

REPULIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTRE DE L'ENSEIGNEMENT SUPERIER ET DE  
LA RECHERCH SCIENTIFIQUE  
UNIVERSITE MOULOUDE MAMMERI DE TIZI OUZOU  
FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE  
DEPARTEMENT INFORMATIQUE

# Mémoire

En vue de l'obtention du diplôme de master  
Recherche option system informatique

# Thème

Conception et Réalisation d'une application JEE

CAS :comptabilité générale de l'entreprise ENIEM

DIRIGE PAR :

M<sup>R</sup>: KIRBICHE AHMED

REALISE PAR :

M<sup>R</sup> :BELHOUS YAHIA

M<sup>elle</sup> :BERKANE RADIA

Promotion 2015/2016



# *Remerciement*

*Nous tenons à témoigner notre reconnaissance à dieu tout puissant, qui*

*Nous A aidé et béni par sa volonté durant toute cette période.*

*Nous tenons à exprimer nos sincères remerciements à notre Promoteur*

*monsieur Kerbiche pour l'intérêt et l'aide qu'il nous a donné.*

*Nous tenons également à remercier le comptable et l'informaticien de*

*l'ENIEM qui nous ont pris en charge tout au long De stage pratique.*

*Nos remerciements vont également aux membres du jury pour*

*L'honneur qu'ils nous font en acceptant d'examiner et de Juger notre*

*travail.*

*Nous remercions aussi tous ceux, et celles qui ont contribué*

*De près ou de loin pour l'accomplissement de Ce modeste travaille*



# DÉDICACE

*A la mémoire de mon ancle said et mon grand père mohand*

*A mes parent et chère grande mère*

*A mes chère sœurs nono nina feriel*

*A mes chere frères mohand,redouane,lyes,madjid,momoh*

*Sam,moukran ainsi que son épouse souhila.*

*Et cousines cici, kikiz,faroudja,souhila ,lamia ,kahina ,lyly et  
leures enfants amar ,foufoud,mina.amine*

*A tout mes ami(e)s souad,samira,nacira,marilia,thasadith ,  
Salima...*

*A Mon oncle et mes tantes*

*A mon binôme ainsi que sa famille*

*A TOUT MES AMI(E)S DE LA SECTION MATH INFO*



# DÉDICACE

*A la mémoire de mon père et ma grande mere et mon grand  
père ali*

*A ma plus chere mere*

*A mes chère freres lounes et marzouk*

*A mes chere oncles saïde, rachide, hmidouche, bouleme, et mes  
tantes faroudja, naima, fatma*

*A ma meilleure amie hamida*

*A tous mes cousins*

*A tout mes ami(e)s mahdi, yacine, idir, akila, lydia, et tous les amis  
de la fac*

*A mon binôme ainsi que sa famille*

*A TOUT MES AMI(E)S DE LA SECTION MATH INFO*

# Tables des matière

Introduction générale.....	1
----------------------------	---

## Chapitre :Technologie JEE

I. Introduction.....	2
I.1 Architecture logicielle.....	2
I.2 Critères de qualité logicielle .....	2
I.3 Architecture client/serveur.....	4
I.3.1 Fonctionnement d'un système client /serveur .....	4
I.3.2 objectif de cette architecture .....	5
I.3.3 Caractéristique de l'architecture client /serveur.....	5
I.3.4 Evolution de l'architectures client/serveur .....	5
I.3.4.1 Le client/ serveur de premier génération.....	5
I.3.4.2 Le client/serveur de deuxième génération .....	5
I.3.4.2 Le client /serveur de troisième génération .....	6
I.3.5 Avantage et inconvénient du modèle Client/serveur .....	6
I.4 présentation de la technologie J2EE .....	7
I.4.1 Introduction a J2EE.....	7
I.4.1 Fonctionnement interne.....	7
I.4.2 Architecture .....	7
I.4.5 Les service de J2EE .....	9
I.4.5.1 Les service d'infrastructures .....	9
I.4.6 Composants de l'architecture JEE .....	9
I.4.6.1 Composants WEB .....	9
I.4.6.2 Pages JSP .....	9
I.4.6.3 Les servleT.....	11

I.4.6.3.1 le fonctionnement d'une servlet (cas de l'utilisation de http) .....	11
I.4.6.3.2 API servlet.....	12
I.4.7.le conteneur de composant web .....	12
I.4.7.1 les composant EJB .....	12
I.4.7.2 Les différent types d'EJB.....	13
I.4.7.3 le serveur EJB.....	14
I.4.7.4client EJB.....	14
I.8 Technologies J2EE.....	15
I.8.1 Java name and Directory interface .....	15
I.8.2 Exemple d'utilisation de JNDI.....	16.
I.8.3 JDBC .....	16
I.8.4 Structure d'un programme GDBC .....	16
I.9Conclusion .....	17

## **Chapitre II :L'organisme d'accueille**

II.1Introduction .....	18
II.2.2 Historique .....	18
II.2.1 Mission. ....	18
II.2.2 Objectif .....	19
II.3 Mode d'organisation .....	19
II.3.1 organisation générale .....	19
II.3.1.1 unité froid.....	19
II.3.1.2 unité cuisson .....	20
II.3.1.3 unité climatisation .....	20
II.3.1.4 unité de prestation technique .....	20
II.3.1.5 unité commercial .....	20

II.3.1.5 unité sanitaire.....	20
II.3.1.6 unité FILAPM.....	21
II.3.2 les direction .....	21
II.3.2 .1direction générale.....	21
II.3.2. 2 direction industrielle .....	21
II.3.2.3 direction finance et comptabilité .....	22
II.3.2.4 directions de marketing et de la communication .....	22
II.3.2.5 direction développement et partenariat.....	23
II.3.2.6 direction des ressources humaines .....	23
II.3.2.7direction planification et contrôle de gestion .....	23
II.4 présentation de l'unité commerciale .....	25
II.4.1 Historique et fonction .....	25
II.4.2 organisation générale .....	25
II.4.1.2.1 département de vente .....	25
II.4.2.2 département distribution .....	25
II.4.2.3 département marketing .....	26
II.4.2.4 département service après vente .....	26
II.4.2.5 département finance et comptabilité.....	26
II.4.2.6 département ARCH .....	26
II.4.2.7 département contrôle de gestion .....	26
II.4.3 organigramme de l'unité commerciale .....	27
II.5 LA Comptabilité .....	28
II.5.1 Introduction .....	28
II.5.2 Organisation pratique .....	28
II.6 conclusion .....	28

### Chapitre III : Analyse et conception

III.1 Introduction .....	29
III.2.1 les diagramme de structure ou statique qui sont au nombre de sept .....	29
III.3 une démarche pour l'analyse et conception de notre projet .....	32
III.3.1 phase d'analyse .....	32
III.3.2 phase de conception .....	32
III.3.3 phase de réalisation .....	32
III.4 cahier de charge de notre application .....	31
III.5 analyse et conception de notre system .....	33
III.5.1 analyse .....	33
III.5.1.1Spécification des besoin .....	33
III.5.1.2 cas d'utilisation .....	33
III.5.1.3 spécification des tache.....	34
III.5.1.4diagramme de cas d'utilisation globale .....	34
III.5.2conception .....	37
III.5.2.1 élaboration de diagramme de séquence .....	37
III.5.2.2diagramme d'activité.....	42
III.5.2.3 diagramme de classe .....	48
III.6 conclusion .....	50

## Chapitre VI :Réalisation

IV Introduction .....	51
IV.1 description des langage et des outils de développent.....	51
IV.1.1 Le langage de programmation (java) .....	51
IV.1.2 IDE (NetBeans).....	51
IV.1.3 le serveur d'application(Glassfish) .....	52
IV.1.4 le SGBD( Mysql) .....	52



IV.2 présentation de l'application .....	52
IV.2.1 Implémentation de différentes partie de l'application .....	53
IV.2.2création de la base de donnée .....	53
IV.3 présentation de quelque interfaces de l'application web réalisé .....	55
IV.3.1 les page statique .....	55
IV.4 conclusion .....	61

## Table des figures

figure I.1 fonctionnement de model client/serveur .....	4
figure I.2 architecture et les composant de JEE .....	8
figure I.3 architecture généré d'un EGB .....	13
figure I.4 l'architecture de la technologie JNDI.....	15
figure II.1 organisation de l'entreprise ENIEM .....	24
figure II.2 organigramme de l'unité commercial .....	27
figure III.1 classification des diagramme UML .....	31
figure III.2 spécification des taches.....	34
figue III.3 diagramme de cas d'utilisation pour l'espace client .....	35
figure III.4 Diagramme de cas d'utilisation de comptable pour l'espace comptabilité générale	36
Figure III.5 Diagramme de cas d'utilisation pour le profil société .....	37
Figure III.6 Diagramme de séquence de cas d'utilisation « authentification ».....	38
Figure III.8 Diagramme de séquence de cas d'utilisation « Enregistrement facteur» .....	39
Figure III. 9 diagramme de séquence de cas d'utilisation « changer le mot passe » .....	40
Figure III.10 Diagramme de séquence de cas d'utilisation « création de compte ».....	41
Figure III.11 Diagramme de séquence de cas d'utilisation « voir les compte » .....	42
Figure III.12 Diagramme d'activité de cas d'utilisation « S'authentifier » .....	44
Figure III.13 Diagramme de cas d'utilisation « édition des facteurs » .....	
III.14 Diagramme d'activité de cas d'utilisation « changer le mot de passe».....	45
III.15 Diagramme d'activité de cas d'utilisation « création de compte » .....	46
III.16 Diagramme d'activité de cas d'utilisation « voir les comptes ».....	47
III.17 diagramme de classe globale de notre application .....	48
Figure VI.1 interface de L'IDE Netbeans. ....	51
Figure IV.2 l'arborescence de l'application .....	52

# Introduction générale

## Introduction générale

pour de nombreux développeurs, JEE est souvent synonyme de « Entreprise java Beans » En fait, JEE est beaucoup plus que cela .en simplifiant ,nous pouvons dire que JEE est une collection de composants ,de conteneurs et de service permettant de créer et de déployer des application distribuées au sein d'une architecture standardisée .

JEE est un ensemble de composant standardisées facilitant le déploiement des application ,des interfaces définissant la façon dont les modules logiciels peuvent être interconnectés ,et les services standards ,avec leur Protocol associer ,grâce auxquels ces module peuvent communiquer .

Le web est de plus en plus le support privilégié des applications d'entreprise ,la réforme est devenue plus que une nécessité pour remédier a tout le dysfonctionnements et rendre les entreprise algérienne plus performante, complétive et attractive et lui permettre de répondre aux grand défis de modernisation et de l'évolution rapide.

Le mémoire est organisé en quatre chapitre :

Dans le premier chapitre ,nous présenterons la technologie JEE qui est la solution utilisé dans le demain entreprise dans le second chapitre l'organisme d'accueille aux sien le l'entreprise ENIEM(Entreprise Nationale des Industrie en Electroménager).le chapitre trois seras consacré a la conception de l'application décrite Par les diagramme UML. Dans le dernier chapitre nous décrirons le logiciel réalisé ainsi que son utilisation a travers quelques interfaces .Enfin en conclurons en donnant quelque perspectives .

# Chapitre I

## Technologie JEE

## I. Introduction

De nos jours ,le génie logiciel nous offre des nouvelles technologies qui ont les potentialités de révolutionner le monde des conceptions des Systems informatiques.

Un bon logiciel ne verra jamais le jour sans utiliser une bonne architecture logicielle qui respecte les critères de fiabilité ,sécurité, portabilité...

Dans ce chapitre on va étudier une approche de l'architecture logicielle ( j2EE) qui est nécessaires ou bien indispensables pour la réalisation des applications des entreprises.

### I.1 Architecture logicielle [8]

L'architecture logicielle décrit d'une manière symbolique et schématique les différents éléments d'un ou plusieurs Systems informatiques , leurs interrelations et leurs interactions .Contrairement aux spécification produite par l'analyse fonctionnelle, le modèle d'architecture ,produit lors de la phase de conception ,ne décrit pas ce que doit réaliser un system informatique mais plutôt doit être réalisé de manière a répondre aux spécification L'analyse décrit le « quoi faire »alors que l'architecture décrit le « comment le faire ».

### I.2 Critères de qualité logicielle [8]

- ✓ **L'interopérabilité extrinsèque** : exprime la capacité de logiciel a communiquer et a utiliser les ressources d'autres logiciels comme , par exemple , les documents créés par certaine application.
- ✓ **L'interopérabilité intrinsèque** :exprime le degré de cohérence entre le fonctionnement des commandes modules a l'intérieur d'un système ou d'un logiciel.
- ✓ **La portabilité** :exprime la possibilité de compilé le code source et/ou d'exécuter le logiciel sur des plates-formes (machines system d'exploitation .environnements) différents.
- ✓ **La compatibilité** :exprime la possibilité ,pour un logiciel, de fonctionner correctement dans un environnement ancien (compatibilité descendante) ou plus récent (compatibilité ascendante) .
- ✓ **La validité** :exprime la conformité des fonctionnalités du logiciel avec celle décrites dans le cahier des charge .
- ✓ **La vérifiabilité** : exprime la simplicité vérification de la validité.
- ✓ **L'intégrité** :exprime la faculté du logiciel a protéger ses fonction et ses données non

Autorisés.

- ✓ **La fiabilité** : exprimé la faculté d'un logiciel a géré correctement ses propre erreurs de fonctionnement en cours d'exécution .
- ✓ **La maintenabilité** : exprime la simplicité de correction et de modification de logiciel, et même parfois la possibilité de modification d'un logiciel en cours d'exécution.
- ✓ **Réutilisabilité** :exprime la possibilité d'étendre simplement les fonctionnalité d'un logiciel sans compromettre son intégrité et sa fiabilité.
- ✓ **L'efficacité** : exprime la capacité de logiciel à exploiter au mieux les ressources offertes par les machines ou le logiciel seras implanté.
- ✓ **L'autonomie** : exprime la capacité de contrôle son exécution ,de ses données et de ses communications.
- ✓ **La transparence** : exprime la capacité pour un logiciel de masquer a l'utilisateur (humain ou machine) les détails inutiles a l'utilisation de ses fonctionnalités.
- ✓ **La compossibilité** :exprime la capacité d'un logiciel de combiner des informations provenant de sources différentes.
- ✓ **La convivialité** :décrit la facilité d'apprentissage et d'utilisation du logiciel par les usagers.

### I.3 Architecture client/serveur[5]

Le client/serveur est un modèle de communication entre deux processus .Le premier appelé client , demande des services aux deuxième processus qui est le serveur ,ce dernier exécute la requête de client et envoie des résultat en routeurs . Client et serveur sont en pratique deux logiciels différents communiquant au moyen d'un protocole ,a travers un réseau local ou bien a travers un réseau étendu.

#### Notion de base :

- ✓ **Requête :** message transmit par un client a un serveur décrivant l'opération a exécuté pour le compte du client.
- ✓ **Réponse :** requête transmit par le serveur a un client suite a l'exécution d'une opération contenant les paramètre de retour de l'application
- ✓ **Client :** processus demandant l'exécution d'une opération a un autre processus par envoi d'un message contenant le descriptif de l'opération a exécuter et attendant la réponse a cette opération par un message en retour .Ce client émet cette requête ver le serveur grâce a son adresse IP et son port ,qui désigne un service particulier du serveur .
- ✓ **Serveur :** processus accomplissant une opération sur demande d'un client et transmettant la réponse a ce client a l'aide de l'adresse de sa machine .il est par conséquent initialement passif car il est toujours en attente prêt a traiter les requêtes envoyées par les clients.

#### I.3.1 Fonctionnement d'un système client /serveur :

Un system client /serveur fonction selon le schéma suivant :

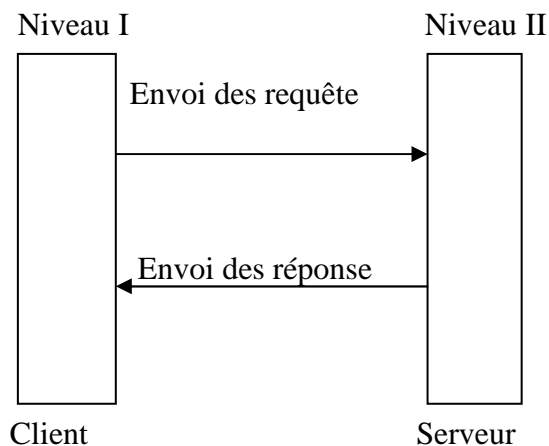


Figure I.1 : fonctionnement de model client/serveur

Le client émet une requête vers le serveur grâce a son adresse et le port ,qui désigne un service particulier du serveur. Le serveur reçoit la demande et répond a l'aide de l'adresse de la machine et son port .



**I.3.2 objectif de cette architecture :**

- une grande flexibilité /souplesse.
- Une grande sécurité (la sécurité peut être définie pour chaque service ).
- de meilleure performances (les tache son partagé )

**I.3.3 Caractéristique de l'architecture client /serveur**

❖ **Service** : le modèle client /serveur est essentiellement une relation entre les processus. Le processus serveur est le fournisseur de service et le client est le consommateur. Le modèle client /serveur établit ainsi une opération claire de rôle a partir de notion de service .

- ❖ **Partage de ressources** : un serveur est censé savoir traiter plusieurs demandes clients a la fois et leur accès aux ressources.
- ❖ **Asymétrie des protocoles** :la relation entre le client et le serveur est de type plusieurs ver un .toute foi le client est le déclencheur de dialogue .en demandant un service alors que le serveur attend passivement les requêtes.
- ❖ **Assemblage multi vendeur** : le client serveur est indépendant de la Plateforme matérielle de system d'exploitation .on doit toujours pouvoir mélanger et apparier les plat forme client/serveur .
- ❖ **Echange de message** :le client et le serveur sont des system a liaison épisodique qui interagissent au moyenne de message .le message est mécanisme d'émission de demandes de service et de réponses a celle-ci .
- ❖ **Encapsulation des service** : le serveur est spécialiste ,un message lui indique quel service est requis c'est a luis de décider comment rendre ce service .les serveurs peuvent être mise a niveaux sans effet sur les client tant que l'interface des messages reste la même .
- ❖ **Intégrité** :le code et les données du serveur sont gérés de façon centralisé, ce qui garantit un moindre cout de maintenance et une meilleure intégrité des données tandit que les clients restent individuels et indépendants.
- ❖ **Souplesse et adaptabilité** :on peut modifier le module serveur sans toucher aux module client et vice versa ,si une station est remplacée par un modèle plus récent, on modifier le module client sans modifier le module serveur .

**I.3.4 Evolution de l'architectures client/serveur [5]**

Le modèle client/serveur constitue une évolution majeure de l'informatique .le principe de base est de composer un processus informatique on aux mois deux taches mois complexe (le client et le serveur ) associer a un mécanisme de communication leurs permettant de coopérer .

**I.3.4.1 Le client serveur de premier génération :**

Né a la fin des années 80 et basé sur des outils client des SGBD relationnels. Le développent S'effectue sur le serveur de la base de données et sur le client pour l'application .  
L'inconvénient de cette architecture est que tout le code applicatif est exécuté sur le client.

**I.3.4.2 Le client/serveur de deuxième génération**

Né au milieu des années 90 cette génération est caractérisée par :

- La possibilité de développer des traitements applicatifs au sien de serveur de données Sous forme de procédure déclenchées par l'application ou lors d'événement sur la base de données .
- L'utilisation de l'approche orienté objet .
- La facilité de déploiement des applicatifs avec partitionnement de code applicatif entre le client et le serveur.

#### **I.3.4.3 Le client serveur de troisième génération [6]**

avec l'application d'internet et le web ,les architecture client\_ serveur sont évoluer vers des architectures a trois niveaux

- ✓ Le client est responsable de la présentation .il utilisent pour cela des navigateur web comme internet Explorer .
- ✓ Le serveur d'application exécute le code applicatif .
- ✓ Le serveur de données supporte le SGBD qui gère éventuellement des types de données très ,comment les données multimédias
- ✓ Le serveur d'application et le serveur de données peuvent être regroupés sur la même machine
- ✓ L'architectures a trois\_tiers ,appelée encore client/serveur web est aujourd'hui bien adaptée aux system répartie autour d'un réseaux local et/ou internet .
- ✓ Elle permet a de multiple poste ou satins de travaille distribuer sur la planète de partager les même données
- ✓

#### **I.3.5 Avantage et inconvénient du modèle Client/serveur [4]**

##### **A. Avantage**

Les principe avantage de ce modèle sont :

- ✓ Des ressources centralisées : Etant donné que le serveur est au centre du réseau ,il peut gérer des ressources communes a tout les utilisateurs ,comment par exemple une base de données centralisée afin d'éviter les problèmes de redondance et d'incohérence.
- ✓ Une meilleure sécurité : car le nombre de points d'accès aux données est moins important
- ✓ Une administration aux niveaux serveur :les client ont moins importance dans ce modèle .ils ont besoin d'être administrés .
- ✓ Un réseaux évolutif :grâce a cette architecture ,on peut supprimer ou ajouter des client sans perturber le fonctionnement du réseau et sans modifications majeures

##### **B.inconvénients :**

L'architecture client/serveur a tout de même quelques inconvénients permet les quels :

- ✓ Un cout élevée du a la technicité du serveur (bien plus puissant que une machine de bureau ).
- ✓ Un maillon faible : le serveur est le seul maillon faible du réseau client/serveur étant donné que tout le réseaux est architecturé autour de lui.

## I.4 PRESENTATION DE LA TECHNOLOGIE J2EE

### I.4.1 Introduction a JEE

J2EE (aujourd'hui appelé JEE-java Entreprise Edition ) est une spécification pour le langage de programmation java de Sun destinée aux application d'entreprise . la première Version des librairies J2EE a été mise à la disposition des développeurs en 1999 .J2EE offre une plate-forme de développement de déploiement en langage java pour les application distribuer à plusieurs niveaux.

### I.4.2 Fonctionnement interne [9]

le langage java ,sur lequel les librairies J2EE sont utilisées ,met a la disposition un compilateur et une machine virtuelle (JVM-Java Virtual Machine) qui de créé un environnement standard pour le lancement de l'application sur tout type de system opérationnel. Le compilateur compile le code source et produit le bytecode, soit un code intermédiaire qui seras ensuite lit par la machine virtuelle java. Chaque system opérationnel majeur possède une JVM expressément codée.

### I.4.3 Architecture [10]

J2EE ajout de nombreuse couche de niveau entreprise au-dessus de la plate -forme J2SE (Java Standard Edition).chaque couche est conçue pour supporter une différente technologie de développement .

- **Technologie web application :** technologie liées a la production des interfaces web dynamiques ,par exemple jsp (Java Servlet Page) et servlet .
- **Technologie Enterprise Application :** technologies liées a la logique de business :EJB (Entreprise Java Bean),java Mail , JMS(Java Message Service) ,JTA (Java Transaction),etc.
- **Technologie web service :**technologie utiles au développement des applications adhérentes au paradigme SOA (Service Oriented Architecture) :web services JAX-WS (java API for XML-based web services ) .JAX -RPC (java API for XML-Based RPC ).
- **Technologie Management and Security :** technologie liée a la gestion de la technologie entreprise afin de réaliser l'accès et l'échange d'information entre services distribuer : JAAS (java Authentification and Autorisation service) JCA (Java Connector Architecture).

