'analyse et facilite la compréhension de représentations abstraites complexes. Son caractère polyvalent et sa souplesse en font un langage universel. [11]

### **1.2.2. Les points faibles d'UML :**

La mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation.

La nécessité de s'accorder sur des modes d'expression communs est vitale en informatique. UML n'est pas à l'origine des concepts objets, mais en constitue une étape majeure, car il unifie les différentes approches et en donne une définition plus formelle. [12]

### **1.3. Présentation générale des diagrammes d'UML**

Un modèle est une représentation simplifiée d'un problème .UML permet d'exprimer les modèle objet à travers un ensemble à de diagramme. Ces derniers sont des moyens de description des objets ainsi des liens qui les relient.

Un diagramme est une représentation graphique qui s'intéresse à un aspect précis du modèle .UML offre 12 types de diagramme .chaque type de diagramme offrent une vue complète du système. [12][10]

Les diagrammes UML peuvent être classés sous deux grandes catégories [12] :

- Les diagrammes de structure ou statique
- Les diagrammes de comportement ou dynamique

#### **1.3.1. Représentation statique du système (structurel) :**

- Le diagramme de classes.
- Le diagramme d'objets.
- Le diagramme de composants
- Le diagramme de déploiement.
- Le diagramme de packages.
- Le diagramme de CAS d'utilisation.
- Le diagramme de structure composite.

#### **1.3.2. Représentation dynamique du système (comportemental):**

- Le diagramme d'activité.
- Le diagramme de séquence.
- Le diagramme d'état-transition.

## Chapitre3 : Modélisation et réalisation

- Le diagramme de collaboration.
- Le diagramme de communication.

Pour la modélisation des besoins de notre système, nous utilisons les diagrammes UML suivant :

Diagramme de cas d'utilisation, diagramme de séquence, et diagramme d'activité ainsi que le diagramme de classe.

### 1.4. Diagramme de cas d'utilisation:

Le Diagramme de CAS d'utilisation est le premier diagramme du modèle UML utilisé pour la modélisation des besoins des utilisateurs. Les cas d'utilisations décrivent le comportement du système étudié du point de vue de l'utilisateur, et décrivent les possibilités d'interactions fonctionnelles entre le système et les acteurs, ils permettent de définir les limites et les relations entre le système et son environnement. Il est destiné à structurer les besoins des utilisateurs et les objectifs par rapport au système. C'est donc l'image d'une fonctionnalité en réponse à la simulation d'un acteur externe. [9]

Il s'agit de la solution UML pour représenter le modèle conceptuel. [13]

- Les cas d'utilisations permettent de structurer les besoins des utilisateurs et les objectifs correspondants d'un système.
- Ils centrent l'expression des exigences du système sur ses utilisateurs: ils partent du Principe que les objectifs du système sont tous motivés.
- Ils se limitent aux préoccupations "réelles" des utilisateurs.
- Ils identifient les utilisateurs du système (acteurs) et leur interaction avec le système.
- Ils permettent de classer les acteurs et structurer les objectifs du système.
- Ils servent de base à la traçabilité des exigences d'un système dans un processus de développement intégrant UML.

#### 1.4.1. Identification des acteurs :

Un acteur est une entité externe qui agit sur le système, il peut consulter l'état du système en mettant ou en recevant des messages susceptibles d'être porteurs de données. [13]

Dans notre application, on à Un seul acteur c'est l'opticien qui interagit avec le système.

## Chapitre3 : Modélisation et réalisation

### 1.4.2. Identification des cas d'utilisations:

Un cas d'utilisation centrent l'expression des exigences du système sur ces utilisateurs ils se limitent aux préoccupations « réelles » des utilisateurs; ils ne présentent pas de solutions d'implémentation et ne forment pas un inventaire fonctionnel du système. Ils identifient les utilisateurs du système et leur interaction avec celui-ci. C'est est un ensemble de séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable pour un acteur particulier du système, il permet de décrire ce que le futur système devra faire sans spécifier comment il le fera.[13]

#### ✓ **Voici les cas d'utilisation de notre application :**

##### **Authentication:**

L'application vérifie que c'est bien l'utilisateur (opticien) qui veut utiliser le système et lui donne ensuite l'autorisation d'accès, et ouvrir a lui une session.

##### **Vérification d'affiliation d'assuré :**

L'utilisateur (opticien) entre le numéro d'assurance ou le nom et prénom pour vérifier l'existence d'assuré. S'il existe le système affiche leur information, ainsi que leur ayant-droit.

##### **Consultation des ayant-droit :**

Si les ayant-droit existe, une table s'affiche contient une liste de ces derniers, l'utilisateur peut choisit l'ayant droit concerné, après la consultation de ces informations, il vérifier les conditions dans les quels il peut bénéficier de cet offre.

