



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE
DEPARTEMENT D'INFORMATIQUE



Memoire de fin d'études

Pour l'obtention du diplôme de Master II en Informatique

Option : Ingénierie des systèmes d'information

*Conception et réalisation
d'une application N-tiers
J2EE*

Réalisé par :

M^r Lardjani Smail

M^{elle} Yousfi Kassia

Encadré par :

M^{me} Bourkache Ghenima

Promotion 2011/2012



Remerciements

Nous remercions le bon Dieu de nous avoir donné la force et permis de faire ce modeste travail.

Nous tenons à exprimer notre profonde gratitude et nos sincères remerciements à notre promotrice qui nous a fait l'honneur de diriger ce travail.

Nos plus vifs remerciements vont à tous le personnel du service des ressources humaines de l'unité froid de l'entreprise nationale des industries de l'électroménager ENJEM, qui nous ont généreusement aidés durant notre stage.

Que les membres de jury trouvent ici nous remerciements les plus vifs pour avoir accepté d'honorer par leur jugement notre travail.

Nos sincères sentiments vont à tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce projet, en particuliers nos chères familles.



Dédicaces

A mes chers parents Smina et Ramdane,

A ma Sœur Lydia et son mari Omar,

A mes frères Brahim et Aghiles,

A mon binome Smail et sa famille,

Kassia

*A mes chers parents, mes Sœurs, mes frères et a ma
binome kassia et sa famille.*

Smail

Partie I :

Introduction général

I. Généralités

Introduction.....	P2
1. Logiciel.....	P3
1.1. Définition d'un logiciel.....	P3
1.2. Les couches logicielles.....	P3
1.2.1. Les logiciels systèmes.....	P3
1.2.2. Les logiciels d'application.....	P4
1.3. Les types de logicielles.....	P4
1.4. Les PGI.....	P4
1.4.1. Définition.....	P4
1.4.2. Couverture fonctionnelle des PGI.....	P4
1.4.3. Structure des PGI.....	P5
2. L'architecture logicielle.....	P5
2.1. Définition d'une architecture logicielle.....	P5
2.2. Les architectures distribuées.....	P6
2.3. Le modèle Client/Serveur.....	P7
2.3.1. Définition du client/serveur.....	P7
2.3.2. Les principes généraux du modèle client/serveur.....	P7
2.3.3. La répartition des tâches.....	P8
2.3.4. Paradigme client/serveur.....	P8
2.3.5. Principe de fonctionnement client/serveur.....	P9
2.3.6. Modèle de dialogue client/serveur.....	P10
2.3.7. Modèle de connexion client/serveur.....	P11
2.3.8. Gestion des requêtes.....	P12
2.3.9. Mise en œuvre du client/serveur.....	P13
2.3.10. Les modèles client/serveur.....	P14
2.3.11. Architecture client/serveur.....	P17
2.3.12. Les middlewares.....	P22
2.3.12.1. Définition d'un middleware.....	P22
2.3.12.2. Les services du middleware.....	P22
2.3.12.3. Exemples de middleware.....	P23
2.3.12.4. Les middlewares objets.....	P23
2.3.13. Avantages et inconvénients de l'architecture client/serveur...	P24
2.3.13.1. Avantages de l'architecture client/serveur.....	P24
2.3.13.2. Inconvénients de l'architecture client/serveur.....	P24
3. J2EE.....	P25
3.1. Définition.....	P25
3.2. Fonctionnement interne.....	P25
3.3. Architecture.....	P25
3.3.1. Technologie web application.....	P25

Sommaire

3.3.2. Technologie entreprise application.....	P25
3.3.3. Technologie web services.....	P26
3.3.4. Technologie managed and security.....	P26
3.3.1.1. JSP et Servlet.....	P28
3.3.1.2. Les composants EJB.....	P28
3.3.5. Le serveur d'application.....	P29
3.4. J2EE et les architectures distribués.....	P30
3.4.1. Couche présentation.....	P30
3.4.2. Couche de service.....	P30
3.4.3. Couche d'objets métiers.....	P31
3.4.4. Couche d'accès aux données.....	P31
4. La sécurité informatique.....	P32
4.1. Définition.....	P32
4.2. Les principaux défauts de sécurité.....	P33
4.3. Protections.....	P33
4.4. J2EE et la sécurité	P37
Conclusion	P37

Partie II :

Introduction	P38
I. Etude préliminaire.....	P39
II. Présentation de l'organisme d'accueil.....	P40
Introduction.....	P40
1. Présentation de l'ENIEM.....	P40
1.1. Situation géographique de l'ENIEM.....	P40
1.2. Historique.....	P40
1.3. La gamme des produits fabriqués par l'ENIEM.....	P41
1.4. Missions et objectifs.....	P42
1.5. Description de la structure organisationnelle de l'ENIEM.....	P42
III. Etude de l'existant (champs d'étude).....	P46
1. Délimitation et définition du domaine d'étude.....	P47
2. Présentation des différents composants de l'unité froid.....	P47
3. La liste des documents manipulés par le champ d'étude.....	P49
4. Etude des postes.....	P50
5. Etude des documents	P55
IV. Présentation du projet.....	P80
1. Présentation de notre travail.....	P80
2. Grand choix technique.....	P80
3. Identification des acteurs.....	P80
4. Diagramme de contexte.....	P81
V. Capture des besoins fonctionnels.....	P82

Sommaire

Introduction.....	P82
1. Identification des cas d'utilisations.....	P82
1.1.Cas d'utilisations.....	P82
1.2.Diagramme des cas d'utilisation globale.....	P83
1.3. Description des cas d'utilisation.....	P85
1.4. Les scénarios par les diagrammes de séquence et d'activité.....	P94
1.1.1. Structuration des cas d'utilisations dans des packages.....	P100
1.1.2. Identification des classes candidates.....	P101
VI. Développement du modèle statique.....	P105
1. Analyse statique.....	P105
2. Dictionnaire de données.....	P105
3. Liste des classes.....	P107
4. Liste des classes d'association.....	P108
5. Diagramme de classe générale	P109
VII. Développement du modèle dynamique.....	P110
1. Description des scénarios.....	P110
VIII. Conception détaillé.....	P115
1. Conception.....	P115
2. Affinage des diagrammes des classes.....	P115
3. Les méthodes des classes.....	P115
4. Les méthodes des classes d'association.....	P117
5. Le diagramme des classes final.....	P120
6. Le passage vers le modèle relationnel.....	P120
6.1.Le modèle relationnel.....	P120
Conclusion	P122
PARTIE III :	
Réalisation :	
1. Environnement de programmation.....	P123
1.1. Langage de programmation JAVA.....	P123
1.2.L'IDE NetBeans.....	P124
2. Réalisation	P125
2.1.Présentation de la base de données.....	P125
2.2.Fonctionnement de l'application.....	P126
2.2.1. Page d'accueil Administrateur	P126
2.2.2. Connexion du gestionnaire emploi et formation.....	P126
2.2.3. Modifier le mot de passe du gestionnaire emploi	P127
2.2.4. Ajouter un contrat.....	P127
Conclusion générale.....	P128

Sommaire

Partie III :

1. Annexe A : présentation de 2TUP et d'UML.....P129
2. Annexe B : présentation du SGBD PostgreSQL.....P134
3. Annexe C : présentation des procédures de gestion des ressources humaines.....P139

.

INTRODUCTION GENERALE

Introduction générale

Avec la mondialisation, les entreprises s'agrandissent de jour en jour, en augmentant ainsi la taille de leur système d'information, ainsi l'informatique est devenue incontournable de nos jours. Donc il est plus que nécessaire d'avoir des systèmes extrêmement bien conçus et réalisés.

La direction des ressources humaines de l'ENIEM a des faiblesses au sein de son système de gestion telles que :

- La difficulté d'accès aux informations à cause des moyens d'archivage utilisés.
- La lenteur de recherche d'informations.
- Manque de contact entre les différents services.
- Utilisation des moyens traditionnels (papier,.....).
- Perte d'informations, perte du temps dans la réalisation des différentes tâches.

Dans l'objectif de remédier à ces soucis nous avons décidé de développer une application Client/serveur 3-tiers, en utilisant PostgreSQL comme SGBD, ainsi que la plateforme java J2EE, le serveur GlassFich pour le déploiement de notre application et le langage java pour le développement.

Pour mener à terme ce projet, nous avons divisé notre travail en trois parties : dans la première partie nous avons présenté des généralités sur le logiciel et son architecture, dans la seconde partie nous avons fait une étude de l'existant, une conception de notre application en suivant les étapes de la méthode 2TUP et en utilisant les diagrammes UML, et enfin en dernier y a la partie réalisation ou nous expliquons principalement le fonctionnement de l'application avec quelques interfaces ainsi que les moyens logiciels utilisés pour son développement.

Ce projet se termine par une conclusion générale, ainsi que des annexes où nous présentons la méthode 2TUP, UML, PostgreSQL et les différentes procédures des ressources humaines.

DEVELOPPEMENT DU MODELE DYNAMIQUE

PARTIE I

GENERALITES

Introduction

Les années 90 ont été marquées par l'arrivée de l'informatique Client/Serveur. Cette dernière est arrivée essentiellement pour combler les lacunes des systèmes centraux, en proposant un environnement ouvert, souple. Il permet d'améliorer la flexibilité des systèmes, et procurer certains avantages à l'entreprise telle que la souplesse et la capacité d'adaptation face au changement.

Fin des années 90, début 2000 arrive enfin la plateforme J2EE, elle occupe aujourd'hui une place prépondérante dans l'évolution des systèmes informatiques. Elle permet de prendre en charge et d'améliorer les systèmes complexes existant ainsi que de simplifier leur exploitation et leur maintenance.

1. Logiciel :

1.1. Définition d'un logiciel :

- ✓ Ensemble des programmes, procédés et règles et éventuellement de la documentation, relatifs au fonctionnement d'un ensemble de traitement de l'information (arrêté du 22 décembre 1981).
- ✓ Un logiciel peut être représenté par :
 - Un ensemble d'instructions ordonnées.
 - Un objectif précis.
 - Un langage propre au monde de l'informatique.

Un logiciel est conçu pour un environnement déterminé, les logiciels peuvent être perçus comme des couches successives qui s'intercalent entre l'utilisateur et le matériel.

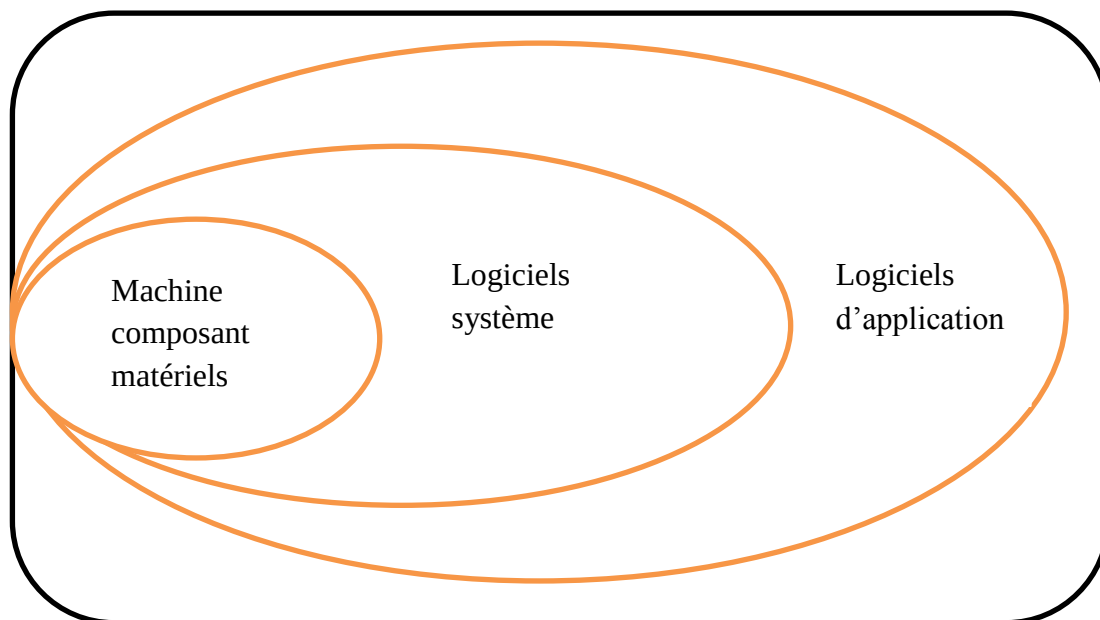


Figure 1 : les couches logicielles.

1.2. Les couches logicielles : On distingue deux couches de logiciels

1.2.1 Les logiciels systèmes :

- les logiciels systèmes se placent entre le matériel et les autres logiciels.
- Ils permettent aux logiciels d'application une indépendance vis-à-vis du matériel.
- Ils réalisent les fonctions de base du système (impression, visualisation, écriture mémoire...etc.).
- Traitent les demandes d'accès au matériel des logiciels d'application.

Sans eux chaque logiciel d'application serait écrit spécifiquement pour chaque configuration matérielle, les logiciels systèmes sont le système d'exploitation et le bios.

1.2.2 Les logiciels d'application :

On a différents types de logiciels d'application :

- Les utilitaires.
- Les outils de développement d'application.
- Les systèmes de gestion de base de données.
- Les logiciels applicatifs.

1.3. Les types de logicielles :

On peut classer les logiciels en quatre types qui sont :

- ✓ **Les logiciels commerciaux** : vendus dans le commerce.
- ✓ **Les sharewares** : mis à disposition gratuitement pour essais ou tests et qui doivent être payés pour une utilisation régulière.
- ✓ **Les freewares** : mis à disposition gratuitement et qui peuvent être utilisés librement sans aucune contribution.
- ✓ **Les logiciels libres** : qui sont fournis normalement avec le code source et qui peuvent être modifiés librement, (pour certains, la licence GPL définit les conditions).

1.4. Les PGI :

1.4.1. Définition :

Un progiciel de gestion intégrée (PGI) plus connu sous le sigle ERP (Entreprise Resource Planning) est un logiciel paramétrable construit autour d'une base de données unique, susceptible de couvrir la majorité des besoins en traitement d'information des différentes fonctions de l'entreprise grâce à un ensemble de modules compatibles. Ces PGI sont de plus en plus adoptés par les organisations comme épine dorsale de leur système d'information de gestion.

1.4.2. Couverture fonctionnelle des PGI :(document Laure Ferbourg) [1]

Les PGI sont conçus pour fournir aux entreprises et administrations de toutes les tailles et de tous les secteurs un outil intégrant toutes les activités de l'entreprise dans un système globale

qui couvre l'ensemble des opérations de planification, de gestion et de suivi. Le cœur des fonctionnalités est constitué d'applications standards : gestion comptable, contrôle de gestion, gestion de la production, achats et stocks, management de la qualité, maintenance, administration des ventes, gestion du personnel et gestion de projets. Le principe de base des données unique permet de représenter des données internes et externes à l'entreprise. Facilitant la prise de décisions stratégique de tous les domaines de l'organisation. Ces modules sont complétés par des modules transversaux et par des modules métiers : aérospatiale et défense, assurance, banque, chimie, pharmaceutique, télécoms, etc.

1.4.3. Structure des PGI [1]:

Bien que conçus de manière intégrée, les différents modules peuvent être utilisés séparément et complétés progressivement au gré des besoins de l'entreprise. L'entreprise s'appuie alors sur une structure informatique globale, qui ne tient plus compte du cloisonnement entre services. Les PGI ne connaissent pas les frontières liées aux organisations ou à la géographie : services centraux, sites de productions, succursales et filiales sont fédérés par un réseau global de traitement intégré des processus de gestion.

2. L'architecture logicielle :

L'architecture logicielle se consacre à structurer et à concevoir une application à partir de ses spécifications fonctionnelles, elle structure et décompose de façon logique chaque application en couches et introduit les notions et concepts de découpage en couches, modules, composants, design patterns et Framework

2.1. Définition d'une architecture logicielle :

Selon P. Kruchten, l'architecture logicielle se représente selon « 4+1 » vues :(livre blanc)

- **Vue Logique** : elle se concentre sur la modélisation des principaux éléments d'architecture et mécanismes logiciels, la définition des éléments du domaine (objets métier, diagrammes d'états de ces objets...). Elle comprend en particulier les modèles d'analyse et de conception du système, c'est la vue des équipes de conception et développement.
- **Vue Composants (ou vue Implémentation)** : elle permet d'identifier les modules (composants logiciels) qui implémentent les éléments définis dans la vue logique, de

les regrouper en composants logiciels, d'identifier les dépendances d'intégration entre ces composants ; c'est la vue des équipes d'intégration.

- **Vue Processus** : en environnement multi-tâches, cette vue de l'architecture permet de définir les processus, la coordination et la synchronisation des processus, les threads d'exécution... Cette vue est optionnelle et n'est utile que dans le cas d'architectures complexes multi-tâches.
- **Vue Déploiement** : elle précise l'architecture de production (ressources matérielles, implantation des composants, pilotage...) Cette vue permettra par exemple de s'assurer que l'application répond aux contraintes de déploiement, aux exigences de qualité de service (montée en charge, temps de réponse, haute-disponibilité...) et s'intègre aux infrastructures de supervision, etc.
- **Vue Use-Cases (vue « +1 »)** : elle se concentre sur un sous-ensemble des cas d'utilisation qui ont une influence significative sur l'architecture du système. Ces use-cases structurants permettent d'identifier les fonctionnalités et contraintes importantes, les risques majeurs de l'architecture, ce sont eux qui guident l'élaboration des quatre autres vues de l'architecture, de la conception à la mise en production de l'application.

2.2. Les architectures distribuées :

Une application distribuée est une application (typiquement de grandes dimensions) pour laquelle les différents composants ont été distribués sur différents niveaux de l'environnement d'élaboration du réseau par rapport aux fonctions qu'elles offrent dans l'architecture globale. Ces applications sont complexes et sont constituées de nombreux programmes software qui sont exécutés sur différents supports hardware. Les « niveaux » (tiers) d'une application distribuée sont les niveaux entre sa propre application et les données qu'elle gère. Ces niveaux respectent les règles suivantes :

- Chaque niveau ne communique qu'avec le niveau sous-jacent et dépend seulement de ce dernier
- Chaque niveau n'a pas connaissance d'aucun niveau en dehors du niveau sous-jacent

2.3. Le modèle client/serveur :

2.3.1. Définition du client/serveur :

L'architecture client/serveur est un modèle de fonctionnement logiciel qui peut se réaliser sur tout type de d'architecture matérielle (petites ou grosses machines) à partir du moment où ces architectures peuvent être interconnectées.

On parle de fonctionnement logiciel dans la mesure où cette architecture est basée sur l'utilisation de 2 types de logiciels ; à savoir un logiciel serveur et un logiciel client s'exécutent normalement sur 2 machines différentes. L'élément important dans cette architecture est l'utilisation de mécanismes de communication entre les deux applications.

Le dialogue entre les applications peut se résumer par :

- ✓ Le client demande un service au serveur.
- ✓ Le serveur réalise ce service et renvoie le résultat au client.

L'un des principes fondamental est que le serveur réalise un traitement pour le client.

2.3.2. Les principes généraux du modèle client/serveur :

- ✓ **Service** : le serveur est fournisseur de services. Le client est consommateur de services.
- ✓ **Protocole** : c'est toujours le client qui déclenche la demande de service. Le serveur attend passivement les requêtes du client.
- ✓ **Partage des ressources** : un serveur traite plusieurs clients en même temps et contrôle leurs accès aux ressources.
- ✓ **Localisation** : le logiciel client/serveur masque aux clients la localisation du serveur.
- ✓ **Hétérogénéité** : le logiciel client/serveur est indépendant des plates-formes matérielles et logicielles.
- ✓ **Redimensionnement** : il est possible d'ajouter et de retirer des stations clientes. Il est possible de faire évoluer les serveurs.
- ✓ **Intégrité** : les données du serveur sont gérées sur les serveurs de façon centralisée, les clients restent individuels et indépendants.
- ✓ **Souplesse et adaptabilité** : on peut modifier le module serveur sans toucher au module client. La réciproque est vraie, si une station est remplacée par un modèle plus récent, on modifie le module client sans modifier le module serveur.

2.3.3. La répartition des tâches :

Dans l'architecture client/serveur, une architecture est constituée de ces trois parties suivantes :

- ✓ L'interface utilisateur.
- ✓ La logique des traitements.
- ✓ La gestion des données.

Le client n'exécute que l'interface utilisateur ainsi que la logique des traitements (formuler la requête), laissant au serveur de BDD la gestion complète des manipulations de données.

La liaison entre le client et le serveur correspond à tout un ensemble complexe de logiciels appelés middleware qui se charge de toutes les communications entre les processus.

2.3.4. Paradigme client/serveur : [11]

La majorité des applications réseaux modernes se fondent sur le paradigme client/serveur. Le terme client/serveur est souvent utilisé sous deux aspects :

- ✓ **Physique :**

Ou on donne le qualificatif de client ou de serveur à des systèmes physiques : on assimile les postes de travail au « client » et les systèmes partagés au « serveur ». Il existe en effet deux modes d'implémentation :

- **Mode symétrique (horizontal) :** un poste de travail physique ou un serveur physique peuvent se comporter indifféremment comme client ou serveur ce cas de figure est illustré par les fonctions « voisinage réseau » fournies par Windows ; ces fonctions permettent à tout poste de travail autorisé d'émettre des demandes de services d'accès aux fichiers d'autres postes du réseau, cette fonction illustre le fait qu'un poste de travail physique peut se comporter comme client et comme serveur de fichier.
- **Mode d'implantation hiérarchique (verticale) :** un serveur physique peut par exemple jouer simultanément le rôle de serveur de fichier et de client pour l'accès aux fichiers d'un autre serveur physique, on utilise par fois le terme de modèle client/serveur pour désigner cette implantation.

✓ **Logique :**

Où le modèle client/serveur concerne le domaine du logiciel, il implique une répartition des traitements entre deux systèmes, sans qu'il y ait obligatoirement identité ce qu'il est convenu d'appeler un client ou un serveur physique et les fonctions client/serveur du traitement réparti. Dans l'implantation logique on peut avoir une application client/serveur qui tourne sur une seule machine physique qui jouera le rôle de client et de serveur en même temps.