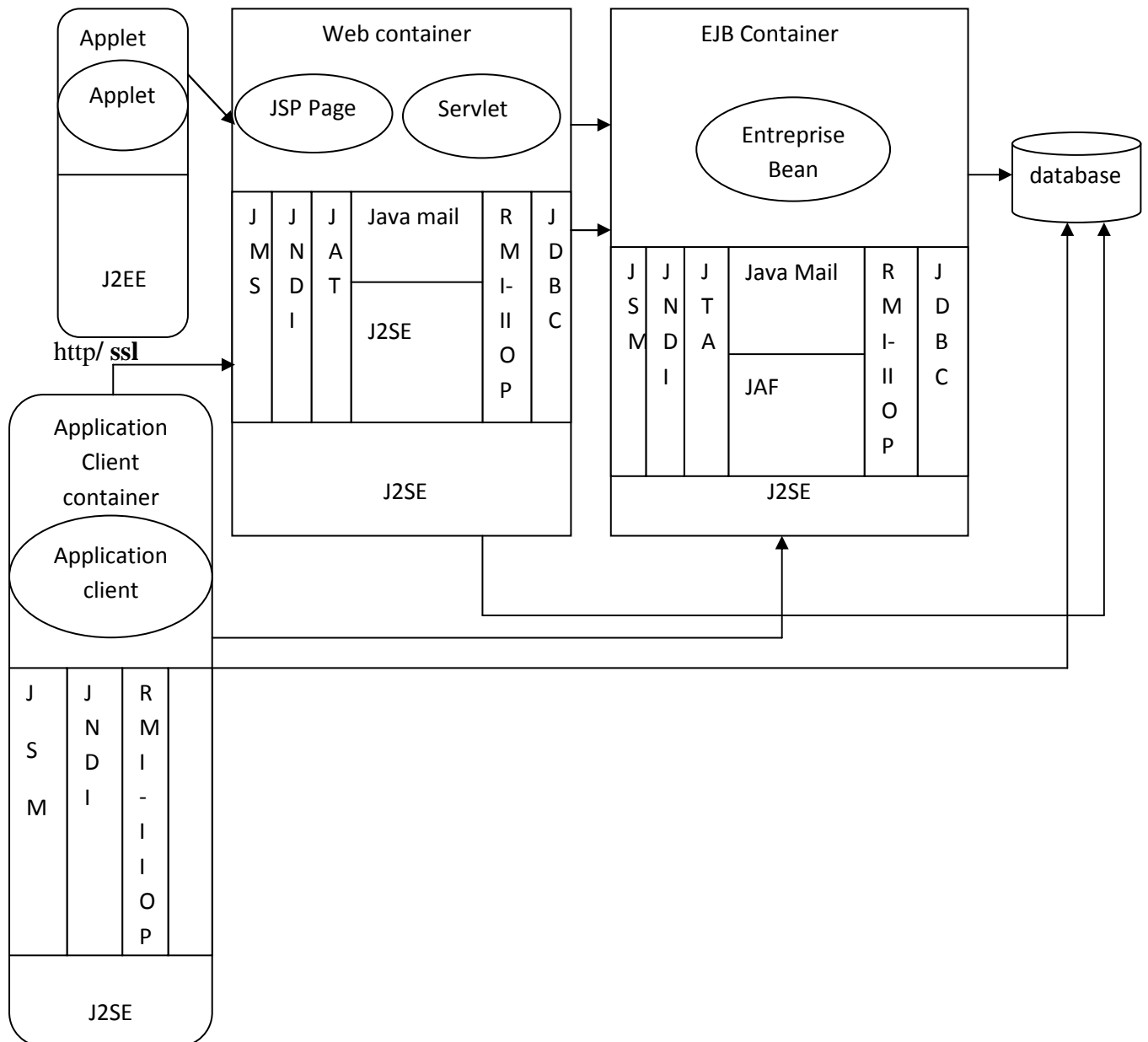


Figure I.2 :Architecture et les composants de JEE .

pour expliquer l'utilisation de ses technologies on peut imaginer que les technologies entreprise sont utilisées pour gérer l'accès aux données (généralement une ou plusieurs database), les technologies web application sont utilisées pour montrer les données aux utilisateurs génériques. Dans un contexte business to business, les technologies web service seront utilisées pour échanger les informations avec les partenaires commerciaux et les technologies de gestion gèrent tout les processus informationnels assurant la sécurité des transactions.

#### **I.4.5 Les services de JEE [11]**

##### **I.4.5.1 Les services d'infrastructures :**

- **JDBC** (Java Database Connectivity) est l'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.
- **JTA / JTS** (Java Transaction API/Java Transaction Services) est une API définissant des interfaces standard avec un gestionnaire de transactions.
- **JCA** (J2EE Connector Architecture) est une API de connexion aux systèmes d'information de l'entreprise, tels que les ERP.
- **JMX** (Java Management Extension) fournit des applications permettant de développer des applications web de supervision d'applications).

#### **I.4.6 Composants de l'architecture JEE :**

##### **I.4.6.1 Composants WEB**

##### **I.4.6.2 Pages JSPs [11]**

Java Server Page est une technologie Java qui permet de générer dynamiquement, aussi bien côté client que serveur, du code HTML, XML ou autre, donc des pages dont le contenu variera en fonction du contexte.

**Page JSP :** programme destinée à produire des pages HTML.

**Page JSP** (seconde interprétation) : page HTML produite par l'interprétation d'une page JSP (première interprétation).

Les JSP comme les Servlet sont considérées comme des composants car leur composant peut être paramétré via un conteneur.

Une page JSP est conceptuellement substituable à un Servlet. (une page est potentiellement compilée à un Servlet).

Une page JSP ne présente pas comme un fichier contenant du code source HTML entrecoupée de code Java intégrée via différentes sortes de balises.

Il existe trois types de tags :

- **Tags de directives** : ils permettent de contrôler la structure de la servlet générée.
- **Tags de scripting** : ils facilitent l'utilisation des composants.

### 1. Les tags de directives < %@...%>

Les directives permettent de préciser des informations globales sur la page JSP. Les spécifications JSP définissent trois directives :

- **Page** : permet de définir des options de configuration.
- **Include** : permet d'inclure les fichiers statiques dans la JSP avant la génération de la servlet.
- **Taglib** : permet de définir les tags personnalisés.

Leur syntaxe est la suivante :

`<% @ directive attribut="valeur" ...%>`

### 2. Les tags de scripting

Ces tags permettent d'insérer du code Java qui sera inclus dans la servlet générée à partir de JSP. Il existe trois tags pour insérer du code Java.

- **Le tag de déclaration** : le code Java est inclus dans le corps de la servlet générée. Ce code peut être la déclaration de la variable d'instances ou de classes ou la déclaration de la méthode. Sa syntaxe est `<%!declarations %>`
- **Le tag d'expression** : évalue une expression et insère le résultat sous forme de chaîne de caractères dans la page web générée. Sa syntaxe est : `<%=expression %>`
- **La tag de scriptlets** : par défaut le code Java est inclus dans la méthode `service()` de la servlet. Sa syntaxe est la suivante : `<%=code java %>`
- **Variables prédéfinies jsp**

- Request (HttpServletRequest)
- response (HttpServletResponse)
- out ( PrintWriter) utilisée pour écrire dans la page réponse
- session (HttpSession) la session (si elle existe) associée à la requête
- application (ServletContext) identique à `getServletConfig().getContext()`.
- config (ServletConfig)
- page (this)
- pageContext. Accès aux classes JSP spécifiques (javax.servlet.jsp)

**I.4.6.3 Les servlet [12]**

Une servlet est une class java qui permet de crée dynamiquement des données au sien d'un serveur http .Ces données sont le plus généralement présentées au format HTML ,mais elle peuvent également l'être au format XML ou toute autre format destiné aux navigateur web. Les servelet utilisent l'API java servelet (package javax.servlet).

Une servlet s'exécute dynamiquement sur le serveur web et permet l'extension des fonction de ce dernier ,typiquement :accès a des base de données ,transactions d'e-commerce ,etc.une servlet peut être chargé automatiquement lors du démarrage du serveur web ou lors de la premier requête du client . une fois chargées, les servlets restent actives dans l'attente d'autre requêtes de client. L'utilisation de servlet se fait par le biais d'un conteneur de servlets ( framework) coté serveur .celui ci constitue l'environnement d'exécution de la servlet et lui permet de persister entre les requêtes des clients. l'API définit les relation entre le conteneur et la servlet . le conteneur reçoit la requête de client ,et sélectionne la servlet qui aura a la traiter .le conteneur fournit également tout un ensemble de services standards pour simplifier la gestion des requêtes et des sessions.

**I.4.6.3.1 le fonctionnement d'une servelet (cas de l'utilisation de http)**

un serveur d'application permet de charger et d'exécuter les servlets dans une JVM.C'est une extension de serveur web .Ce serveur d'application contient entre autre un moteur de servlets qui se charge de manager les servlets qu'il contient.

Pour exécute une servlet ,il suffit de saisir une URL qui désigne la servlet dans un navigateur .

Le serveur reçoit la requête http qui nécessite une servlet de la part de navigateur c'est la premier sollicitation de la servlet ,le serveur l'instancie.les servlets sont stockées( sous forme de fichier .class)dans un répertoire particulier de serveur .ce répertoire dépend de serveur d'application utilisé .la servlet reste en mémoire jusqu'à l'arrêt du serveur . Certains serveur d'application permet aussi d'instancier des servlet dès le lancement du serveur .

La servlet dans la mémoire peut être appelée par plusieurs threads lancés par le serveur pour chaque requête .ce principe de fonctionnement évite d'instancier un objet de type servlet a chaque requête et permet de maintenir un ensemble de ressources actives l'el qu'une connexion a la base de données.

### I.4.6.3.2 API servlet

Les servlets sont conçues pour agir selon un modèle requête /réponse. Tout les protocoles utilisant ce modèle peuvent être utilisé tel que http,ftp,etc.

L'API servlet est une extension de la jdk de base ,et en tant que tel elle est regroupée dans des package préfixés par javax .

L'API servlet regroupe un ensemble de class dans deux package :

- javax.servlet :contient des class pour développer des servlet génériques indépendantes d'un protocole.
- javax.servlet.http : contient des class pour développer des servlets qui reposent sur le protocole http utilisé é par les serveur web.

### I.4.7 Le conteneur de composant web

Les composant web sont hébergés dans des conteneurs de servlet ,conteneurs de JSP et conteneurs web .En su des fonctionnalités normales d'un conteneur de composants , un conteneur de servlets,(servlet container) fournit les services réseaux par lesquels les requêtes et les réponse sont émise .il decode également les requête et formate les réponse dans le format a approprié . Tout les conteneurs de servlet doivent supporter le protocole et peuvent aussi supporter le protocole HTTPS .Un conteneur de JSP (JSP container) fournit les mêmes services qu'un conteneur de servlets .Ces conteneur sont généralement appelés conteneurs web (web containers).

#### I.4.7.1 Les composants EJB [12]

La technologie EJB définit un modèle pour le développement et le déploiement de Composants de serveur java réutilisables également appelé **composant EJB** .

Un composant EJB est un composants de serveur non visuel dont les méthodes fournissent généralement une logique de gestion dans les applications distribuer . un client distant ,appelé client EJB peut appelé ses méthode qui sont généralement a l'origine de la mise a jour de la base de données .

L'architecture EGB se présente de la façon suivante :



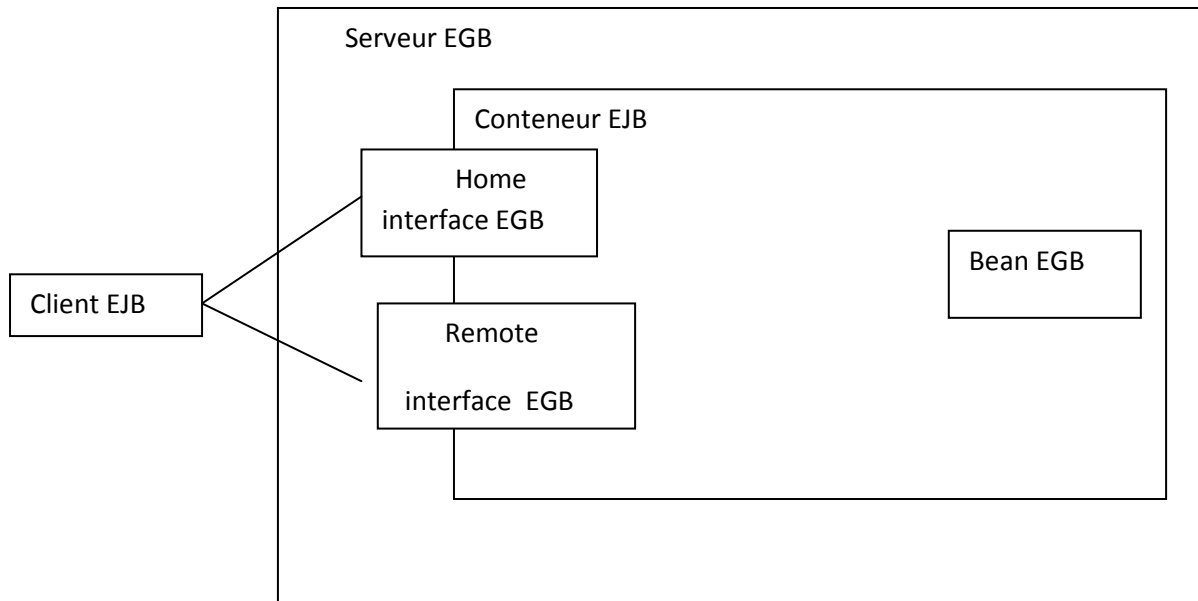


Figure I.3 :l'architecture génère d'un EGB

#### I.4.7.2 Les différents types d'EJB [13]

- **Les beans session** : Les beans session comme leur nom l'indique ,ont une durées de vie correspondant a une " conversation " ou " session " entre l'application cliente et le composant .ces beans servent a fournir a l'application cliente divers services conçus par le développeur. Selon le choix de celui-ci, un bean session peut maintenir a un état pendant toute la durée de la session .(c'est-à-dire conserver l'état des attributs internes aux objet de façon a maintenir une conversation avec le client ) ou au contraire sans état (stateless),ce qui signifie qu'il fournisse accès a des méthodes implémentant la logique applicative (comme RMI ,mais ne conserve aucun résultat auquel le client pourrait faire référence ultérieurement.
- **Les beans entités** : les beans entités représente des objets métier de domaine de l'application tels que client ,factures ,produit, etc.ces objets persiste de façon pouvoir être réutilisés . Le conteneur de l'architecture JEE s'occupe de tout les détails de cette tâche persistance. Rappelons que la persistance correspond à l'utilisation d'une base de donnée qui stock la valeur des attribut dans ces beans entités .avec les beans entités ,il n'est pas de tout nécessaire de maitriser le langage SQL ainsi que la connectivité GDBC .De fait ,la base de donnée de type relationnelle devient une base technologique .le développeur passe généralement beaucoup de temps a la gestion de la base de données .Avec un bean entité ,le développeur ne voit pas de tout la base de données et donc ne s'en occupe pas ;il peut alors passé tout son temps sur l'application elle-même.
- **Les beans contrôlés par les messages** :le troisième type d'EJB ,le bean contrôlé par message ,fournit un modèle de composant permettant d'écouter un services de

messages .la plateforme JEE définit une queue de message ,qui est une sorte de file d'attente dans laquelle les applications peuvent placer des messages .elle peuvent également s'abonner à une queue de messages. L'avantage de cette architecture et que les composants qui utilisent les message n'ont pas besoin de savoir qui les a envoyés. Il leur suffit d'identifier la queue qui contenant les messages .c'est une sorte de flot d'informations que les composants ont en commun sans que ces dernières soit dédiée a un composant en particulier .un system de gestion de portefeuilles d'action est un exemple d'utilisation d'une telle queue .les cours des actions sont envoyés a une queue sous forme de message ,qui sont consommés par les composants qui ont besoin de connaître ces cours ,avec les beans contrôlé par ces messages, il est possible de créer des composants répondant aux messages concernant les cours en prenant automatiquement certaines décisions en fonction de leur variation .

#### **I.4.7.3 serveur EJB [13]**

EJB server intègre le conteneur EJB assurant les services requis par le composant EJB.

#### **I.4.7.4 Client EJB**

Le client EJB fournit généralement la logique de l'interface utilisateur sur la machine cliente .il passe les appels aux composants EJB distants hébergés sur un serveur et doit savoir comment trouver le serveur EJB et interagir avec les composants EJB .

Un composant EJB peut faire office de client EJB en appelant les méthodes d'un autre composant EJB .

Un client EJB ne communique pas directement avec un composant EJB .

Un conteneur fournit des objets proxy mettant en œuvre les Home interface et Remote Interface des composants .le Remote Interface des composants définit les méthodes de gestion pouvant être appelées par le client .Ce dernier appelle les méthodes de Home Interface pour créer et détruire les proxies pour le Remote Interface.

#### **L'interface Remote**

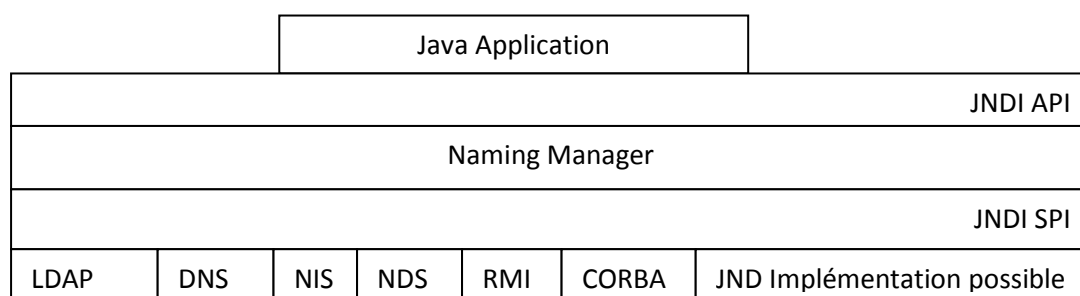
L'interface remote permet de définir les méthodes qui contiendront les traitements proposés par le bean .cette interface doit étendre l'interface javax.ejb.EJBObject.

## I.8 Technologies J2EE

### I.8.1 Java name and Directory interface .

JNDI est. l'acronyme de java Naming and Directory Interface. Cet API fournit une interface unique pour utiliser les différents services de nommage ou d'annuaires :

- LDAP (Lightweight Directory Access protocol )
- DNS (Domain Naming Service)
- NIS(Network Information Service) de sun
- Service de nommage CORBA
- Service de nommage RMI



**Figure I.4 l'architecture de la technologie JNDI**

Un service de nommage permet d'associer un nom unique à un objet et faciliter ainsi l'obtention de cet objet. Un annuaire est un service de nommage qui possède un ou plus représentation hiérarchique des objets qui contient et un mécanisme de recherche.

pouvoir utiliser autant de services différents possédant des protocoles d'accès différents.

JNDI utilise des pilote SPI (Service Provider). Trois de ces pilotes sont fournis en standard :

- ✓ LDAP (Lightweight Directory Access protocol).
- ✓ Service de nommage CORBA (COS).
- ✓ Service de nommage RMI.

la technologie JNDI est utilisée dans le JEE pour localiser les objets sur un serveur et accéder aux objets externe à partir des composants J2EE.

Chaque conteneur stock une référence aux objets qu'il peut créer et instancier ces objets. À la demande des clients ou des applications qui fonctionnent sur le serveur.

Le conteneur met aussi à la disposition des composants un jeu de ressources JND initial, issu de la configuration de serveur et /ou des applications web (via les descripteurs de déploiements).

Un objet initialContext est créé par le conteneur lorsqu'une application web est déployée. Cet objet est accessible par les composants, en lecture seulement.

L'autre rôle de JNDI dans une application J2EE est la location des interfaces distante des beans.

### **I.8.2 Exemple d'utilisation de JNDI**

Exemple d'accès a une ressource JDBC par JNDI :

```
//obtain our environment naming context

ContextinitCtx =new InitialContext();

Context envCtx =(Context)InitCtx.lookup("java:comp/env") ;

//look up our data source

DataSource ds =(DataSource) ;

envCtx.lookup("jdbc/employeeDB");

//allocate and use a connexion fram the pool

Connection conn =ds.getConnection();

... use this connection to access the database ...

Conn.close();
```

### **I.8.3 JDBC**

JDBC est une API java (ensemble de class et d'interfaces définit par sun et les acteur de domaine BD) permettant d'accéder aux base de données a laide de langage java via les requête SQL. Cette API permet d'attendre de manière quasi-transparente des bases Sybase,oracle,Informix ... avec le même programme java JDBC. En faite cette API est spécification de ce que on doit implanter un constructeur de DB pour que celle-ci soit interrogeable par JDBC .De ce fait dans la programmation JDBC en utilise essentiellement des références d'interface (Connexion ,statement,ResultSet, ...).

### **I.8.4 Structure d'un programme JDBC**

Un code JDBC est de la forme :

- recherche et chargement de driver approprié a la DB .
- établissement de la connexion a la base de données.
- construction de la requête SQL.
- envoi de cette requête et récupération des réponse .

-parcours des réponses .

### **I.9 Conclusion**

L'architecture JEE est le résultat de la collaboration des technologie de l'information, qui se sont entendus sur un certain nombre de protocoles et d'approche qui favorisent l'interopérabilité entre les plateforme ,les system d'exploitation et les langage de programmation cette technologie elle est suffisamment développées pour que les développeurs les utilisent maintenant dans tout les domaines de l'informatique ,afin de récolter les divers bénéfices de la technologies .ils représentent aujourd'hui la technologies la plus adapté pour le développement des system informatique distribuer sur Internet.Nous allons présenter dans le chapitre suivant l'organisme d'accueille de l'entreprise ENIEM.





## Bibliographie

[1] :www Wikipidia.org

[2]Alin Lefebvre, « L'architecture client\_serveur »,Edition ARMAND COLIN 2006

[3]Paolo Zanella & yves Ligier, « Architecture et technologie des ordinateurs »Dunod 4ème Edition 2005.

[4]George et Olivier Gardarin, « le client/serveur» Edition EYROLLES 2000.

[5] : <http://www.dyalo.com/coursinformatique/introductionj2EE.HTML>

[6] : documentation j2EE <http://java.sun.com/j2ee/>

[ 7] : Développement en java :par jean Michel DOUDOUX

[8] : <http://www.jmdoudoux.fr/java/dej/indexavecframes.html>



# Chapitre II

## Organisme d'accueil

---

## II.1 Introduction :

Dans ce chapitre, nous allons faire une courte présentation de l'entreprise ENIEM, en commençant par un bref historique, ensuite nous allons présenter son organigramme structurel puis le déroulement de son activité commerciale.

## II.2 Historique :

L'entreprise nationale des industries de l'électroménager ENIEM est issue de la restructuration organique de la SONELEC (société nationale de fabrication et montage du matériel électrique et électronique).

Elle a été créée en vertu du décret N°83-19 du 02 janvier 1983. L'entreprise a été chargée de la production et la commercialisation des produits électroménagers et disposait à sa création de :

- ✓ Complexe d'appareils électroménagers (CAM) de tizi ousou, entré en production en juin 1977.