##### **Création et impression des factures :**

L'utilisateur remplit la facture selon des critères bien définis, et enfin imprimer la facture .

##### **Déconnexion :**

L'utilisateur ferme la session.

### ✓ La présentation de notre diagramme de cas d'utilisation :

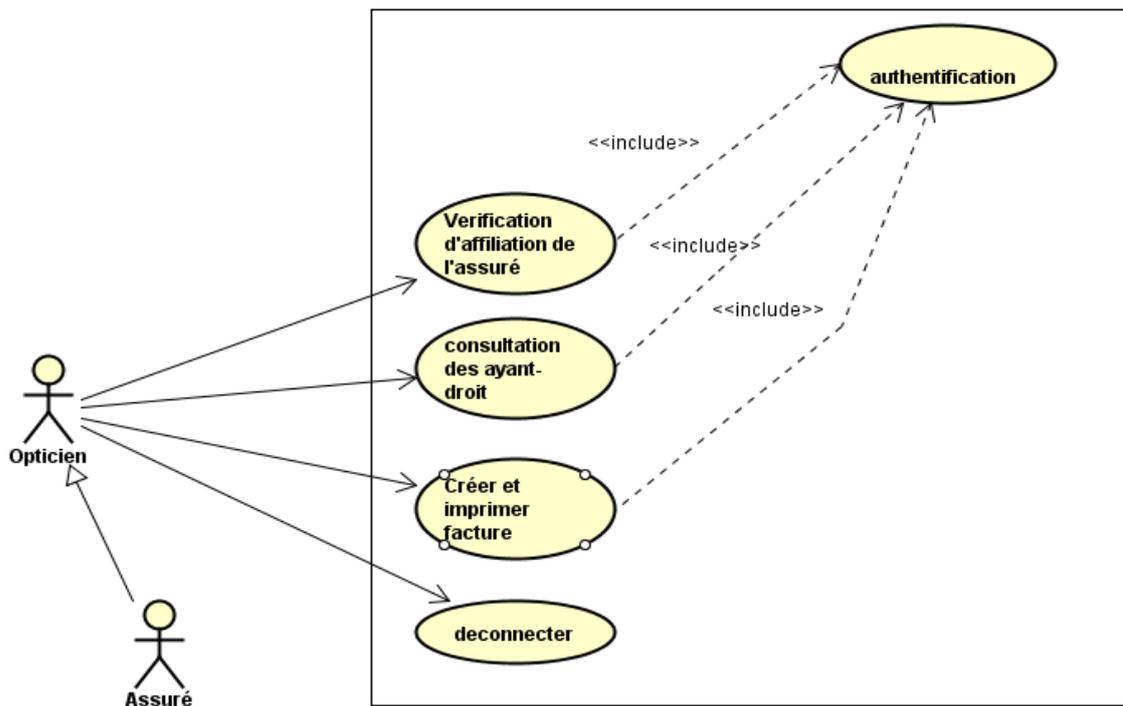


Figure 3.1- Diagramme de CAS d'utilisation

### 1.5. Diagramme de séquence:

Le diagramme de séquence suit le diagramme de CAS d'utilisation car il le complète. Il permet de décrire les scénarios (déroulement des traitements entre les éléments du système et les acteurs) de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets. En particulier, il montre aussi les objets qui participent à l'interaction par leur "ligne de vie" et les messages qu'ils échangent présentés en séquence dans le temps. Voici quelques notions de base du diagramme : **[14]**

- Scénario: une liste d'actions qui décrivent une interaction entre un acteur et le système.
- Interaction: un comportement qui comprend un ensemble de messages échangés par un ensemble d'objets dans un certain contexte pour accomplir une certaine tâche.
- Message: Un message représente une communication unidirectionnelle entre objets qui transporte de l'information avec l'intention de déclencher une réaction chez le récepteur.

La présentation de notre diagramme de séquence :