2.3.5. Principe de fonctionnement client/serveur :

Le client/serveur est un mode de communication entre deux processus. Le premier appelé client, demande des services au deuxième processus qui est le serveur, ce dernier exécute les requêtes du client et envoie les résultats en retour.

- ✓ **Client :** processus demandant l'exécution d'une opération à un autre processus serveur par l'envoi d'un message contenant le descriptif de l'opération à exécuter et attendant la réponse à cette opération par un message en retour.
- ✓ **Serveur :** processus accomplissant une opération sur demande d'un client et transmettant la réponse à ce dernier.
- ✓ **Réponse :** message transmis d'un serveur à un client suite à l'exécution d'une opération contenant les paramètres de l'opération.
- ✓ **Requête :** message transmis d'un client à un serveur décrivant l'opération à exécuter.

Les appels mis en en jeu sont au nombre de quatre :

- ✓ **Send Request () :** permet au client d'émettre le message décrivant la requête à une adresse correspondante à la porte d'écoute du serveur.
- ✓ **Receive Request () :** permet au serveur de recevoir la requête sur sa porte d'écoute.
- ✓ **Send Reply () :** permet au serveur d'envoyer la réponse sur la porte d'écoute du client.
- ✓ **Receive Reply () :** permet au client de recevoir la réponse en provenance du serveur.

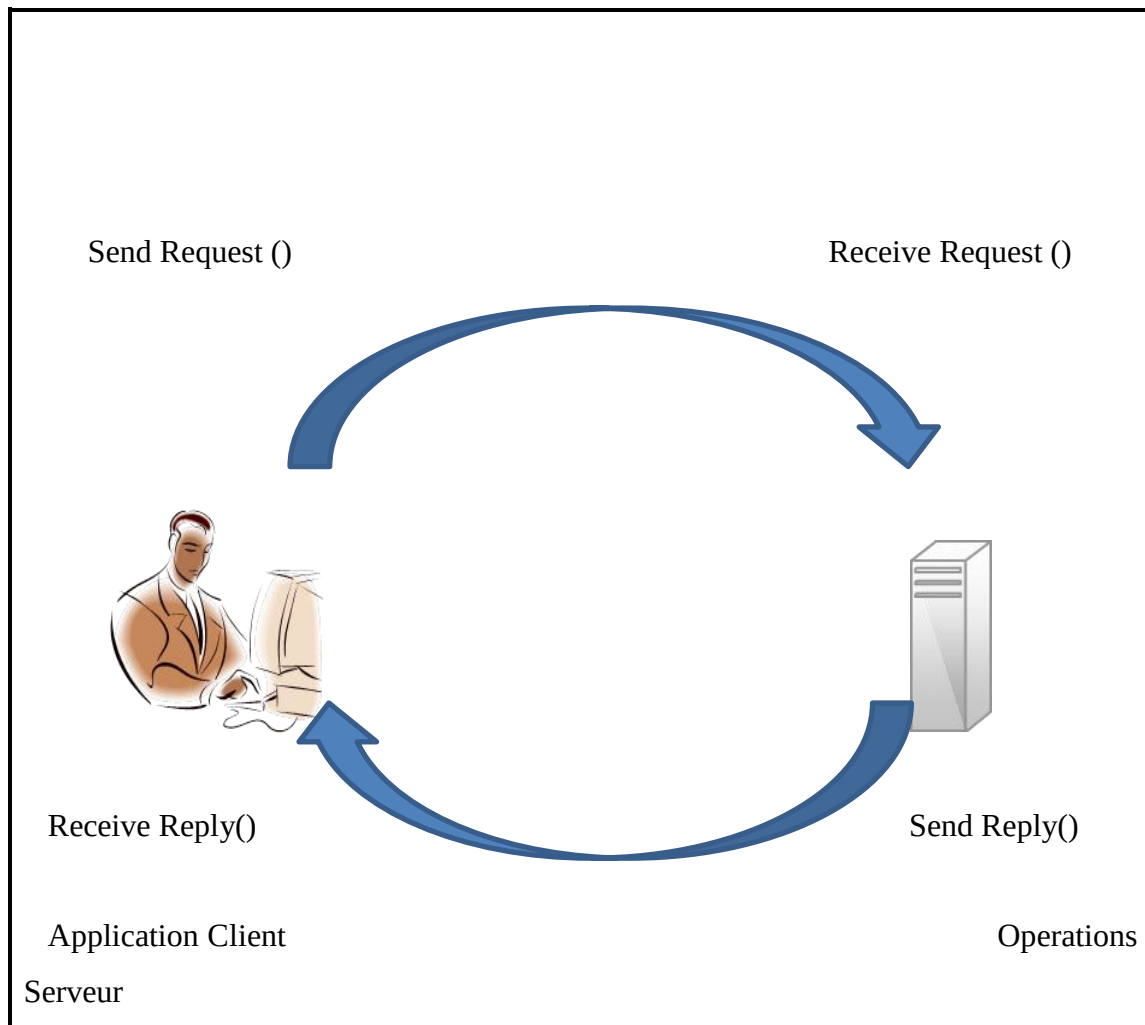


Figure 2 : principe de fonctionnement Client/Serveur.

2.3.6. Modèle de dialogue client/serveur :

La communication entre le client et le serveur peut être en mode synchrone ou asynchrone.

✓ **Mode synchrone :**

Dans ce type de dialogue, le client ne peut pas émettre d'autre requête pendant que le serveur exécute une opération pour lui. Autrement dit, les commandes d'émission et de réception sont bloquantes.

✓ **Mode asynchrone :**

Dans ce type de dialogue, au moins une des commandes d'émission ou de réception est non bloquante, il est donc géré par une file d'attente. Ce dialogue permet

au client d'effectuer une autre requête pendant que le serveur exécute une opération pour lui.

2.3.7. Mode de connexion Client/Serveur :

- ✓ **Mode non connecté** : Il est caractérisé par :
 - Pas d'établissement préalable d'une connexion.
 - Adapté aux applications pour lesquelles les réponses aux requêtes des clients sont courtes.
 - Protocole de transport utilisé : UDP.
 - Mode d'échange par messages : le récepteur reçoit les données suivant le même découpage que celui effectué par l'émetteur.

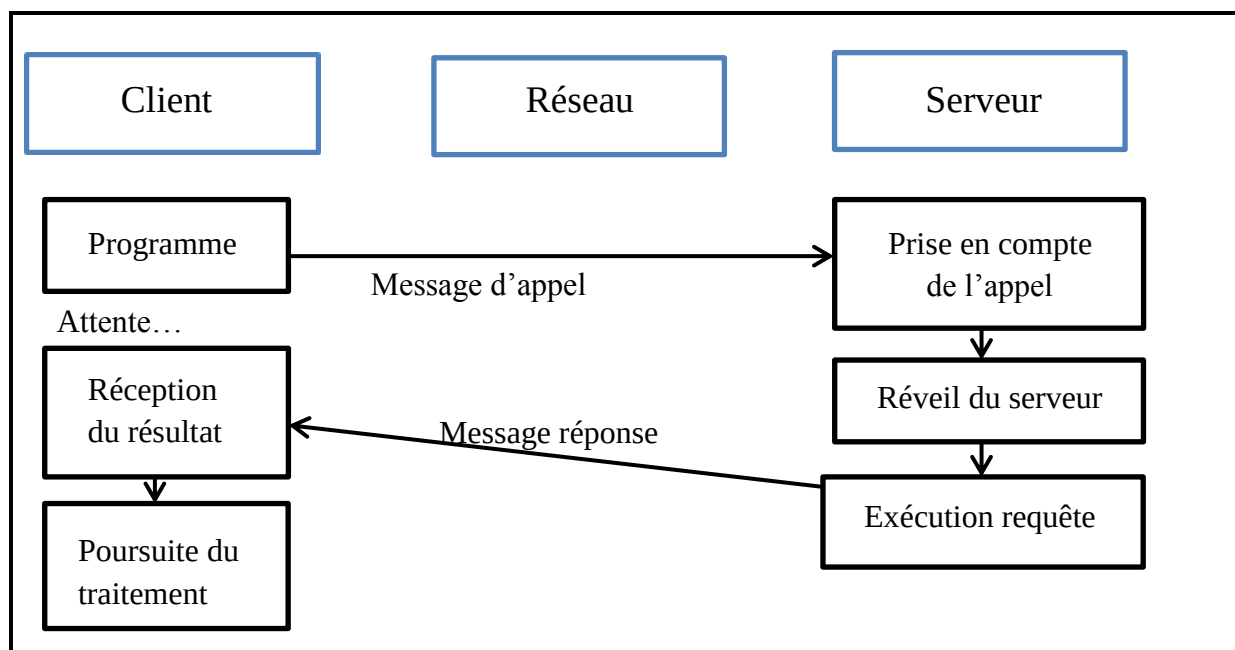


Figure 3 : mode de connexion non connecté.

- ✓ **Mode connecté** : Il est caractérisé par :
 - Etablissement préalable d'une connexion : le client demande au serveur s'il accepte la connexion.
 - Fiabilité assurée par le protocole de transport utilisé : TCP.
 - Mode d'échange par flots d'octets : le récepteur n'a pas connaissance du découpage des données effectué par l'émetteur.
 - Possibilité d'émettre et de recevoir des caractères urgents.

- Après initialisation, le serveur est “passif”, il est activé lors de l’arrivée d’une demande de connexion d’un client.
- Un serveur peut répondre aux demandes de services de plusieurs clients: les requêtes arrivées et non traitées sont stockées dans une file d’attente.

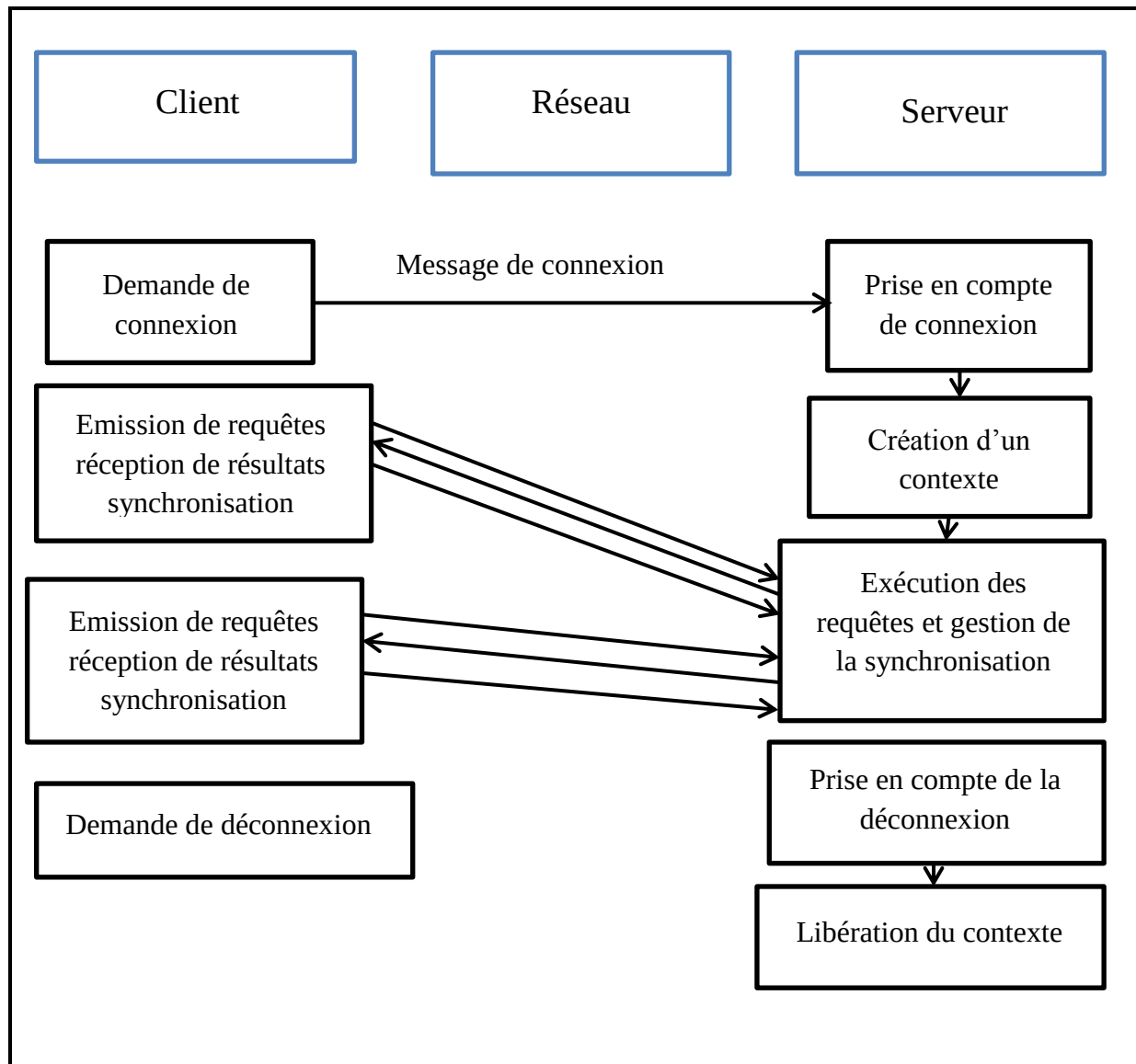


Figure 4: mode de connexion connecté.

2.3.8. Gestion des requêtes :

Les termes itératif et concurrent sont utilisés pour définir la gestion des requêtes.

- ✓ **Accès itératif :** Un seul processus gère toutes les requêtes de tous les clients :
 - Le processus traite séquentiellement les requêtes,
 - Adapté aux requêtes qui peuvent s'exécuter rapidement,

- Souvent utilisé en mode non connecté (recherche de la performance).
- ✓ **Accès concurrent** : Un processus gère chaque requête. Le serveur accepte les requêtes puis les délègue à un processus fils (traitement de plusieurs clients) :
 - Adapté aux requêtes qui demandent un certain traitement (le coût du traitement est suffisamment important pour que la création du processus fils ne soit pas pénalisante).
 - Souvent utilisé en mode connecté.

2.3.9. Mise en œuvre du client/serveur :

Tandis que le client/serveur est bâti sur l'idée de partage d'une application de chaque côté d'une frontière entre client et serveur pour créer diverses solutions sur des réseaux locaux, sa mise en œuvre s'appuie sur plusieurs mécanismes :

✓ **Les sockets :**

L'interface de programmation nommée sockets est devenue une des interfaces les plus utilisées pour la programmation réseau. Les sockets sont ce que l'on appelle une API « Application Programming Interface » c'est-à-dire une interface entre les programmes d'application et la couche de transport, elles fournissent un accès aux protocoles de transport et forment un point de communication par lequel un processus peut émettre ou recevoir des informations.

✓ **Les TLIs :**

TLI (Transport Layer Interface) fournit une interface à la couche transport du modèle OSI, s'appuyant sur le mécanisme général d'entrée-sortie par streams (image du flot de données établi entre l'utilisateur et un périphérique d'E/S via son « driver »).

✓ **Les RPCs :**

L'appel de procédures à distance ou Remote Procedure Call (RPC) permet d'appeler des procédures situées dans des processus s'exécutant sur d'autres machines. Quand un appel de procédure à distance est effectué, les paramètres de l'appel sont passés à la procédure distante, l'appelant (le client) attend la réponse qui est renvoyée par la procédure appelée. L'utilisation des RPC permet au programmeur d'une application distribuée de ne pas se préoccuper des détails de l'interface réseau. L'indépendance des RPC vis-à-vis du

transport, sépare complètement l'application des éléments logiques et physiques de la communication des données.

✓ **Les MOMs :**

Protocole de communication permettant la gestion d'envoi des messages en environnement réseau. Permet une communication d'égal à égal et notamment la diffusion d'événement vers les acteurs connectés. Les MOMs offrent les avantages suivants :

- Intégration de multiples protocoles et des multiples plateformes.
- Messages définis par les utilisateurs.
- Equilibrage de charges.
- Tolérance de pannes.
- Support pour plateformes hétérogènes.

2.3.10. Les modèles client/serveur : [12]

Certaines applications client/serveur se différencient selon les services qu'elles offrent (i.e. selon les types de serveurs) et d'autres préfèrent les distinguer par la façon dont les fonctions distribuées se partagent entre client et le serveur et utilisent plutôt les termes orientation, orientation serveur.

2.3.10.1. Les types de serveur :

Il existe plusieurs variantes des technologies client/serveur, comme le décrivent les paragraphes suivants :

- **Serveur de fichiers :**

Dans le cas de serveur de fichiers, le client requiert des enregistrements de fichiers en émettant des requêtes au serveur de fichiers. Les serveurs de fichiers sont utiles pour partager les fichiers sur un réseau et ils sont indispensables pour créer des banques de documents, d'images...etc. mais l'obtention de l'information nécessite de nombreux échanges de messages sur le réseau.

- **Serveur de base de données :**

Dans le cas de serveur de base de données, le client émet des requêtes SQL sous forme de message en direction du serveur, le résultat est envoyé au client. Le serveur utilise sa capacité de traitement pour rechercher les données demandées au lieu de transmettre tous les articles au client et de laisser en faire la sélection, donc la puissance est répartie et utilisée de façon beaucoup plus efficace.

- **Serveur de transaction :**

Dans ce modèle, le client invoque des procédures distantes résidant sur le serveur qui comporte un moteur de base de données SQL. Ces procédures distantes exécutent un ensemble d'instructions SQL. L'échange sur le réseau consiste en un seul message requête/réponse (contrairement à l'application serveur de base de données pour laquelle le message requête/réponse est émis pour chaque instruction SQL). Pour ce type de serveurs l'application client/serveur nécessite du code source au niveau du serveur.

- **Serveur d'application web :**

Le World Wide Web est la première application client/serveur intergalactique, ce modèle consiste des clients légers, portables qui communiquent avec de très gros serveurs. Un serveur web renvoie des documents lorsque le client les demande.

- **Serveur groupware :**

Le groupware s'intéresse à la gestion d'informations semi-structurées telles que le texte, l'image, courrier, messagerie et ordonnancement des tâches.

- **Serveur d'application objet :**

Dans le cas d'un serveur objet, l'application client/serveur est écrite sous forme d'un jeu d'objets communicants. Les objets clients communiquent avec les objets serveur au moyen d'un courtier d'objet ou ORB (Object Request Broker). Le client invoque une méthode sur un objet distant, l'ORB localise une instance de la classe, appelle la méthode demandée et renvoie les résultats à l'objet client.

2.3.10.2. Orientation Client et orientation serveur :

Les applications Client/Serveur peuvent se différencier par la façon dont les fonctions réparties se partagent entre le client et le serveur.

- **Le modèle orienté serveur :**

La plupart des fonctionnalités sont placées sur le serveur, les applications qui tournent sur ce type de modèle sont faciles à gérer et à installer.

- **Le modèle orienté client :**

Le gros de l'application se tourne sur le client, c'est le modèle le plus traditionnel, il présente une souplesse et facilite la création d'outils frontaux permettant aux utilisateurs de créer leurs applications. Chacun des deux modèles a son domaine d'application, ces avantages et ses inconvénients.

Les spécialistes des architectures Client/Serveur préfèrent plutôt le terme architecture Client/Serveur à deux niveaux et à trois niveaux que modèle orienté client et orienté serveur, mais la logique est la même et dépend entièrement de la manière dont l'application est partagées entre les entités client et serveur.

2.3.11. Les modèles de client/serveur :

Les différents modèles de client/serveur sont liés aux services qui sont assurés par le serveur. On distingue couramment:

- **Le client/serveur de présentation :**

Sur le poste client sont exécuter des fonctions de dialogue avec l'utilisateur et le serveur gère les données et exécute le code applicatif.

- **Le client/serveur de rhabillage :**

Sur le poste client s'exécute les fonctions de dialogues avec l'utilisateur, le serveur s'occupe de la gestion des données, exécute le code applicatif, assure les dialogues simplifiés avec l'utilisateur.

- **Le client/serveur de données :**

Dans ce cas, le serveur assure les tâches de gestion, de stockage et de traitement de données. C'est le cas le plus connu de client/serveur et qui est utilisé par tous les grands SGBD : la base de données avec tous ses outils (maintenance, sauvegarde,...) et installé sur un poste serveur. Sur les clients un logiciel d'accès est installé permettant d'accéder à la base de données du serveur. Tous les traitements sur les données sont effectués sur le serveur qui renvoie les informations demandées (souvent à travers une requête SQL) par le client.

- **Le client/serveur de traitements :**

Dans ce cas le serveur effectue des traitements à la demande du client. Il peut s'agir de traitements particuliers sur des données, de vérification de formulaire de saisie, de traitement d'alarme... Ces traitements peuvent être réalisés par des programmes installés sur des serveurs mais également intégrés dans des bases de données, dans ce cas la partie donnée et traitement sont intégrés.

2.3.12. Architecture client/serveur [15] :

On distingue plusieurs types d'architecture client/serveur :

- **Architecture à 2 niveaux :** L'architecture à deux niveaux (aussi appelée architecture 2-tier) caractérise les systèmes clients/serveurs pour lesquels le client demande une ressource et le serveur la lui fournit directement, en utilisant ses propres ressources. Cela signifie que le serveur ne fait pas appel à une autre application afin de fournir une partie du service.

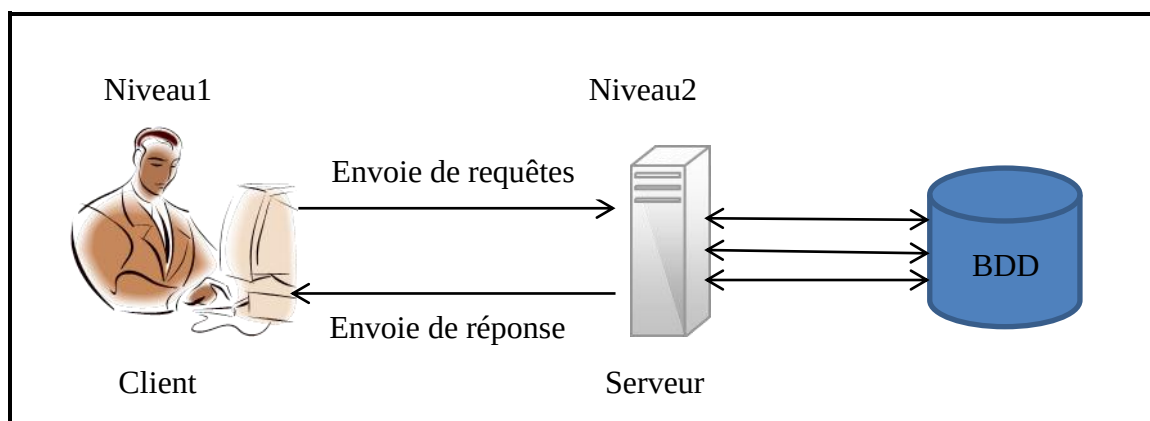


Figure 5: architecture client/serveur à deux niveaux.