- ✓ Unité lampes de Mohammedia(ULM), entrée en production en juin 1979.

Le statut de l'ENIEM est passé d'une entreprise publique et économique (EPE) à celui d'une société par actions (SPA). Ainsi elle est passée à l'autonomie le 08 octobre 1989 avec un capital social de 40.000.00 DA. Depuis 1996, l'entreprise est organisée en unités, et filiale l'unité lampes de Mohammedia.

L'ENIEM est la première entreprise de Maghreb à être certifiée ISO 9002 depuis le premier juillet 1998 par les experts de l'association française de l'assurance de la qualité (AFAQ), puis gratifiée en 2003 de l'ISO 9001 <<version 2000>>. A noter que les produits ENIEM sont 0% cfc (chloro fluoro carbonés), et ce depuis 1997.

Le siège social de l'entreprise se situe au chef lieu de la wilaya de tizi ousou. Les unités froides, climatisation, commercial et prestations technique sont implantées à la zone industrielle AISSAT IDIR de OUAD AISSI, distance de 10 Km du chef lieu de la wilaya. L'unité sanitaire est Miliana, wilaya d'Ain Défila, et la filiale lampes de Mohammedia à la wilaya de Mascara. Missions et objectifs de l'organisme :

### A- Missions

Dans le cadre de développement économique et social, l'ENIEM assure les fonctions suivantes :

La production, le montage, la commercialisation et la recherche dans les différentes branches de l'électroménagers notamment :

- ✓ Les équipements ménagers domestiques.
- ✓ Les équipements industriels.
- ✓ petit appareil ménager

Elle assure également la production :

- ✓ Des appareils réfrigérateurs et congélateurs des différentes capacité à 520L).
- ✓ Des cuisiniers à gaz 4 et 5 feux, dont la production atteint 150 000 appareils par ans.
- ✓ Des climatiseurs types fenêtres et Split système (1CV à 2,5 CV) : à de 500 000 appareils par ans.

## B- Objectifs

Parmi les principaux objectifs de l'ENIEM nous citons :

- ✓ Mettre en place un système de management environnemental selon norme ISO 14001.
- ✓ Développer la formation et la communication.
- ✓ Développer les produits.
- ✓ Augmenter les productions
- ✓ Améliorer les chiffres d'affaires

**NB/** A savoir que l'ENIEM a atteint un taux de production de 97% des objectifs.

## II.3 Mode d'organisation

### II.3.1 Organisation générale

#### 3.1.1. Les unités

L'entreprise s'est organisée par centre d'activités stratégique qui se composent de 04 unités de production, d'une unité commerciale, d'une unité de prestations ainsi que d'une filiale dont le capital est 100% ENIEM. Actuellement, l'entreprise ENIEM est constituée de :

- ✓ La direction générale ;
- ✓ L'unité froid ;
- ✓ L'unité de cuisson ;
- ✓ L'unité de climatisation ;
- ✓ L'unité prestation technique (UPT) ;
- ✓ L'unité commerciale (UC) ;
- ✓ L'unité produit sanitaires ;
- ✓ La filiale lampes (FILAMP).

#### A. Unité froid

La mission globale de cette unité est de produire et développer des produits de froid domestique tels que :

- Réfrigérateur petits (160 à 240 litres) et grands modèles (300 à 520 litres);
- Congélateur vertical ou horizontal.

Le processus de fabrication est assuré par huit (08) ateliers qui sont :

- ✓ Atelier d'injection plastique ;
- ✓ Atelier presses et soudures ;
- ✓ Atelier de refondée et de mise en longueur ;
- ✓ Atelier de traitement et de revêtement des surfaces ;
- ✓ Atelier de fabrication de pièces métalliques ;
- ✓ Atelier de thermoformage ;
- ✓ Atelier de montage final.

Et un laboratoire central composé de trois (03) sections

- ✓ Laboratoire de chimie ;
- ✓ Laboratoire de métallurgie ;
- ✓ Laboratoire d'essais produits.

### **B. Unité cuisson**

Cette unité a pour mission ; la production et le développement des produits de cuisson à gaz, électrique ou mixte et tout produit de technologie similaire, elle produit des cuisinières à gaz 04 et 05 feux. Comporte quatre (04) ateliers de fabrication :

Atelier mécanique : s'occupe de la fabrication de composants d'alimentation en gaz et des différentes grilles de cuisinières ;

- ✓ Atelier tôlerie : s'occupe de la fabrication des différentes pièces en tôle ;
- ✓ Atelier d'assemblage.

Ainsi qu'un laboratoire d'essais gazinières.

### **C. Unité climatisation**

Cette unité fait dans la production et le développement des produits de climatisation, de chauffage et annexes :

- ✓ Equipements de climatisation individuels et collectifs
- ✓ Activités annexes : chauffe eau, chauffe bain et radiateur à gaz butane.

Composée essentiellement de quatre (04) ateliers de fabrication :

- ✓ Atelier tôlerie ;
- ✓ Atelier peinture ;
- ✓ Atelier montage final ;
- ✓ Atelier montage d'appareils de chauffage.

### **D. Unité de prestations techniques**

Chargée principalement de gérer et d'exploiter les moyens communs (production d'énergie et utilités) utilisés dans le processus de production des autres unités, ainsi que de la gestion des totalités des infrastructures communes (bâtiments, voirie, éclairage...).

Cette unité assure également ; les pièces mécaniques nécessaires à l'entretien des équipements de production, la conception et la fabrication de nouveaux moyens (moules, outils, gabarits...). Constituée d'ateliers de mécanique et de deux (02) stations :

- ✓ Station de production d'énergie et des fluides ; elle produit de l'eau surchauffée, de la vapeur et de l'air comprimé.
- ✓ Station de neutralisation ; s'occupe de traitement des rejets industriels avant leur évacuation. Et un laboratoire de métrologie qui se charge de l'étalonnage et de la vérification des instruments de mesure.

### **E. Unité commercial**

Cette unité est chargée de la commercialisation des produits de l'entreprise, de la promotion des exportations et de la gestion du réseau SAV (service Après Vente). Cette unité sera l'objet d'une étude plus détaillée loin dans ce chapitre car elle est notre domaine d'étude.

### **F. Unité produits sanitaires**

L'unité produits sanitaires est acquise par l'entreprise ENIEM en l'an 2000. Elle n'entre pas dans le périmètre de certification de l'entreprise. La mission de l'unité est

---

de produire et de développer des produits sanitaires (baignoires, lavabos et éviers...).

### **G. Unité FILAMP**

L'unité FILAMP de Mohammedia (ULM) a démarré en février 1979 pour fabriquer des lampes d'éclairage domestique ainsi que des lampes de réfrigérateurs. Elle est devenue filiale à 100% ENIEM le 01 janvier 1997.

### **II.3.2 Les directions :**

En plus des unités de productions, l'ENIEM est dotée d'une direction générale et de six directions centrales :

1. Direction générale ;
2. Direction industrielle ;
3. Direction des Finances et Comptabilité ;
4. Direction de Marketing et de la Communication ;
5. Direction Développement et Partenariat ;
6. Direction des Ressources Humaines ;
7. Direction Planification et Contrôle de gestion.

#### **II.3.2.1 Direction générale**

La direction générale, l'unique entité qui est responsable de la stratégie et du développement de l'entreprise. Elle exerce son autorité hiérarchique et fonctionnelle sur l'ensemble des directions et des unités.

#### **II.3.2.2 Direction industrielle**

Elle est chargée de développer et de mettre en place les moyens et l'organisation industrielle nécessaire à la réalisation de la production en agissant sur les approvisionnements, les moyens et les techniques de production.

- ✓ Définit les programmes de production en fonction de la demande commerciale et des capacités installées avec le souci de rentabilité optimale.
- ✓ Veille à l'optimisation et l'adaptation des approvisionnements en utilisant au mieux les capacités financières de l'entreprise pour assurer des stocks homogènes et productifs.
- ✓ Suit la réalisation des programmes de production et préconise des solutions d'adaptation en cas de difficultés.
- ✓ Améliore la gestion de la production en relation avec la structure informatique (GPAO).
- ✓ Entrepren et suscite des études de modernisation, de renouvellement, d'optimisation et d'installation des moyens de production.
- ✓ Prend en charge l'industrialisation des nouveaux produits ou modifiés dans le cadre du développement.
- ✓ Organise et anime l'industrialisation de nouveaux produits.
- ✓ Se tient informée des évolutions des techniques de fabrication des appareils électroménagers et les étudie avec l'opportunité de leur adoption.
  - ✓ Veille au renforcement des dispositifs de contrôle qualité à tout les stades de la préparation technique, de soutien et de la fabrication de produits et ce, en étroite collaboration avec les responsables qualité.
  - ✓ Définit une politique d'amélioration de la maintenance des équipements de productions et en assure le suivi.

### II.2.3 Direction des finances et comptabilité

Garant des obligations légales, des règles comptables et des procédures de l'entreprise, dont elle vérifie l'application par la mise en œuvre d'un contrôle interne. Cependant elle :

- ✓ Assure la comptabilisation, dans les délais, de toutes les opérations permettant l'établissement du bilan, du compte de résultat et des déclarations fiscale de l'entreprise.
- ✓ Recherche et mobilise dans les meilleures conditions de délai et de cout les besoins en ressources financières.
- ✓ Analyse les équilibres financiers de l'entreprise.
- ✓ Etudie et met en place la stratégie financière de l'entreprise (plan de financement à long terme).
- ✓ Définit la politique bancaire et l'orientation budgétaire.
- ✓ Gère la trésorerie (recette et dépenses).
- ✓ Contrôle les déclarations fiscales périodiques.
- ✓ Analyse les couts et les prix de revient.
- ✓ Met à la disposition des responsables opérationnels l'information financière nécessaire.
- ✓ Rédige, vérifie et approuve les dispositions décrites relatives au fonctionnement efficace de son activité.

La direction des finances et comptabilité est auditée, au moins une fois par an, par un commissaire au compte.

### II.2.4 Direction de marketing et de la communication

La direction du Marketing et de la Communication décide en collaboration avec le président directeur général, des politiques commerciales et de communication et les met en œuvre par la conception et l'élaboration des méthodes et outils de gestion nécessaires :

- ✓ Conduit les travaux d'études, d'analyse et de synthèse relative aux tendances et évolutions des marchés intérieurs et extérieurs.
- ✓ Elabore, en conformité avec la politique commerciale de l'entreprise, toute action concernant les schémas de distribution des produits finis, d'implantation d'antennes de vente au niveau national et international.
- ✓ Contribue avec les structures concernées de l'entreprise à l'élaboration des annuels et pluriannuels de production, de commercialisation et de développement.
- ✓ Participe à la politique de détermination des barèmes de prix.
- ✓ Elabore un plan de communication interne et le met en œuvre après approbation de la direction de l'entreprise.
- ✓ Elabore avec la direction commerciale le plan de communication externe et le met en œuvre après approbation de la direction de l'entreprise.
- ✓ Etablit les enquêtes clients en vue de mesurer le niveau de satisfaction de la clientèle.

- ✓ Initie et suscite des actions d'amélioration continue de la communication en
- ✓ relation avec l'environnement externe et médiatique de l'entreprise. Dirige toutes les opérations d'exportation de produits finis vers l'étranger.

### **II.2.5 Direction développement et partenariat**

Responsable des études et du développement des produits finis ainsi que des actions de partenariat et de sous-traitance. Ainsi elle :

- ✓ Définit et supervise les actions de développement du produit existant et l'élargissement de la gamme en fonction du marché.
- ✓ Suit avec direction industrielle les actions de développement des processus defabrication et de modernisation de l'outil de production, en vue de l'amélioration de la rentabilité et des conditions de travail.
- ✓ Participe à la définition de l'organisation de la production dans l'objectif de la flexibilité et de la réduction des couts de fabrication.
- ✓ Définit et concrétise des actions de sous-traitance et de partenariat.
- ✓ Développe d'autres créneaux pour l'utilisation maximale des capacités technologiques de l'entreprise.

### **II.2.6. Direction des ressources humaines**

En cohérence avec la politique qualité de l'entreprise, la fonction Ressources Humaines accroît la mobilisation et la valorisation du personnel dans actions au service du client. Elle pilote le recrutement, l'accueil, l'information et gère le plan de carrière du personnel. Elle conçoit le plan de formation à partir du recueil des besoins collectifs et individuels et s'assure de son exécution. Elle supervise la gestion administrative et légale pour le personnel et les pouvoirs publics en respectant les objectifs de conformité, de fiabilité et délais.

Elle doit en outre :

- ✓ Assure en place et l'adaptation des besoins tant quantitatifs que qualitatifs de l'entreprise en matière des ressources humaines et ce, en fonction de son développement et des conjonctures.
- ✓ Encourager les actions nécessaires à la rationalisation des effectifs et à l'émergence des compétences.
- ✓ Rédiger, vérifier et approuver les dispositions décrites, relatives au fonctionnement efficace de son activité.
- ✓ Définir et exécuter les plans de formation en fonction des besoins de l'entreprise en suivant les niveaux de qualification du personnel. - Gérer les éventuels conflits dans le respect de la loi et préparer les décisions de la direction.
- ✓ Se prononcer sur le choix de l'entreprise en matière d'organisation.
- ✓ Organiser et promouvoir le siège de l'entreprise.

### **II.2.7 Direction planification et contrôle de gestion**

Cette direction est responsable du contrôle de la gestion, de l'audit finance ainsi que du budget de l'entreprise. Cependant elle :

- ✓ Réalise et présente tous les travaux permettant de produire une information complète et cohérente des activités de l'entreprise (production, commercialisation, approvisionnement et finance).

- ✓ Exploite et analyse les informations relatives aux agrégats de gestion afin de préconiser les actions correctives nécessaires avec toute l'anticipation attendue.
- ✓ Planifie un programme annuel d'audits finance et organise sa réalisation.
- ✓ Exploite les résultats des audits finance, les interprète et fait les recommandations

nécessaires.

- ✓ Prépare, établit et suit le budget de l'entreprise.
- ✓ Contrôle et consolide les rapports d'activités.

#### II.4 Organigramme de l'entreprise ENIEM :

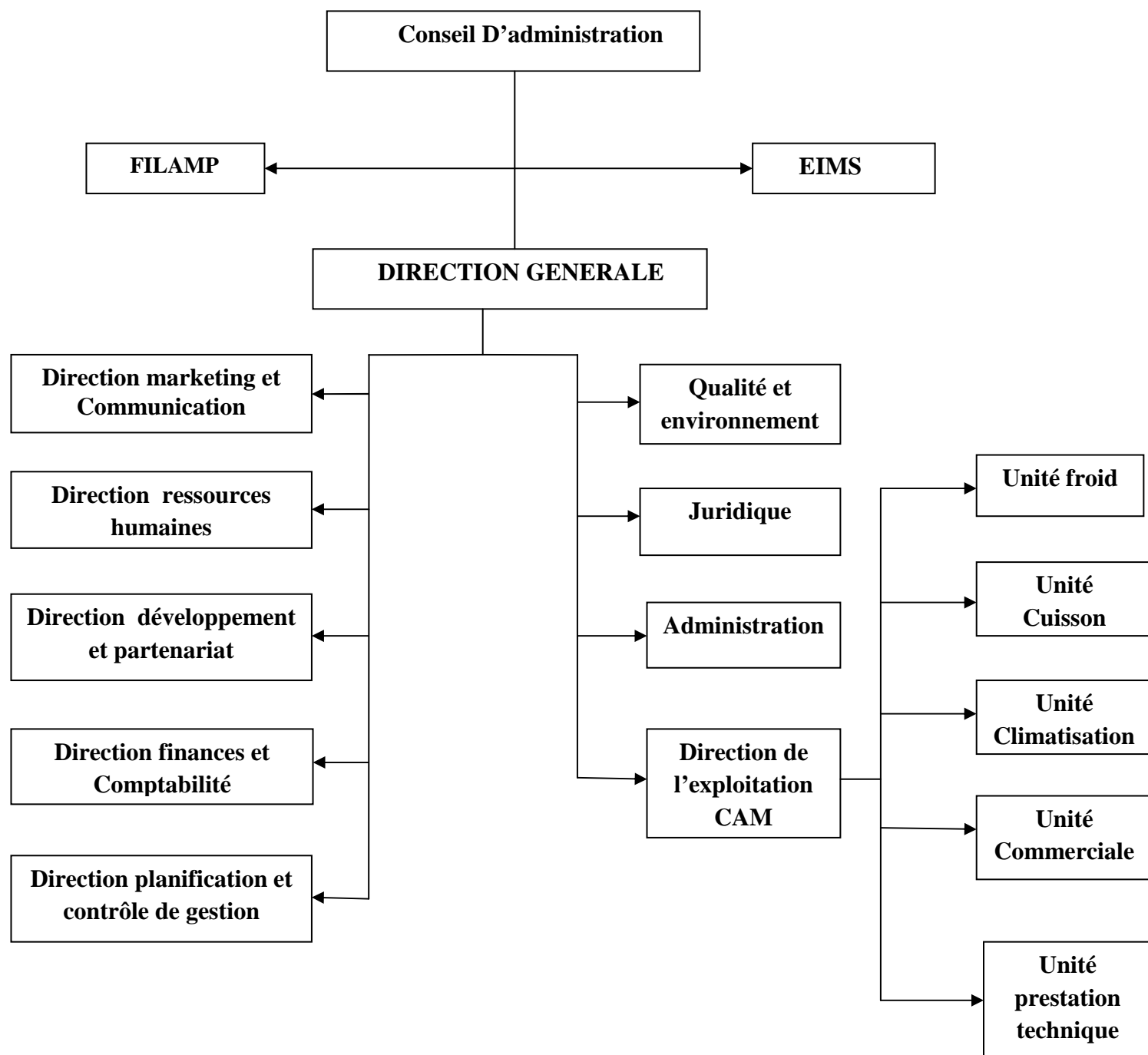




Figure II.1 organigramme générale de l'entreprise ENIEM

## II.4 Présentation de l'unité commerciale

### II.4.1. Historique et fonction

L'unité commerciale créée par l'entreprise le 02 janvier 1989, assure la fonction de commercialisation des produits finis de l'entreprise, de la promotion des exportations et de la gestion du réseau SAV (Service Après Vente) et pour cela, un réseau d'agents agréés a été mise en place.

### II.4.2. Organisation générale

Composée essentiellement, d'une direction commerciale au sommet de la hiérarchie à laquelle sont rattachés le secrétariat, l'assistant qualité et l'assistant juridique. Cette direction a sous sa tutelle sept départements qui collaborent pour mettre en œuvre la stratégie commerciale de l'entreprise. ces départements sont :

#### II.4.2.1. Département Vente

Trois services composent ce département ; un service vente (facturation), un service client et un service synthèse et recouvrement .

##### Service vente

chargé principalement de :

La réception des commandes des clients ;

- ✓ La gestion des facturations ;
- ✓ La gestion des règlements ;
- ✓ Le contrôle des livraisons par rapport aux facturations.

**Service client :** cette structure enregistre les clients et gère le réseau des agents agréés à travers le territoire national, et ce par :

- ✓ La définition des critères de sélection des agents agréés ;
- ✓ La réception et l'étude des dossiers de soumission ;
- ✓ L'établissement des conventions vente ;
- ✓ La classification et le suivi des clients.

##### Service synthèse et recouvrement :

le travail de ce service consiste en le recouvrement des créances lesquelles sont générées soit les ventes à terme, soit par les ventes par facilité, les ventes par traite ou bien par chèque. Ainsi que le traitement et le suivi des marchés.

#### II.4.2.2. Département Distribution

Composé de deux services ; le service magasin produits finis et le service programmation.

**Service magasin produits finis :** sa fonction principale est la gestion des stocks de l'entreprise. En outre il doit :

---

- ✓ Alimenter les stocks sur base de réceptions depuis les unités de production ;
- Gérer le stockage à l'extérieur (dépôts et entre dépôts) ;
- ✓ Réintégrer des produits retournés par le client dans le cadre de la garantie ou depuis les stocks extérieurs.

Notons que le service magasin produits finis est subdivisé en deux sections, en l'occurrence la section bâtiment 4 AS et la section bâtiment 6 qui sont les deux aires de stockage qui composent le magasin central de l'entreprise.

✓ **Service programmation** : ce service assure les livraisons vers le client par le biais de la section expédition qui se charge de l'établissement de bulletin de livraison en se basant sur la facture après avoir servi le client. Tandis que la section programmation, se basant sur les entrées et sorties établit de manière section programmation, se basant sur les entrées et sorties établit de manière périodique les situations des stocks ainsi la balance globale des stocks.

#### **II.4.2.3. Département Marketing**

Ce département met en œuvre la stratégie concurrentielle de l'entreprise, composé de chargés des foires qui s'occupent de préparation et organisation des foires et salons nationaux et internationaux ainsi que la réalisation des maquettes. Et de chargés de marchés qui font la prospection et l'étude du marché.

#### **II.4.2.4. Département Service Après Vente**

Ce département est responsable de la gestion des circuits SAV à travers tout le territoire national. Plus de 200 agents agréés forment le réseau SAV de l'ENIEM, ils sont à la charge du département service vente, qui doit les alimenter en pièces de rechange pour assurer les prestations service qui s'inscrivent dans le cadre de la garantie sur les produits de l'entreprise. Lors des retours clients, le département SAV, à travers une commission technique spécialisée décide de la réception ou remplacement du (des) produits retourné.

#### **II.4.2.5. Département Finance et Comptabilité**

Composé de deux services, un service comptabilité générale et un service finances, ce département assure la comptabilisation de toutes les opérations permettant l'établissement des bilans et des comptes des dépenses des autres départements en matière de prestations techniques à base desquelles sont définis les prix de vente et les taxes sur les produits finis.

#### **II.4.2.6. Département ARGH**

Le département administration générale et ressources humaines est formé du service gestion du personnel qui gère les paies, les emplois, la formation et les correspondants sociaux. Et du service, moyens généraux qui se charge de la gestion des affaires générales, les achats et des relations extérieures.

#### **II.4.2.7. Département Contrôle de Gestion**

Ce département est un organe de contrôle de toute la gestion de l'unité commerciale. Il réalise et présente tous les travaux permettant de produire une information complète et cohérente des activités de l'unité (production, commercialisation, approvisionnement et finance). Il a accès pratiquement à toutes les informations produites par les autres services ou départements. Alors qu'il exploite et analyse l'information relative aux agrégats de gestion afin de

préconiser les actions correctives nécessaires avec toute l'anticipation attendue, il exploite les résultats des audits finance les interprète et fait les recommandations nécessaires.