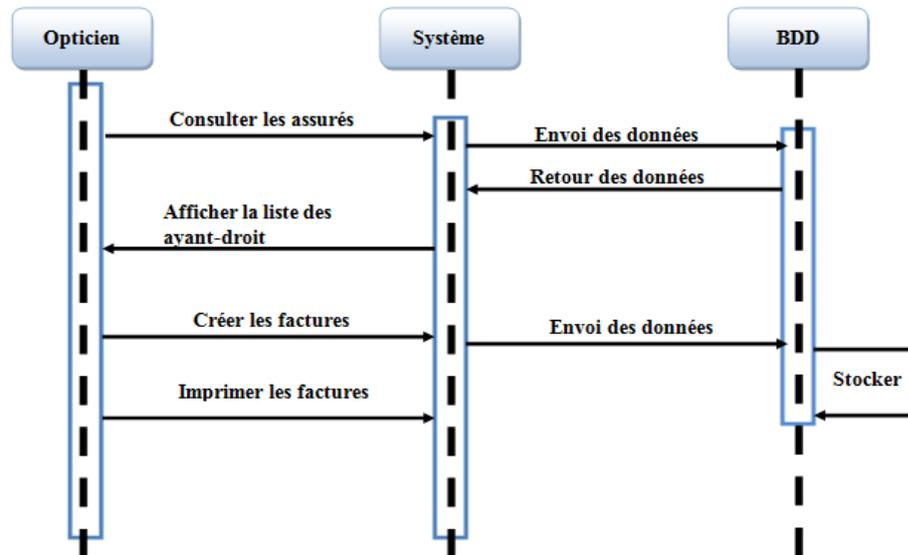


Figure 3.2 - Diagramme de séquence

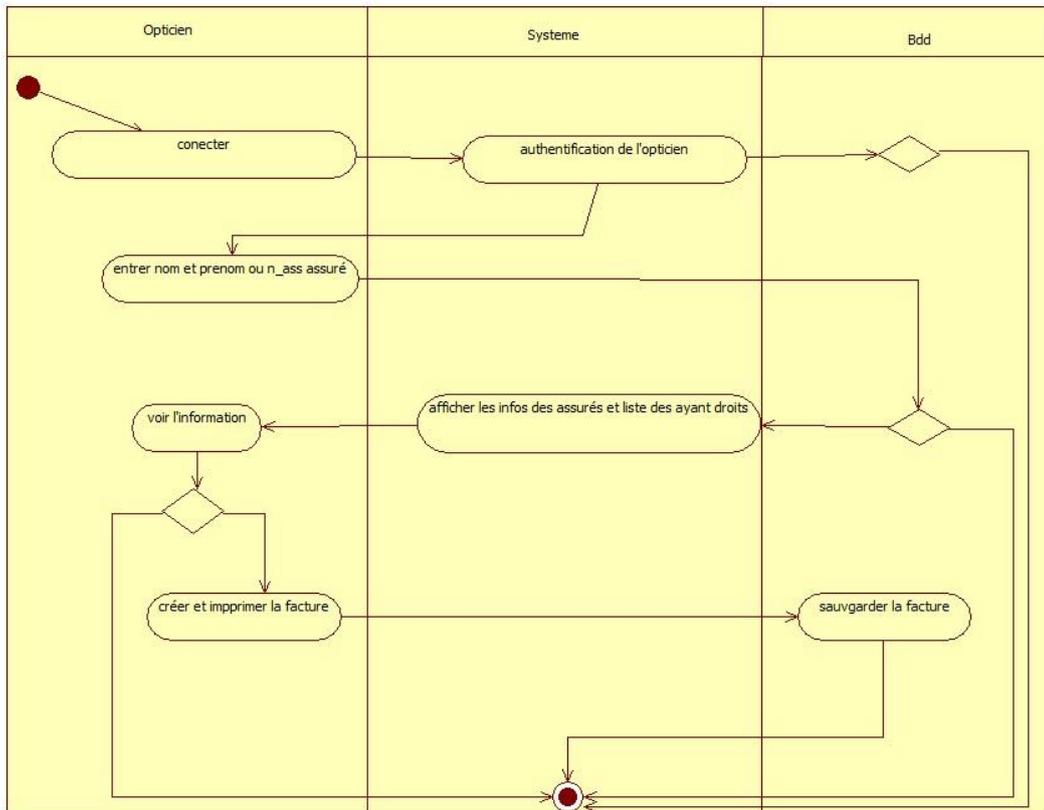
### 1.6. Diagramme d'activité :

Il donne une vision des enchaînements des activités propre à une opération ou à un cas d'utilisation. [15]

Un diagramme d'activités permet de mettre l'accent sur les traitements et est particulièrement adapté à la modélisation du cheminement de flots de contrôle et de flots de données. Il indique la part prise par chaque objet dans l'exécution d'un travail. Il sera enrichi par les conditions de séquence. Dans la phase de conception, les diagrammes d'activités sont particulièrement adaptés à la description des cas d'utilisation en venant, par exemple, illustrer et consolider leur description textuelle. Ainsi leur représentation est sous forme d'organigrammes, ce qui les rend utiles dans la phase de réalisation. [16]

## Chapitre3 : Modélisation et réalisation

### La présentation de notre Diagramme d'activité :



*Figure 3.3- Diagramme d'activité*

### 1.7. Diagramme de classes :

Le diagramme de classe constitue un élément très important de la modélisation : il permet de définir quelles seront les composantes du système final. [15]

Il représente les classes intervenant dans le système. Une classe décrit les responsabilités, le comportement et le type d'un ensemble d'objets, les éléments de cet ensemble sont les instances de la classe.