- **Architecture à 3 niveaux :**

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

- ✓ Un client, c'est-à-dire l'ordinateur demandeur de ressources, équipée d'une interface utilisateur (généralement un navigateur Web) chargée de la présentation.
- ✓ Le serveur d'application (appelé également middleware), chargé de fournir la ressource mais faisant appel à un autre serveur.
- ✓ Le serveur de données, fournissant au serveur d'application les données dont il a besoin.

Cette architecture est très souvent utilisée sur Internet :

- Le client possède un logiciel de type navigateur. Le serveur middleware fournit une page web au format html. Il génère cette page directement depuis un code source html ou à partir d'un script PHP. Il s'agit souvent d'un serveur apache ou apache2.
- Le serveur middleware utilise les services d'un serveur de base de données (exemple : MySQL) pour stocker les données utilisées.

Il est possible et souvent probable que ces deux serveurs soient une seule et même machine, auquel cas on parle plus de service que de serveur. Dans ce cas, on retombe sur une architecture 2-tier.

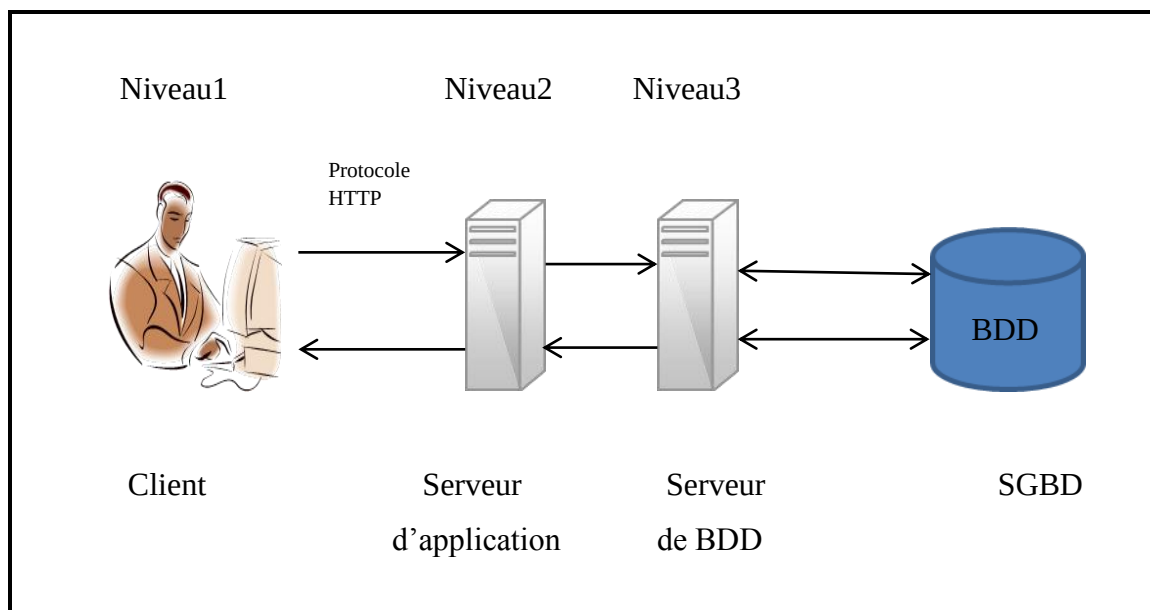


Figure 6 : l'architecture c/s à trois niveaux.

- **Comparaison des architectures à deux et à trois niveaux :**

	Deux Niveaux	Trois Niveaux
Administration du système	Complexe (plus de logique à gérer côté client)	Moins complexe (l'application peut être gérée de façon centrale et les programmes d'application sont rendus visibles aux outils standard de gestion du système)
Sécurité	Faible (sécurité au niveau des données)	Importante (réglage fin au niveau du service, de la méthode ou du type d'objet)
Encapsulation des données	Faible (les tables de données sont accessibles)	Importante (le client invoque des services ou des méthodes)
Performance	Faible (de nombreuses instructions SQL transitent sur le réseau ; les données sélectionnées doivent être chargées sur le client pour analyse)	Bonne (seules les requêtes de service et les réponses sont échangées entre client et serveur) importante (concentre les sessions entrantes ; peut répartir la charge sur de nombreux serveurs)
Facilité de changement d'échelle	Faible (gestion limitée des liens de communication des clients)	Importante (concentre les sessions entrantes ; peut répartir la charge sur de nombreux serveurs)
Réutilisation possible	Faible (l'application client est monolithique)	Importante (réutilisation possible des services et des objets)
Facilité de développement	Importante	En voie d'amélioration (des outils standards peuvent être utilisés pour créer les clients)

		et on voit apparaître des outils capables de développer les côtés serveur et client de l'application)
Infrastructure serveur-serveur	Non	Oui (via le middleware côté serveur)
Intégration des applications du patrimoine	Non	Oui (via des passerelles encapsulées par les services ou les objets)
Support de l'internet	Faible (les limitations de la bande passante internet rendent difficile le chargement des clients et exacerbent celles déjà constatées)	Excellent (de petits clients sont facilement téléchargés comme applets ou beans ; les appels de services distants répartissent la charge sur le serveur)
Support de bases de données hétérogènes	Non	Oui (les applications peuvent utiliser plusieurs bases de données)
Variété des choix de communication	Non (uniquement appels synchrones orientés connexion)	Oui (support d'appels type RPC mais aussi échange de message sans connexion, file d'attente, publication/abonnement et diffusion)
Disponibilité	Faible	Excellente (relance possible du niveau intermédiaire sur d'autres serveurs)

Tableau 1 : Comparaison des architectures à deux et à trois niveaux.

- **Architecture multi niveaux :**

Dans l'architecture à 3 niveaux, chaque serveur (niveaux 2 et 3) effectue une tâche (un service) spécialisée. Un serveur peut donc utiliser les services d'un ou plusieurs autres

serveurs afin de fournir son propre service. Par conséquent, l'architecture à trois niveaux est potentiellement une architecture à N niveaux...

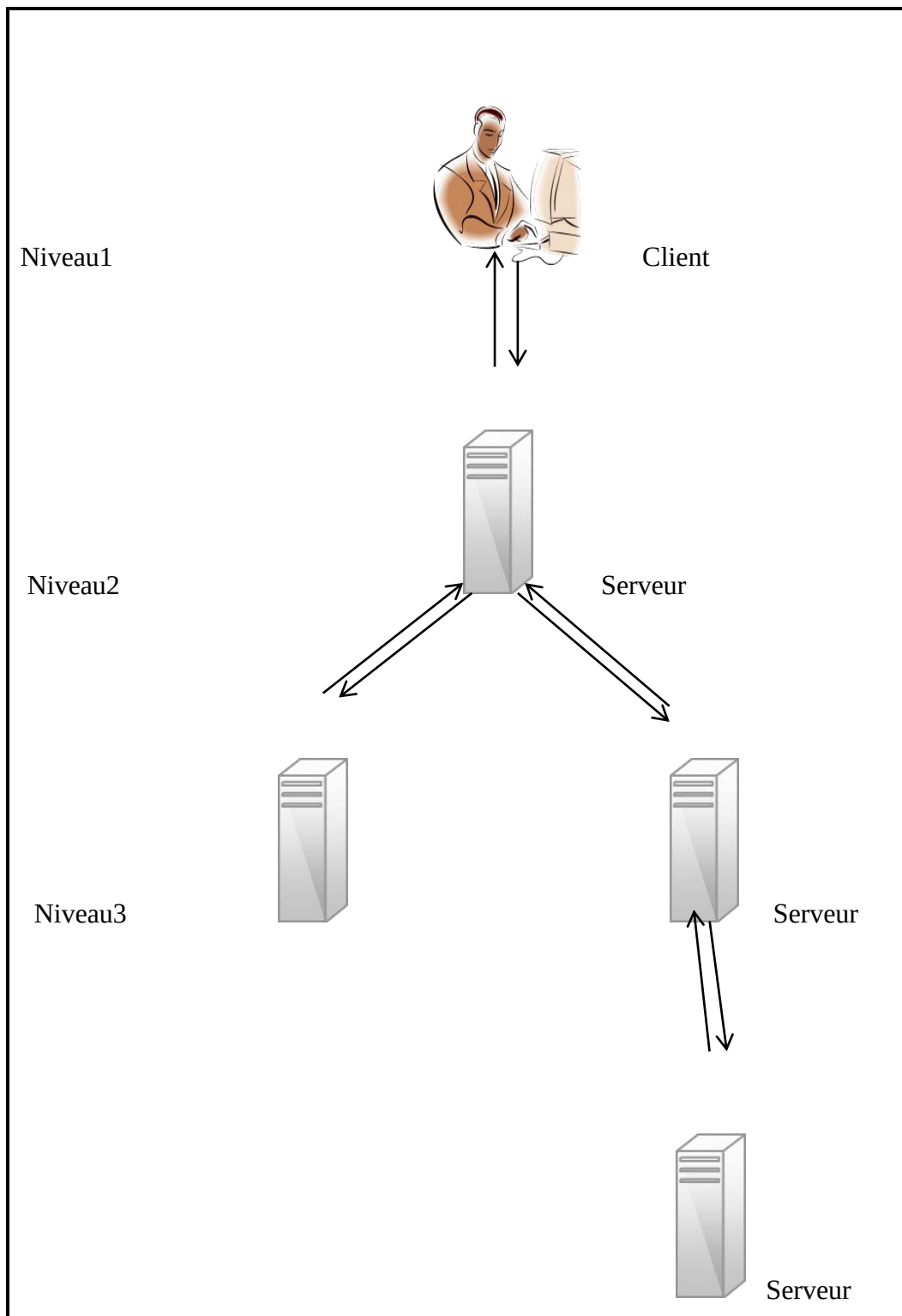


Figure 7 : architecture multi-niveaux.

2.3.13. Les middlewares :

2.3.13.1. Définition du middleware :

Le middleware (ou logiciel médiateur en français), ou bien littéralement « élément du milieu », est l'interface de communication entre les processus. C'est l'ensemble de couches réseaux et de services logiciel qui permettent le dialogue entre les différents composants d'une application répartie .ce dialogue se base sur un protocole applicatif commun, défini par l'API du middleware. Le middleware représente la clef de voute de toute application client/serveur. L'objectif principal du middleware est d'unifier, pour les applications, l'accès et la manipulation de l'ensemble des services disponibles sur le réseau, afin de rendre l'utilisation de ces derniers presque transparente.

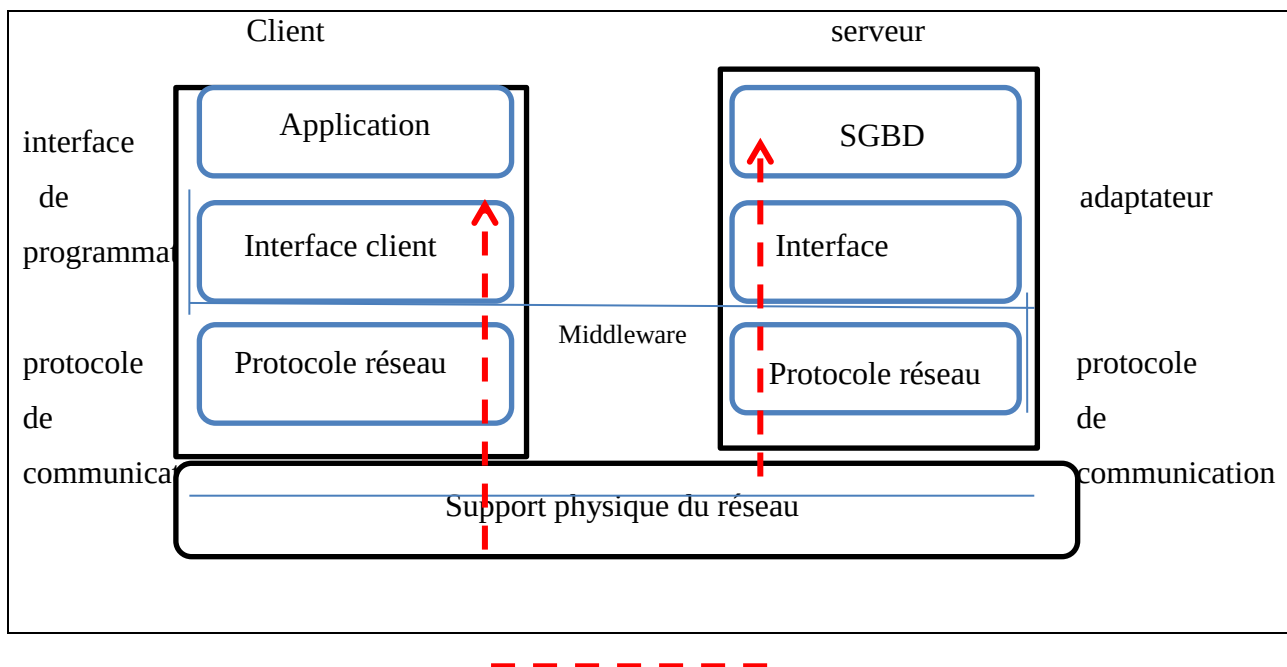


Figure 8: représentation du middleware.

2.3.13.2. Les services du middleware :

Un middleware rend les services suivants :

- **Conversion:** service utilisée pour la communication entre machines mettant en œuvre des formats de données différents.
- **Adressage:** permet d'identifier la machine serveur sur laquelle est localisé le service demandé afin d'en déduire le chemin d'accès. Dans la mesure du possible.

- **Sécurité** : permet de garantir la confidentialité et la sécurité des données à l'aide de mécanismes d'authentification et de cryptage des informations.
- **Communication** : permet la transmission des messages entre les deux systèmes sans altération ; ce service doit gérer la connexion au serveur, la préparation de l'exécution des requêtes, la récupération des résultats et la déconnexion de l'utilisateur.

Le middleware masque la complexité des échanges inter-applications et permet ainsi d'élever le niveau des API utilisés par les programmes. Sans ce mécanisme la programmation d'une application client/serveur serait complexe et difficilement évolutive.

2.3.13.3. Exemples de middleware :

Le choix d'un middleware joue un rôle important dans la structuration du système d'information. Parmi les principaux middlewares on trouve les suivants :

- **DCE** : (Distributions Computing Environment) permet l'appel à des procédures distantes depuis une application. Correspond à RPC (Remote Procedure Call) qui permet d'exécuter des procédures distantes.
- **SQL*Net** : interface propriétaire permettant de faire dialoguer une application cliente avec une base de données oracle. Ce dialogue peut aussi bien être le passage de requêtes SQL que l'appel de procédures stockées.
- **ODBC** : (Objet Data Base Connexion) interface standardisée isolant le client du serveur de données. C'est l'implémentation par Microsoft d'un standard défini par le SQL Access Group. Elle se compose d'un gestionnaire de driver standardisé, d'une API s'interfaçant avec l'application cliente (sous MS Windows) et d'un driver correspondant au SGBD utilisé.

Pour certaines applications devant accéder à des services hétérogènes, il est parfois nécessaire de combiner plusieurs middlewares. On en vient à la notion de client lourd.

2.3.13.4. Les middlewares objet :

Pour permettre la répartition d'objets entre machines et l'intégration des systèmes non objets, il doit être possible d'instaurer une communication entre tous ces éléments. Ainsi est né le concept de middleware objet qui a donné naissance à plusieurs spécifications, dont l'architecture CORBA (Common Object Request Broker Architecture) préconisé par l'OMG (Object Management Group) et DCOM développée par Microsoft.

Ces middlewares sont constitués d'une série de mécanismes permettant à un ensemble de programmes d'inter-opérer de façon transparente. Les services offerts par les applications serveurs sont présentés aux clients sous la forme d'objets. La localisation et les mécanismes mis en œuvre pour cette interaction sont cachés par le middleware.

La communication entre objets gomme la différence entre ce qui est local ou distant, les appels de méthodes d'objet à objet sont traités par un mécanisme se chargeant d'aiguiller les messages vers les objets (locaux ou distants).

2.3.14. Avantages et inconvénients du client/serveur :

2.3.14.1. Avantages de l'architecture Client/Serveur :

Le modèle Client/Serveur est particulièrement recommandé pour des réseaux nécessitant un grand niveau de fiabilité, ses principaux atouts sont :

- **Des ressources centralisées :**

Etant donné que le serveur est au centre du réseau, il peut gérer des ressources communes à tous les utilisateurs, comme par exemple une base de données centralisée, afin d'éviter des problèmes de redondance et de contradiction.

- **Une meilleure sécurité :**

Car le nombre de points d'entrée permettant l'accès aux données est moins important.

- **Une administration au niveau d'un serveur :**

Les clients ayant peu d'importance dans ce modèle, ils ont moins besoin d'être administrés.

- **Un réseau évolutif :** grâce à cette architecture il est possible de supprimer ou rajouter des clients sans perturber le fonctionnement du réseau et sans modifications majeures.

2.3.14.2. Inconvénients de l'architecture Client/Serveur :

L'architecture Client/Serveur a tout de même quelques lacunes parmi lesquelles :

- Un coût élevé : dû à la technicité du serveur.
- Un maillon faible : le serveur est le seul maillon faible du réseau Client/Serveur, étant donné que tout le réseau est architecturé autour de lui ; Heureusement que le serveur a une grande tolérance aux pannes.

3. J2EE

3.1. Définition :

J2EE (aujourd'hui appelé JEE – Java Enterprise Edition) est une spécification pour le langage de programmation Java de Sun destinée aux applications d'entreprise. La première version des librairies J2EE a été mise à disposition des développeurs en 1999. J2EE offre une plate-forme de développement et déploiement en langage Java pour les applications distribuées à plusieurs niveaux. La version courante du langage est la 1.6 (ou Java 6).

3.2. Fonctionnement interne [2] :

Le langage Java, sur lequel les librairies J2EE sont utilisées, met à disposition un compilateur et une machine virtuelle (JVM – Java Virtual Machine) qui se charge de créer un environnement standard pour le lancement de l'application sur tout type de système opérationnel. Le compilateur compile le code source et produit le byte Code, soit un code intermédiaire qui sera ensuite lu par la machine virtuelle Java. Chaque système opérationnel majeur possède une JVM expressément codée.

3.3. Architecture [2]:

J2EE ajoute nombreuses couches 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:**

Technologies liées à la production des interfaces web dynamiques, par exemple JSP (Java Servlet Pages) et servlet.

- **Technologie Enterprise application:**

technologies plus directement liées à la logique de business : EJB (Enterprise Java Bean), JavaMail, JMS (Java Message Service), JTA (Java Transaction), etc.

- **Technologie web services:**

technologies 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:**

technologies liées à la gestion de la technologie entreprise afin de réaliser l'accès et l'échange d'information entre machines et services distribués : JAAS (Java Authentication and Authorization Service), JCA (Java Connector Architecture)

Pour expliquer l'utilisation de ces technologies on peut imaginer que les technologies entreprise sont utilisées pour gérer l'accès aux données (généralement un ou plusieurs data bases), 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 commerciales et les technologies de gestion gèrent tous les processus informationnels assurant la sécurité des transactions.

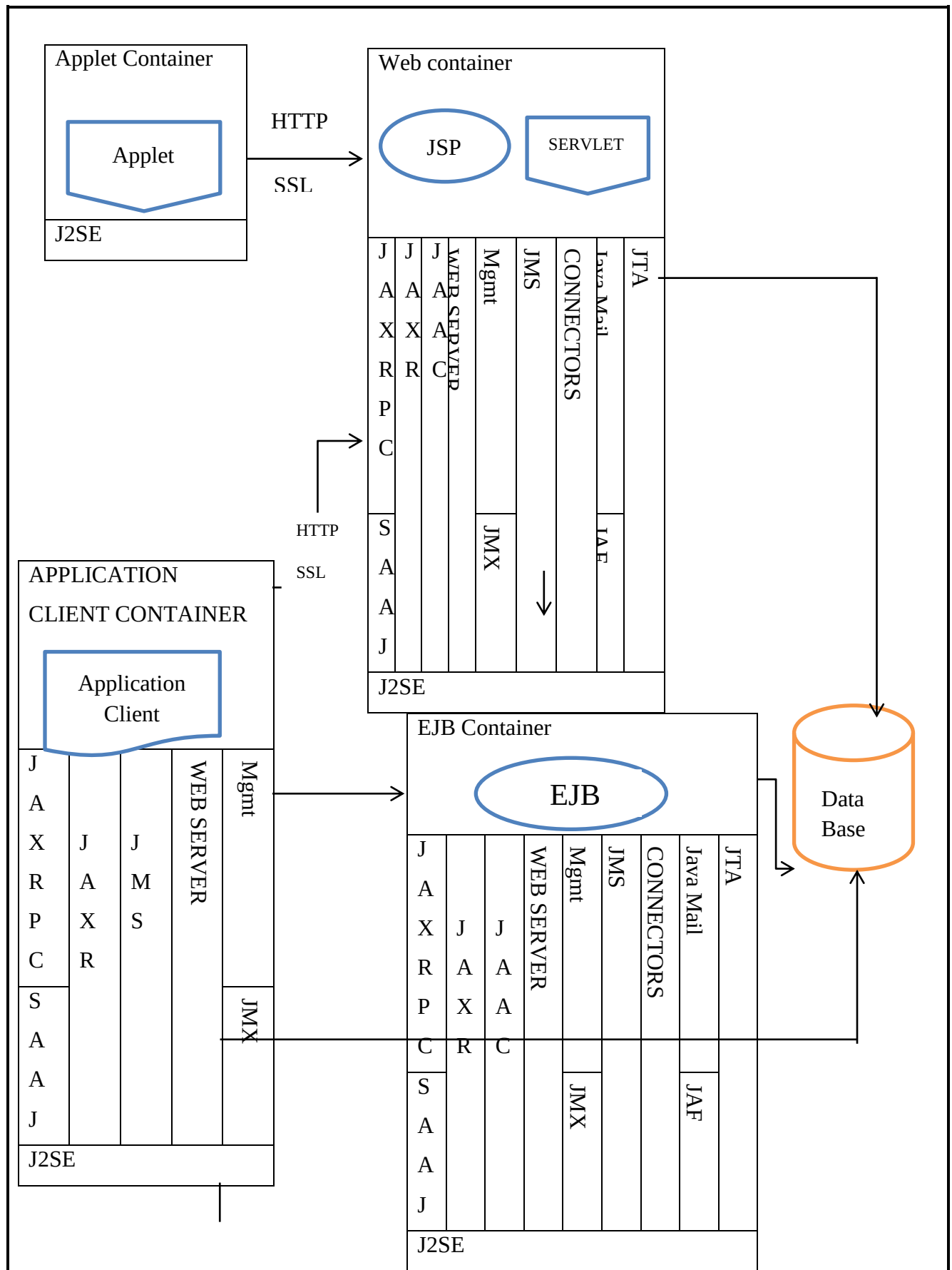


Figure 9: architecture J2EE.