#### II.4.3.organigramme de l'unité commerciale

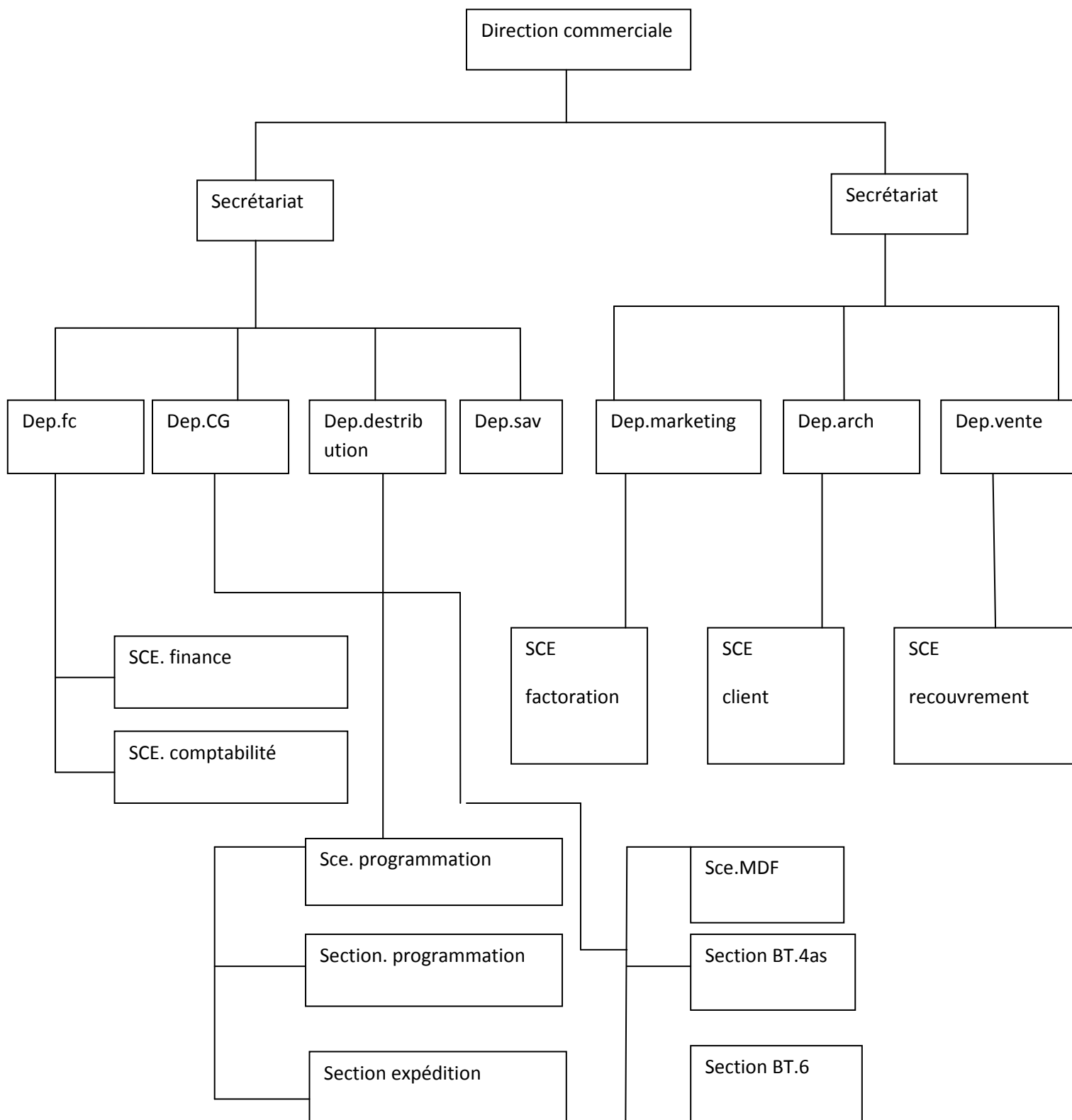


Figure II.2 organigramme de l'unité commercial

### **II.5 LA Comptabilité**

#### **II.5.1 Introduction**

Toute entreprise a besoin de connaître à tout moment l'état de ses affaires et de son patrimoine. La comptabilité répond à ce besoin; elle peut être définie comme une méthode d'observation des faits économiques qui se produisent dans l'entreprise, tels que des achats et des ventes de marchandises, des paiements à des fournisseurs ou émanant de clients, ... .

#### **II.5.2 Organisation pratique**

les achats (marchandises, services, ...)

- les ventes ;
- les opérations financières (paiements, ...) ;
- les autres, dites opérations diverses ou OD (imputation des salaires, opération d'inventaire, ...).

A chacun de ces groupes vont correspondre des journaux auxiliaires spécialisés :

- journal des achats ;
- journal des ventes ;
- journaux financiers (banque, caisse, ...) ;
- journal des OD.

Périodiquement (mois, semaine, jour), voire immédiatement, les totaux de ces journaux sont retranscrits dans le livre central et dans les comptes du grand livre.

Cette opération s'appelle la centralisation. Notez que dans les systèmes informatisés, l'écriture dans les comptes s'effectue au fur et à mesure.

#### **Conclusion**

la présentation de l'organisme d'accueil nous a permis de mettre en évidence notre domaine d'étude . L'étude de ce dernier sur le plan information ainsi que les insuffisances et les anomalies qui entravent le bon fonctionnement de service concerné .

# Chapitre III

## Analyse et conception

### III.1 Introduction

La conception de toute solution logicielle doit être traitée avec précision et détail, précédé d'une analyse profonde et bien réfléchi, car elle est le reflet du système avant même sa concrétisation. Dans le but d'avoir une meilleure analyse et de rendre la conception de notre application plus complète, nous avons adopté le langage UML (Unified Modeling Language) qui permet de bien présenter l'aspect statique et dynamique d'une application par une série de diagrammes qu'il offre.

### III.2 présentation des diagramme UML.[18]

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

Un diagramme est une représentation graphique qui s'intéresse à un aspect précis du modèle.

UML offre 13 types de diagrammes. Chaque type de diagramme offre une vue d'un système. Combinés, les différents diagrammes offrent une vue complète d'un système.

Les diagrammes UML peuvent être classés sous deux grandes catégories :

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

#### III.2.1 les diagrammes de structures ou statique

**1. Les diagrammes de classes :** Sans doute les diagrammes les plus utilisés d'UML, ils décrivent les types des objets qui composent un système et les différents types de relations statiques qui existent entre eux.

Les diagrammes de classes font abstraction du comportement du système.

**2. Les diagrammes d'objet :** Représente un instantané des objets d'un système à un moment donné. Ces diagrammes sont souvent appelés diagrammes d'instances car ils représentent des instances et non pas des classes.

**3. Les diagrammes de composants :** décrivent l'architecture interne d'une classe. Ils sont très utilisés pour représenter l'architecture physique et statique d'une application en termes de modules (fichiers, sources, librairie, exécutables,...) pour montrer la mise en œuvre physique des modèles statiques avec l'environnement de développement.

**4. Les diagrammes de déploiement :** représentent l'agencement physique d'un système montrant sur quels composants matériels les différents composants logiciels s'exécutent.

Les principaux éléments sont des nœuds connectés par des voies de communication. Un nœud représentera une unité qui va héberger un logiciel, un équipement matériel ou un ordinateur,

Un environnement d'exécution, un système d'exploitation ... etc.

Les nœuds contiennent des artefacts (implémentation d'un composant) tel que des fichiers (.exe, .dll, .scripts, ... etc).

**5.les diagramme de structure composite :** font parties des nouveautés ramenées par UML 2.0 Ils permettent de décomposée hiérarchiquement une classe en une structure interne .

**6.les diagrammes de package :**permettent de grouper des éléments dans des unité de plus haut niveau .En effet ,il peuvent servir a organiser des élément UML selon leur types ,leurs fonctionnalités ou leur architectures.

Ils sont plus souvent utilisés pour regrouper des classes.

**III.2.2 les diagrammes de comportement :**qui sont au nombre de sept :

1.les diagrammes d'activités : décrivent le comportement d'une méthodes, le déroulement d'un cas d'utilisation ,les enchainement d'activités une activité désigne une suite d'action .le passage d'une action vers une autre est matérialisé par une transition les transition sont déclenchées par la fin d'une action et provoque le début immédiat d'une autre (elle sont automatique ).

**2.les diagramme de cas d'utilisation :** les cas d'utilisations (en anglais use cases) ont été introduits par Ivar jacobson dans sa méthode **OOSE** (Oriented Object Software Engeneering)

Ils constituent une technique qui permet de déterminer les besoins des utilisateurs et de capturer les exigences fonctionnelles d'un system .en d'autres termes .ils décrivent le comportement d'un système de point du vue de d ses utilisateurs .ils décrivent les interaction entre les utilisateurs d'un système et le Systems lui-même .et **un diagramme de cas d'utilisation** permet de représenter graphiquement les cas d'utilisation . le faite q'un acteuere déclenche un cas d'utilisation est représenter par une flèche entre ces deux derniers .

3.les diagrammes de machines d'état : appelés diagrammes d'état-transitions dans UML 1.x permet de décrire le comportement d'un objet durant son cycle de vie .

Ils permet plus précisément de décrire les changements d'états d'un objet ,en réponse aux interaction avec d'autre objets ou acteur .

**4.les diagramme de séquence :** permettent de représenter les interaction entre objets selon un point de vus temporel. L'accent est mis sur la chronologie des envois de messages.

**5.les diagrammes de communication :**appelés diagrammes de collaboration dans UML 1.X ,ils décrivent l'interaction .

6.les diagrammes de vue d'ensemble des interactions :donne une vus d'ensemble des interactions en combinant des diagrammes d'activités et des diagrammes de séquences .En mettant l'accent sur les liaisons des données entre les différents participants a l'interaction .

**7.les diagrammes de timing :**permettent de définir des contraintes temporelles pour un ou plusieurs objets .

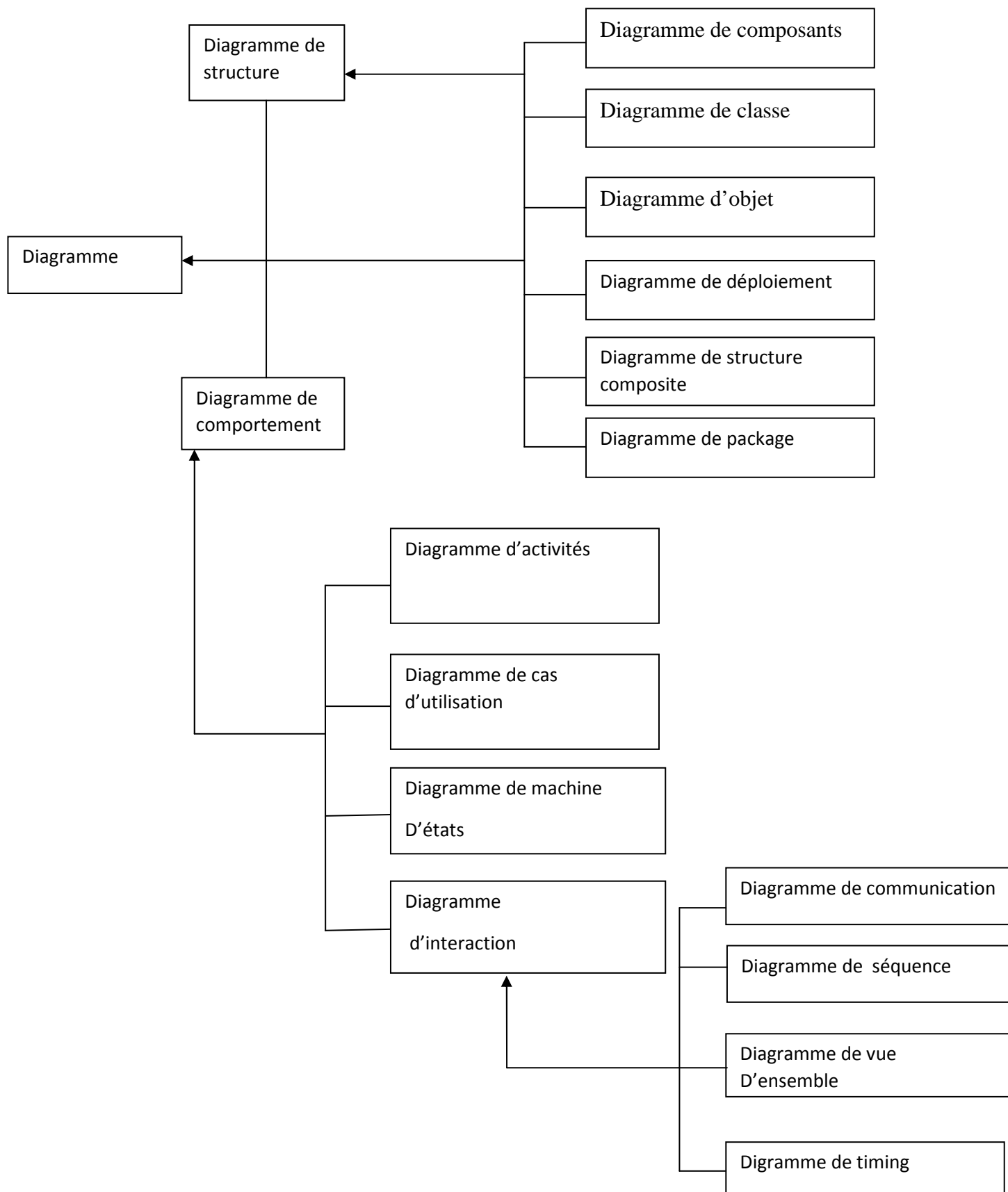


Figure III.1 Classification des diagrammes UML( v.2.0)



Nous remarquons dans la figure III.1 extraite de (Fowler, 2004), que les diagrammes de comportement sont aux mêmes classes en deux types de diagrammes. On distingue les diagrammes d'interaction représentés par : les diagrammes de séquences d'activité, de cas d'utilisation. Nous pouvons considérer que les diagrammes de cas d'utilisation et d'activité décrivent plus le fonctionnement de système alors qu'un diagramme de machine d'état décrit la dynamique d'un objet en termes de changements d'états.

### III.3 une démarche pour l'analyse et la conception du projet.

#### III.3.1 phase d'analyse

- **Analyse des besoins :** cette phase consiste à étudier l'existant en recensant toute l'information nécessaire sur le champ d'étude.
- **Diagramme de cas d'utilisation :** les cas d'utilisation permettant de décrire les fonctionnalités de système et leur interaction avec le ou les utilisateurs. La description des interfaces pourrait très bien se faire à partir de ce point.

#### III.3.2 phase de conception

- **Diagrammes de séquences :** ces schémas permettant de détailler tous les scénarios possibles d'un cas d'utilisation en précisant les objets du système impliqués dans le scénario. L'affichage des interfaces peut se faire à ce niveau, aux bien aux niveaux de diagramme de collaboration.
- **Diagramme d'activité :** permet de modéliser l'aspect dynamique du système et en donnant une représentation des différents processus faisant intervenir les acteurs concernés et manipulant les ressources nécessaires.
- **Diagramme de classe :** permettra d'établir un schéma conceptuel de la partie statique du système d'information en décrivant les classes et les associations entre classes d'information.

#### III.3.3 phase de réalisation

C'est à partir du diagramme de classe que l'on pourra dériver le modèle relationnel de la base de données à implémenter. Les diagrammes d'activité permettant de mettre en œuvre les interfaces logicielles nécessaires à l'exploitation de la base de données.

### III.4 cahier de charge de notre application :

Notre projet porte sur la mise en place d'une application de comptabilité en basant sur l'architecture JEE. Elle assurera un environnement interactif afin de rendre les tâches de gestion accessibles via un réseau local dans le but de maîtriser les différentes fonctions de base du système qui sont :

- S'authentifier.
- accéder à un espace privé.
- changement des mots de passe.
- ✓ **pour le client**
  - enregistrement d'un client
  - modification de paramètres d'un client
  - suppression d'un client
  - consultation des listes clients

**✓ pour le fournisseur**

- enregistrement d'un fournisseur
- modification de paramètre d'un fournisseur
- suppression d'un fournisseur
- consultation des fournisseur

**✓ comptabilité générale**

- création de compte
- modification de paramètre d'un compte
- suppression d'un compte
- consultation des compte
- validation des facture
- faire le schéma d'écritures
- impression d'une facture

**✓ profil société**

- ajouter une période
- ajouter journal
- ajouter centralisateur client
- ajouter centralisateur fournisseur
- supprimer centralisateur fournisseur
- supprimer journal
- supprimer centralisateur client

**III.5 Analyse et conception de notre system**

Consiste a déterminer ce que l'application devras faire ,l'expression de son comportement et de son architecture ainsi que l'examen des cas d'utilisation et leurs scénarios .

**III.5.1 Analyse :**

Cette activité commence par la mise en évidence des différents acteurs intervenant dans le system cible ainsi que leurs besoins. Ensuite la phase conception donnera la modalisation des objets a atteindre en s'appuyant sur les résultat de la phase analyse .

**III.5.1.1 Spécification des besoins**

Aux cors de notre passage par déferrent services de l'entreprise de l'ENIEM ,difficulté dans la recherche d'information l'importance charger dans les travaux et l'exécution quotidienne des même traitement .suite a ces problèmes nous avons pu tracer l'objectif d'améliorer le fonctionnement et l'organisation de system actuels.

**III.5.1.2 cas d'utilisation :**

Les cas d'utilisation sont des outil formels qui permettre de consigner et d'exprimer les interaction et les dialogues entre le system et ces utilisateurs l'ensemble des cas l'utilisation donne une description détaillé de comportement de system et renonce sollicitations de ces de derniers. Un cas d'utilisation doit exprimer le system ce que doit faire sans préjuger de façon dont cela seras fait .

avant de décrire les différentes cas d'utilisation, il est nécessaire de mettre en évidence les acteurs utilisant le système ainsi que leur tâche respective et les scénarios qui les décrivent. On regroupe l'ensemble de ses scénarios nous obtenant les différents cas d'utilisation.

### III.5.1.3 Spécification des tâches

Le comptable effectue certaines tâches qui sont résumées dans le tableau ci-dessous

Acteur	Tâches
Comptable	T1 : Authentification pour accéder à un espace privé T2 : création de compte T3 : suppression des comptes T4 : modification de compte T5 : changer le mot de passe T6 : consultation de compte T7 : établissement des factures T8 : faire le schéma d'écritures T9 : ajouter des clients T10 : suppression d'un client T11 : modifier les paramètres d'un client T12 : consultation des paramètres client T13 : ajouter des fournisseurs T14 : suppression des fournisseurs T15 : impression des factures T16 : ajouter période T18 : ajouter centralisateur client T19 : suppression centralisateur client T20 : ajouter centralisateur fournisseur T30 : suppression centralisateur fournisseur T31 : ajouter journal T32 : suppression journal

## III.2 Spécification des tâches

### III.5.1.4 Diagramme des cas d'utilisation globale

Il décrit le comportement du système du point de vue utilisateur sous la forme d'action et de réactions. Il existe deux concepts fondamentaux dans la modélisation par cas d'utilisation :

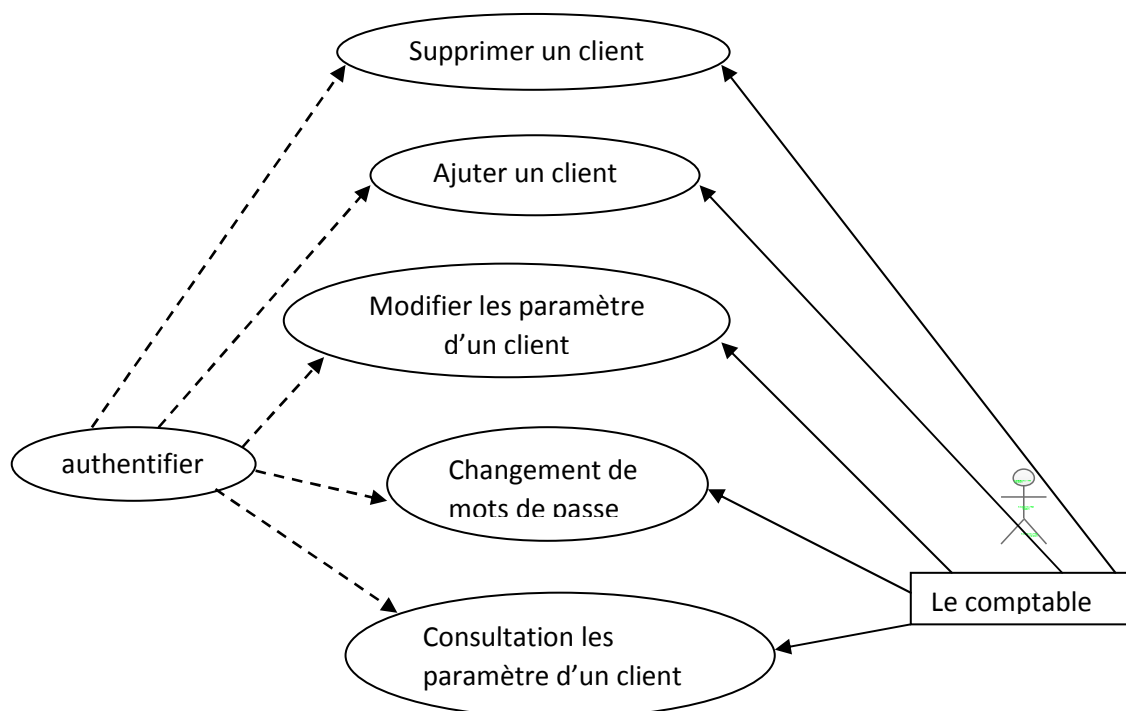
- les acteurs qui utilisent le système
- les cas d'utilisation qui représentent l'utilisation de système par les acteurs.