#### 1.7.1 Son utilisation :

Le diagramme de classe est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces d'un système ainsi que les différentes relations entre celle-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques.

## Chapitre3 : Modélisation et réalisation

### 1.7.2 Identification des classes :

Une classe est une description d'un groupe d'objets partageant un ensemble commun de propriétés (les attributs), de comportements (les opérations) et de relations avec d'autres objets (les associations et les agrégations). [13]

Une classe contient : [16]

- Des attributs :(ou champs, ou variables d'instances) : Les attributs d'une classe est une caractéristique d'un objet, décrivent la structure de ses instances (les objets). Un attribut souligné correspond à un attribut de classe.
- Des méthodes:(ou opérations de la classe) : Les méthodes décrivent les opérations qui sont applicables aux instances de la classe. C'est un service dont un objet peut demander l'exécution. La Multiplicité: sert à compter le nombre minimum et maximum de possibilité que chaque classe contient dans la relation liant deux ou plusieurs classes. Une agrégation : Il s'agit d'une relation entre deux classes, spécifiant que les objets d'une classe sont des composants de l'autre classe. C'est donc une association qui, lorsqu'elle est lue dans un sens signifie "est une partie de" et lorsqu'elle est lue dans l'autre sens elle signifie "est composé de".

### La présentation de notre Diagramme de classe :

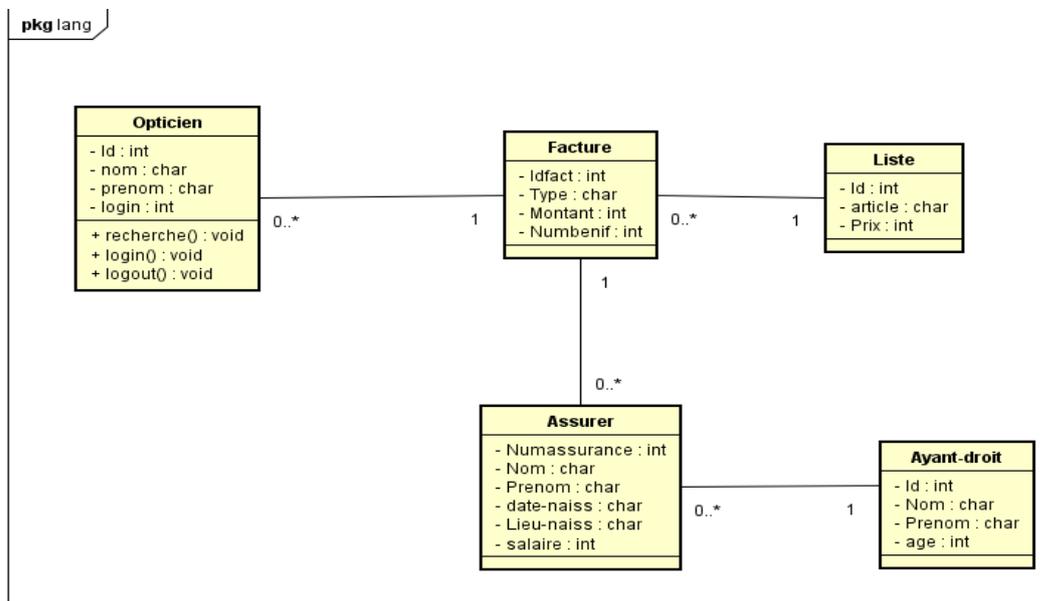


Figure 3.4- Diagramme de classes

### 2. Réalisation de l'application :

Dans cette partie de chapitre, nous allons présenter les outils choisis pour la réalisation de notre application, ainsi les autres aspects mis en place et quelques captures d'écrans de notre outil.

Nous avons utilisée le langage de programmation Java, l'IDE netbeans 8.0.2 et comme base de données Java DB intégrer dans Netbeans. Avant d'entamer la phase réalisation, nous allons présenter les technologies et les outils utilisés durant cette phase.

#### 2.1. Langage de Programmation Java :

Le langage Java est un langage généraliste de programmation synthétisant les principaux langages existants lors de sa création en 1995 par Sun Microsystems .Il permet une programmation orientée-objet, modulaire et reprend une syntaxe très proche de celle du langage C.