3.3.1.1. JSP : Java Server Page

JSP est une technologie pour le développement de pages Web incluant du contenu dynamique. Contrairement à une page HTML qui ne contient que du contenu statique qui reste par définition toujours le même, JSP peut changer selon l'identité du visiteur, de son navigateur Internet, de l'heure, de la configuration du système, des actions du visiteur, etc. Une page JSP contient des balises standards, comme du HTML (ou du WML, XML...), comme toute page web normale. Pourtant, une page JSP contient aussi des éléments JSP spécifiques (scriptlets), permettant au serveur l'insertion dynamique de contenu (contenu de BDD, préférences du visiteur...). Lorsqu'un utilisateur accède à une page JSP, le serveur exécute les éléments JSP, fusionne les résultats avec les parties statiques de la page, et envoie le tout au navigateur. JSP définit des outils standards utiles pour n'importe quelle application web comme l'accès à des composants JavaBeans, les contrôles d'accès entre pages, le partage d'information entre requêtes, pages et utilisateurs.

3.3.1.1.Servlet

Technologie Java utilisée pour effectuer des traitements coté serveur en réponse aux requêtes provenant en général de poste clients distants. Bien que les Servlets puissent répondre à n'importe quel type de requête, elles sont généralement employées pour répondre à des requêtes de type HTTP et qui permettent de retourner dynamiquement des pages HTML.

3.3.1.2.les composants EJB (Entreprise Java Bean) :

Cette architecture propose un cadre pour créer des composants distribués (déployés sur des serveurs distants) écrit en langage de programmation Java hébergés au sein d'un serveur applicatif permettant de représenter des données (EJB dit entité), de proposer des services avec ou sans conservation d'état entre les appels (EJB dit session), ou encore d'accomplir des tâches de manière asynchrone (EJB dit message). Notions d'annotations. EJB est grossièrement équivalent à COM/DOM de Microsoft dans son architecture, mais comme toute architecture basée sur Java, elle peut être déployée sur quasi-tous les OS.

✓ EJB entité :

Un Bean entité à l'inverse est un objet persistant qui modélise les données à l'intérieur de l'entrepôt de données, c.à.d. qu'il encapsule les données dans un objet. Par rapport au beans de session qui peuvent être utilisés par n'importe quel client, les beans entités peuvent être employés en même temps par plusieurs clients, mais ils doivent conserver une identité unique

via une clé primaire .en réalité, avec l'architecture de conteneurs J2EE, vous pouvez choisir soit de laisser le conteneur gérer l'état persistant du Bean entité pour vous, soit de mettre en œuvre vous-même cet état à l'intérieur du Bean.

✓ **EJB session :**

Il existe également deux formes de beans de session, ceux qui possèdent un état (stateful) et ceux qui n'en possèdent pas (stateless) .un Bean session qui possède un état est un objet transitoire permettant de représenter l'interaction d'un client avec le système : le Bean de session exécute les requêtes clients au sein de l'application, en accédant à la base de données, etc. et , lorsque les opérations clientes sont effectuées, il est détruit (c.à.d. que sa durée de vie est celle de la session cliente). A l'inverse un Bean de session sans état ne conserve pas d'état entre les requêtes clientes. En général, ce type de Bean de session permet de mettre en œuvre un service spécifique ne nécessitant pas d'état client.

✓ **EJB message :**

3.3.5. Le serveur d'application

Les API J2EE ont été projetés pour travailler avec un particulier type de serveur appelé J2EE Application Server. Un serveur d'application est une couche software résident sur une machine serveur qui fournit les services que la technologie J2EE nécessite. Il y a plusieurs applications serveurs. Parmi les produits commerciaux on rappelle Bea WebLogic, IBM Websphere, Sun Application Server, Pramati, etc. Parmi les produits libres le plus connu est JBOSS. Les différences principales entre les différents serveurs d'applications sont relatives aux opérations de déploy, clustering, etc... Par contre toutes les fonctionnalités qui concernent strictement le fonctionnement des applications J2EE adhèrent aux spécifications proposées par la Sun. Un serveur d'application s'installe et se lance comme un normal serveur web (de fait il met à disposition des services typiques d'un serveur web). En plus il possède un panneau d'administration accessible par un poste distant à travers lequel on peut définir les différents services. On peut considérer le serveur d'application comme une application Java standalone exécutée par une J2SE qui dessert les requêtes provenant des différents clients.

3.4. J2EE et les architectures distribuées :

3.4.1. Couche de présentation :

La couche de présentation est la face visible de notre application. Les clients se divisent en deux catégories :

- ✓ **Client léger** : les navigateurs HTML car ils sont préinstallés sur la plupart des plateformes avec le système d'exploitation.
- ✓ **Client lourd** : utilisant une interface graphique à base de formulaires riches et de contrôles graphiques complexes. avec J2EE pour le client léger la technologie J2EE possède les APIs standard Java Servlet et Java Server Pages. Le servlet container de l'application server permet le déploiement de ces composants web sur tout type de systèmes opérationnels. Des Framework comme Struts sont mis à disposition pour améliorer l'architecture de cette couche. Pour le client lourd les bibliothèques Java mettent à dispositions les bibliothèques Swing (propre à J2EE) et AWT. La portabilité d'AWT et de Swing provient de la conception interne des classes existantes. Toutes les plates-formes possèdent des systèmes d'affichage vidéo différents: Xwindows Motif pour Unix, Win32/GDI pour Windows, Apple/MAC. Sun a donc mis en place la notion de Composant Peer simplifiant le portage vers une plate-forme cible. Cette partie est donc native (code C/C++) et utilise des bibliothèques liées à chaque plate-forme. Quant à la conception du modèle objet Swing, elle est basée sur le concept de conteneur/composant et utilise intensivement le design pattern MVC (Modèle Vue Controller).

3.4.2. Couche de service : La couche de service est la partie de notre architecture contenant les traitements applicatifs. On a plusieurs alternatives au développement de services dans cette couche. Soit on prend l'entière responsabilité de l'implémentation des composants sans l'utilisation des services du Framework , soit on fait appel au Framework pour :

- ✓ La gestion des transactions automatiques
- ✓ Le cache d'objets (Pooling)
- ✓ La montée en charge et le multithreading
- ✓ Les composants asynchrones, ...

La responsabilité du conteneur est de nous fournir tous ces services en nous proposant un canevas dans lequel on implémentera nos composants. Avec J2EE Le choix de l'implémentation de cette couche réside dans deux approches : utiliser les EJB ou ne pas les utiliser. Les composants concernés par cette couche sont les EJB Session. Sun spécifie un certain nombre d'éléments dans J2EE allant de la gestion des transactions en passant par la sécurité à la synchronisation des tâches. Encore une fois, Java se distingue par l'ouverture de ces APIs avec comme maître mot : la portabilité du serveur.

3.4.3. Couche d'objets métier :

Les objets métiers sont représentés par l'ensemble des objets persistants du domaine de l'application. Une facture, un bon de commande ou tout autre objet nécessitant d'être stocké en base. Cette couche assure l'indépendance totale entre le client et le type de stockage utilisé (SGBDR, SGBDO, fichiers XML, ...). Le client doit posséder uniquement une vue sur un objet avec l'ensemble de ces attributs et non un éventuel curseur (ResultSet et RecordSet) pointant vers une ligne d'une quelconque base. Cette couche est en étroite collaboration avec la couche de persistance qui assure la création des objets métier de manière totalement transparente. Avec J2EE ce type d'objet est représenté par les EJB Entity. Dans ce cas, vos entités métiers sont couplées aux interfaces du Framework . Une autre approche consiste à développer des objets métiers simples n'héritant d'aucune interface ou classe du Framework en utilisant des outils de persistance fonctionnant par introspection. C'est à dire que le mapping entre nos objets et nos tables est réalisé via des fichiers de configuration externes à notre application. C'est le cas de la spécification JDO qui cible ce genre de besoin avec des produits comme Castor, JUDO, KODO etc.... Ainsi, on ne couple pas nos objets du domaine à notre Framework de persistance.

3.4.4. Couche d'accès aux données :

Cette couche est responsable de la création, destruction et chargement des objets métier de manière totalement transparente. Son rôle est de masquer entièrement l'opération de création et de manipulation de tables (SGBDR) ou autre types de stockages spécifiques (Fichiers à plats, ...). En règle général, nous avons deux alternatives : Implémenter cette couche nous-même par l'intermédiaire de classes jouant le rôle d'usines à objets métier (pattern Factory), ou laisser le soin à un Framework tiers de réaliser cette tâche. Tout dépend de la nature de nos besoins et de la complexité des données manipulées pour

mettre en œuvre le mapping objet/relationnel. Cette partie, souvent délaissée dans la mise en place d'un projet est le tendon d'Achille d'une architecture. La conception de la couche de persistance doit se faire avec soin et en tenant compte de multiples facteurs tels que la montée en charge, la complexité de mise en œuvre mais aussi l'impact d'évolutions futures de la base. Voyons ce que proposent J2EE dans ce domaine. Dans le monde Java, l'API JDBC (Java Database Connectivity) est en charge de la communication entre un client et un SGBDR. JDBC s'adresse uniquement aux bases de données SQL.

Si on prend le choix de gérer nous-même la persistance des données, il nous suffira d'implémenter des classes utilisant JDBC. Dans le cas où nos besoins sont plus complexes, il faudra faire l'acquisition d'un produit (gratuit ou payant) réalisant cette tâche à notre place. Il en existe un certain nombre s'interfaçant avec les spécifications EJB (EJB Entity). Le but étant de fournir un composant standard qui utilise les services des outils de mapping objet/relationnel de manière totalement transparente.

Il existe deux types d'EJB : BMP (Bean Managed Persistance) et CMP (Container Managed Persistance). Dans les deux cas, l'API utilisée pour communiquer avec les bases de données relationnelles est JDBC. Cette API est spécifiée par Sun, intégrée à J2EE et dispose de plusieurs implémentations à travers de multiples drivers JDBC disponibles sur le marché (Oracle, SQL Server, DB2, ...). JDBC permet aussi de convertir le résultat de requêtes SQL en XML (JDBC 3.0).

4. La sécurité informatique :

1.1. Définition de la sécurité informatique [4] :

La sécurité informatique c'est l'ensemble des moyens mis en œuvre pour réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles. Il convient d'identifier les exigences fondamentales en sécurité informatique. Elles caractérisent ce à quoi s'attendent les utilisateurs de systèmes informatiques en regard de la sécurité :

- ✓ **disponibilité** : demande que l'information sur le système soit disponible aux personnes autorisées.
- ✓ **Confidentialité** : demande que l'information sur le système ne puisse être lue que par les personnes autorisées.
- ✓ **Intégrité** : demande que l'information sur le système ne puisse être modifiée que par les personnes autorisées.

La sécurité recouvre ainsi plusieurs aspects :

- intégrité des informations (pas de modification ni destruction)
- confidentialité (pas de divulgation à des tiers non autorisés)
- authentification des interlocuteurs (signature)
- respect de la vie privée (informatique et liberté). Du point de vue de la sécurité informatique, une menace est une violation potentielle de la sécurité. Cette menace peut-être accidentelle, intentionnelle (attaque), active ou passive.

1.2. Principaux défauts de sécurité :

Les défauts de sécurité d'un système d'information les plus souvent constatés sont :

- Installation des logiciels et matériels par défaut.
- Mises à jour non effectuées.
- Mots de passe inexistants ou par défaut.
- Services inutiles conservés (NetBIOS...).
- Traces inexploitées.
- Pas de séparation des flux opérationnels des flux d'administration des systèmes.
- Procédures de sécurité obsolètes.
- Eléments et outils de test laissés en place dans les configurations en production.
- Authentification faible.
- Télémaintenance sans contrôle fort.

1.3. Protections :

✓ Formation des utilisateurs

On considère généralement que la majorité des problèmes de sécurité sont situés entre la chaise et le clavier

Discrétion : la sensibilisation des utilisateurs à la faible sécurité des outils de communication et à l'importance de non divulgation d'informations par ces moyens est indispensable. En effet il est souvent trop facile d'obtenir des mots de passe par téléphone ou par e-mail en se faisant passer pour un membre important de la société.

Virus : plusieurs études récentes montrent que 1/3 des utilisateurs ouvriraient encore une pièce jointe d'un courrier nommée « i love you » et que la moitié ouvriraient une pièce nommée « ouvrez-ça » ou similaire ; L'information régulière du personnel est nécessaire, attention toutefois aux rumeurs (hoax).

Charte : l'intérêt principal d'une charte d'entreprise est d'obliger les employés à lire et signer un document précisant leurs droits et devoirs et par la même de leur faire prendre conscience de leur responsabilité individuelle.

✓ **Poste de travail**

Le poste de travail reste un maillon faible de la sécurité. On améliore sa sécurité en dotant le PC d'une puce dédiée à la sécurité. Elle sera chargée de vérifier l'intégrité du BIOS, du chargement de l'OS, de sauvegarder les clés et certificats (PKI) et connaîtra les protocoles de cryptage (RSA, DES...).

✓ **Antivirus**

Principale cause de désagrément en entreprise, les virus peuvent être combattus à plusieurs niveaux. La plupart des antivirus sont basés sur l'analyse de signature des fichiers, la base des signatures doit donc être très régulièrement mise à jour sur le site de l'éditeur (des procédures automatiques sont généralement possibles). On a deux modes de protection :

- 1 Généralisation de l'antivirus sur toutes les machines, il faut absolument prévoir une mise à jour automatique de tous les postes via le réseau.
- 2 Mise en place d'un antivirus sur les points d'entrée/sortie de données du réseau après avoir parfaitement identifiés tous ces points. La rigueur de tout le personnel pour les procédures doit être acquise.

✓ **Pare-feu (fire wall) ou garde barrière**

C'est une machine dédiée au routage entre LAN et Internet. Le trafic est analysé au niveau des datagrammes IP (adresse, utilisateur, contenu...). Un datagramme non autorisé sera

simplement détruit, IP sachant gérer la perte d'information. Une translation d'adresse pourra éventuellement être effectuée pour plus de sécurité (protocole NAT Network Address Translation RFC 1631+2663). Mais cependant il faut toutefois faire attention un firewall est inefficace contre les attaques ou les bévues situées du côté intérieur et qui représentent 70% des problèmes de sécurité.

✓ **Authentification et cryptage**

L'authentification est basée sur les 3 principes :

- Savoir : login, mot de passe...
- Être : biométrie (empreintes...)
- Avoir : clés USB, carte à puce, « token ».

Une authentification est dite forte lorsqu'elle utilise deux mécanismes différents (carte à puce avec mot de passe par exemple). "Nom + mot de passe + date" sont cryptés avec des clés publiques et privées. Le cryptage de la date évite la réutilisation éventuelle du message par un pirate. Par le cryptage on peut identifier de manière sûre l'utilisateur connecté. Pour éviter l'espionnage, la modification du contenu, l'ajout de message... on pourra utiliser la signature électronique (CRC crypté en fin de message) ou crypter toute l'information. Les infrastructures PKI (Public Key Infrastructure) devraient se développer. Pour l'instant, le protocole SSL (Secure Socket Layer) domine toujours largement le marché de l'authentification sur les sites marchands. Radius, Tacacs ou IPSec (qui comporte un processus d'authentification dans son en-tête) constituent encore la solution retenue par la majorité des entreprises.

1 Cryptage symétrique

Une même clé est utilisée pour crypter et décrypter le message, très efficace et assez économe en ressources CPU cette technique pose le problème de la distribution des clés dans un réseau étendu (exemple DES, triple DES ou le récent AES).

2 Cryptage asymétrique

Chaque utilisateur dispose d'un jeu unique de clés, dont l'une est privée (secrète) et l'autre publique (exemple RSA). Pour recevoir des documents protégés, le détenteur d'un jeu de clés envoie sa clé publique à ses interlocuteurs, qui l'utilisent pour chiffrer les données avant de les lui envoyer. Seul le destinataire et détenteur des clés peut lire les informations en associant

sa clé privée à sa clé publique. Cette technique nécessite des clés plus longues pour une sécurité équivalente.

3 Protocoles courants

- SSL (Secure Socket Layer)
- SET (Secure Electronic Transaction)
- C-SET (Chip Secure Electronic Transaction)
- S/MIME (Secure Multipurpose Internet Mail Extension)
- PGP (Pretty Good Privacy)

4 PKI (Public Key Infrastructure)

L'infrastructure PKI repose sur la notion de chiffrement asymétrique. Pour s'authentifier, en revanche, le détenteur des clés utilise un certificat, sorte de document électronique faisant office de carte d'identité électronique. Inséré dans un message, lors d'un paiement sur Internet par exemple, ce certificat joue le rôle de signature numérique. Il contient des informations relatives à l'identité du détenteur, son champ d'application (date de validité, types d'applications, etc.) et la clé publique. Un tiers de confiance garantit l'association entre un individu et les données contenues dans le certificat. La gestion des certificats en interne implique des infrastructures lourdes afin d'enregistrer les demandes, de vérifier la validité des certificats, de gérer les pertes ou les vols (risques d'autant plus importants lorsque le certificat est inclus dans un support physique tel qu'une carte à puce). Il faudra, de plus, assurer la protection des serveurs contre le piratage.

✓ Messageries

Les messageries sont très utilisées et posent quelques problèmes de sécurité particuliers. De plus la majorité des virus utilisent actuellement ce vecteur.

1 Sécurité des messages

- Confidentialité : seul le chiffrement peut l'assurer.
- Intégrité : le message reçu est identique à celui émis, le scellement et la signature électronique sont nécessaires.
- Contrôle d'accès : uniquement les personnes autorisées peuvent émettre des messages
- Non répudiation : utilisation d'un tiers de confiance.

✓ Détection d'intrusion

Même si l'intrus parvient à franchir les barrières de protection (coupe-feu, système d'authentification, etc.), il est encore possible de l'arrêter avant qu'il n'attaque. Placés sur le réseau de l'entreprise, les outils de détection d'intrusion décèlent tout comportement anormal ou trafic suspect.

1.4. J2EE et la sécurité :

Vu la complexité de ces serveurs d'application il est très difficile d'assurer la protection contre les menaces et les intrusions dans ce type d'architecture, donc il faut faire attention aux risques que peut produire un serveur mal configuré et essayer de fermer les portes d'entrées que peuvent laisser de telles serveurs pour diminuer les risques d'attaque et protéger ce type d'application

Conclusion :

Dans cette partie nous avons présenté le paradigme Client/Serveur, sa mise en œuvre et son fonctionnement, nous avons parlé de la plateforme J2EE, de son architecture, de ses composants. Ainsi que des menaces qui pèsent sur les systèmes informatiques et de la manière de les protéger.

PARTIE II

ANALYSE ET CONCEPTION

Introduction

Pour développer des applications informatiques, il est nécessaire d'utiliser une démarche méthodologique et rigoureuse. Alors le choix d'une méthode est très important.

Dans le cadre de notre projet nous avons optés pour une méthode 2TUP (Annexe A) basée sur le langage de modélisation unifié (UML) qui est comme son nom l'indique est l'unification des méthodes de modélisation par objet à savoir (OMT, BOOCH, OOSE).

Plusieurs raisons nous ont conduits à adopter UML dans notre modélisation. En effet :

- Il a été normalisé par l'OMG, qui est une organisation internationale se chargeant de la standardisation des technologies objets, c'est donc un langage standard compréhensible par tout le monde.
- Malgré son évolutivité, il est stable et facilite la compréhension du système grâce à ses représentations simples.
- Il permet d'utiliser les principes et les concepts objet pour enrichir la phase de la conception des systèmes.

ETUDE PRELIMINAIRE

Introduction

L'étude préliminaire est la toute première étape de notre processus de développement 2TUP. Elle consiste à effectuer un premier repérage des besoins fonctionnels et opérationnels, en utilisant principalement le texte, ou diagrammes très simples. Elle prépare les activités plus formelles de capture des besoins fonctionnels et de capture des besoins techniques.

Pour se faire, on a suivi les points suivant :

- Présentation de l'organisme d'accueil
- Analyse de l'existant (champ d'étude)
- Présentation du projet à réaliser.
- Recueil des besoins fonctionnels.
- Choix techniques.
- Identification des acteurs.
- Identification des messages.
- Modélisation de contexte.

**PRESENTATION DE
L'ORGANISME
D'ACCUEIL
« ENIEM »**

Introduction :

Pour bien cerner notre projet il est essentiel de bien comprendre le contexte dans lequel il est posé et d'avoir une vue générale sur l'organisme d'accueil, ce qui permet de préciser les frontières du domaine d'étude.

I. Présentation de l'ENIEM**1. situation géographique :**

L'Entreprise ENIEM (Entreprise Nationale des Industries de Electroménagers) se trouve au sein de la zone industrielle AISSAT_IDIR OUED_AISSI a10 Km de TIZI OUZOU, elles s'étale sur une surface totale de 5.5 Hectares, sa direction générale se trouve au Chef-lieu de TIZI_UOZOU à proximité de la gare ferroviaire.

2. Historique :

ENIEM résulte d'un contact « produit en main » établi dans le cadre du premier plan quadriennal, et signé le 21 Aout 1971 avec un groupe d'Entreprises allemandes présentées par le chef de file D.I.A.G (Société allemande) pour une valeur de 400 millions de dinars les travaux de génie civil ont été entamés en 1977 et la réception des bâtiments avec tous les équipements nécessaires au lieu en juin 1977

En 1983, l'ENIEM essuie la restructuration de SONELEC en 1983, elle est donc une entreprise au statut de société national.