Chaque cas d'utilisation indique une fonctionnalité du système déclenchée par un acteur

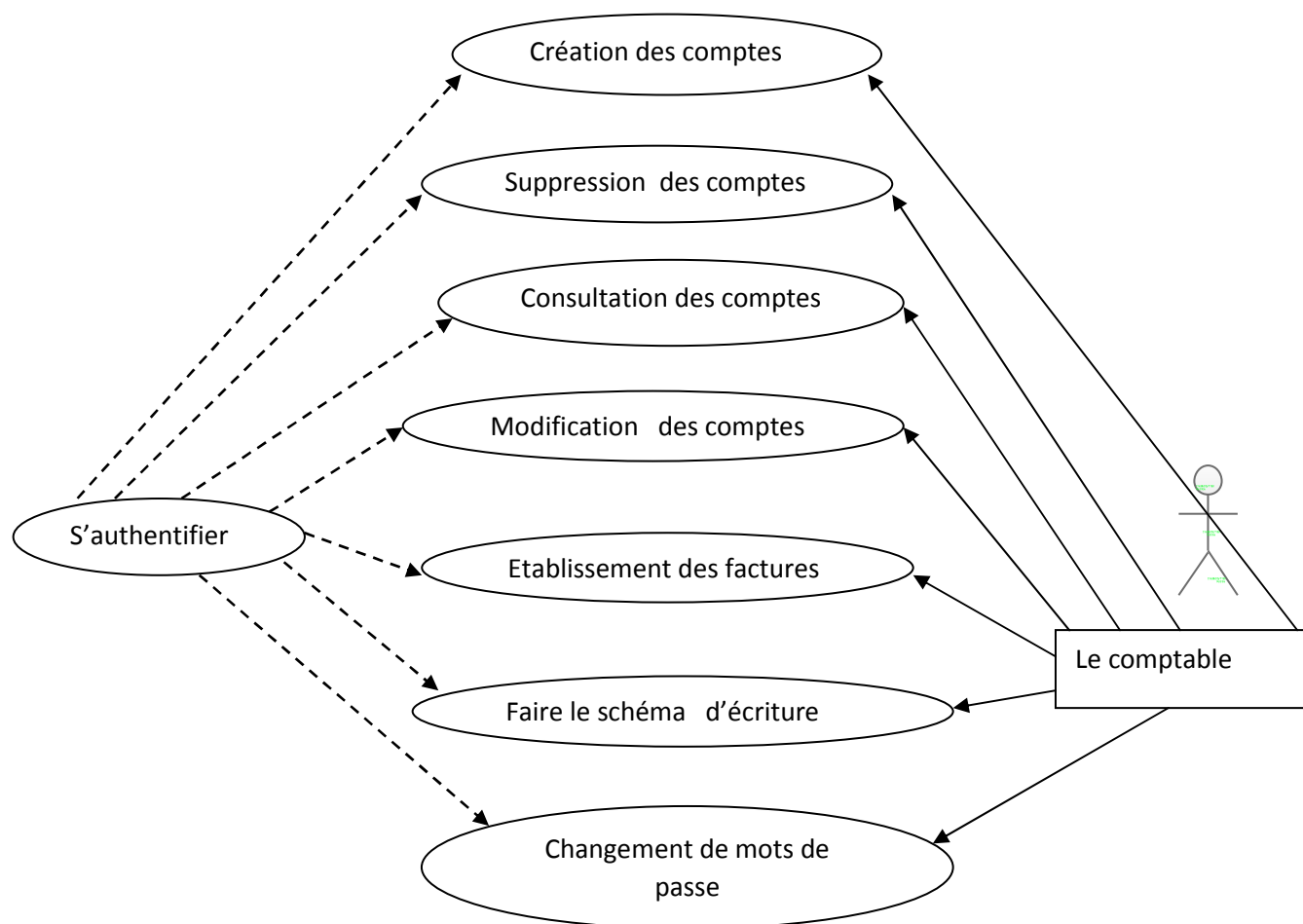
externe de système .ce genre de diagramme permet de permet en place et de comprendre les besoins des utilisateurs .

les cas d'utilisation peuvent être structurer .En plus de la relation de communication ,qui consiste au déclenchement d'un cas d'utilisations par un acteur, nous pouvons citer deux types de liens ou relations qui sont les plus utilisés [1] :

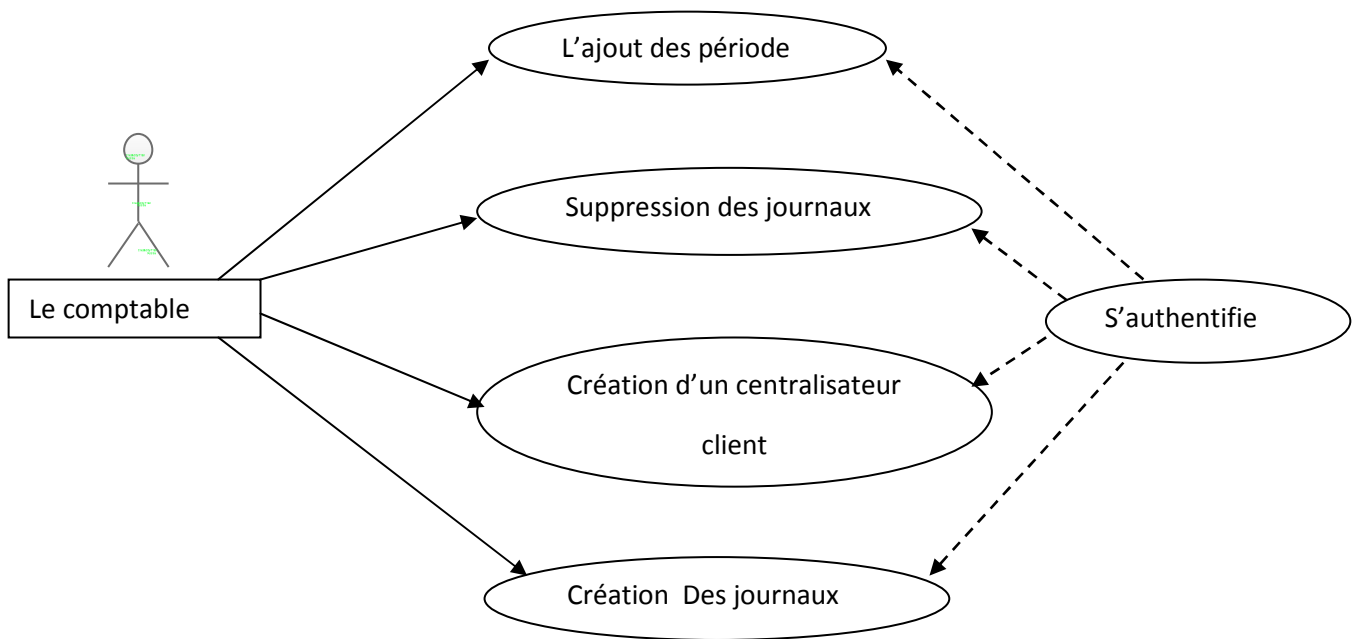
- **le lien d'utilisation** : ce lien nommé « utilise» (ou « include » en anglais) indique que le cas d'utilisation source contient le comportement décrit dans le cas d'utilisation destination
- **le lien d'extension** : ce lien indique que le cas d'utilisation source « étend » (en anglais « extend ») ou précise le cas d'utilisation destination .



**Figure III.4 diagramme de cas d'utilisation pour l'espace client**



#### III.4 Diagramme de cas d'utilisation de comptable pour l'espace comptabilité générales



**Figure III.5 Diagramme de cas d'utilisation pour le profil société**

### III.5.2 Conception

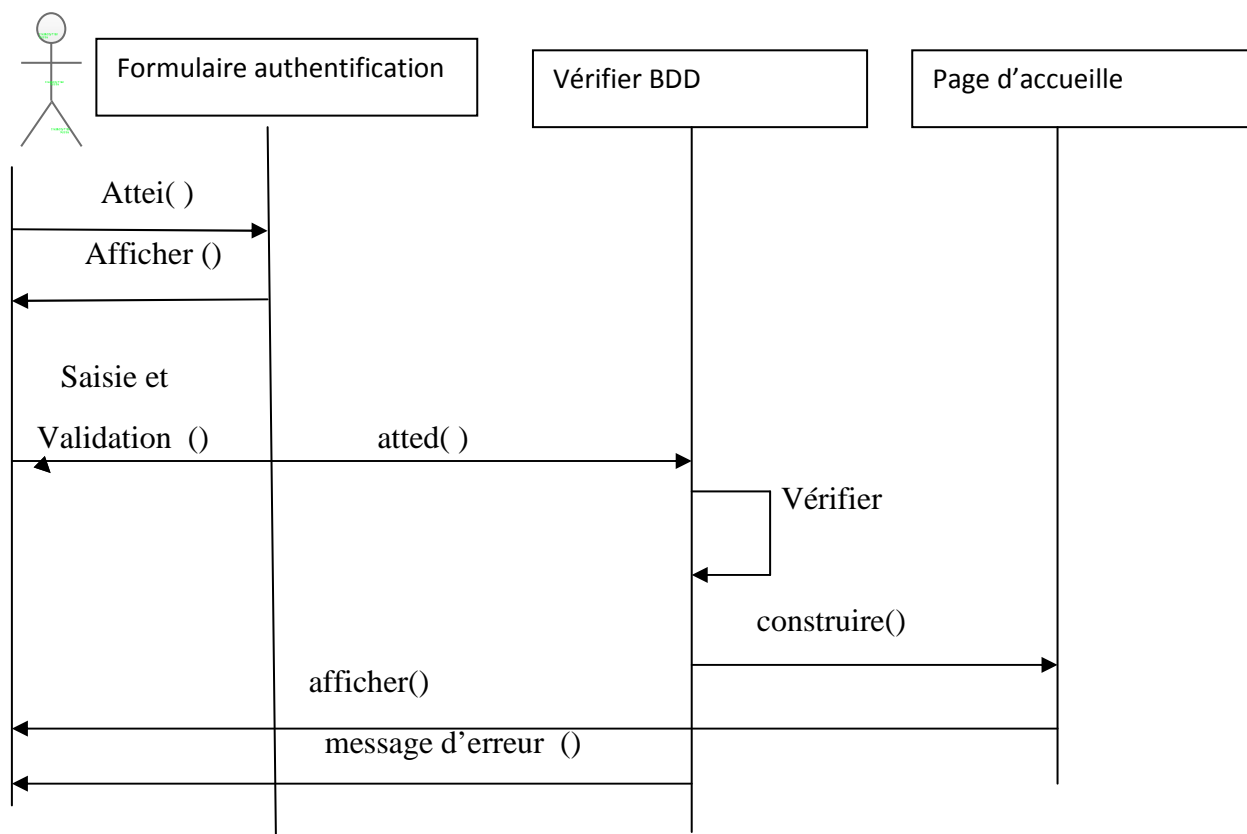
Après avoir spécifier les besoin de l'application et détermine les cas d'utilisation nous allons passe a la phase de conception du projet qui consiste a affiner les spécification d'après leurs contenus .

La conception des application web se distingue de celle d'autre système par deux activité majeurs :

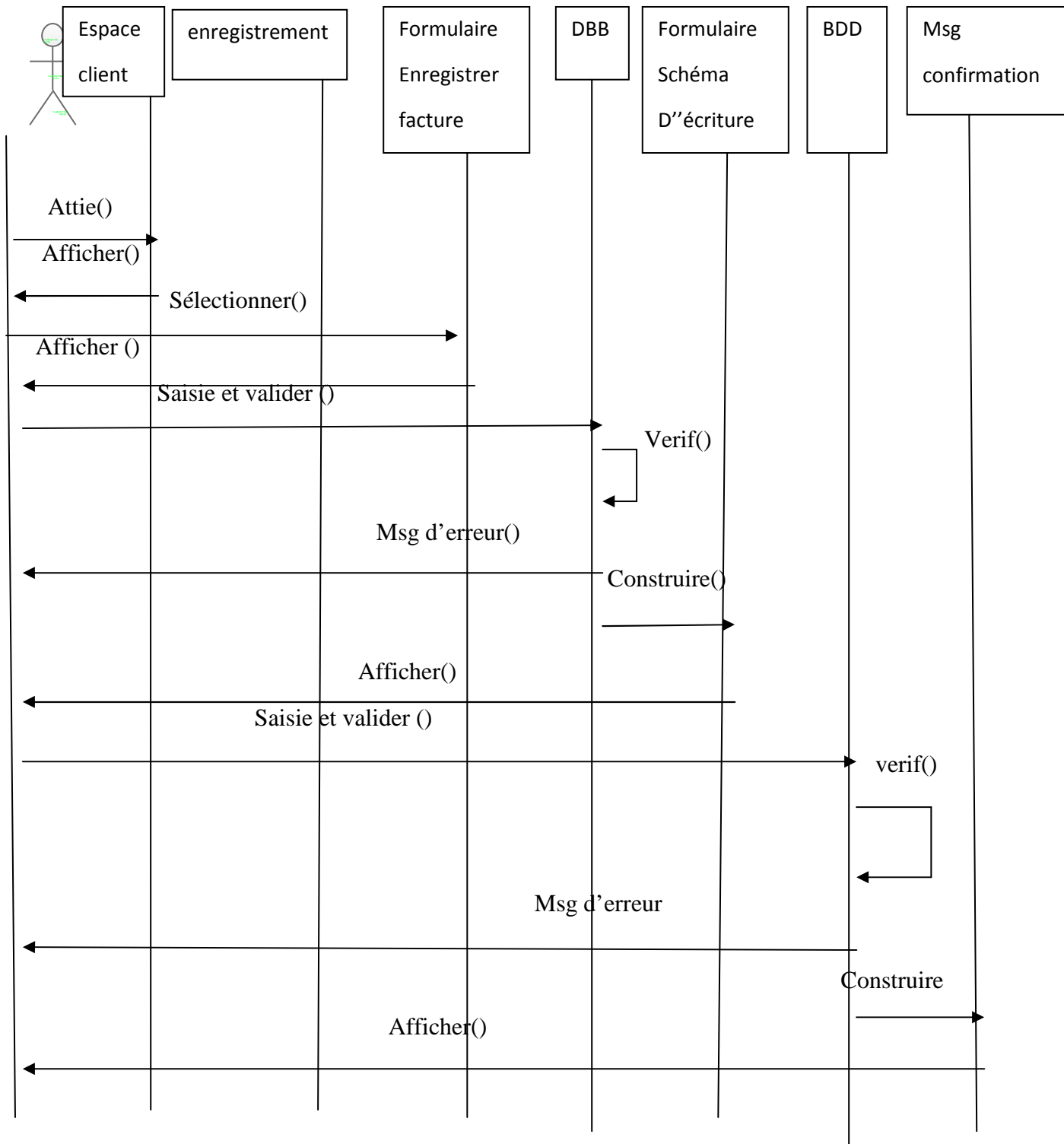
#### III.5.2.1Elaboration de diagramme de séquence

Dans l'étape suivante nous avons choisi de modéliser l'interaction entre le system et les acteur grâce aux diagramme se séquence .vue le nombre élevé de cas d'utilisation recensés, et afin d'éviter qu'ils s'étalent sur plusieurs pages ,nous avons décidé d'en etudier quelque diagramme.

Diagramme de séquence du cas l'utilisation « authentification » de comptable :

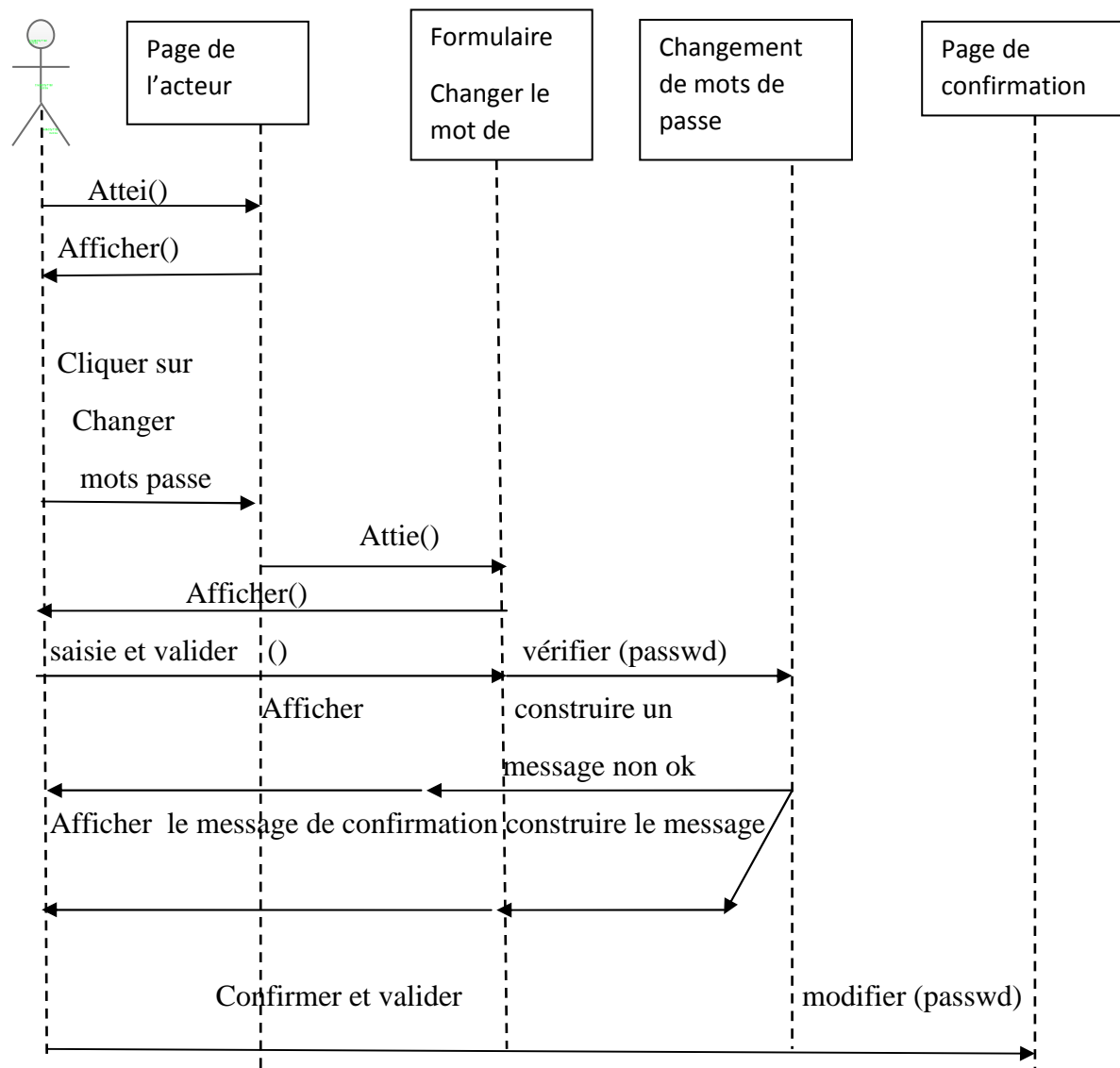


### III.6 Diagramme de séquence de cas d'utilisation « authentification »

**Diagramme de séquence de cas d'utilisation « enregistrement facture »**Figure III.7 **Diagramme de séquence de cas d'utilisation « enregistrement facture »**



## diagramme de séquence de cas d'utilisation « changer le mot passe »



## III. 9diagramme de séquence de cas d'utilisation « changer le mot passe »

Diagramme de séquence de cas d'utilisation « création de compte » :

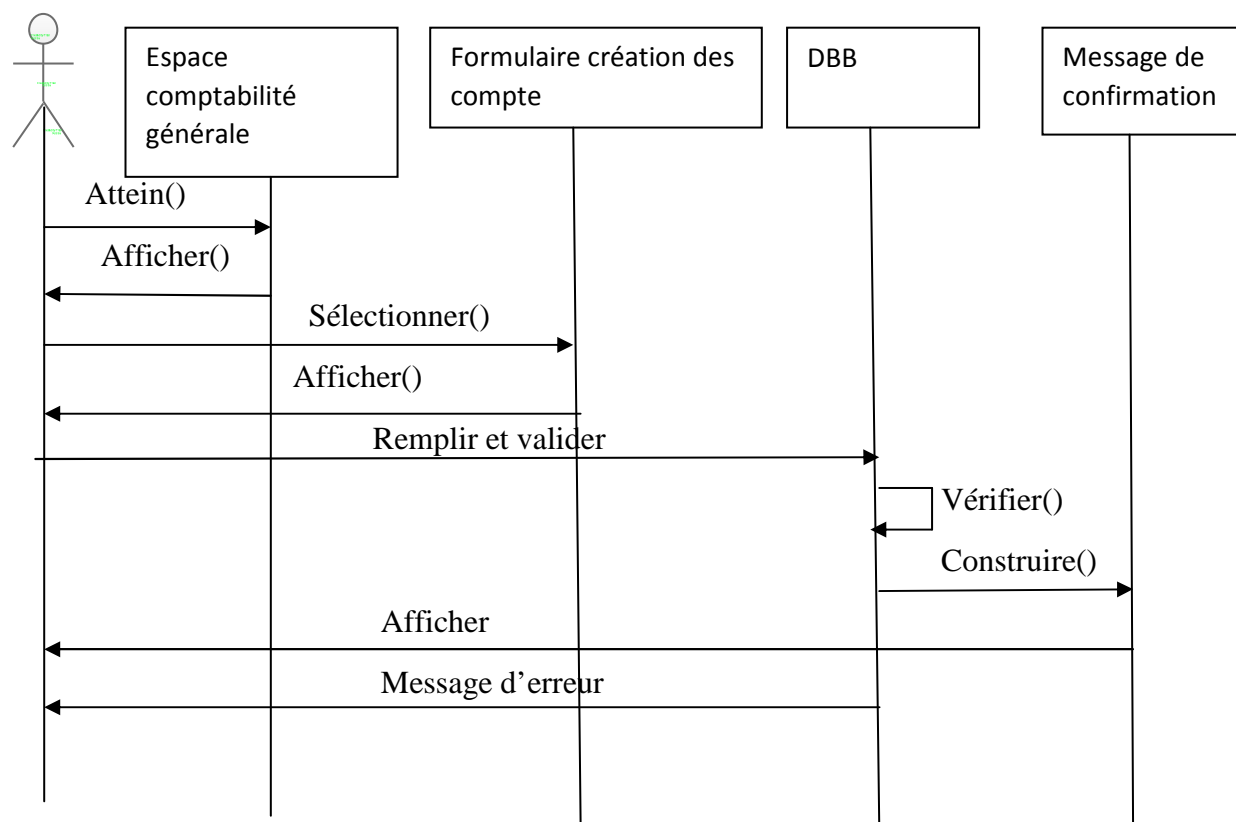


Figure III.10 Diagramme de séquence de cas d'utilisation « création de compte »

Diagramme de cas d'utilisation « voir les compte »