Outre son orientation objet, le langage Java a l'avantage d'être modulaire (on peut écrire des portions de code génériques, c.-à-d. utilisables par plusieurs applications), rigoureux (la plupart des erreurs se produisent à la compilation et non à l'exécution) et portable (un même programme compilé peut s'exécuter sur différents environnements). En contrepartie, les applications Java ont le défaut d'être plus lentes à l'exécution que des applications programmées en C par exemple. [17]

#### 2.2. Netbeans

L'EDI NetBeans est un environnement de développement, un outil pour les programmeurs pour écrire, compiler, déboguer et déployer des programmes. Il est écrit en Java - mais peut supporter n'importe quel langage de programmation. Il y a également un grand nombre de modules pour étendre l'EDI NetBeans. [18]



*Figure 3.5: logo netbeans*

### **2.3. La base de donnée java DB :**

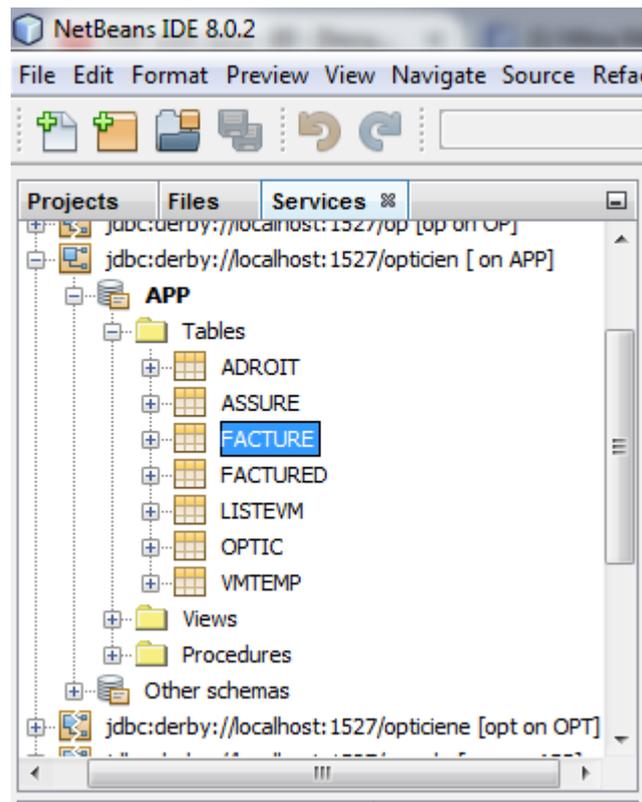
#### **2.3.1. Définition :**

Java DB est pris en charge la distribution de Oracle de la base de données open source Apache Derby. Il prend en charge la norme SQL ANSI / ISO à travers les JDBC et API Java EE. Java DB est inclus dans le JDK .

#### **2.3.2. Avantages :**

- Complet et facile à utiliser
- Transaction protégée et crash-recouvrable
- Intégrable dans les applications
- Inclus dans le JDK
- Compact (2,6 Mo)

## Chapitre3 : Modélisation et réalisation



*Figure 3.6 : Notre base de données*

### 2.4. Présentation de l'application :

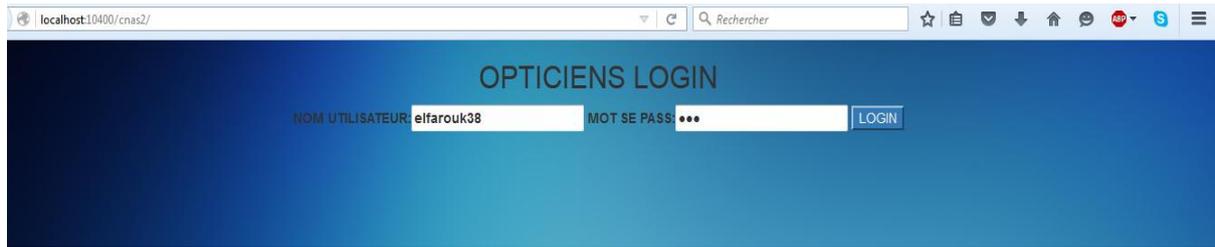
Nous présentons dans cette section les principales fonctionnalités de l'outil application à travers des captures écran illustrant les interfaces de ce dernier. Pour bien décrire ces interfaces, nous considérons un scénario d'exécution, à partir de déploiement de l'application, jusqu'à l'impression.

#### 2.4.1. Authentification :

Cette page la (figure 3.7) constitue la page d'authentification de notre application. A travers cette fenêtre l'opticien s'authentifie impérativement pour entrer dans le site.

Cette étape met en valeur l'aspect sécurité : nous verrons la disponibilité du compte utilisateur et nous lui attribuons les droits et les privilèges nécessaires, pour qu'à la suite on lui affiche les fonctionnalistes sur lesquelles il a les droits d'accès.