En 1989, l'ENIEM est passée à l'autonomie, les premières réformes ont été engagées et dans ce cadre l'ENIEM fut dotée de tous les organes de gestion légaux :

- Une assemblée générale.
- Un conseil d'Administration
- Un capital Social.

Le champ d'activité de l'entreprise ENIEM consiste à la production, le développement, la recherche dans le domaine de l'électroménager, ainsi que la prise en charge de la fonction commerciale, la promotion des exportations et du service après-vente.

Actuellement l'entreprise ENIEM est constituée de :

- La direction générale.
- Unité froid.
- Unité cuisson.
- Unité climatisation.
- Unité prestations techniques (UPT).
- Unité commerciale (UC).
- Unité produits sanitaires.
- La filiale FILAMP (Mohammedia).

3. la gamme des produits fabriqués par l'ENIEM :

- **Appareils de réfrigération et de congélation :** Réfrigérateur 160 L. Réfrigérateur 200 L. Réfrigérateur 240 L. Réfrigérateur 300 L (à deux portes). Réfrigérateur 330 L. Combiné 290 L. Congélateur 220 L.
- **Appareils de climatisation :**
Types fenêtre : 123/124 de puissance frigorifique de : 2.62, 163/164 de puissance frigorifique de : 4.49, 243/244 de puissance frigorifique de : 4.36
Split système : S/320 de puissance frigorifique de : 3.27, S/430 de puissance frigorifique de : 4.38, S/530 de puissance frigorifique de : 5.23, S/560 de puissance frigorifique de : 5.23, SPM/230 de puissance frigorifique de : 5.23, SPM/ 300 de puissance frigorifique de : 3.50, SPM/ 360 de puissance frigorifique de : 3.60
- **Appareils de cuisson :** Cuisinière à 4feux, Cuisinière à 5feux.
- **Appareils de chauffage :** Chauffe à eau à 5 L, Chauffe bain à 10 L.
- **Lampes à incandescence tous types confondus.**

A l'origine, ENIEM fabrique ses produits sous les marques et licences suivantes :

- Réfrigérateurs PM : licence BOSCH (Allemagne), réfrigérateur GM : MITSUI et TOSHIBA (Japon).
- Cuisinières : TECHNOGAZ (Italie).
- Climatiseurs : activités démarrées avec licence AIR WELL.
- Machines à laver et chauffage eau : sous contrats avec plusieurs marques.

4. Missions et objectifs

La mission de l'ENIEM est d'assurer la production, le montage, la commercialisation, le développement et la recherche dans les différentes branches de l'électroménager notamment :

- Les appareils de réfrigération et de congélation par l'unité froid.
- Les appareils de cuisson par unité cuisson.
- Les appareils de climatisation par l'unité climatisation.
- Les produits sanitaires par unité d'AIN DEFLA.

Et a pour principaux objectifs

- L'amélioration de la qualité des produits.
- La maîtrise des coûts de production.
- L'augmentation des capacités d'études et de développement.
- L'amélioration de la maintenance de l'outil de production des installations.
- La valorisation des ressources humaines.

- L'augmentation des taux d'intégration (interne et externe).
- L'augmentation du volume de production.
- Demande (marché et externe).

5. Description de la structure organisationnelle de l'entreprise ENIEM

Direction générale : est 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 unités. Situé à proximité de l'ex-gare routière de TIZI OUZOU.

Direction planification et contrôle de gestion : assure le contrôle de la gestion. De l'audit finance ainsi le budget de l'entreprise. Elle assure également les tâches suivantes :

- La réalisation et la présentation de tous les travaux permettant de produire une information complète et cohérente des activités de l'entreprise.
- Préparation, établissement et le suivi du budget de l'entreprise.
- Planification et organisation du programme annuel d'audit finance ainsi sa réalisation.
- Le contrôle de rapport d'activités.
- L'exploitation et l'interprétation des résultats des audits finances, et faire les recommandations nécessaires.

Direction développement et partenariat :

La direction assure l'étude et le développement du produit fini ainsi que les actions de partenariat et de sous-traitance.

- Elle définit et supervise les actions de développement des produits existants.
- Suit avec la direction industrielle les actions de développement des processus de fabrication et de modernisation de l'outil de production.
- Participe à la définition de l'organisation de la production dans l'objectif de flexibilité, de réduction des coûts de fabrication.

Direction des finances et comptabilité :

Ses missions principales sont :

- Assure la comptabilisation dans les délais de toutes les opérations permettant l'établissement du bilan, du compte de résultat et des déclarations fiscales de l'entreprise.
- Analyse les équilibres financiers de l'entreprise.
- Définit la politique bancaire et l'orientation budgétaire.
- Gère la trésorerie (recette et dépenses) et contrôle les rédactions fiscales périodiques.
- Rédige, vérifie et approuve les dispositions décrites relatives au fonctionnement efficace de son activité.

Direction industrielle

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

- Définit les programmes de production en fonction de la demande commerciale et des capacités installés avec le souci de la rentabilité.
- Améliore la gestion de production en relation avec la structure informatique (GPAO).
- Organise et anime l'industrialisation de produits nouveaux
- Définit une politique d'amélioration de la maintenance des équipements de production et en assurer le suivi.

Direction des ressources humaines (champs d'étude):

Elle pilote le recrutement, l'accueil l'information et gère le plan de carrière de personnel et les pouvoirs publics.

- Définit et exécute les plans de formation des besoins de l'entreprise et suivant les niveaux de qualification du personnel.
- Gère les éventuels conflits collectifs dans le respect de la loi et préparer les décisions de direction.
- Organise et promouvoir la médecine de travail.

Les unités :

Unité froid : C'est l'unité la plus importante du point de vue effectif, elle produit plusieurs modèles de réfrigérateurs et de congélateurs. Ces activités sont :

- Transformation de la tôle ;
- Traitement et revêtement de surface (peinture, plastification).
- Fabrication de pièces métalliques (condensateurs, évaporateur).
- Isolation
- Thermoformage
- Assemblage

Unité cuisson : Cette unité est spécialisée dans la production de différents types de cuisinières. Ces activités sont :

- Transformation de la tôle.
- Traitement et revêtement de surface (émaillage, zingage, chromage).
- Assemblage de cuisinières

Unité climatisation : Elle est spécialisée dans la fabrication et montages de plusieurs types de climatiseurs. Ces activités sont :

- Transformation de la tôle.
- Traitement et revêtement de surface (peinture).
- Assemblage de climatiseurs.

En plus de ces activités de réalisation, les unités de production, (froid, cuisson et climatisation) assurent, en leurs seins respectifs, les activités suivantes :

- Etudes\méthodes de fabrication.
- Achats.
- Contrôle (réception, en cours de fabrication, final).
- Stockage (magasin, atelier)
- Maintenance.
- Sécurité industrielle.

Unité prestation technique : Cette unité est chargée de fournir et d'exploiter les moyens techniques communs ainsi que la gestion de la totalité des infrastructures communes (bâtiments, voiries, éclairages etc. ...).

Elle assure également la réalisation de pièces de rechanges pour la production, la conception et la fabrication d'outillage (modules, outils etc...), assure toutes les activités informatiques des unités.

Unité commerciale : Elle est chargée de la commercialisation de produit de l'entreprise, de la gestion du réseau et de service après-vente.

Unité Filamp (filiale) : Sa mission globale est de produire des lampes d'éclairages domestiques ainsi que des lampes de réfrigérateurs.

Unité produit sanitaire : Elle fabrique du matériel sanitaire (baignoire, évier, lavabo...).

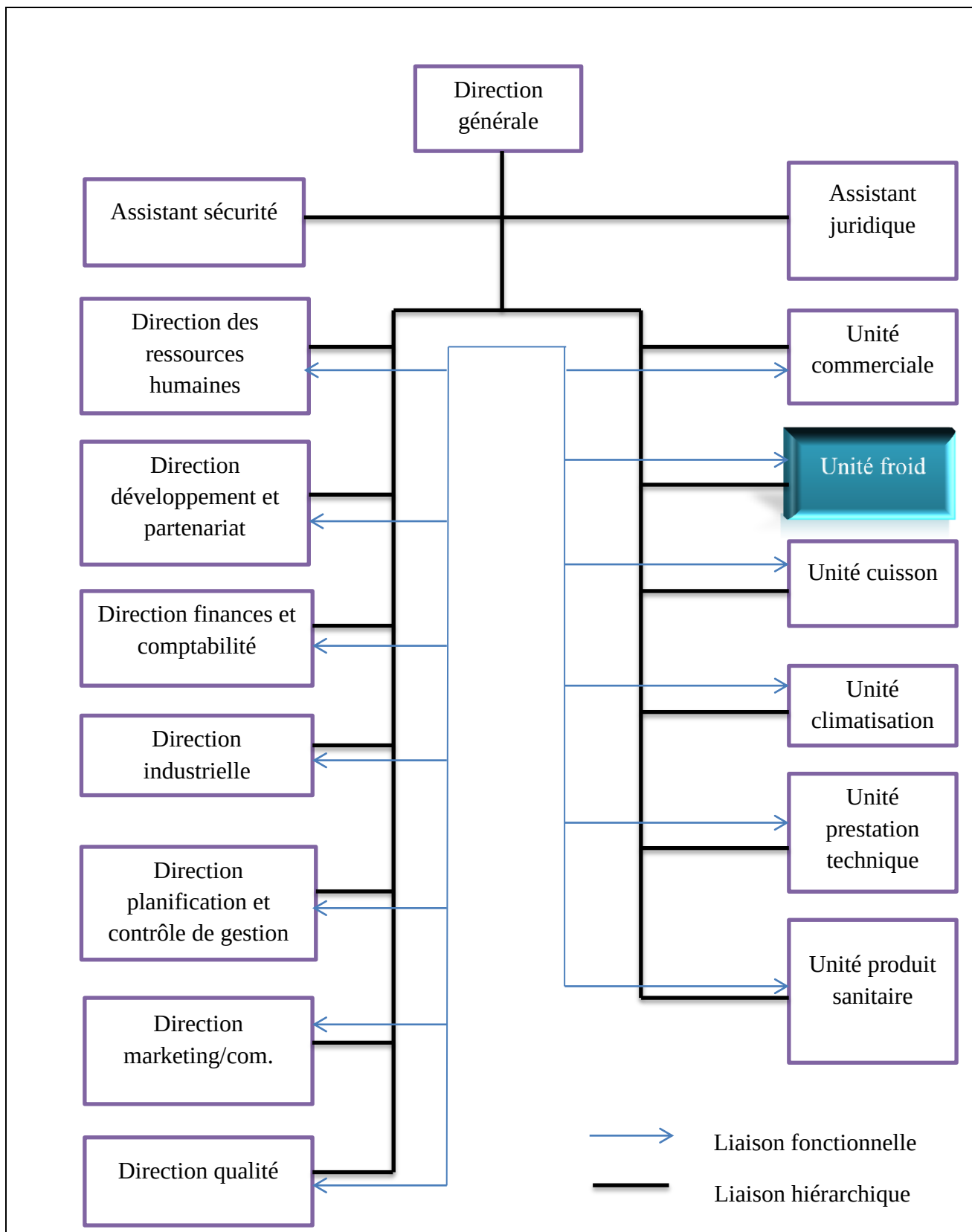


Figure 10 : Organigramme général de l'entreprise ENIEM.

ETUDE DE L'EXISTANT

« UNITE FROID »

Introduction

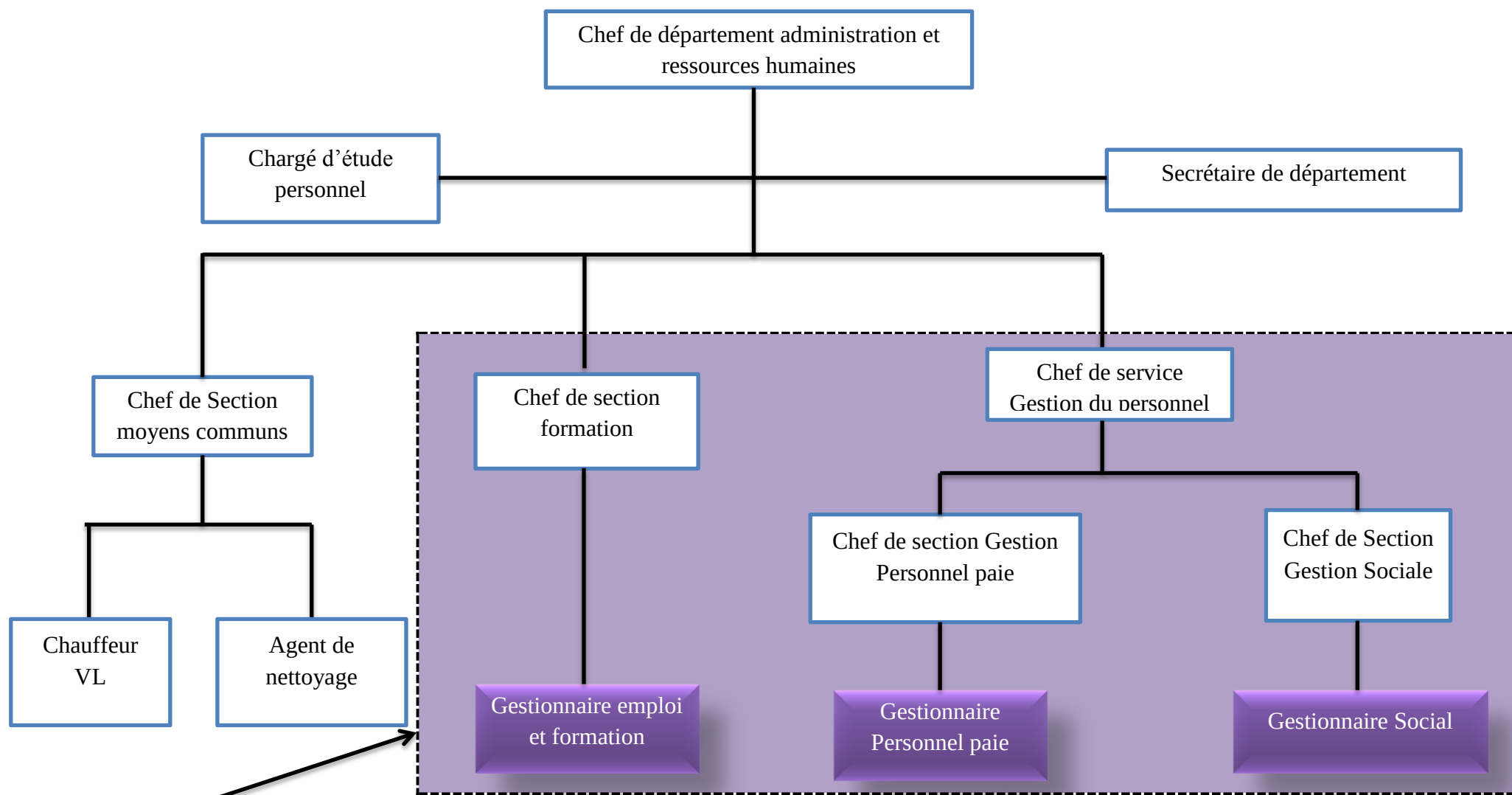
L'analyse de l'existant a pour objectif l'étude du système d'information existant, son diagnostic, la détermination des objectifs du nouveau système et l'ébauche de solution et aussi délimiter le champ d'étude. Pour ce faire, il faut étudier le système existant lui-même ainsi que l'environnement des lequel il baigne.

1. Délimitation et définition de domaine d'étude :

L'unité froid : cette unité est chargée de la production des appareils électroménagers dans le domaine de froid domestique et de veiller sur l'amélioration de la qualité de ces produits et la maintenance des outils de fabrication et c'est au sein de cette unité que notre domaine d'étude est situé.

2. Présentation des différentes composantes de l'unité froid :

- **Le département production :** responsable de la coordination des services de production.
- **Le département commercial :** responsable d'un département regroupant les activités : achat, gestion des stocks et sous-traitance d'une unité.
- **Le département technique :** a comme tâche de planifier et superviser les études de modification des modèles des produits fabriqués et la mise au point des procédés de fabrication correspondant.
- **Le département qualité :** supervise, organise, décide sur l'ensemble des opérations de contrôle qualitatif des matières et composants de fabrication achetés, des sous-ensembles fabriqués et des produits finis, et analyse les résultats enregistrés.
- **Le département finance et comptabilité :** chargé de la définition et la coordination de l'application de la politique financière fiscale et comptable.
- **Le département maintenance :** est responsable de la maintenance des machines, équipement et outillage en soutien de la production.
- **Le département laboratoire central :** s'occupe de l'assurance, la planification et l'organisation des activités du laboratoire central.
- **Le département sécurité industrielle :** est placé sous la responsabilité directe du directeur de l'unité, il s'occupe de l'animation, la coordination et le contrôle des activités liées à la prévention et l'intervention en matière d'hygiène et sécurité au sein de l'unité.
- **Le département administration et ressources humaines :** chargé de l'animation, la coordination, l'orientation et le contrôle des activités d'administration (emploi et formation) et la gestion des moyens communs de l'unité, et c'est là où se situe notre domaine d'étude.



Champ d'étude

Organigramme de département administration des ressources humaines (unité froid)

3. La liste des documents manipulés dans notre champ d'étude :

01	Certificat de cessation de paiement
02	Décision d'absolution de paiement
03	Décision d'affectation
04	Décision de confirmation
05	Décision de détachement
06	Décision de licenciement
07	Décision de mise en disponibilité
08	Décision de mise en retraite
09	Décision de mutation
10	Décision de promotion
11	Décision de radiation (démission)
12	Décision de réintégration
13	Décision de sanction
14	Fiche d'appréciation des candidats à la promotion
15	Fiche d'entrée en fonction
16	Visite médicale aptitude professionnelle
17	Décision de formation
18	Attestation de formation
19	Facture de formation
20	Convention de formation
21	Attestation de stage
22	Attestation d'apprentissage
23	Demande de stage pratique
24	Convention de stage
25	Certificat de travail
26	Fiche d'évaluation de période d'essai
27	Attestation de travail
28	Relevé des émoluments
29	Déclaration de reprise de travail
30	Feuille de soin médicale + ordonnance
31	Déclaration d'accident de travail
32	Certificat médical initial
33	Fiche familiale
34	Certificat de scolarité
35	Attestation de salaire
36	Fiche de paie
37	Demande de démission
38	Demande de mutation
39	Demande de mise en disponibilité
40	Proposition de sanction
41	Décision de recrutement
42	Demande d'absolution de sanction
43	Décision d'absolution de stage
44	Proposition de promotion
45	Proposition de dégradation
46	Décision de dégradation

47	Demande de réintégration
48	Décision de réintégration
49	Fiche de circuit
50	Déclaration de reprise de travail
51	Déclaration d'accident de travail
52	Déclaration de rechute d'accident de travail
53	Solde de tout compte
54	Déclaration de reconstitution de carrière
55	Déclaration de maladie professionnelle
56	Convention de recrutement
57	Note d'instruction
58	Lettre de démission
59	Lettre d'engagement
60	P.V d'installation
61	Contrat de travail
62	Fiche de présence
63	Demande de recrutement
64	Extrait de naissance
65	Titre de congé
66	Décision d'intérim
67	Contrat temporaire
68	Notification d'attribution d'une pension de retraite
69	Demande de congé
70	Proposition d'intérim
71	Décision d'infirmité
72	Convocation
73	Mise en demeure
74	Liste des permanences

4. Etude des postes

L'étude des postes de travail permet de détecter les relations qui existent entre ces différents postes, ainsi que les tâches exécutées et les documents manipulés par ces derniers.

A cet effet, une fiche d'analyse de poste est dressée pour chacun des postes de travail figurant dans la liste ci-dessous :

1. Chef de service gestion Personnel.
2. Gestionnaire personnel et paie.
3. Gestionnaire sociale.
4. Gestionnaire emploi et formation

Poste N°1 :

Fiche d'analyse du poste de travail N°1	
Désignation : chef de service gestion personnel.	
Nombre d'effectif : 1	
La responsabilité de poste : Supervise, coordonne, contrôle l'activité de gestion du personnel (Administration du personnel, emploi, gestion sociale et paie) (unité froid), participe à l'élaboration du plan de formation et le met en œuvre conformément aux procédures en vigueur (Unité Prestations Techniques et Unité Commercial).	
Les taches principales du poste • Supervise, coordonne et contrôle les activités du service en matière de paie, gestion du personnel, gestion sociale. • Veille à la bonne tenue des dossiers et du fichier du personnel. • Veille à l'application de la législation et du règlement intérieur en matière de discipline générale. • Met en application les procédures de Gestion des ressources humaines. • Suit le développement ou l'évolution des carrières des travailleurs tant sur le plan professionnel que social (formation, perfectionnement.....). • Tient à jour les statistiques du personnel. • Sensibilise son personnel sur la norme ISO 14001 et veille à l'application des exigences du SME et de la législation environnementale, unité prestations techniques et unité commerciale. • Participe à l'élaboration du plan de formation de l'unité conformément à la procédure en vigueur. • Met en œuvre le plan de formation une fois approuvé par la direction des ressources humaines conformément à la procédure en vigueur. • Sensibilise son personnel sur la norme ISO 14001 et veille à l'application des exigences du SME et de la législation.	