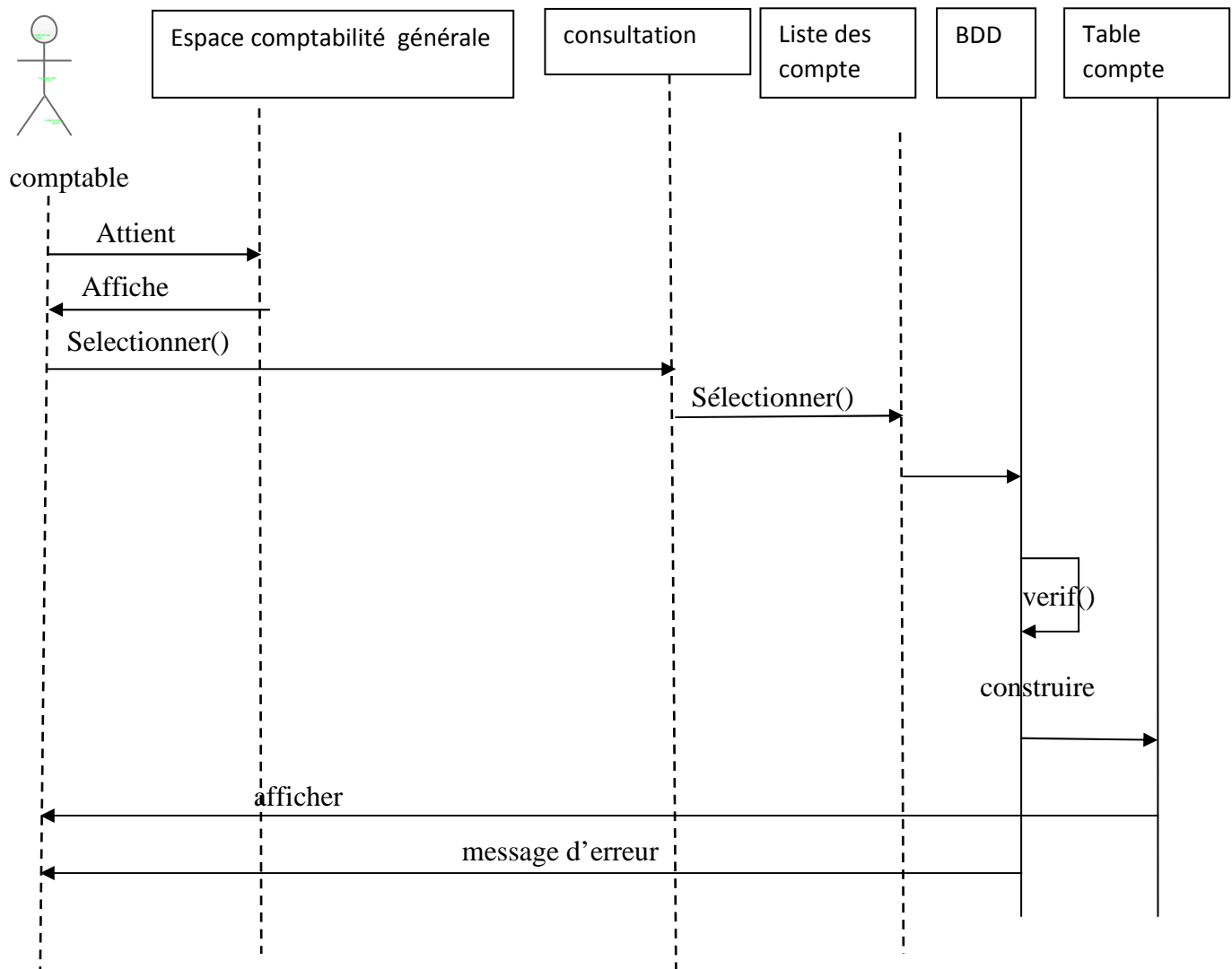
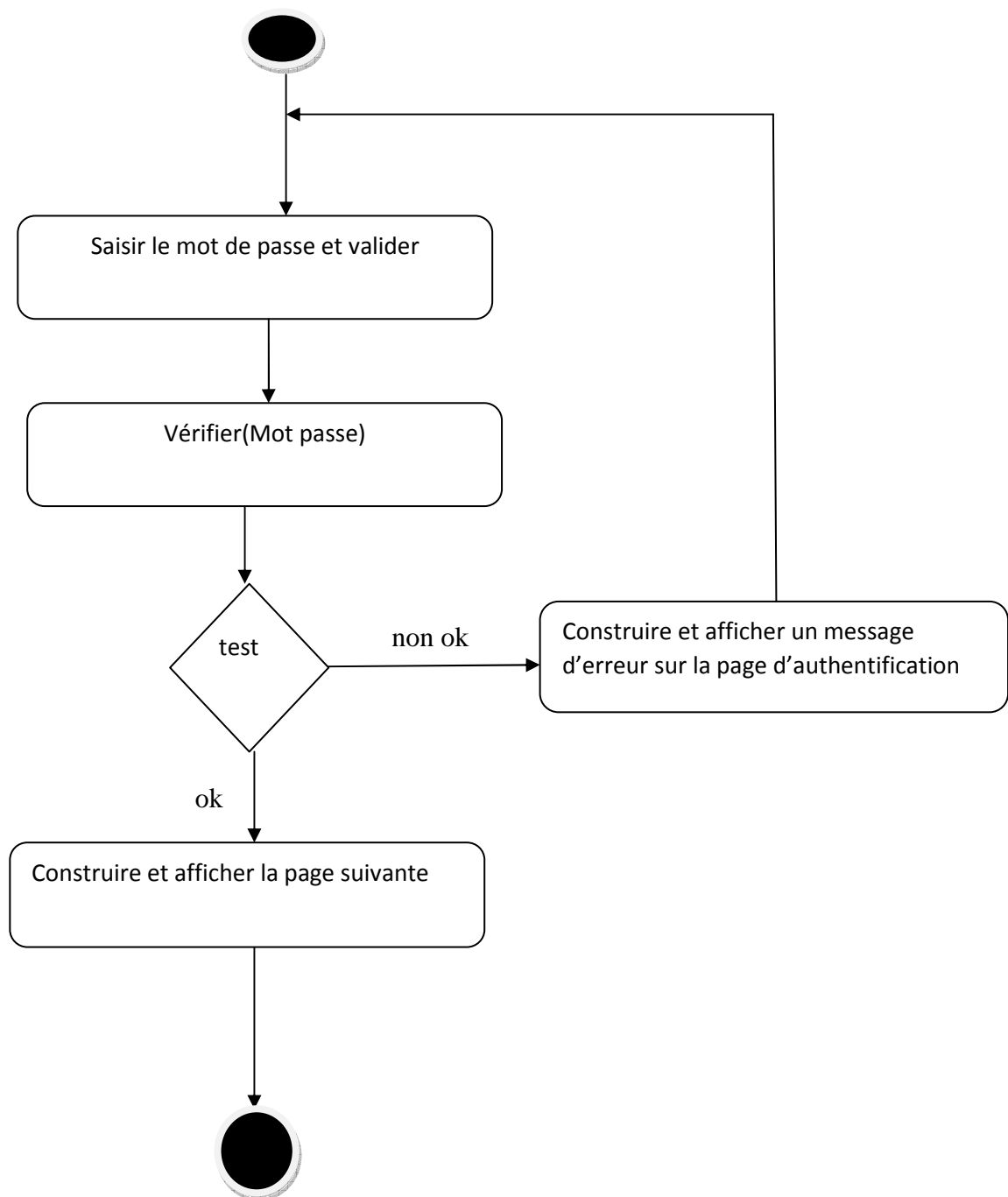


Figure III.11 Diagramme de séquence de cas d'utilisation « voir les compte »

### III.5.2.2 Diagramme d'activité :

Le diagramme d'activité est l'un des diagrammes d'UML qui sont conçus pour la modélisation et aspect dynamique de system .un diagramme d'activité représente l'état de l'exécution d'un mécanisme ,sous forme d'un déroulement d'étapes regrouper séquentiellement dans des branche parallèles de flux de contrôle.

**Diagramme d'activité de cas d'utilisation « authentification »****Figure III.12 Diagramme d'activité de cas d'utilisation « S'authentifier »**

## Diagramme d'activité de cas d'utilisation « Edition des facteur»

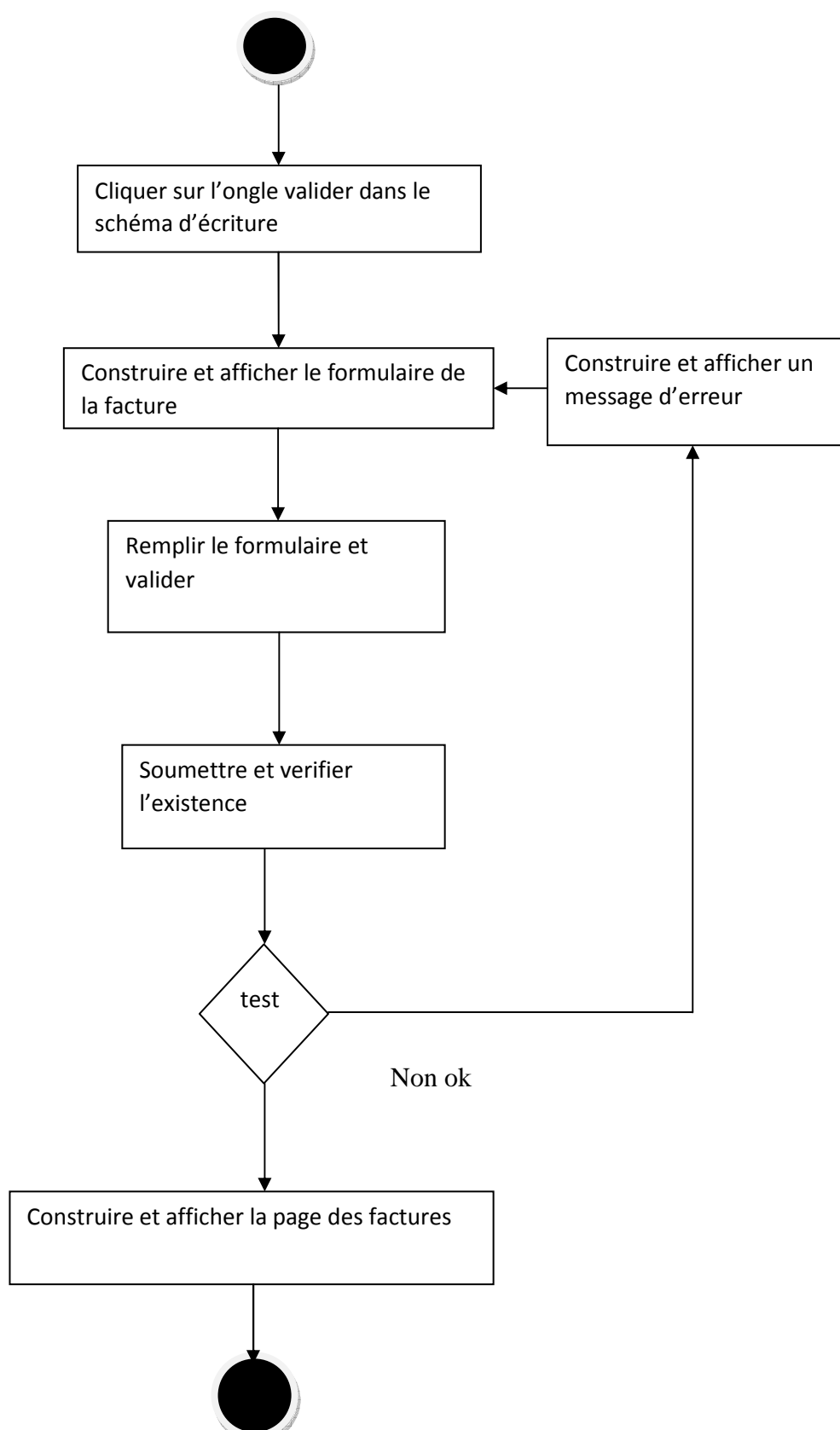
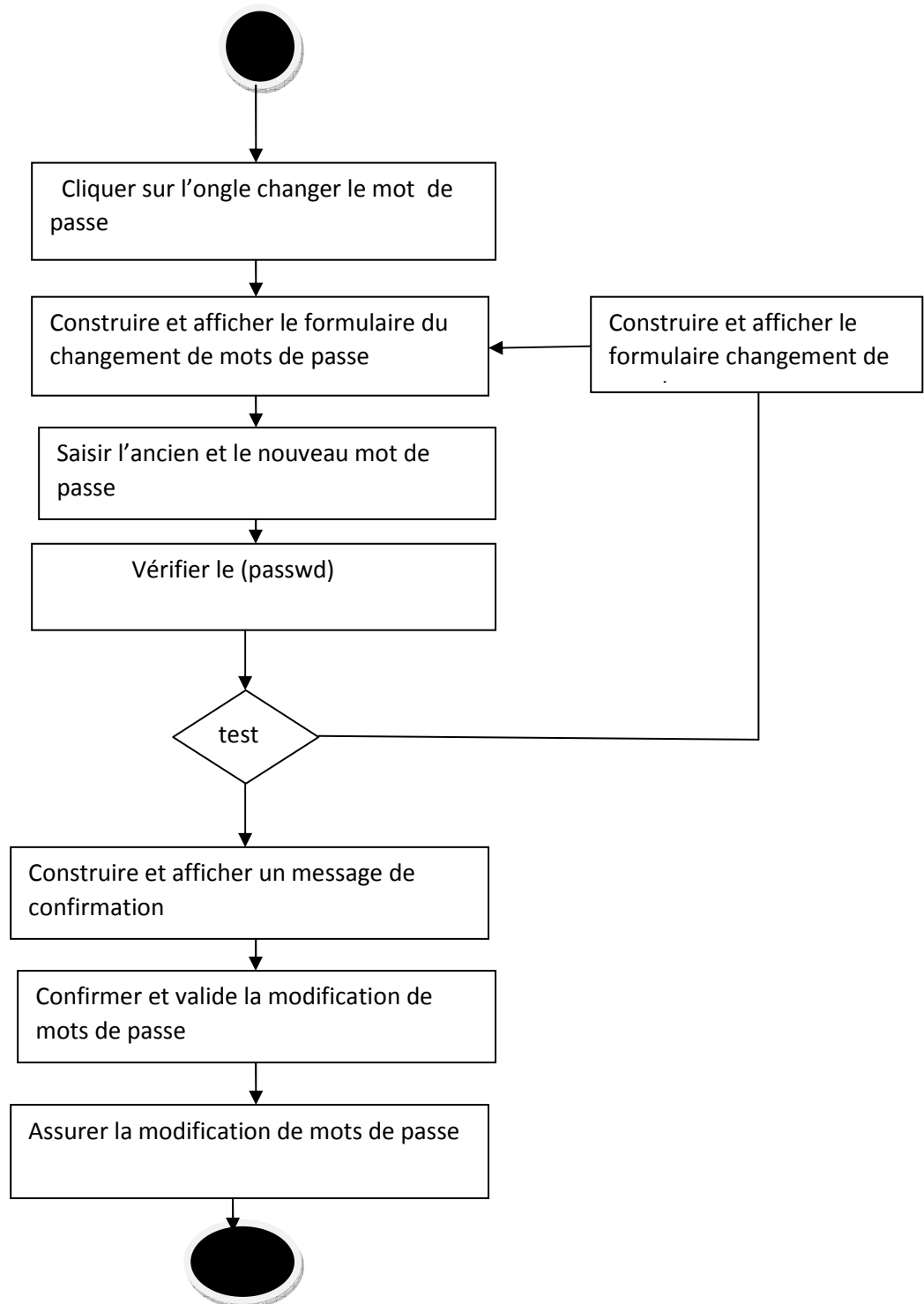
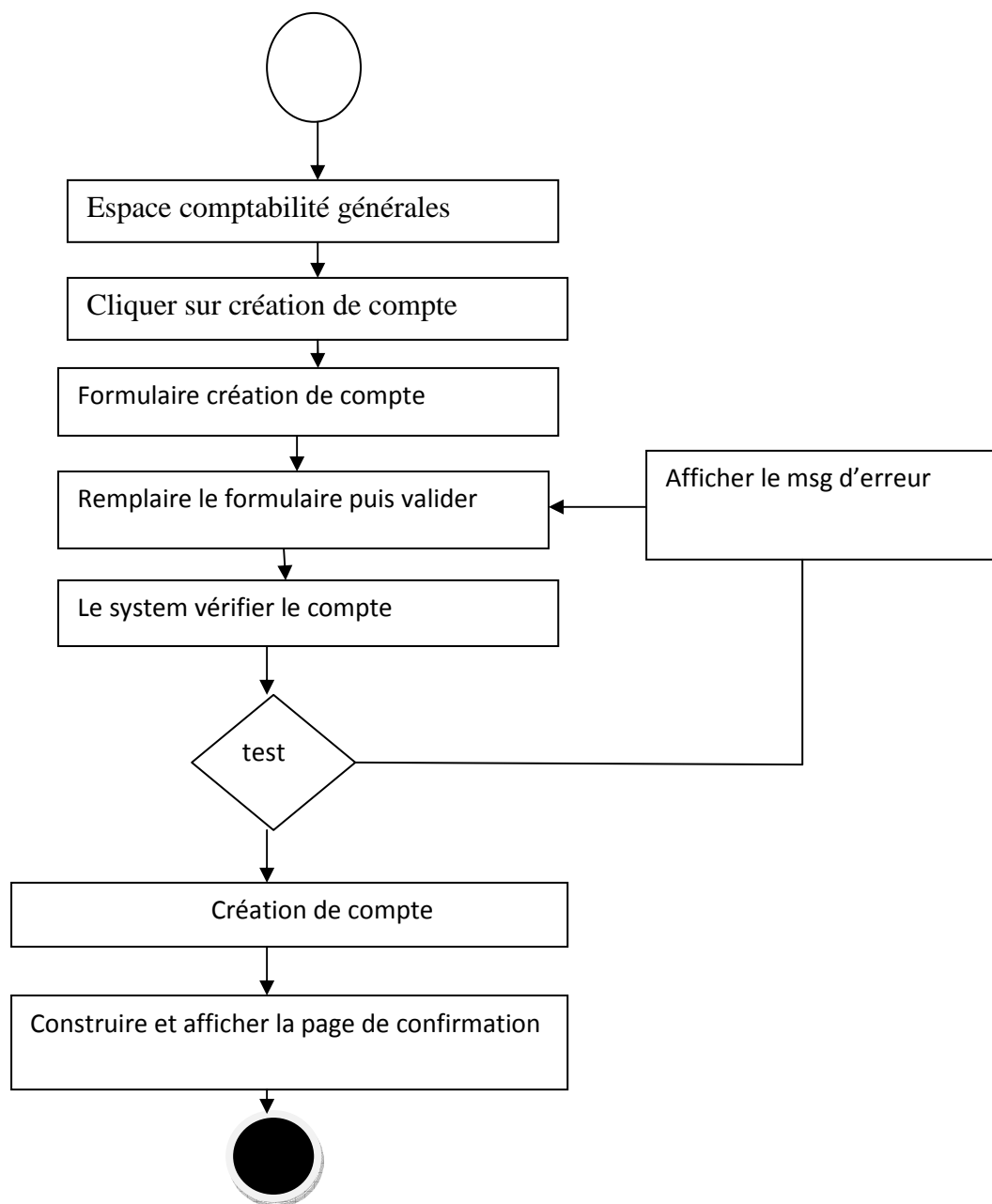
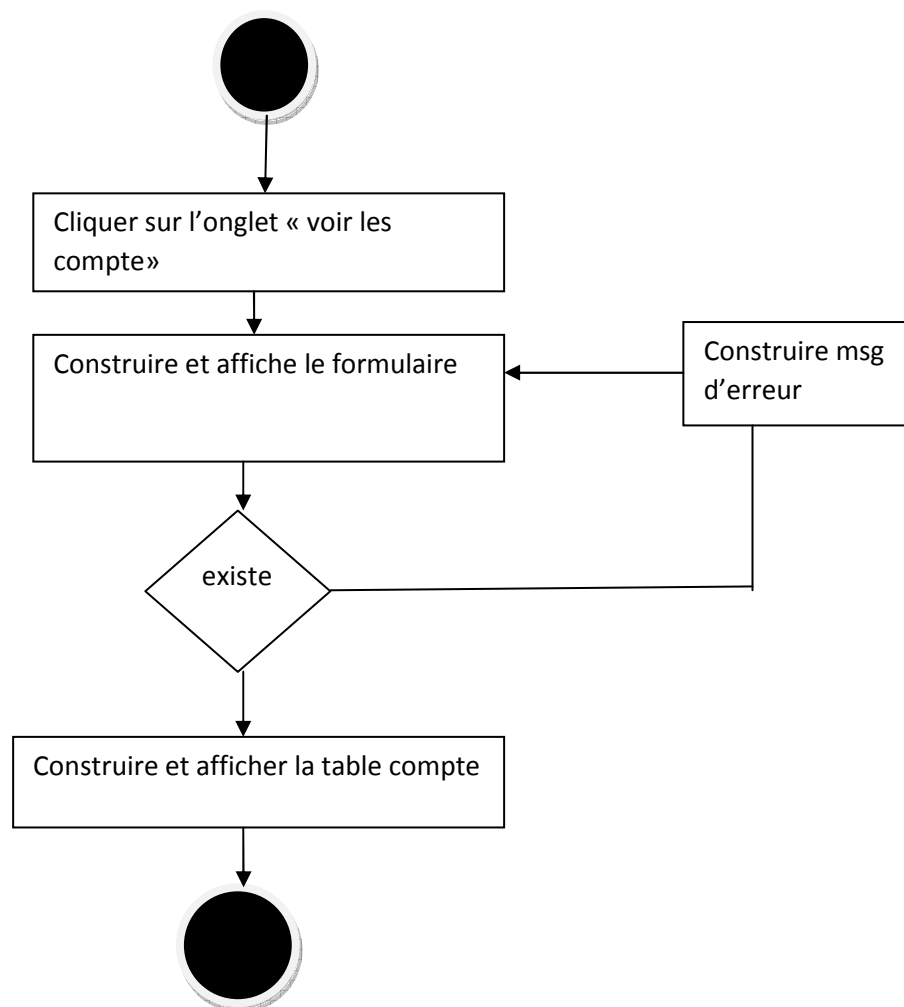


Figure III.13 Diagramme de cas d'utilisation « édition des facteurs »

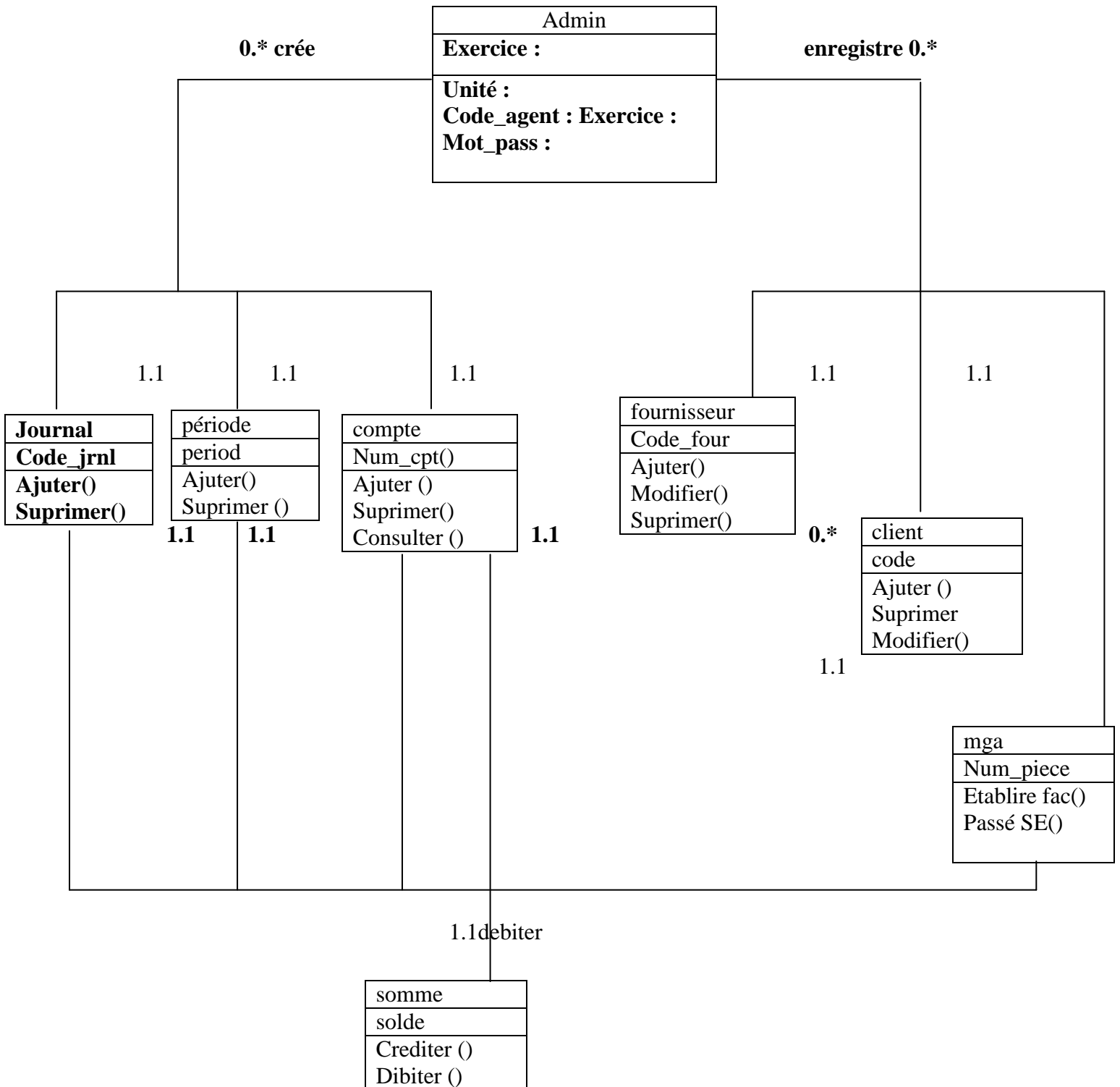
**Diagramme d'activité de cas d'utilisation « changer le mots de passe »****III.14 Diagramme d'activité de cas d'utilisation « changer le mot de passe»**