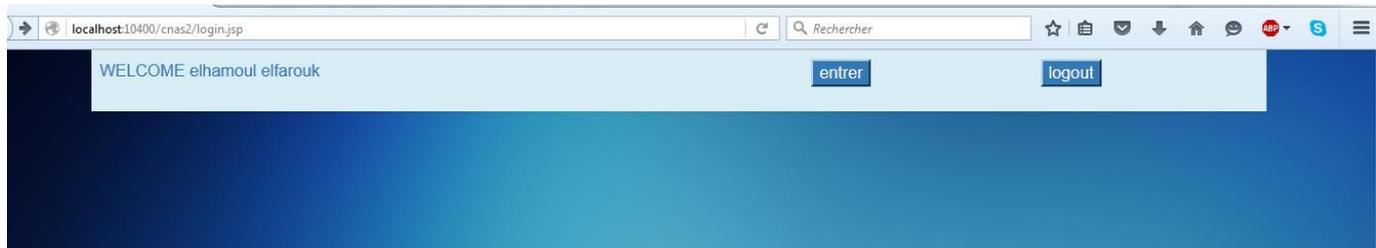
## Chapitre3 : Modélisation et réalisation



*Figure 3.7 : page d'authentification*

### 2.4.2 : page d'accueil :

Cette page est une intermédiaire, elle vient après l'authentification, qui permet à l'opticien d'accéder à la page d'assuré à travers le bouton entrer, ou fermer la session on click sur le bouton log out.



*Figure 3.8 : page d'accueil*

### 2.4.3 : Page assuréInfo :

Comme apparu dans l'écran ci-dessous (figure 3.9), cette page permet à l'opticien de faire la recherche des clients assuré et leur ayant-droit avec deux options :

- La recherche par nom et prénom d'assuré.
- La recherche par le numéro d'assurance.

La sélection d'un ayant-droit envoie l'opticien à une autre page.

## Chapitre3 : Modélisation et réalisation

NUM  
le numero d'assurance      le nom d'assure      le prenom d'assure      Logout

search

assure id: 2  
first name: bentamra  
last name: hichem  
adresse: rue ibn khaldoun  
salaire: 26540  
daten: 1980-10-05  
lieun: tiaret

NUMERO	FirstName	LastName	Age	
2	bentamra	oussama	12	4

*Figure 3.9 : Page assuréInfo*

### 2.4.4. Page d'ayant-Droit :

Cette page contient des informations d'ayant-droit sélectionné dans la page précédente et une liste déroulante contient les types du verre et montures et leurs prix.

A partir de cette liste l'utilisateur remplit la facture puis l'enregistrer on clique sur le bouton imprimer