Les documents manipulés par le poste N°1 :

	désignation	Provenance
Documents entrants	Demande de congé	Gestionnaire Personnel et Paie
	Demande de démission	Gestionnaire Personnel et Paie
	Demande de rattachement	Gestionnaire Personnel et Paie
	Demande de promotion	Gestionnaire emploi et formation
	Proposition d'intérim	Gestionnaire emploi et formation
	Contrat temporaire	Gestionnaire emploi et formation
	Fiche d'entrée en fonction	Gestionnaire emploi et formation
	Demande de mutation	Gestionnaire emploi et formation
	Demande de mise en disponibilité	Gestionnaire emploi et formation
	Demande de réintégration	Gestionnaire emploi et formation
	Courier externe	Secrétaire
	Désignation	Destination
	Contrat temporaire	Gestionnaire Personnel et Paie
	Fiche d'entrée en fonction	Gestionnaire Personnel et Paie
	Demande de mutation	Gestionnaire emploi et formation
	Demande de mise en disponibilité	Gestionnaire emploi et formation

Documents sortants	Demande de réintégration	Gestionnaire emploi et formation
	Proposition de promotion	Gestionnaire emploi et formation
	Proposition d'intérim	Gestionnaire emploi et formation
	Proposition de sanction	Gestionnaire emploi et formation
	Demande de détachement	Gestionnaire Personnel et Paie
	Demande de démission	Gestionnaire Personnel et Paie
	Demande de congé	Gestionnaire Personnel et Paie

Tableau : liste des documents manipulés par le poste N°01**Poste N°2 :**

Fiche d'analyse de poste N°2		
Désignation : gestionnaire personnel et paie.		
Nombre d'effectif : 1		
Responsabilité de poste : assure la gestion administrative du personnel et paie conformément aux dispositions légales et procédures en vigueur.		
Les tâches principales spécifiques au poste N°2 :		
- Assure la constitution et la tenue des dossiers administratifs des travailleurs du recrutement jusqu'à leur départ. - Etablit à la demande des travailleurs les documents administratifs dont ils ont besoin (attestation de travail, relevé des émoluments, etc...). - Tient à jour le fichier du personnel dont il a la charge. - Etablit tout document relatif aux salaires (ATS, STC, etc.....). - Suit l'évolution des carrières des travailleurs. - Reçoit et traite les doléances et les réclamations des travailleurs portant sur les éléments de salaire. - Suit l'absentéisme des travailleurs et signale les abandons de poste. - Contrôle les fiches de présence et les cartons de pointage. - Suit les droits aux congés annuels (Registre légal).		
Les documents manipulés par le poste N°2		
Documents entrants	Désignation	provenance
	Demande de recrutement	DG
	Demande de réintégration	Employé
	Demande de mise en disponibilité	Employé
	VMAP confirmé par le médecin	Médecin
	Notification d'attribution d'une pension de retraite.	Caisse nationale des retraités
	Décision de mise en disponibilité	Secrétaire
	décision de réintégration	Secrétaire
	Décision de sanction	Secrétaire
	Décision de licenciement	Secrétaire
	Décision de recrutement	Secrétaire
	Décision d'intérim	Secrétaire
	Décision de promotion	Secrétaire
	Décision de mutation	Secrétaire
	Demande de mutation	Employé
	Proposition d'intérim	Structure
	Proposition de promotion	Structure

	Proposition de sanction	Structure
	Fiche d'entrée en fonction	Directeur de l'unité
	Contrat temporaire	Directeur de l'unité
	Décision de confirmation	Secrétaire
	Décision d'infirmité	Secrétaire
	Fiche d'évaluation période d'essai	Directeur de l'unité
	Dossier de retraite	Gestionnaire personnel et paie
Documents sortants	Désignation	Destination
	VMAP	Médecin
	Contrat temporaire	Chef de service gestion personnel Directeur de l'unité Caisse nationale des retraités
	Proposition de promotion	Chef de service gestion personnel
	Décision de promotion	Employé, secrétaire, DU
	Décision de mutation	Employé, secrétaire, DU
	Fiche d'évaluation période d'essai	Directeur de l'unité
	Notification d'attribution d'une pension de retraite.	Gestionnaire personnel et paie
	Décision d'intérim	Employé, secrétaire, DU
	Certificat de cessation de paiement et d'activité	Caisse nationale des retraités
	Décision de mise en disponibilité	Chef de service gestion personnel
	décision de réintégration	Caisse nationale des retraités
	Décision de licenciement	Employé, secrétaire, DU
	Fiche d'entrée en fonction	Chef de service gestion personnel Directeur de l'unité Caisse nationale des retraités
	Demande de mutation	Chef de service gestion personnel
	Demande de mise en disponibilité	Chef de service gestion personnel
	Demande de réintégration	Chef de service gestion personnel
	Proposition de sanction	Chef de service gestion personnel
	Proposition d'intérim	Chef de service gestion personnel
	Demande de recrutement	Chef de service gestion personnel
	Convocation	Candidat
	Décision de confirmation	Employé
	Décision d'infirmité	Employé
	Dossier de retraite	Caisse nationale des retraités

Tableau : liste des documents manipulés par le poste N°02

Poste N°3 :

Fiche d'analyse de poste N°3		
Désignation : Gestionnaire Social		
Nombre d'effectif : 1		
Définition sommaire : Gère les dossiers relatifs aux frais médicaux, allocations familiales, mutuelles et autres assurances sociales des assurés de l'Unité.		
Les taches principales spécifiques au poste N°3 :		
• Reçoit, vérifie, complété et établit tous les documents nécessaires aux dossiers relatifs aux allocations familiales, allocations sociales, arrêt de travail, mutuelles et autres assurances sociales. • Reçoit et traite les documents relatifs aux remboursements et les transmet à la gestion du personnel pour mise sur paie. • Assure toutes correspondances avec les organismes compétents en la matière sociale. (CNR, CNAS, CAAR). • Tient à jour les fichiers des adhérents aux différentes caisses de sécurité sociale. • Etablit mensuellement les états de rapprochement concernant les allocations familiales, les frais médicaux, etc.		
Documents manipulés par le poste N°3		
	Désignation	provenance
Les documents entrants	Feuilles de soin médical + ordonnances	Service intéressé
	Déclaration d'accident de travail	Service intéressé
	Déclaration de rechute d'accident de travail	Service intéressé
	Déclaration de maladie professionnelle	Médecin de travail
	Certificat de scolarité	Intéressé
	Fiche familial	Intéressé
	Désignation	Destination
Les documents sortants	Déclaration de reprise de travail	CNAS
	Déclaration de reconstitution de carrière	CNAS, CNR
	Bulletin d'adhésion à l'assurance groupe	SAA
	Liste des adhérents à l'assurance groupe	CNR
	Attestation de travail et de salaire	CNAS

Tableau : liste des documents manipulés par le poste N°03

Poste N°4 :

Fiche d'analyse de poste N°4		
Désignation : Gestionnaire emploi et formation.		
Nombre d'effectif : 1		
Définition sommaire : exécute tous les mouvements du personnel (recrutement, mutation, promotion et redéploiement).		
Les tâches principales spécifiques au poste N°4		
• Réalise et suit les opérations d'apprentissage conformément aux textes réglementaires. • Suit les stages pratiques qui se déroulent sur site. • Elabore le bilan de l'apprentissage et des stages pratiques. • Exécute les opérations de recrutement, de mutation, de promotion conformément aux procédures en vigueur à l'entreprise. • Saisit l'ensemble des mouvements survenus durant le mois sur les registres prévus à cet effet. • Contribue à la mise en œuvre du SME.		
Documents manipulés par le poste N°4		
	Désignation	provenance
Les documents entrants	Demande de mise en disponibilité	Intéressé
	Demande de mutation	Service intéressé
	Demande d'absolution de sanction	Service intéressé
	Demande de réintégration	Service intéressé
	Proposition de dégradation	Service intéressé
	Proposition de promotion	Service intéressé
	Désignation	Destination
Les documents sortants	Décision d'absolution de sanction	Service intéressé Intéressé Dossier
	Décision de mutation	Service intéressé Gestionnaire personnel et paie Intéressé Dossier
	Décision de promotion	Intéressé Service
	Décision d'absolution de sanction	Gestionnaire personnel et paie Service intéressé Dossier
	Fiche d'entrée en fonction	Dossier
	Décision de confirmation au poste	Gestionnaire personnel et paie. Intéressé. Service intéressé. Dossier.
	Fiche d'appréciation des candidats à la promotion	Gestionnaire personnel et paie.

Tableau : liste des documents manipulés par le poste N°04.

5. Etude des documents :

- ✓ **Introduction :** cette étude nous permet de faire l'inventaire sur tous les documents utilisés dans notre champ d'étude.
- ✓ **Liste des documents :**
- ✓ **Analyse des documents :**

1. Certificat de cessation de paiement :

Fiche d'analyse du document N° 01 :			
1- Caractéristiques du document :			
-Désignation : Certificat de cessation de paiement et d'activité. -Code : C.C.P.A -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : employé, dossier, CNAS et CNR. -Rôle : Justifier la cessation de paiement de l'employé. -Nature du document: Externe -Nombre d'exemplaires : 04			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	JJ/MM/AAAA.
-Prénom de l'employé.	A	20	
-Date de naissance de l'employé.	D	10	
-Lieu de naissance de l'employé.	A	20	
-Fonction.	A	30	JJ/MM/AAAA.
-Date de cessation de paiement.	D	10	
-Motif de cessation de paiement.	A	30	
-signature du directeur.	/	/	

Tableau : Fiche d'étude du document N°01

2. Décision de confirmation :

Fiche d'analyse du document N° 02 :			
1- Caractéristiques du document :			
-Désignation : Décision de confirmation. -Code : DC -Poste d'origine : Secrétaire. -Poste destinataire : GEF, Employé. -Rôle : Officialise la confirmation. -Nature du document: Externe -Nombre d'exemplaires : 02			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Code poste.	A	04	
-Catégorie.	AN	03	
-Echelon.	N	02	
-Salaire de base.	N	05	En dinars Algérien
-Signature du directeur de l'unité	/	/	

Tableau : Fiche d'étude du document N°02

3. Décision de détachement :

Fiche d'analyse du document N° 03 :			
1- Caractéristiques du document :			
-Désignation : Décision de détachement. -Code : D.D -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : Employé, dossier. -Rôle : Pour officialiser le détachement de l'employé. -Nature du document: Externe. -Nombre d'exemplaires : 02.			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	

-Fonction.	A	30	JJ/MM/AAAA.
-Objet de décision.	A	20	
-Date de détachement.	D	10	
-Durée de détachement.	AN	08	
-Motif du détachement.	A	30	
-Visa du directeur.	/	/	

Tableau : Fiche d'étude du document N

4. Décision de licenciement :

Fiche d'analyse du document N° 04 :			
1- Caractéristiques du document :			
-Désignation : Décision de licenciement. -Code : D.L -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : L'employé, dossier. -Rôle : Officialisation du licenciement de l'employé. -Nature du document: Interne -Nombre d'exemplaires : 02			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Fonction.	A	30	
-Date de licenciement.	D	10	JJ/MM/AAAA.
-Motif du licenciement.	A	30	
-Visa du directeur.	/	/	

Tableau : Fiche d'étude du document N°04

5. Décision de mise en disponibilité :

Fiche d'analyse du document N° 05 :			
1- Caractéristiques du document :			
-Désignation : Décision de mise en disponibilité. -Code : DMED -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : Employé, dossier. -Rôle : officialiser la mise en disponibilité de l'employé. -Nature du document: interne -Nombre d'exemplaires : 02			
3- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Lieu de l'entreprise.	AN	10	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Fonction.	A	30	
-Durée de la mise en disponibilité.	AN	08	
-Date de la décision	D	10	JJ/MM/AAAA
-Visa du directeur.	/	/	

Tableau : Fiche d'étude du document N°05.

6. Décision de mutation :

Fiche d'analyse du document N° 06 :			
1- Caractéristiques du document :			
-Désignation : Décision de mutation. -Code : DM. -Poste d'origine : Chef de service gestion personnel. -Poste destinataire : Employé, dossier. -Rôle : Officialisé la mutation de l'employé. -Nature du document: Externe. -Nombre d'exemplaires : 02.			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom et prénom de l'employé.	A	40	
-Code poste.	A	04	

-Catégorie.	N	02	JJ/MM/AAAA.
-Nombre d'échelon.	N	02	
-Salaire de base de l'employé.	N	05	
-Date de la décision.	D	10	
-Visa du directeur.	/	/	

Tableau : Fiche d'étude du document N°06.

7. Décision de promotion :

Fiche d'analyse du document N° 07 :			
1- Caractéristiques du document :			
-Désignation : Décision de promotion. -Code : DA -Poste d'origine : Chef de service gestion personnel. -Poste destinataire : intéressé, dossier. -Rôle : Officialisé la promotion de l'employé. -Nature du document: Externe. -Nombre d'exemplaires : 02			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom et prénom de l'employé.	A	40	[1.....20] En dinars Algérien JJ/MM/AAAA.
-Code poste.	A	04	
-Catégorie.	AN	03	
-Nombre d'échelon.	N	02	
-Salaire de base de l'employé.	N	05	
-Date de promotion.	D	10	JJ/MM/AAAA.
-Visa du directeur.	/	/	

Tableau : Fiche d'étude du document N°7

8. Décision de démission :

Fiche d'analyse du document N° 08 :			
1- Caractéristiques du document :			
-Désignation : Décision de démission. -Code : D.D. -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : L'employé, dossier. -Rôle : Officialisation de la démission -Nature du document: Interne -Nombre d'exemplaires : 02			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Fonction.	A	30	
-Visa du directeur.	/	/	

Tableau : Fiche d'étude du document N°08.**9. Décision de réintégration :**

Fiche d'analyse du document N° 09 :			
1- Caractéristiques du document :			
-Désignation : Décision de réintégration. -Code : DR. -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : L'employé, dossier. -Rôle : Officialiser la réintégration de l'employé. -Nature du document: Interne -Nombre d'exemplaires : 02			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Code poste.	A	04	JJ/MM/AAAA.
-Date de réintégration.	D	10	
-Visa du chef de service.	/	/	

Tableau : Fiche d'étude du document N°09.

10. Décision de sanction :

Fiche d'analyse du document N° 10 :			
1- Caractéristiques du document :			
-Désignation : Décision de sanction. -Code : D.S. -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : L'employé, dossier. -Rôle : Officialiser la sanction de l'employé. -Nature du document: Interne. -Nombre d'exemplaires : 02			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	JJ/MM/AAAA.
-Prénom de l'employé.	A	20	
-Fonction.	A	30	
-Date de sanction.	D	10	
-Motif de la sanction.	A	30	
-Durée de la sanction.	AN	08	
-Visa du directeur.	/	/	

Tableau : Fiche d'étude du document N°10**11. Fiche d'entrée en fonction :**

Fiche d'analyse du document N° 11 :			
1- Caractéristiques du document :			
-Désignation : Fiche d'entrée en fonction. -Code : FEF -Poste d'origine : GEF -Poste destinataire : Dossier, employé, CNR, CNAS -Rôle : Confirme l'installation de l'employé. -Nature du document: Externe -Nombre d'exemplaires : 04			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	JJ/MM/AAAA.
-Date de naissance de l'employé.	D	10	

-Lieu de naissance de l'employé.	A	25	Marié(e), divorcé(e), célibataire, veuf (veuve). JJ/MM/AAAA.
-Situation familiale.	A	01	
-Adresse de l'employé.	AN	40	
-Date d'entrée en fonction.	D	10	
-Code poste.	A	04	
-Catégorie.	AN	03	
-Nombre d'échelon	N	02	
-Appréciation finale.	A	11	
-Visa de la hiérarchie.	/	/	
-Signature de l'intéressé	/	/	
-Visa pour le département administratif et ressources humaines.	/	/	
-Visa pour le service de sécurité.	/	/	
-Avis du médecin de l'unité.	/	/	

Tableau : Fiche d'étude du document N°1

12. Visite médicale d'aptitude professionnelle :

Fiche d'analyse du document N° 12 :			
1- Caractéristiques du document :			
-Désignation : Visite médicale d'aptitude professionnelle. -Code : V.M.A.P -Poste d'origine : Centre médical de l'entreprise. -Poste destinataire : L'intéressé, dossier. -Rôle : Confirmation de la bonne santé de l'employé. -Nature du document: Externe. -Nombre d'exemplaires : 01			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'intéressé.	A	15	JJ/MM/AAAA.
-Prénom de l'intéressé.	A	20	
-Date de naissance de l'intéressé.	D	10	
-Lieu de naissance de l'intéressé.	A	20	JJ/MM/AAAA.
-Date d'entrée en fonction.	D	10	
-Fonction.	A	30	
-Avis du médecin de l'unité.	A	15	

Tableau : Fiche d'étude du document N°12

13. Certificat de travail :

Fiche d'analyse du document N° 13 :			
1- Caractéristiques du document :			
-Désignation : Certificat de travail. -Code : C.T. -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : L'employé, dossier. -Rôle : Certifier que l'employé a réellement exercé dans l'entreprise pour une durée limitée. -Nature du document: Externe -Nombre d'exemplaires : 02			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Date de naissance de l'employé.	D	10	
-Fonction.	A	30	
-Date d'entrée en fonction.	D	10	
-Date de fin du contrat.	D	10	
-Signature du directeur de l'unité.	/	/	JJ/MM/AAAA.

Tableau : Fiche d'étude du document N°13.

14. Fiche d'évaluation de période d'essai :

Fiche d'analyse du document N° 14 :			
1-Caractéristiques du document :			
-Désignation : Fiche d'évaluation période d'essai. -Code : F.E.P.E -Poste d'origine : Gestionnaire emploi et formation. -Poste destinataire : Dossier. -Rôle : Evaluation de l'employé après recrutement. -Nature du document: interne. -Nombre d'exemplaires : 01.			
2-Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Nom de l'employé.	A	20	
-Prénom de l'employé	A	20	
-Fonction.	A	30	
-Date de promotion.	D	10	

-Durée de la période d'essai.	AN	05	
-Catégorie.	AN	03	
-Nombre d'échelon.	N	02	
-Date d'expiration de la période d'essai.	D	10	
-Appréciation finale.	/	/	

Tableau : Fiche d'étude du document N°14.

15. Attestation de travail :

Fiche d'analyse du document N° 15 :			
1- Caractéristiques du document :			
-Désignation: attestation de travail. -code : A.T -poste d'origine : Gestionnaire personnel et paie -Poste destinataire : L'employé -Rôle : justifier l'exercice d'un fonctionnaire au sein de l'unité. -Nature du document : Externe -Nombre d'exemplaires : 01			
2-Description du document :			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Adresse de l'entreprise.	AN	40	JJ/MM/AAAA.
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Date de naissance de l'employé.	D	10	JJ/MM/AAAA.
-Lieu de naissance de l'employé.	A	20	JJ/MM/AAAA.
-Date d'entrée en fonction.	D	10	
-Fonction.	A	30	
-Signature du responsable du personnel.	/	/	

Tableau : Fiche d'étude du document N°15.

16. Demande de démission :

Fiche d'analyse du document N° 16 :			
1- Caractéristiques du document :			
-Désignation : Demande de démission. -Code : D.D. -Poste d'origine : Employé. -Poste destinataire : GPP. -Rôle : Demander une démission. -Nature du document: Externe. -Nombre d'exemplaires : 01.			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Objet de la demande.	A	20	
-Fonction.	A	30	
-Motif de la demande.	A	40	
-Signature de l'employé.	/	/	

Tableau : Fiche d'étude du document N°16**17. Demande de mutation :**

Fiche d'analyse du document N° 17 :			
1- Caractéristiques du document :			
-Désignation : Demande de mutation. -Code : D.M -Poste d'origine : Employé -Poste destinataire : GEF -Rôle : Demander une mutation. -Nature du document: Externe -Nombre d'exemplaires : 01			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
Prénom de l'employé.	A	20	
-Objet de la demande.	A	20	

-Code poste.	A	04	
-Signature de l'employé.	/	/	

Tableau : Fiche d'étude du document N° 17.

18. Demande de mise en disponibilité :

Fiche d'analyse du document N° 18 :			
1- Caractéristiques du document :			
-Désignation : Demande de mise en disponibilité. -Code : D.M.E.D -Poste d'origine : Employé -Poste destinataire : GEF -Rôle : Demander une mise en disponibilité. -Nature du document: Externe. -Nombre d'exemplaires : 01.			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Objet de la demande.	A	20	
-Fonction.	A	30	
-Motif de la demande.	A	20	
-Signature de l'employé.	/	/	

Tableau : Fiche d'étude du document N° 18.

19. Proposition de sanction :

Fiche d'analyse du document N° 19 :			
1- Caractéristiques du document :			
-Désignation : Proposition de sanction. -Code : PS -Poste d'origine : La structure. -Poste destinataire : GPP. -Rôle : Proposer une sanction. -Nature du document: Externe. -Nombre d'exemplaires : 01.			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	

-Date d'effet du document.	D	10	JJ/MM/AAAA
-Nom et prénom de l'employé.	A	40	
-Fonction.	A	25	
-Motif de la sanction.	A	30	
-Visa du directeur.	/	/	
-Visa et avis du chef de service.	/	/	
-Visa et avis du chef de département.	/	/	

Tableau : Fiche d'étude du document N° 19

20. Décision de recrutement :

Fiche d'analyse du document N° 20 :			
1- Caractéristiques du document :			
-Désignation : Décision de recrutement. -Code : D.R -Poste d'origine : Gestionnaire personnel et paie. -Poste destinataire : Employé, dossier. -Rôle : Officialisation de recrutement d'un employé. -Nature du document: Externe. -Nombre d'exemplaires : 02.			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	JJ/MM/AAAA.
-Prénom de l'employé.	A	20	
-Date d'entrée en fonction.	D	10	
-Code poste.	A	04	
-Signature de l'intéressé.	/	/	
-Signature du directeur	/	/	

Tableau : Fiche d'étude du document N° 20.

21. Proposition de promotion :

Fiche d'analyse du document N° 21 :			
1- Caractéristiques du document :			
-Désignation : Proposition de promotion. -Code : PP -Poste d'origine : Gestionnaire emploi et formation. -Poste destinataire : Service de l'employé.			

-Rôle : Promouvoir ou muter un employé a un poste déterminé. -Nature du document: Interne. -Nombre d'exemplaires : 01			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA. JJ/MM/AAAA. En dinars Algérien En dinars Algérien JJ/MM/AAAA. JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Matricule de l'employé.	AN	06	
-Fonction.	A	30	
-Date d'entrée en fonction.	D	10	
-Date de promotion.	D	10	
-Fonction.	A	30	
-Code poste.	N	03	
-Catégorie.	AN	03	
-Nombre d'échelons.	N	02	
-Salaire de base.	N	05	
-Département de la situation proposé.	A	30	
-Service de la situation proposé.	A	25	
-Fonction de la situation proposée.	A	30	
-Code poste.	N	03	
-Catégorie de la situation proposée.	AN	03	
-Salaire de base de la situation proposé.	N	05	
-Nombre d'échelons de la situation proposée.	N	02	
-Avis clairement formulé de la hiérarchie.	A	30	JJ/MM/AAAA. JJ/MM/AAAA.
-Visa du service actuel.	/	/	
-Date du visa.	D	10	
-Avis clairement formulé.	A	30	
-Visa du service d'accueil.	/	/	
-Date du visa.	D	10	JJ/MM/AAAA.
-Avis du DAGRH.	A	30	

Tableau : Fiche d'étude du document N° 21.