**Diagramme d'activité de cas d'utilisation « création de compte »****III.15 Diagramme d'activité de cas d'utilisation « création de compte »**

**Diagramme d'activité de cas d'utilisation voir les compte****III.16 Diagramme d'activité de cas d'utilisation « voir les comptes »**



### III.5.2.4 Diagramme de classe globale



III.17 diagramme de classe globale de notre application

**III.2.3.5 Implémentation de la base de données :**

on se basant sur le diagramme de classe globale que nous avons présenté ,nous aboutissons a l'élaboration du modèle physique de donnée composé des tables suivantes :

Table admin

Nom du champ	Description	Type	Clés
Unité	Nom de l'unité	Char(10)	
Mot_pass	Mot de passe pour acceder	Char(5)	
Code_agent	code agent	Varchar(3)	
Exercice	Mois	Entier(2)	primaire

Table centralisateur client

Nom du champ	Description	Type	Clés
Compte_cl_comptable	Nom Compte client comptable	char(20)	
Code_centralisateur_cl	Code centralisateur client	Integer( 3)	primaire

Table Centralisateur fournisseur

Nom du champ	Description	Type	Clés
Compt-four	Nom de fournisseur	Char(20)	
Code_centralisateur_four	Code centralisateur fournisseur	Integer(3)	primaire

Table compte

Nom du champ	Description	Type	Clés
Num_cpt	Numéro de compte	Integer(3)	primaire
Indicateur_cat	Indicateur catégorie	Integer(1)	
Indicateur_anal	Indicateur analytique	Integer(1)	
Libellé	Désignation	Varchar(50)	

Table journal

Nom du champ	Description	Type	Clés
Nom_jrnl	Nom de journal	Char(30)	
Code_jrnl	Code de journal	Integer(3)	primaire

Table mga

Nom du champ	Description	Type	Clés
Mantant	Mantant	Double(22)	
Date	date	Date(10)	
Libelé	Désignation	Varchar(30)	
Code_jrnl	Code de journal	Integer(3)	Etrangère
Period	mois	integer(2)	
Num_cpt	Numéro de compte	Integer(3)	Etrangère
Exercice	mois	Integer(2)	Etrangère
Num_piece	Numéro de pièce	Integer(11)	primaire
Code	code		

Table tiere

Nom du champ	Description	Type	Clés
Code	Code de client	Varchar(3)	primaire
Nom	Nom de client	Char(20)	
Prenom	Prénom de client	Char(20)	
Adress	Adresse de client	Varchar(50)	
Numtel	Numéro de téléphone	Integer(11)	
Pays	Pays de client	Char(20)	
Codecent_cl	Code centralisateur de client	Varchar(3)	Etrangère

Table période

Nom du champ	Description	Type	Clés
Period	Mois	Integer(2)	primaire
Etat	Etat de période (ouverte,fermer)	Integer(1)	
Date_deb	Date de début	Date(10)	
Date_fin	Date de la fin	Date(10)	
exercice	exercice	Integer(3)	Etrangère

### Conclusion

Dans ce chapitre nous nous sommes intéressés à la conception nous avons d'abord présenté les diagrammes UML utilisés ainsi que la modélisation graphique de quelques cas d'utilisation dans notre application .

.

# Chapitre IV

## Réalisation

## IV .Introduction

Dans l'étude conceptuelle nous avons expliqué notre système et nous allons le détailler avec l'étape de la réalisation. En outre, elle concerne aussi le déploiement de l'application dans un environnement réel et par la suite la présentation des langages SGBD utilisés, à la fin les différentes interfaces de l'utilisateur.

### IV.1 Description des langages et des outils de développements

#### IV.1.1 langage de programmation (java)

Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton de Sun Microsystems. Mais c'est également un environnement d'exécution. Java peut être séparé en deux parties : d'une part le programme écrit en langage Java, d'autre part, une machine virtuelle (JVM) qui va se charger de l'exécution du programme Java. C'est cette forme qui garantit la portabilité de Java. Il suffit qu'un système ait une machine virtuelle Java pour tout programme écrit en ce langage puisse fonctionner.

#### IV.1.2 IDE(NetBeans)

C'est un environnement de développement intégré (IDE) pour Java, placé par open source par Sun en juin 2000 sous licence CDDL (Common Development and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, XML, et le HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projet multi-langage, refactoring, éditeur graphique d'interface et des pages web). NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OS X et Open VMS.

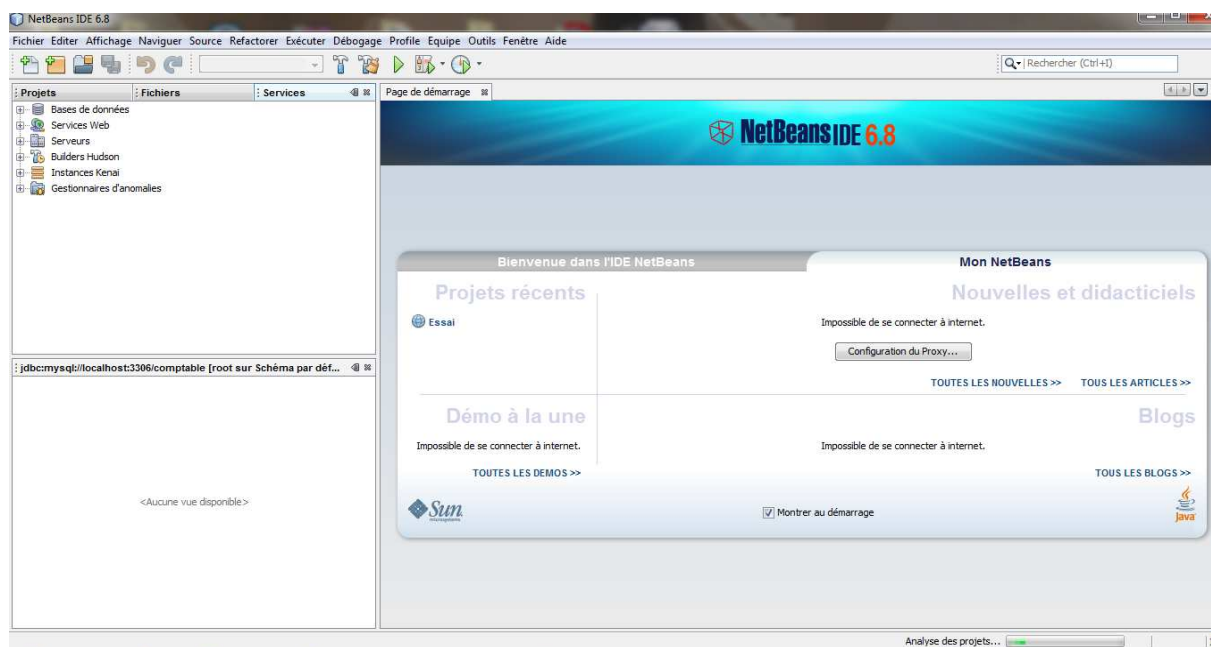


Figure VI.1 interface de L'IDE Netbeans.

### VI.1.3 Le serveur d'application (Glassfish)

le Glassfish est le nom de serveur d'application « open source » java EE sert de fondation au produit « oracle GlassFish Server » (anciennement Sun Java System Application de Sun Microsystems ). Sa partie Toplink persistance provient d'oracle . c'est la réponse aux développeurs java désireux d'accéder aux sources et de contribuer au développement des serveur d'application de nouvelle génération .

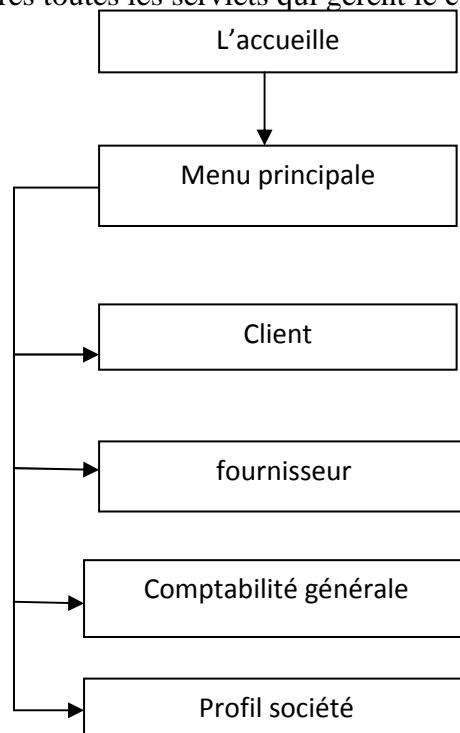
### IV.1.4 le SGBD (Mysql)

MySQL est un gestionnaire de base de donnée libre ,il est très utilisé dans les projets libre et dans le milieu industriels .MySQL est un SGBD relationnel développé dans un souci de performance élevées ,il est multi thread ,multi utilisateurs .on a opté pour le. car cette outil n'est pas indispensable mais très efficace permet de manipuler MYSQL d'une manière très efficace et très simple .cette interface graphique est en faite une couche de manipulation MySQL.

### IV.2présentation de l'application

1.préparer L'arborescence du site .

2. créer l'interface web statique avec le logiciel dreamwever(dans un dossier dans mes document qui va contenir un dossier d'image et les page statiques)
- 3.créer un projet web dans netbeans, copier tous le dossier d'image et les pages statiques qu'on crée dans ce dernier .
- 4.il reste a faires toutes les servlets qui gèrent le coté dynamique.

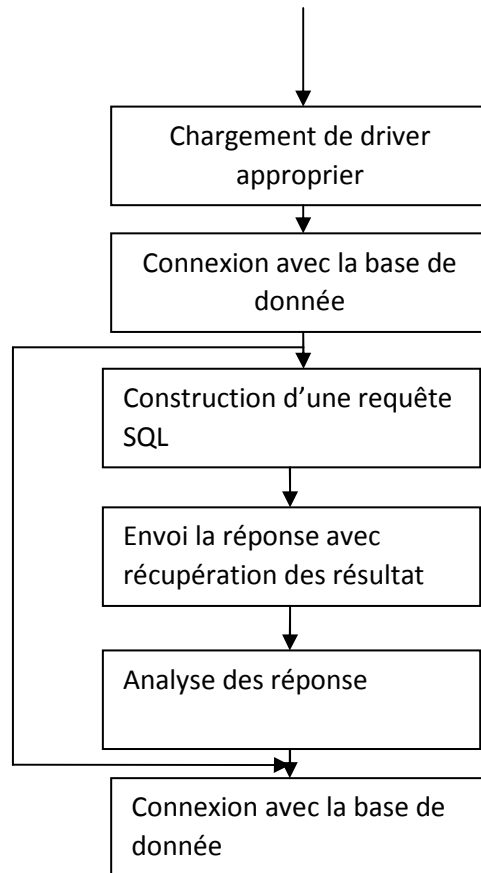


**Figure IV.2 l'arborescence de l'application**

### III. Implémentation de déférente partie de l'application

#### III.1création de la base de donnée

La réalisation d'un programme JDBC permettent l'accès a la base de données est montre par l'algorithme suivant voir (*algorithme1Structure d'un programme JDBC*).



Algorithme 1structure d'un programme GDBC

- Chargement d'un pilote JBDC qui a des méthodes d'accès à la base de données.
- Etablissement de la connexion pour rentrer pour la première fois en contact avec la base de données.
- Création et exécution d'une requête SQL en utilisant des objets spécifiques pour récupérer les résultats (les résultats sont récupérés en mode enregistrement par enregistrement).
- Traitement des résultats.
- Fermeture de la connexion pour éviter de maintenir des ressources non utilisées. Le tableau suivant résume les sept étapes d'accès aux bases de données MySQL :

Les étapes	MySQL
Chargement d'un pilote JDBC	<code>Class.forName("com.mysql.jdbc.driver")</code>
Définition d'une URL de connexion	<code>Jdbc : mysql://localhost :3306/comptable</code>
Etablissement de la connexion	Connexion <code>Connexion=DriverManager.getConnection(URL,username,password)</code>
Creation d'une instruction	<code>Statement commande = connexion.createStatement();</code>
Exécution de la requête	<code>ins.executeUpdate("insert into journal value (" + no1 + ", " + no2 + ")");</code> <code>ResultSet resultat=commande.executeQuery(query);</code>
Traitement des resultants	<code>// Afficher l'enregistrement complet de la table client</code> <code>While (resultat.next()){</code> <code>System.out.println(resultat.getString(1)+"-"</code> <code>" +resultat.getString(2)) :</code>
Fermeture de la connexion	<code>Connexion.close ()</code>

**Tableau VI. 1-** Accès à la base de données MySQL via le JDBC “ *com.mysql.jdbc.driver*”

Exemple :

Notre base de donnée est nommée comptable ;elle est constitué Des tables suivantes

Voici un (admn journal tiere ,periode ,compte facture...)

Voici l'exemple de création de table :

Create table tiere

Primary key Code varchar(3)not null,

Nom char(20) not null,

Prenom char(20) not null,

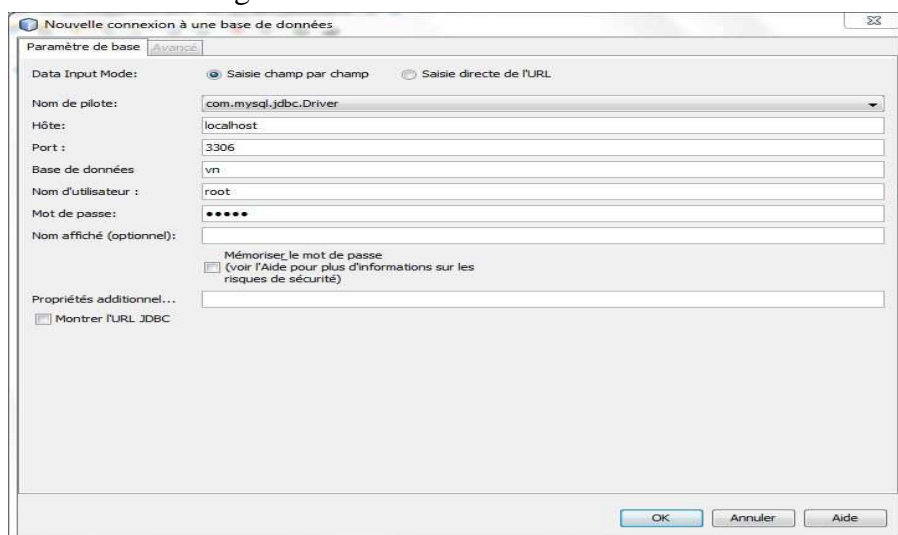
Adres varchar(50) not null,

Numtel integer(10) not null,

Pays char(20) not null,

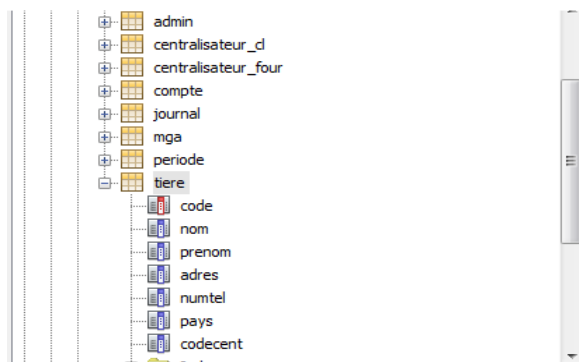
Codecentralisateur\_cl varchar(3)) not null;

Après avoir créé cette base de donnée on a établie une connexion NetBeans/MYSQL à l'aide de pilote JDBC .L'image suivante montre la connexion établie .





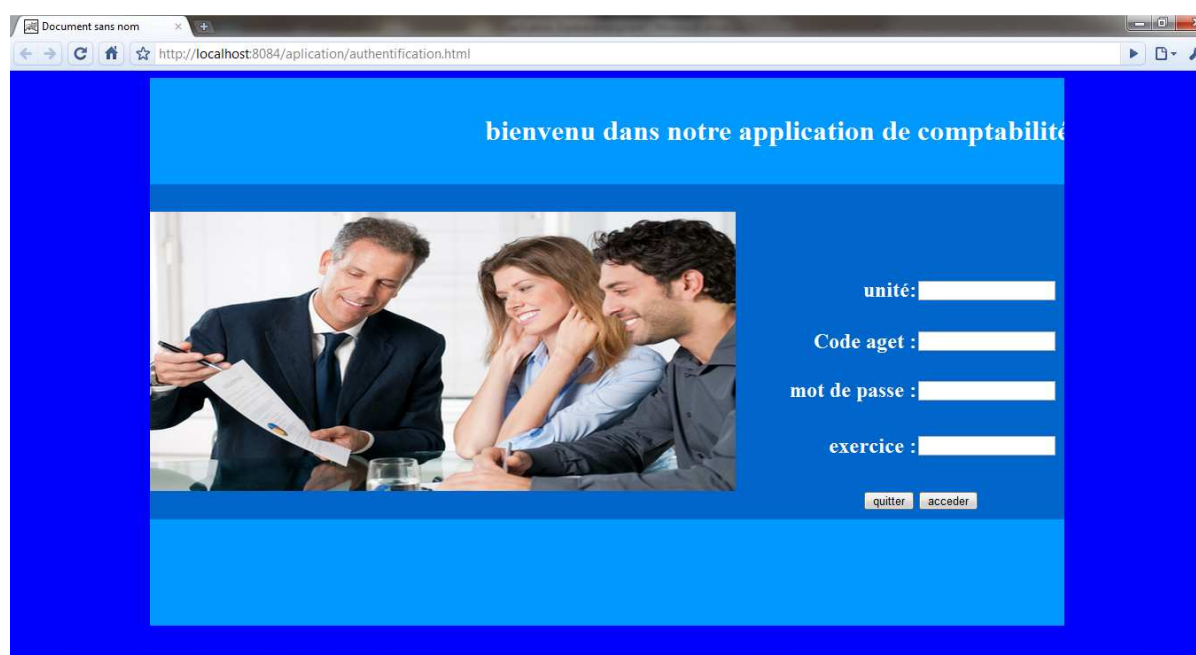
Après la connexion voici quelque table crie dans notre BDD



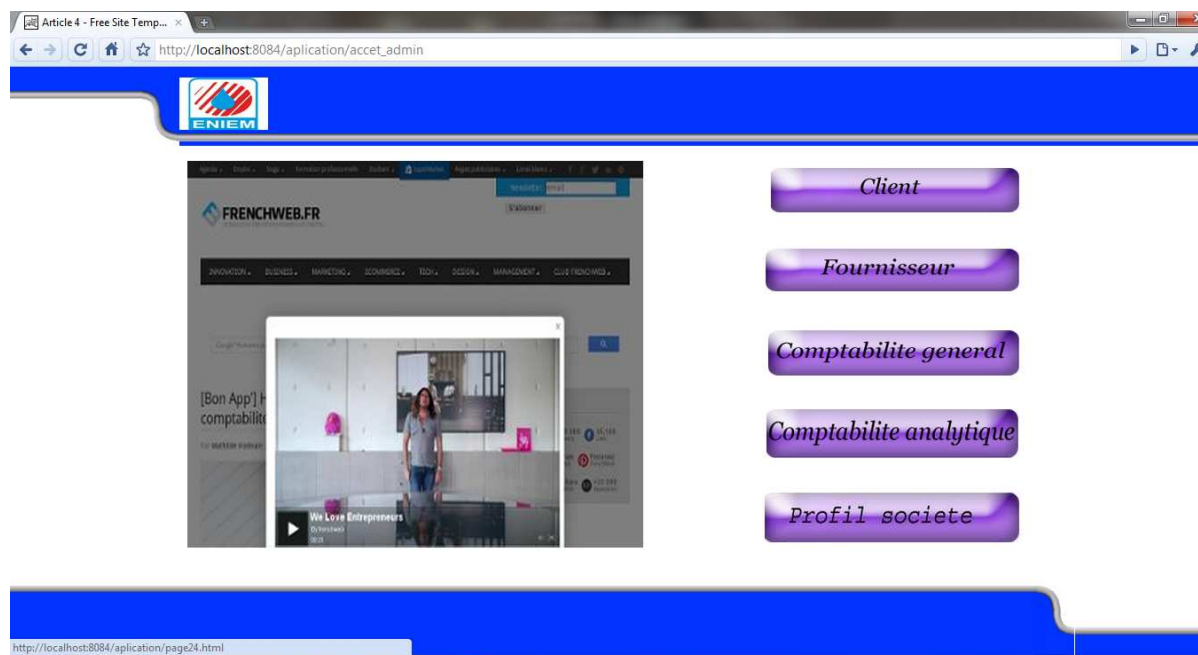
### IV.3Présentation de quelques interfaces de l'application web réalisée

#### IV.3.1 les pages statiques (authentification)

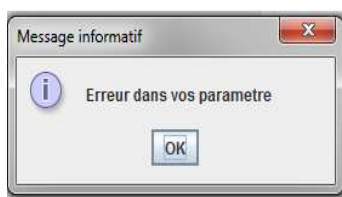
A partir de cette page, le client (ou le comptable) pourra voir sans menu principale après avoir saisi tous les champs