## Chapitre3 : Modélisation et réalisation

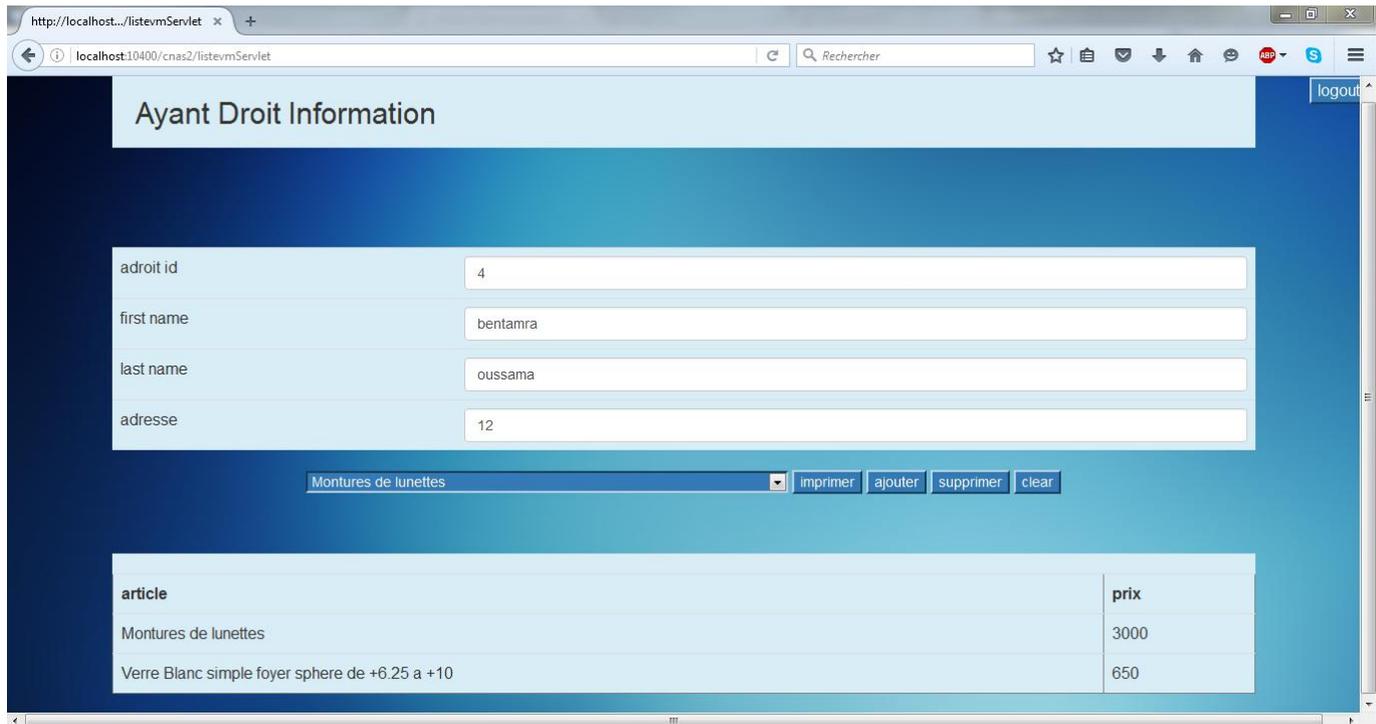


Figure 3.10 : Page d'ayant-droit

### 2.4.5. L'impression :

A partir de cette liste l'utilisateur remplit la facture puis l'enregistre on clique sur le bouton imprimer on va à la page facture, depuis cette dernière nous pouvons télécharger la facture sous forme d'un document PDF d'une seule page.

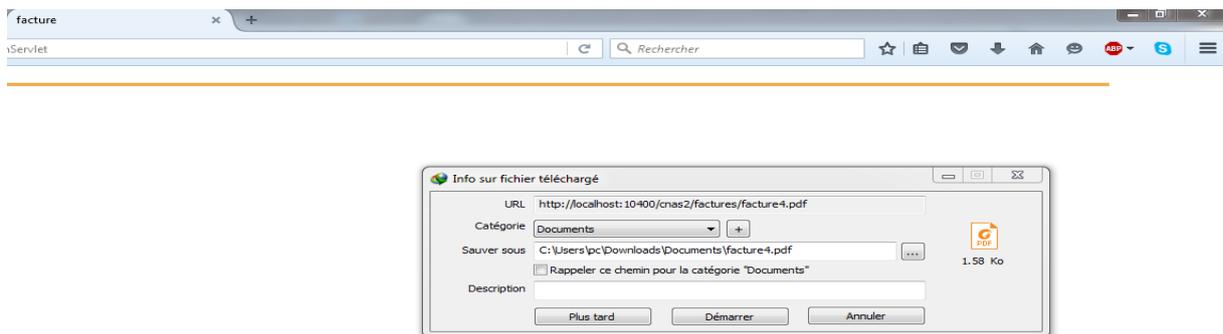


Figure 3.11 : Page assuré Info

## Chapitre3 : Modélisation et réalisation

Fri Jun 17 16:05:56 WAT 2016

Numero d'assurance : 4  
Nom Benaïdissier : bentamra  
prenom benaïdissier : oussama  
Nom Opticien : elhamoulprenom opticien : elfarouk

**FACTURE n:4**

NUM	CODE	ARTICLE	PRIX
0	1	Montures de lunettes	3000
1	2	Verre Blanc simple foyer sphere de -6 a +6	300
2	4	Verre Blanc simple foyer sphere de +6.25 a +10	650
		Total =3950	Total CL=86

Signature :

*Figure 3.12 : la facture*

## Conclusion

Ce chapitre était consacré à l'analyse et la conception de notre application en utilisant la modélisation UML. On a réalisé ces démarches qui concernent les fonctionnalités et les traitements de chaque partie de notre application ou chacune de ces parties est illustrée par un ou plusieurs diagrammes appropriés.

Dans ce chapitre, nous avons abordé l'implémentation et le déploiement de notre Application web. Les fonctionnalités de notre application répondent aux problématiques évoquées dans le chapitre I.

## **Conclusion générale**

### **Conclusion générale**

Une application répartie est constituée d'un ensemble de processus (d'objets, d'agents, d'acteurs) s'exécutant sur plusieurs sites et reliés par un réseau de communication. Ces processus communiquent entre eux via un réseau de communication pour remplir une fonction identifiable. Dans ce sens nous avons envisagé dans le cadre du projet fin d'étude PFE Master 2 à concevoir et réaliser une application répartie qui répond aux closes de la convention entre la CNAS et l'opticien lunetterie pour faciliter leur tâches aux profils des ayants droit en utilisant les différents composants de l'architecture Java EE comme les JSP, Servlet, les EJB.