22. Demande de réintégration :

Fiche d'analyse du document N° 22 :
1- Caractéristiques du document :
-Désignation : Demande de réintégration. -Code : D.R. -Poste d'origine : Employé. -Poste destinataire : GEF. -Rôle : Demander une réintégration. -Nature du document: Externe. -Nombre d'exemplaires : 01

2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	JJ/MM/AAAA.
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Objet de la demande.	A	20	
-Motif de la demande.	A	40	
-Fonction.	A	30	
-Date de détachement.	D	10	JJ/MM/AAAA.
-Signature de l'employé.	/	/	

Tableau : Fiche d'étude du document N° 22.

23. Fiche de circuit :

Fiche d'analyse du document N° 23 :			
1- Caractéristiques du document :			
-Désignation : Fiche de circuit. -Code : F.C. -Poste d'origine : gestionnaire personnel et paie. -Poste destinataire : Dossier, employé -Rôle : Connaitre l'état des dettes de l'employé. -Nature du document: Externe. -Nombre d'exemplaires : 01			
2- Description du document			
Rubriques	Type	Taille	Observation

-Libellé de l'entreprise.	A	56	
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	
-Libellé de service.	A	28	
-Date d'effet du document.	D	10	JJ/MM/AAAA.
-Nom de l'employé.	A	20	
-Prénom de l'employé	A	20	
-Fonction.	A	30	
-Date du départ.	D	10	JJ/MM/AAAA.
-Motif du départ.	A	30	
-Date d'observation.	D	10	JJ/MM/AAAA.
-Visa d'observation.	/	30	
-Observation du service comptabilité générale.	A	30	
-Date d'observation.	D	10	JJ/MM/AAAA.
-Visa d'observation.	/	/	
-Observation du service œuvre sociale.	A	30	
-Date d'observation.	D	10	JJ/MM/AAAA.
-Visa d'observation.	/	/	
-Observation du service documentation.	A	30	
-Date d'observation.	D	10	JJ/MM/AAAA.
-Visa d'observation.	/	/	
-Observation du service moyens généraux.	A	30	
-Date d'observation.	D	10	JJ/MM/AAAA.
-Visa d'observation.	/	/	
-Observation du service gestion du personnel.	A	30	
-Date d'observation.	D	10	JJ/MM/AAAA.
-Visa d'observation.	/	/	
-Observation du service finance.	A	30	
-Date d'observation.	D	10	JJ/MM/AAAA.
-Visa d'observation.	/	/	
-Signature du chef de service gestion personnel.	/	/	

Tableau : Fiche d'étude du document N° 23

24. Titre de congé :

Fiche d'analyse du document N° 24 :			
1- Caractéristiques du document :			
- Désignation : Titre de congé.			
- Code : TC.			
- Poste d'origine : Gestionnaire personnel et paie			
- Poste destinataire : L'employé, dossier.			
- Rôle : Confirmation du congé demandé par l'employé.			
- Nature du document : Externe.			
- Nombre d'exemplaires : 02.			
2- Description du document :			
Rubriques	Type	Taille	Observation
-Libellé de l'entreprise.	A	56	
-Libellé de l'unité.	A	17	
-Libellé de département.	A	58	

-Libellé de service.	A	28	JJ/MM/AAAA.
-Matricule employé.	AN	06	
-Nom de l'employé.	A	20	
-Prénom de l'employé.	A	20	
-Code poste.	N	03	
-Fonction.	A	30	
-Nature du congé.	A	20	
-Date début congé.	AN	08	
-Date fin congé.	D	10	
-Adresse durant le congé.	AN	40	
-Signature de directeur.	/	/	

Tableau : Fiche d'étude du document N° 24

25. Notification d'attribution d'une pension de retraite :

Fiche d'analyse du document N° 25 :			
1- Caractéristiques du document :			
-Désignation : Notification d'attribution d'une pension de retraite. -Code : NADPR -Poste d'origine : Gestionnaire sociale. -Poste destinataire : C.N.R -Rôle : C'est pour faire une demande de retraite. -Nature du document: Externe. -Nombre d'exemplaires : 01			
2- Description du document			
Rubriques	Type	Taille	Observation
-Libellé de la caisse.	A	30	JJ/MM/AAAA.
-Agence de la wilaya.	A	20	
-Numéro du dossier.	AN	04	
-Date d'enregistrement.	D	10	
-Signature du responsable de l'agence.	/	/	
-Nom de l'assuré(e).	A	20	JJ/MM/AAAA.
-Prénom de l'assuré(e).	A	20	
-Nom du père de l'assuré(e).	A	20	
-Prénom du père de l'assuré(e).	A	20	
-Nom de la mère de l'assuré(e).	A	20	
-Prénom de la mère de l'assuré(e).	A	20	
-Date de naissance de l'assuré(e).	D	10	
-Lieu de naissance de l'assuré(e).	AN	20	
-Wilaya.	AN	15	
-Sexe.	A	01	
-Nationalité.	A	15	M: masculine, F: féminin.
-adresse de l'assuré(e).	AN	40	C: célibataire, V : veuf, M : marié, D : divorcé.
-Situation familiale.	A	01	
-Code postal.	N	05	
-Nom du conjoint.	A	20	
-Prénom du conjoint.	A	20	

-Nom du père du conjoint.	A	20	JJ/MM/AAAA. M : masculin, F : féminin.
-Prénom du père du conjoint.	A	20	
-Nom de la mère du conjoint.	A	20	
-Prénom de la mère du conjoint.	A	20	
-Date de naissance du conjoint.	D	10	
-Sexe.	A	01	
-Nationalité.	A	15	
-Fonction.	A	30	
-Ressource.	A	20	
-Numéro de la qualification.	N	04	
-Montant annuel de la pension de retraite.	AN	08	JJ/MM/AAAA.
-Organisme payeur de la pension de retraite.	A	20	
- Nom de la personne prise en charge.	A	20	
-Prénom de la personne prise en charge.	A	20	
-Date de naissance de la personne prise en charge.	D	10	
-Lieu de naissance de la personne prise en charge.	AN	40	
-Sexe.	A	01	
-Nationalité.	A	15	
-Fonction.	A	30	
-Ressource.	A	20	
-Nom de l'enfant à charge.	A	20	JJ/MM/AAAA.
-Prénom de l'enfant à charge.	A	20	
-Date de naissance de l'enfant à charge.	D	10	
-Lieu de naissance de l'enfant à charge.	AN	20	
-Situation familiale de l'enfant à charge.	A	01	
-Situation scolaire.	A	10	
-Profession de l'assuré en Algérie.	A	30	
-Période de travail.	AN	27	
-Nom de l'employeur.	A	20	
-Prénom de l'employeur.	A	20	
-Adresse de l'employeur.	AN	40	JJ/MM/AAAA.
-Caisse d'affiliation.	A	10	
-Numéro de sécurité sociale.	N	12	
-Profession de l'assuré à l'étranger.	A	30	
-Nom du pays.	A	20	
-Nom de l'employeur.	A	20	
-Prénom de l'employeur.	A	20	
-Adresse de l'employeur.	AN	40	
-Période de travail.	AN	27	
-Caisse d'affiliation.	A	10	
-Numéro de sécurité sociale.	N	12	JJ/MM/AAAA.
-Activité no salariale.	A	20	
-Activité exercé	A	20	
-Période.	AN	27	
-Profession	A	30	
-Adresse.	AN	40	JJ/MM/AAAA.
-Date d'effet du document.	D	10	
-Lieu d'effet du document	AN	10	

-Signature de l'assuré.	/	/	
-------------------------	---	---	--

Tableau : Fiche d'étude du document N° 25**✓ Etude des fichiers et des registres :**

Introduction : Cette étude consiste à analyser les fichiers et les registres ayant une relation avec le personnel et son suivi.

Liste de fichiers utilisés :

- 1- Fichier du personnel.

Liste des registres utilisés :

- 1- Registre mouvement du personnel.
- 2- Registre des congés.
- 3- Registre courrier arrivé externe.
- 4- Registre courrier départ externe.
- 5- Registre courrier arrivé interne.
- 6- Registre courrier départ interne.

Analyse des fichiers :**1. Fichier du personnel :**

Fiche d'analyse du fichier N° 01 :			
1- Caractéristique du document :			
Désignation : Fichier du personnel. Code : F.P Support : Chemise Localisation : Armoire métallique Type d'opération réalisée sur ce fichier : Enregistrement, consultation, modification. Poste qui utilise ce fichier : GPP, GEF			
2- Description du document :			
Rubrique	Type	Taille	Observation
-Libellé de l'entreprise.	AN	20	JJ/MM/AAAA. C: célibataire, M: marié, V: veuf, D: divorcé. F : féminin, M : masculin.
-Matricule employé.	N	06	
-Nom.	A	20	
-Prénom.	A	20	
-Date de naissance.	D	10	
-Adresse de l'employé.	AN	40	
-Situation familiale.	A	01	
-Fonction.	A	30	
-Sexe.	A	01	

-Date d'entrée en fonction.	D	10	JJ/MM/AAAA.
-Nombre d'échelon.	N	02	
-Catégorie.	AN	03	
-Code de la paie.	N	02	
-Code d'agence.	N	03	
-Numéro personnel.	N	02	
-Numéro de sécurité sociale.	AN	15	
-Numéro de compte.	AN	12	
-Adresse du compte.	AN	16	

Tableau : Fiche d'étude du fichier N° 01

Analyse des registres :

1. Registre mouvement du personnel :

Fiche d'analyse du registre N° 01			
1- Caractéristiques du document :			
Désignation : Registre mouvement du personnel. Code : R.M.P Support : Registre. Localisation : Armoire métallique. Type d'opération : Enregistrement et consultation Poste qui utilise ce registre : C.S.G.P, G.E.F			
2- Description du document			
Rubriques	Type	Taille	Observation
-Nom	A	15	M: masculin, F: féminin JJ/MM/AAAA.
-Prénom	A	15	
-Sexe	A	01	
-Date de naissance.	D	10	
-Lieu de naissance.	AN	20	
-Adresse.	AN	40	JJ/MM/AAAA. JJ/MM/AAAA.
-Code poste	A	04	
-Date d'entrée en fonction	D	10	
-Date de cessation de la relation de travail	D	10	
-Motif de cessation	A	30	
-Numéro de sécurité sociale	AN	15	Permanent, apprentis. EX : Handicapé
-Nature de la relation de travail	A	10	
-Observation	AN	/	

Tableau : Fiche d'étude du registre N° 01.

PRESENTATION DE PROJET

1. Présentation de notre travail

➤ Problématique

Le suivi de la carrière d'un employé, de la date de son recrutement jusqu'à la date de cessation de travail, est une tâche très compliquée et difficile vu le volume de dossier à traiter

➤ Objectif de notre travail

L'objectif de notre travail consiste à développer une application n-tiers pour faciliter la tâche de suivi de la carrière d'un employé.

2. Grand choix technique

- Méthode de développement 2tup
- Langage de modélisation UML
- SGBD postgresSQL
- Environnement de développement IDE Netbeans 6.0
- Langage de programmation Java
- Architecture logicielle repartie 3-tiers
- Plate-forme J2EE.

3. Identification des acteurs

Administrateur : personne chargée de la gestion les comptes utilisateurs.

Gestionnaire personnel et paie : la personne chargée de la gestion administrative personnel et paie conformément à la réglementation et procédures en vigueur. .

Gestionnaire social : personne chargé de gérer les dossiers relatifs aux allocations familiales, mutuel et autres assurances sociales des assurés de l'unité.

Chargé de l'emploi et formation : personne chargée de l'exécution des mouvements du personnel (recrutement, mutation, promotion et redéploiement).de l'élaboration et de la mise en œuvre du plan de formation de l'unité et ce conformément à la procédure en vigueur.

4. Diagramme de contexte [MUL 97] :

Le diagramme de contexte est un modèle conceptuel qui permet d'avoir une vision globale des interactions entre le système et les liens avec l'environnement extérieur.il permet aussi de délimiter le champ d'étude.

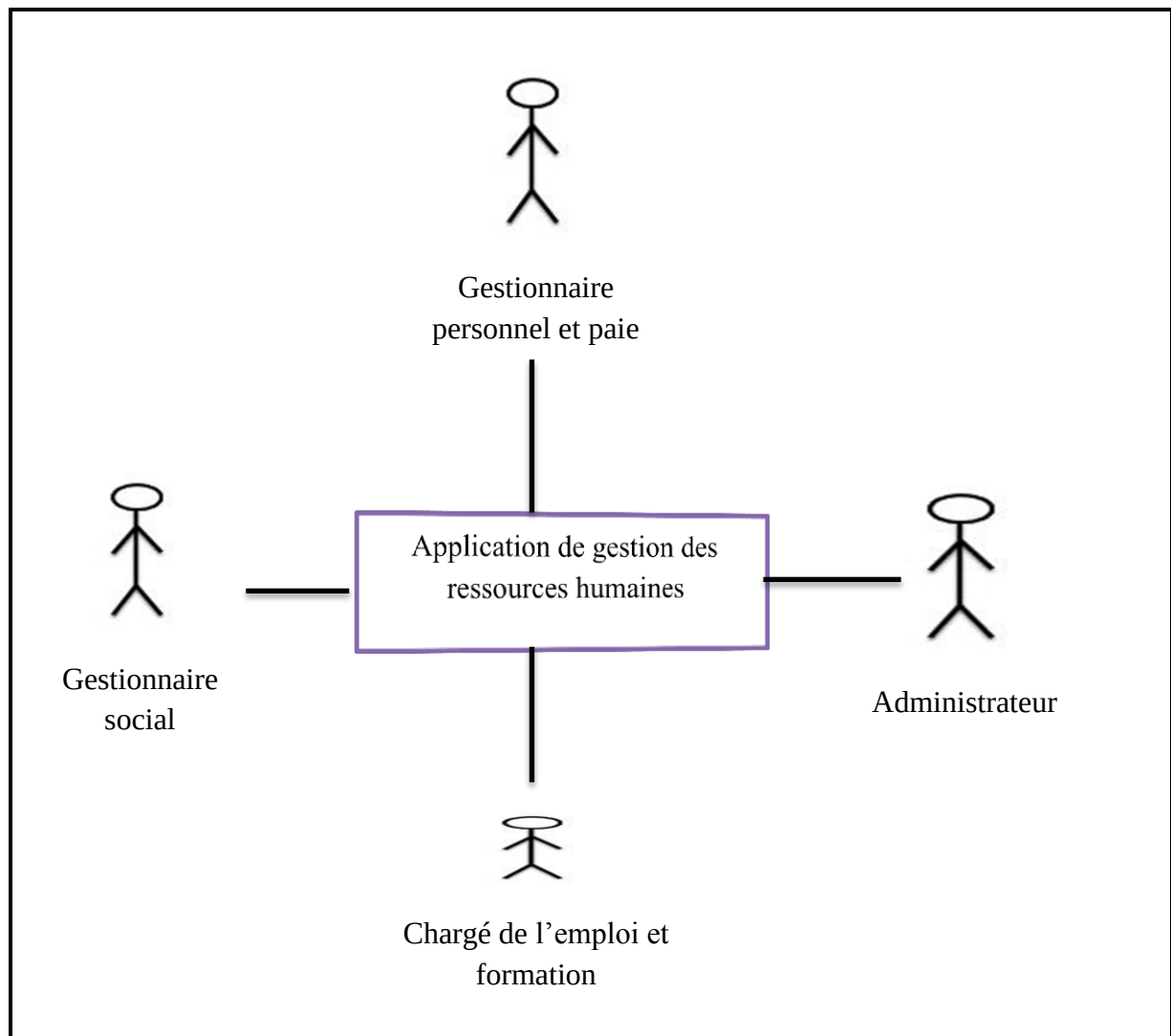


Diagramme de contexte relatif à notre l'application de gestion des ressources humaine.

CAPTURE DES BESOINS FONCTIONNELS

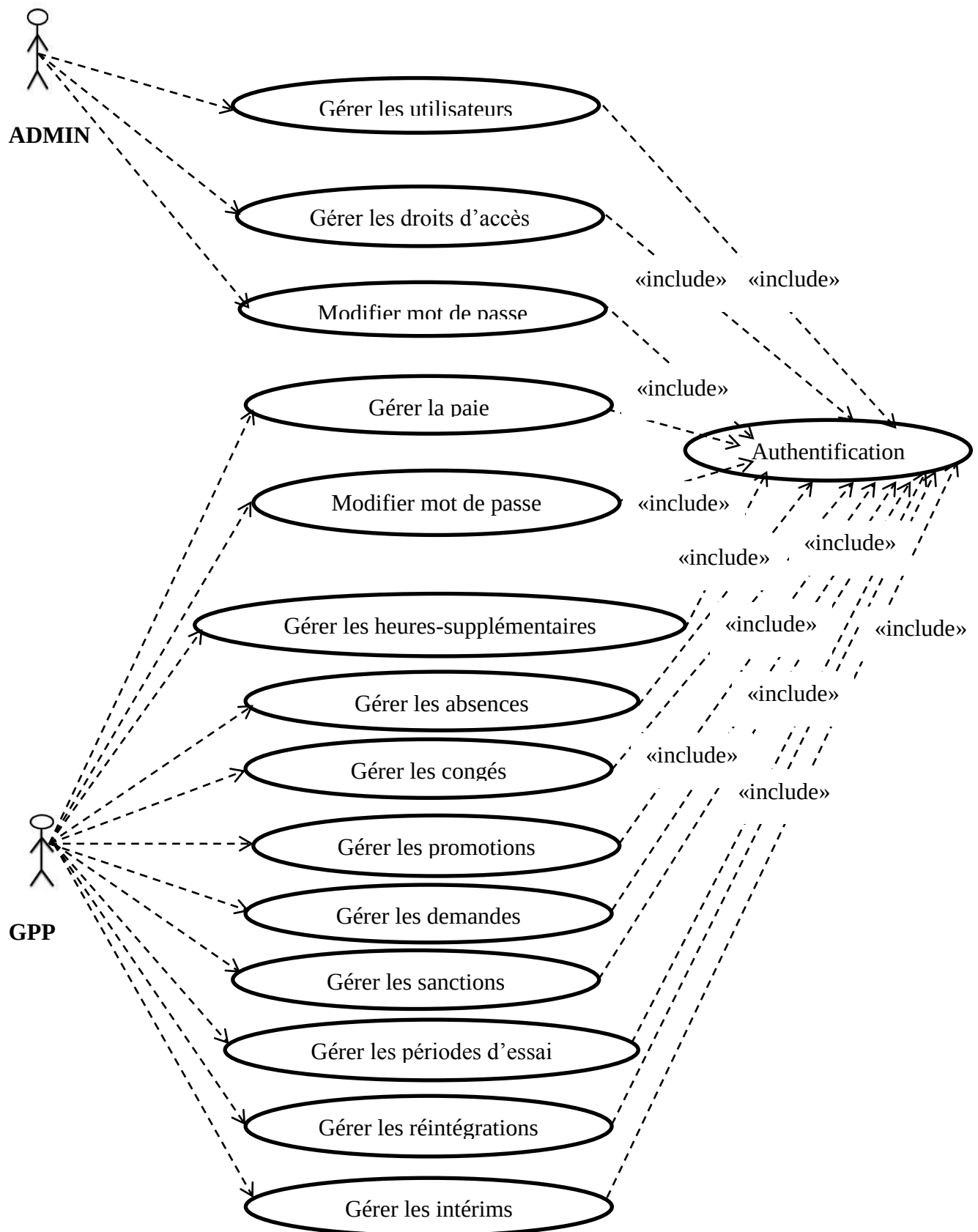
Introduction :

Cette phase représente un point de vue « fonctionnel » de l'architecture système. Par le biais des cas d'utilisation, nous serons en contact permanent avec les acteurs du système en vue de définir les limites de celui-ci, et ainsi éviter de trop s'éloigner des besoins réels de l'utilisateur final.