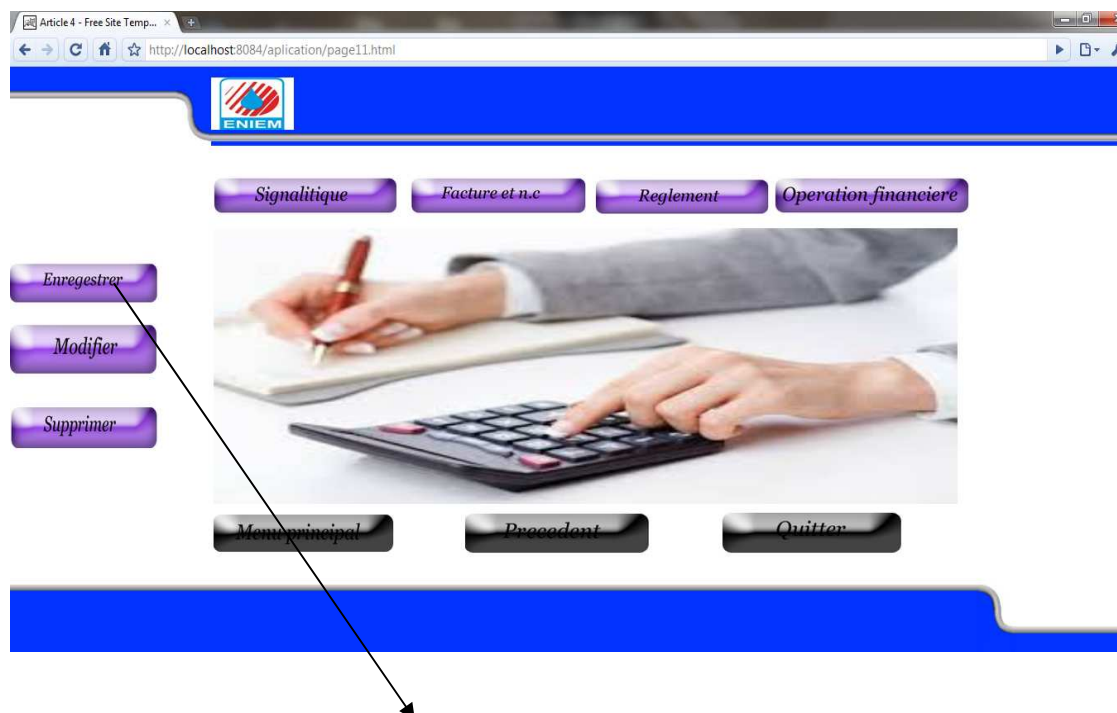
On aura la page suivante si les champs saisis sont corrects



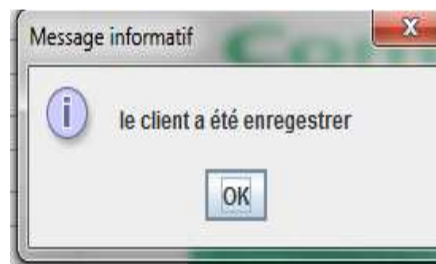
Si les champs saisi son incorrecte on auras :



Si on clique sur client dans le menu principale on auras :

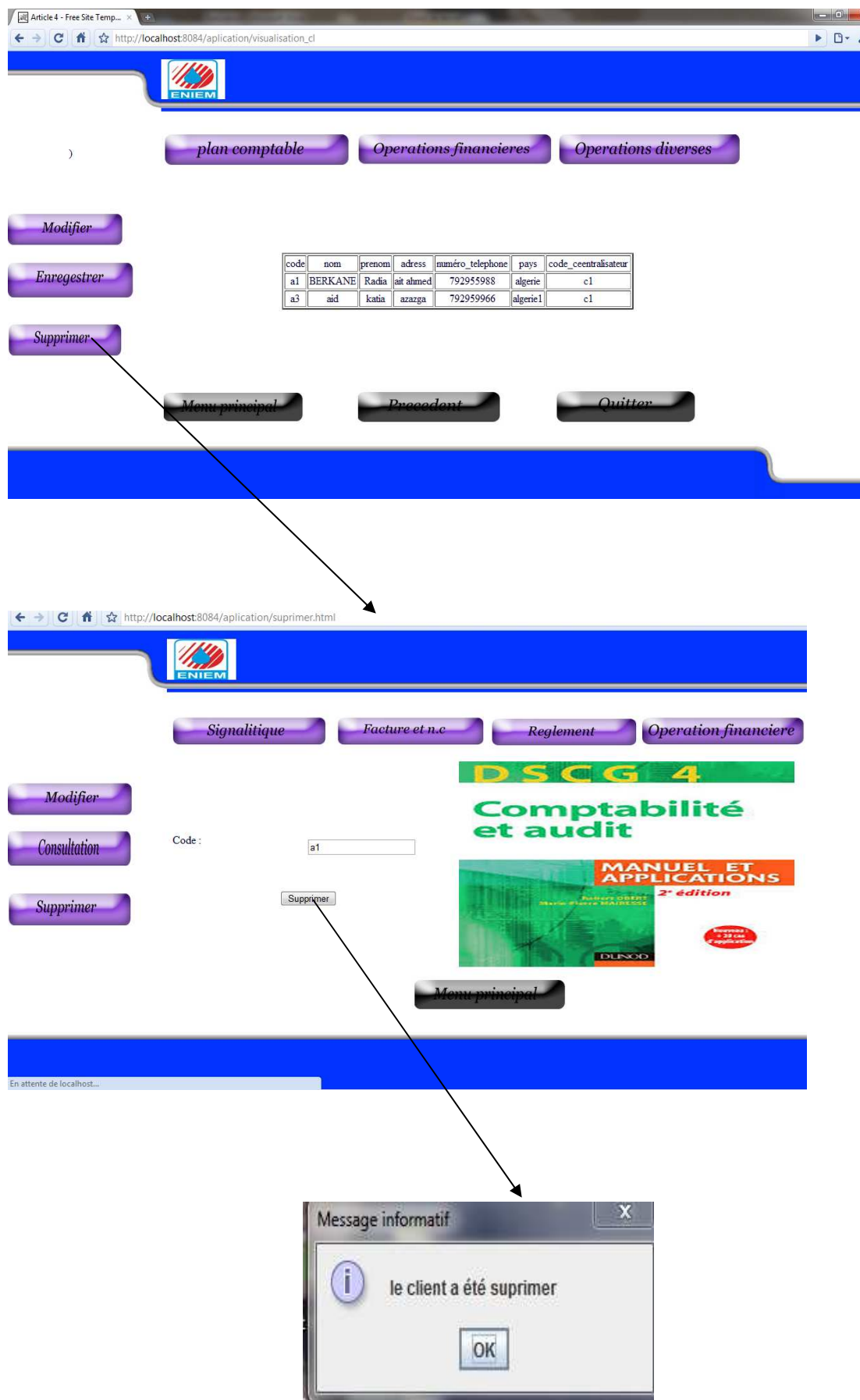


Après avoir rempli les champs et cliquer sur Envoyer en aura :



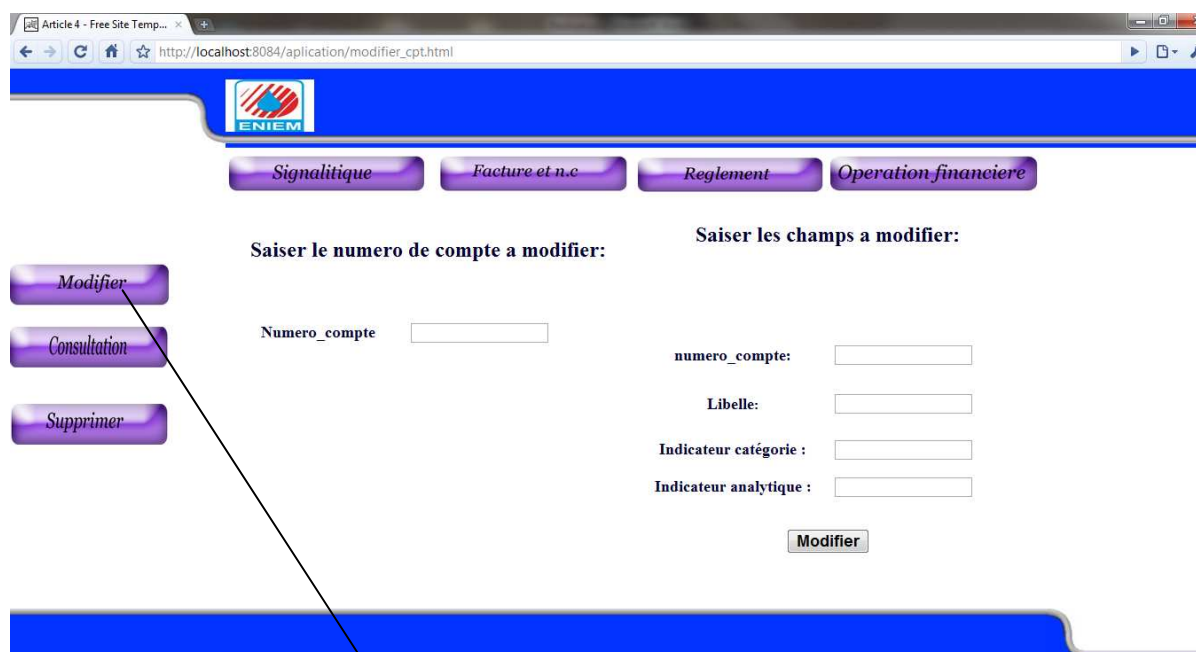
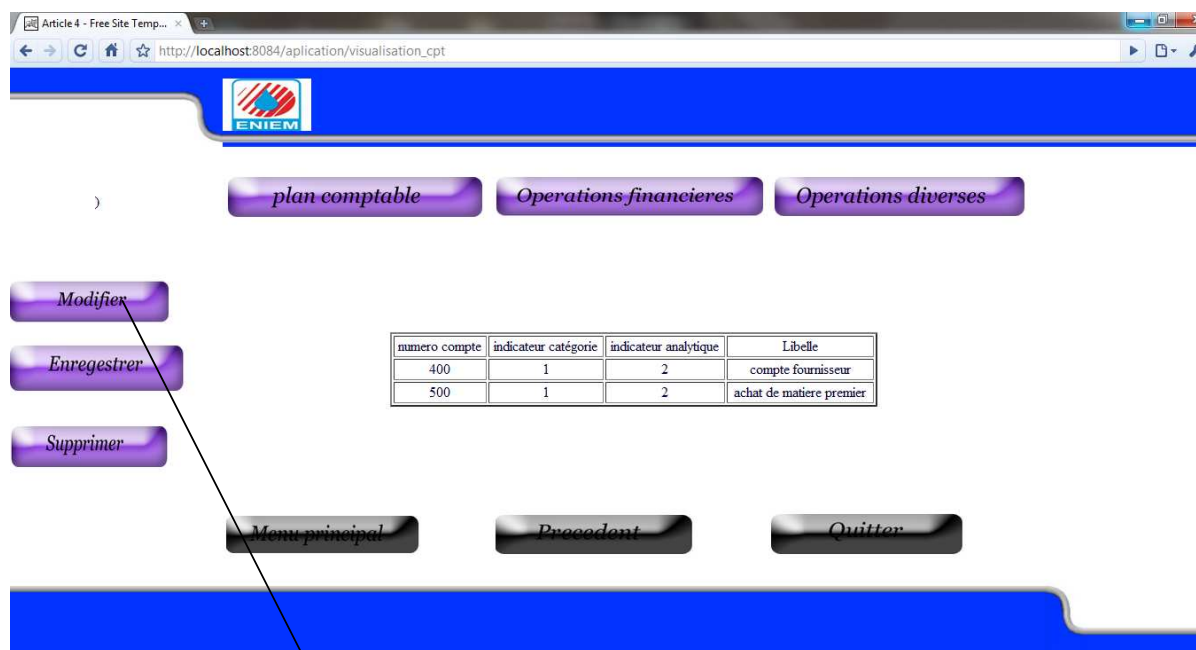
si on veut supprimer on clique sur supprimer et on aura la page suivante :

Avant de supprimer on doit d'abord consulter pour récupérer le code afin de voir qu'il est le client à supprimer



Et pour la modification on va prendre l'exemple de compte

Avant de modifier on doit consulter la table compte pour voir le compte a modifier



Après avoir rempli les champs on aura un message de confirmation



#### IV.4 Conclusion

Dans ce chapitre nous avons, au premier lieu, présenté les différents outils et langages que nous avons utilisé pour implémenter notre application. Par la suite, nous avons présenté quelques interfaces de notre application.

# Conclusion générale



# Conclusion générale

Le but fixé à notre travail était de réaliser et de développer une plateforme assez complète dans son architecture ,simple dans sa composition ,abordable dans son utilisation .Intégrant le plus de fonctionnalités, qui permettent d'assurer une gestion complète de la comptabilité de l'entreprise et offrant aux comptables une panoplie d'outils ,qui permettent à chacun d'accomplir son rôle par notre système .

Vue les avantages que offre la technologie JEE qui deviennent prometteuses ,comme un

Témoignent l'interopérabilité et la réalisation de ces fonctionnalités ,donc il est indispensable de pour une entreprise de faire appel à ces technologies pour laquelle elle assure la fiabilité et la sécurité de l'immense flux d'information .

Ce projet consiste en implémentation d'une application JEE pour rendre les tâches de comptable dynamique en assurant la fiabilité et la sécurité des informations .nous avons d'abord étudié profondément les composants de l'architecture JEE.

Ce projet nous a permis d'approfondir nos connaissances en :

gestion d'application entreprise

interface JDBC

modélisation orientée objet

\_ maîtrise de l'architecture JEE avec tous ses composants .

programmation Java pour le web (EJB ,interface graphique ...)

Enfin nous assurant que notre travail sera un intérêt pour les comptables de l'entreprise ENIEM et un guide efficace pour la nouvelle promotion.

# Bibliographie

## Bibliographie

[1] : [www Wikipidia.org](http://www.Wikipidia.org)

[2] Alin Lefebvre, « L'architecture client\_serveur », Edition ARMAND COLIN 2006

[3] Paolo Zanella & yves Ligier, « Architecture et technologie des ordinateurs » Dunod 4<sup>ème</sup> Edition 2005.

[4] George et Olivier Gardarin, « le client/serveur » Edition EYROLLES 2000.

[5] : <http://www.dyalo.com/coursinformatique/introductionj2EE.HTML>

[6] : documentation j2EE <http://java.sun.com/j2ee/>

[ 7] : Développement en java : par jean Michel DOUDOUX

[8] : [http://www .jmdoudoux.fr/java/dej/indexavecframes.html](http://www.jmdoudoux.fr/java/dej/indexavecframes.html)

# Annex A

UML Unified Modeling Language

# Annexe A

## UML Unified Modeling Language

### A.1 A quoi sert UML ?

UML sert de définir et de visualiser un modèle ,a l'aide de diagramme un diagramme UML est une représentation graphique ,qui s'intéressent a un aspect précis du modèle ; c'est une perspective de modèle .Chaque type de diagramme UML possède une structure (les type des élément de modélisation qui le composent sont préfinis.

Un type de diagramme UML véhicule une sémantique précise (un type de diagramme offre toujours la même vue d'un system .combine les déférent type de diagramme UML .offre une vue complète des aspect statique et dynamique d'un système.

Par extension et abus de langage, un diagramme UML est aussi un modèle (un diagramme modélise un aspect du modèle globale ) .UML est un langage graphique et reposent sur neuf type de diagrammes utilisent le même principe :les concept sont présentent par des ligne qui relient les symboles .dans ce que nous allons présenter les diagramme les plus utilisés dans la modélisation et conception d'un system d'information distribuer sur l'Internet.

### A.2 Utilité d'UML

UML est utilisé pour spécifier, visualiser, modifier et construire les documents nécessaires au bon développement d'un logiciel orienté objet. UML offre un standard de modélisation, pour représenter l'architecture logicielle. Les différents éléments représentables sont :

- Activité d'un objet/logiciel
- Acteurs
- Processus
- Schéma de base de données
- Composants logiciels
- Réutilisation de composants

Grâce aux outils de modélisation UML, il est également possible de générer automatiquement une partie de code, par exemple en langage Java, à partir des divers documents réalisés.

### A.3. Quelques notions de base

**Acteur :** un acteur représente un ensemble de rôles joués par une personne, une machine ou un autre système, il peut interagir avec un cas d'utilisation et participer à son scénario. Il est représenté par un petit bonhomme avec son rôle inscrit dessous.

• **Tache:** est un ensemble de fonctions qu'un acteur bien définit peut accéder.

• **Scénario:**C'est une succession particulière d'enchaînement, s'exécutant du début à la fin du cas d'utilisation.

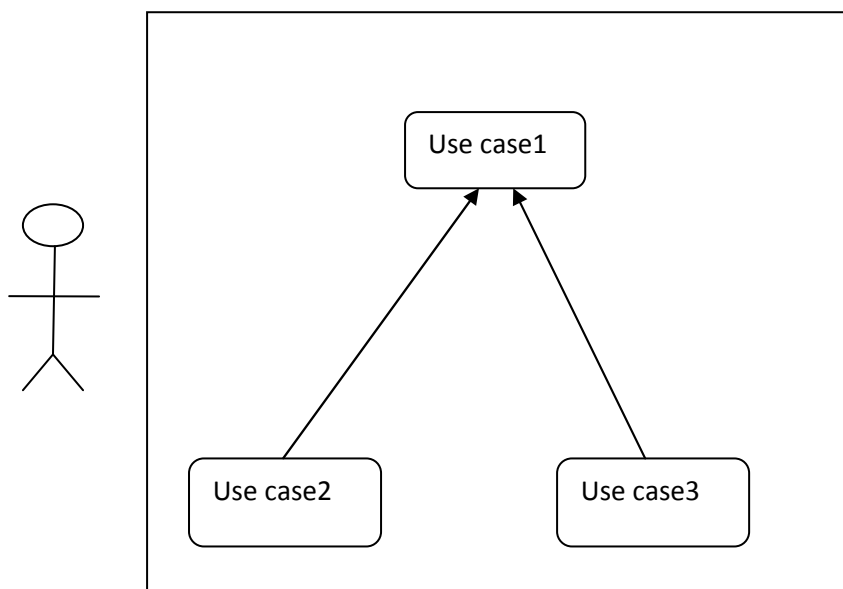
#### A.4 les diagrammes des cas d'utilisation

Les cas d'utilisation décrivent le comportement de système du point de vue de l'utilisateur. Ils permettent de définir les limites du système et les relations entre le système et son environnement. C'est l'image d'une fonctionnalité déclenchée en réponse à la sollicitation d'un acteur externe. Ils permettent de centrer la construction du système sur les besoins des utilisateurs. Ils existent deux principes fondamentaux dans les cas d'utilisation :

- Les acteurs qui utilisent le système :
  - ✓ Acteurs primaires qui sont la raison de l'existence de ce système
  - ✓ Acteurs secondaires qui ont des rôles d'administration et qui lui fournissent toutes les informations nécessaires à son bon fonctionnement.

- **Les use case qui représentent l'utilisation du système par les acteurs.**

Les diagrammes font intervenir les acteurs, les autres systèmes et les cas d'utilisation aux mêmes.



**Figure A : Représentation graphique d'un cas d'utilisation [pamu04]**

Il peut exister des relations entre les cas d'utilisation que l'on représente aux moyennes de stéréotypes :

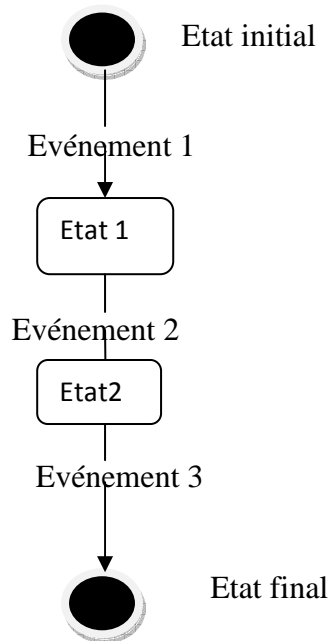
- Le cas d'utilisation peut utiliser un autre cas d'utilisation et on le note par une relation de type « uses »
- Les cas d'utilisation peuvent étendre un autre cas d'utilisation et on le note par une relation de type « Extend »

#### A.3 Les Diagrammes de classe

Ces diagrammes décrivent l'architecture du système. On y représente les classes et les relations entre les classes qu'elles soient d'héritage, d'agrégation, etc.

#### A.4 Les diagrammes d'état

C'est un automate d'état fini, composé de transitions, d'événements et d'activités. Il représente la vue dynamique du système par la détermination du comportement des objets d'une classe en termes d'états et de transitions d'états. Ce diagramme est représenté par un graphe constitué d'un nœud décrivant des états ainsi que les flèches représentant des transitions portant des paramètres et des noms d'événements.

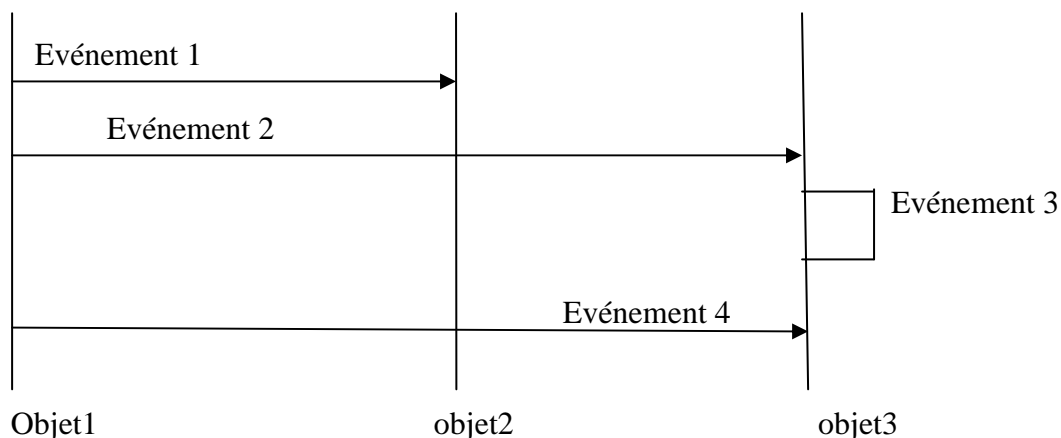


**Figure c :Représentation graphique d'un diagramme d'états**

Le diagramme d'état est propre a une classe donnée, ils doit être construit pour les objet de l'application ayant un comportement non trivial. Un état est une situation .une transition est une relation entre deux état signifiant qu'un passage de l'un a l'autre est possible .En autre termes ,c'est une relation orienté entre deux état, a laquelle est attaché un événement et qui indique qu'un objet dans le premier état passeras dans le second si certaines condition sont remplie

#### **A.5 Les diagramme de séquences**

Les diagramme de séquence illustrent les entités ainsi que les messages qu'elle s'échangent d'une procédure bien précise .donc il y aura autant de scénarios que de procédures .ces diagrammes seront classés par procédures et par acteurs .



**Figure D :Représentation graphique d'un diagramme de séquence**

Le but fixé à notre travail était de réaliser et de développer une plateforme assez complète dans son architecture, simple dans sa composition, abordable dans son utilisation. Intégrant le plus de fonctionnalités, qui permettent d'assurer une gestion complète de la comptabilité de l'entreprise et offrant aux comptables une panoplie d'outils, qui permettent à chacun d'accomplir son rôle par notre système.

Vue les avantages qui offre la technologie JEE qui deviennent prometteuses, comme un

Témoignent l'interopérabilité et la réalisation de ces fonctionnalités, donc il est indispensable de pour une entreprise de faire appel à ces technologies pour laquelle elle assure la fiabilité et la sécurité de l'immense flux d'information.

Ce projet consiste en implémentation d'une application JEE pour rendre les tâches de comptable dynamique en assurant la fiabilité et la sécurité des informations. Nous avons d'abord étudié profondément les composants de l'architecture JEE.

Ce projet nous a permis d'approfondir nos connaissances en :

gestion d'application entreprise

interface JDBC

modélisation orientée objet

\_ maîtrise de l'architecture JEE avec tous ses composants.

programmation Java pour le web (EJB, interface graphique ...)

Enfin nous assurant que notre travail sera un intérêt pour les comptables de l'entreprise ENIEM et un guide efficace pour la nouvelle promotion.