Notre mémoire est articulé sur de parties essentielles. La première partie représenté par la présentation l'aspect théorie sur l'architecture J2EE, et une brève présentation sur la structure de l'établissement CNAS.

La deuxième partie de notre mémoire est la partie pratique, dont laquelle nous avons commencé par la conception de notre modèle de l'application pour la gestion un cabinet d'opticiens afin de faciliter leurs tâche (vente des lunettes médicale) aux profils des ayants droit. L'application crée utilise les pages JSP qui interrogent la servlet. La servlet a pour appeler les couche métiers représentée par les trois (03) types d'EJB pour manipuler la BDD des ayants droit.

Au cours de l'élaboration de notre application, nous avons acquis plusieurs connaissances considérés comment un complément de la formation théorique durant les 4 années passées. Nous avons appris en particulier la modélisation avec le langage UML, et programmation orienté objet et programmation par composants manipulé par l'environnement de développement Netbeans.

## Bibliographie

### Référence Bibliographie

[1] : Introduction à EE 5 avec Netbeans 6.8 et le serveur d'applications Glassfish V3, lelivred'AntonioGoncalves.

[2] (en) Ivar Jacobson, Kurt Bittner, Ian Spence, Use Case Modeling, Addison Wesley Professional, 2002

[3] : Développons en Java, Copyright (C) 1999-2016 Jean-Michel DOUDOUX.

[4] : Hibernate et la gestion de persistance, Claude Duvallet, Université du Havre UFR Sciences et Techniques

[5] : E. Daspet et C. Pierre de Geyer. – PHP 5 avancé.N°12167, 4e édition 2007, 792 pages.

[6] K. Djaafar. – Développement JEE 5 avec Eclipse Europa. N°12061, 2008, 390 pages.

[7] : EJB3, Des concepts à l'écriture du code Guide du développeur, Préface d'Alexis Moussine-Pouchkine

[8] coyote , Créez votreapplication web avecJava EE, Licence Créative Commons 6 2.0

[9] Java Platform, Enterprise Edition : (Java EE) Specification, v6, Sun Microsystem, 6 novembre 2009, 264 p

[10] Mémoire de Master Mastère professionnel en technologie des réseaux des télécommunications, **Conception d'un outil d'administration réseaux** par Walid Zayani Université de Carthage Tunisie 2012

[11] mémoire de Gestion des unités d'enseignement dans le cadre de la réforme LMD à l'université de Lomé, par Komi Klenam ADJAFO-TRETU, Diplôme Universitaire de Technologies informatique 2005

[12] Adaptation et la simulation d'un algorithme de type DTN(delay tolerant network) pour les réseaux de capteurs sans fil, Spécialité : Génie Informatique (Mémoire de fin d'étude 2015 2016)

[13] Christine Gaubert-Macon, Valérie Emin, Olivier Capuozzo , Cas d'utilisation, une introduction, , Travaux de relecture:,13 Mars 2004

[14] lexis Moussine-Pouchkine, EJB3, Des concepts à l'écriture du code Guide du développeur, DUNOD 2006

[15] Antonio Goncalves,Les cahiers du programmeur, java EE5, eyrolles, 2007

[16] Anthony Patricio, Java Persistence et Hibernate, eyrolles, 2008

## **Bibliographie**

[17] C. Delannoy. – Programmer en Java. Java 5 et 6. N°12232, 5e édition, 2007, 800 pages.

[18] J.-P. Retailé. – Refactoring des applications Java/J2EE. N°11577, 2005, 390 pages.

## Webographie

### Référence Webographie

- [19] : <http://www.commentcamarche.net/contents/548-j2ee-java-2-entreprise-edition>.  
Consulté le 20/05/2016
- [20] : chapitre 2 de la modélisation, <http://dspace.univtlemcen.dz/bitstream/112/6325/3/chapitre2.pdf> consulté le 01/05/2016
- [21] : <http://laurent-audibert.developpez.com/Cours-UML> consulté le 30/04/2016
- [22] : <https://www.edrawsoft.com/fr/uml-activity-diagram-solutions.php>
- [23] : chapitre2de l modélisation, <http://dspace.univ-tlemcen.dz/bitstream/112/6325/3/chapitre2.pdf> consulté le 26/05/2016
- [24] <https://docs.oracle.com/javase/7/tutorial/> consulté le 10/06/2016
- [25] : <https://www.edrawsoft.com/fr/uml-activity-diagram-solutions.php> consulté le 01/06/2016
- [26] : <http://laurent-audibert.developpez.com/Cours-UML> consulté le 02/06/2016
- [27] : <http://www.cnas.dz/>