1. Identification des cas d'utilisation :**1.1.Cas d'utilisation :**

Cas d'utilisation	Acteur
Authentification	Administrateur
Gérer les utilisateurs	Administrateur
Gérer les droits d'accès	Administrateur
Gérer les employés	Gestionnaire Formation et Emploi
Gérer les intérim	Gestionnaire Personnel et paie
Gérer les réintégrations	Gestionnaire Personnel et paie
Gérer les contrats	Gestionnaire Formation et Emploi
Gérer période d'essai	Gestionnaire Personnel et paie
Gérer les sanctions	Gestionnaire Personnel et paie
Gérer les décisions	Gestionnaire Formation et Emploi
Gérer les demandes	Gestionnaire Personnel et paie
Gérer les promotions	Gestionnaire Personnel et paie
Gérer les congés	Gestionnaire Personnel et paie
Gérer les absences	Gestionnaire Personnel et paie
Gérer les heures supplémentaires	Gestionnaire Personnel et paie
Gérer la paie	Gestionnaire Personnel et paie
Gérer les départs	Gestionnaire Social
Gérer les arrêts	Gestionnaire Social
Gérer les conjoints	Gestionnaire Social
Gérer les enfants	Gestionnaire Social
Gérer les formations	Gestionnaire Formation et Emploi
Gérer les stagiaires	Gestionnaire Formation et Emploi

1.2. Diagramme de cas d'utilisation globale :



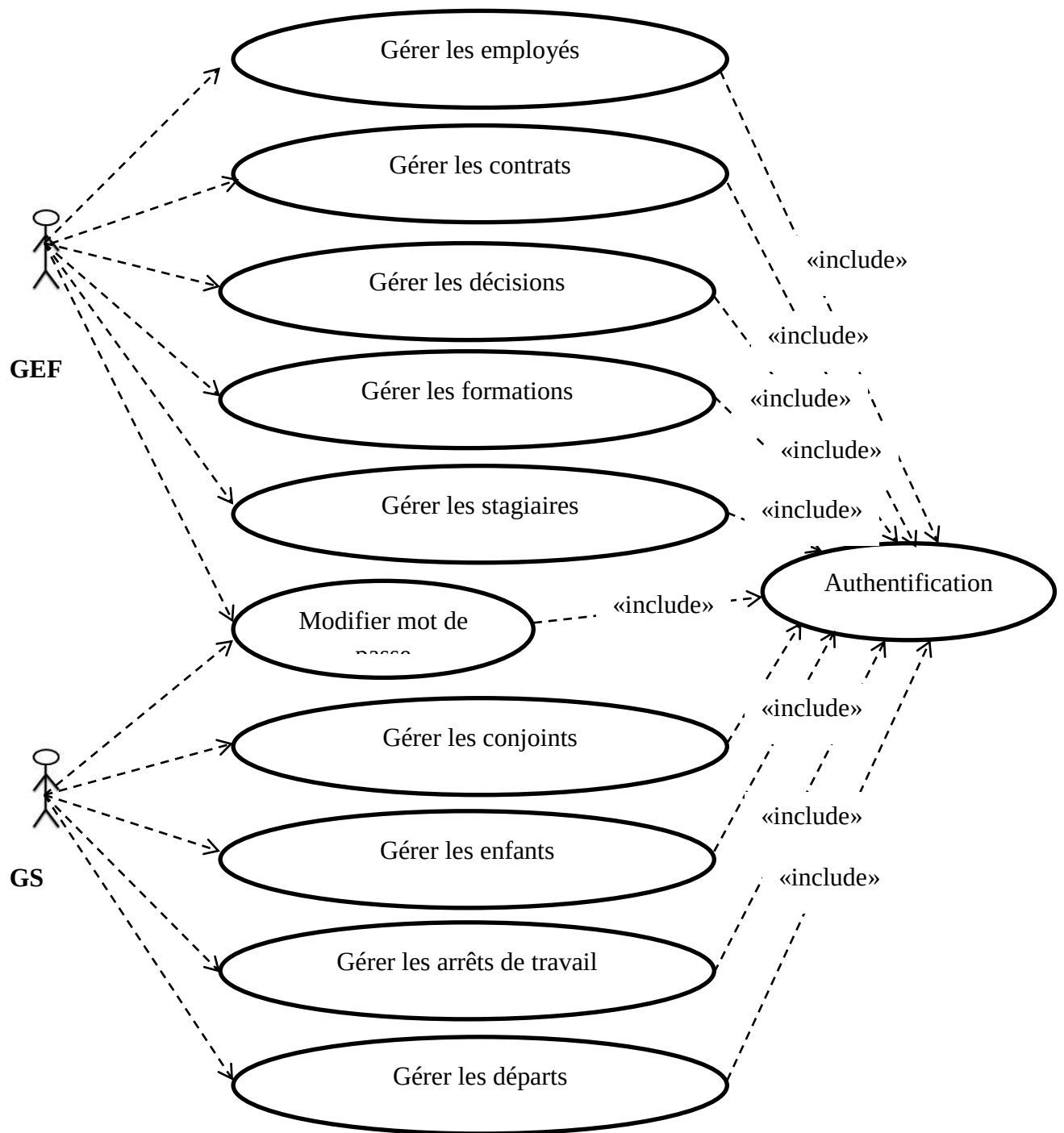


Figure : diagrammes des cas d'utilisations global

1.3.Description des cas d'utilisation :**Remarque :**

Vu le nombre important de cas d'utilisation, on va décrire seulement quelques-uns.

Cas d'utilisation N°1 : « identification d'utilisateur »

Acteur : Utilisateur (tous les utilisateurs du système).

Pré-condition : L'utilisateur doit se connecter.

Post-condition : Accès au système.

Scénario nominal :

1. Demande d'ouverture de session.
2. Le système affiche deux champs de saisie (login, password).
3. L'utilisateur introduit le login, password.
4. Le système vérifie l'existence de l'utilisateur dans le système.
5. Le système affiche l'interface correspondant au profil de l'utilisateur.

Enchaînement alternatif :

A1 : l'identification est provisoirement erronée.

L'enchaînement A1 démarre au point 4 du scénario nominal.

5. Le système indique à l'utilisateur que le code est erroné pour la 1 ère ou 2 eme fois, le scénario nominal reprend au point 2.

Enchaînement d'exception :

E1 : code d'identification définitivement erroné.

5. Le système indique à l'utilisateur que le code est erroné pour la 3 eme fois.
6. Le cas d'utilisation se termine.

Cas d'utilisation N° 2 : « modifier mot de passe »

Acteur : Utilisateur (tous les utilisateurs du système).

Pré-condition : l'utilisateur est connecté (password, login).

Post-condition : mot de passe modifier.

Scénario nominal :

1. L'utilisateur atteint son interface puis ses pages et clique sur le bouton « modifier mot de passe ».
2. Le système lui renvoi un formulaire contenant les champs : ancien password, nouveau password, confirme mot de passe.
3. L'utilisateur remplit le formulaire et soumet la requête.
4. Le système vérifie le mot de passe.
5. Le système lui renvoi un message de confirmation.

Enchaînement alternatif :

A1 : le mot de passe est provisoirement erroné

L'enchaînement A1 démarre au point 4 du scénario nominal.

5. le système indique à l'utilisateur que le code est erronée pour la 1ere ou 2 eme fois le scénario nominal reprend au point 2.

Enchaînement d'exception :

E1 : le mot de passe est définitivement erroné.

5. Le système indique à l'utilisateur que le code est erroné pour la 3 eme fois.
6. Le cas d'utilisation se termine

Cas d'utilisation N°3 : « Gestion des droits d'accès »

Acteur : ADMIN

But : Ajouter nouveau profil ou rôle, pour affecter de nouveaux utilisateurs à ce nouveau profil, et cela pour pouvoir ajouter un groupe (profil) d'utilisateurs

Pré-condition : Authentifier par un login et password.

Post-condition : le nom de profil doit être unique

Description : lorsque l'administrateur veut mettre à jour les droits d'accès des utilisateurs, il doit en premier lieu sélectionner parmi la liste des privilèges affichés par le système, ensuite il doit nommer ce dernier. Si ces deux derniers sont valides le système affiche un message de réussite de l'opération.

Scénario nominal :

1. Le système vérifie les pré-conditions.
2. Le système affiche la liste des privilèges.
3. L'administrateur sélectionne les privilèges nécessaires pour le nouveau profil et lui affecte un nom.
4. L'administrateur valide l'opération.

5. Le système affiche un message de succès de l'opération.

Cas d'utilisation N°4 : « Gérer les employés »**✓ Ajouter employé**

Acteur : GEF.

Pré-condition : GEF doit être connecté (login, password).

Post-condition : employé créé.

Scénario nominal :

1. GEF atteint la page employé.
2. Le système retourne le formulaire de l'employé.
3. GEF saisit et sélectionne les informations.
4. GEF clique sur le bouton « ajouter ».
5. Le système affiche une page de confirmation.

✓ Chercher employé :

Acteur : GEF

Pré-condition : GEF doit être connecté (login, password).

Post-condition : la liste des employés affichés.

Scénario nominal :

1. GEF accède à la page cherché.
2. Le système retourne le formulaire cherché
3. GEF saisit et sélectionne les champs du formulaire.
4. GEF clique sur le bouton chercher.
5. GEF choisit un employé à consulter.
6. Le système affiche les informations de l'employé.

✓ Modifier employé :

Acteur : GEF

Pré-condition : GEF doit être connecté (login, password).

Post-condition : les informations de l'employé sont modifiées.

Scénario nominal :

1. GEF accède à la page cherché.
2. Le système retourne le formulaire cherché
3. GEF saisit et sélectionne les champs du formulaire.
4. GEF clique sur le bouton chercher.
5. Le système affiche la liste des employés trouvé.
6. GEF choisit un employé à modifier.
7. GEF effectue les modifications sur l'employé et les sauvegardent.
8. Le système affiche une page de confirmation.

✓ Supprimer employé :**Acteur :** GEF**Pré-condition :** GEF doit être connecté (login, password).**Post-condition :** employé est supprimé.**Scénario nominal :**

1. GEF accède à la page cherché.
2. Le système retourne le formulaire cherché.
3. GEF saisit et sélectionne et sélectionne les champs du formulaire.
4. GEF clique sur le bouton chercher.
5. Le système affiche la liste des employés trouvés.
6. GEF choisit un employé à supprimer.
7. Le système affiche une page de confirmation.

Cas d'utilisation N°5 : « Gérer les contrats »**✓ Ajouter contrat****Acteur :** GEF**Pré-condition :** GEF doit être connecté (login, password).**Post-condition :** contrat créé.**Scénario nominal :**

1. GEF atteint la page contrat.
2. Le système retourne le formulaire de contrat.
3. GEF saisit et sélectionne les champs du formulaire.
4. GEF clique sur le bouton « ajouter ».
5. Le système affiche une page de confirmation.

✓ Chercher contrat :**Acteur :** GEF**Pré-condition :** GEF doit être connecté (login, password).**Post-condition :** afficher la liste des contrats.**Scénario nominal :**

1. GEF atteint la page de contrat.
2. Le système retourne le formulaire de contrat
3. GEF saisit et sélectionne les champs du formulaire.
4. GEF clique sur le bouton « chercher ».
5. Le système affiche la liste des contrats à consulter.
6. Le système affiche les informations du contrat choisit.

✓ Modifier contrat**Acteur :** GEF**Pré-condition :** GEF doit être connecté (login, password).**Post-condition :** contrat modifié.

Scénario nominal :

1. GEF atteint la page contrat.
2. Le système retourne le formulaire du contrat.
3. GEF saisit et sélectionne les champs du formulaire.
4. GEF clique sur le bouton « chercher ».
5. Le système affiche la liste des contrats trouvés.
6. GEF choisit un contrat à modifier.
7. GEF effectue la modification du contrat et le sauvegarde.
8. Le système affiche une page de confirmation.

✓ Supprimer contrat :**Acteur :** GEF**Pré-condition :** GEF doit être connecté (login, password).**Post-condition :** contrat supprimé.**Scénario nominal :**

1. GEF atteint la page contrat.
2. Le système retourne le formulaire du contrat.
3. GEF saisit et sélectionne les champs du formulaire.
4. GEF clique sur le bouton « chercher ».
5. Le système affiche la liste des contrats trouvés.
6. GEF choisit un contrat à supprimer.
7. Le système affiche une page de confirmation.

Cas d'utilisation N°6 : « Gérer la paie ».**✓ Cessation de paiement :****Acteur :** GPP**Pré-condition :** GPP doit être connecté (login, password).**Post-condition :** certificat de cessation de paiement créé.**Scénario nominal :**

1. GPP atteint la page cessation de paiement.
2. Le système retourne le formulaire de cessation de paiement.
3. GPP saisit et sélectionne les champs du formulaire.
4. GPP clique sur le bouton « ajouter ».
5. Le système affiche une page de confirmation.

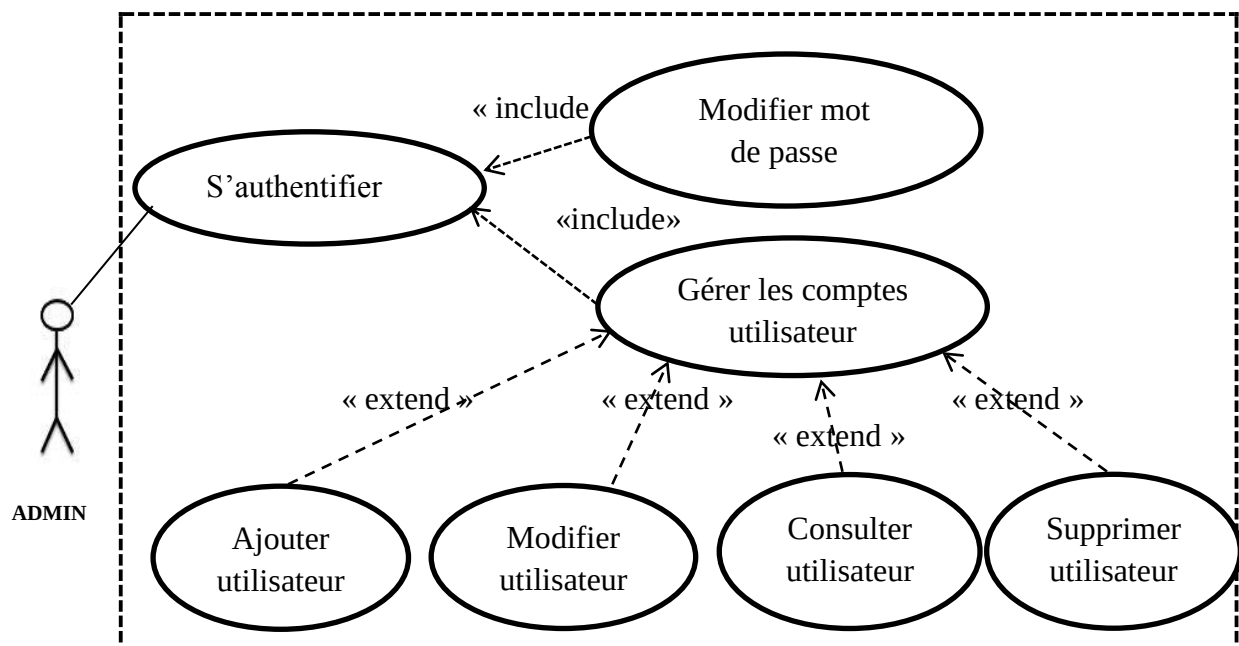
Cas d'utilisation N°7 : « Gérer les congés ».**✓ Titre de congé :****Acteur :** GPP**Pré-condition :** GPP doit être connecté (login, password).**Post-condition :** titre de congé créé

Scénario nominal :

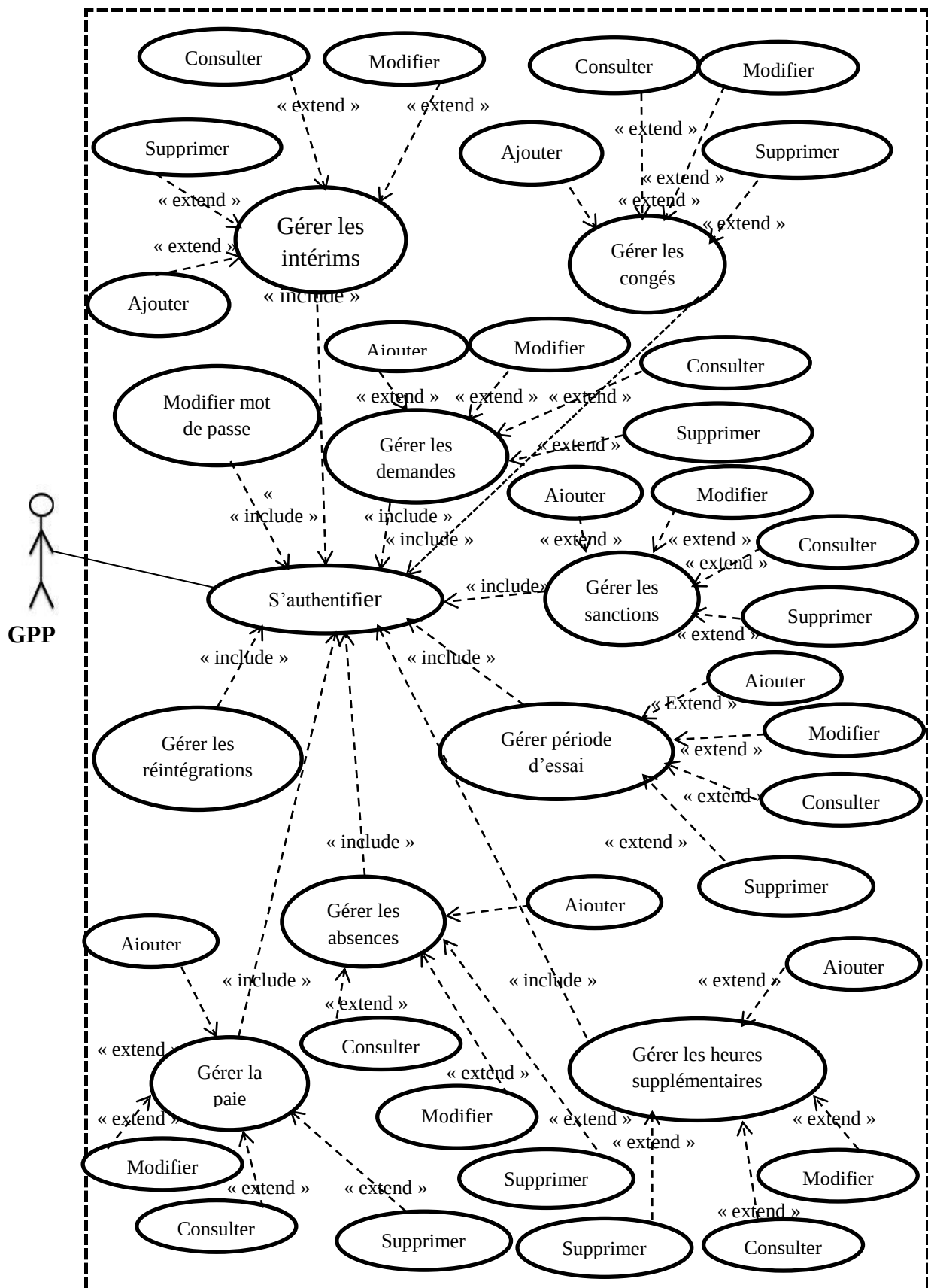
1. GPP atteint la page de congé.
2. Le système retourne le formulaire du titre de congé.
3. GPP saisit et sélectionne les champs du formulaire.
4. GPP clique sur ajouter.
5. Le système affiche une page de confirmation.

Cas d'utilisation N° 8 : « Gérer les enfants ».**✓ Consulter enfant :****Acteur :** GS**Pré-condition :** GS doit être connecté (login, password).**Post-condition :** liste des enfants établis.**Scénario nominal :**

1. GS atteint la page cherché enfant.
2. Le système retourne le formulaire cherché enfant.
3. GS clique sur le bouton chercher
4. Le système affiche la liste des enfants trouvés.
5. GS choisit un enfant à consulter.
6. Le système affiche les informations de l'enfant.

Diagramme de cas d'utilisation détaillé :**✓ La représentation de cas d'utilisation relatif à l'ADMIN :****Figure : Diagramme de cas d'utilisation détaillé relatif à l'ADMIN**

✓ Diagramme de cas d'utilisation relatif à GPP :



✓ Représentation de cas d'utilisation relatif au GS :

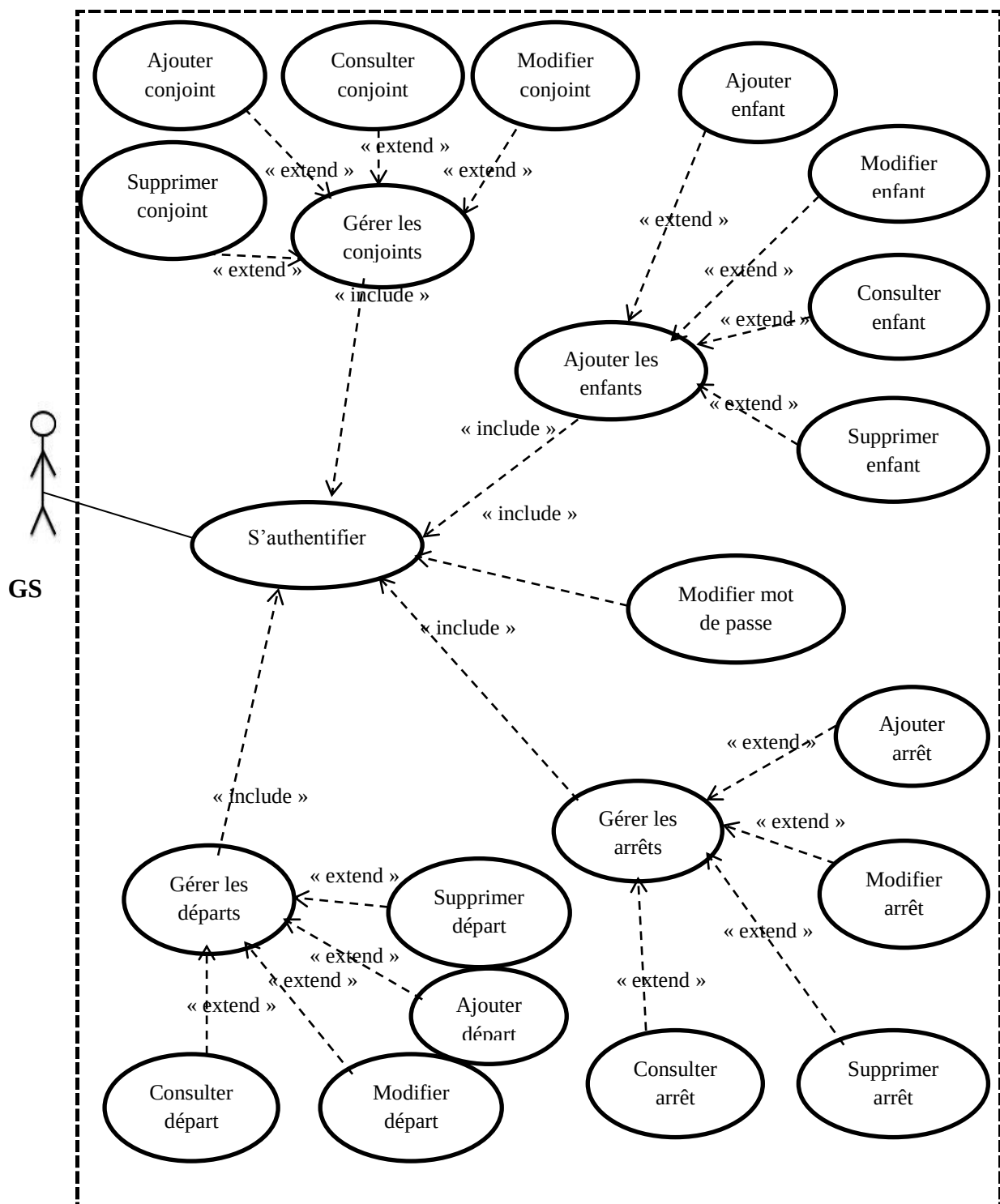


Figure : Diagramme de cas d'utilisation relatif au GS

✓ Représentation de cas d'utilisation relatif au chargé de l'emploi et formation :

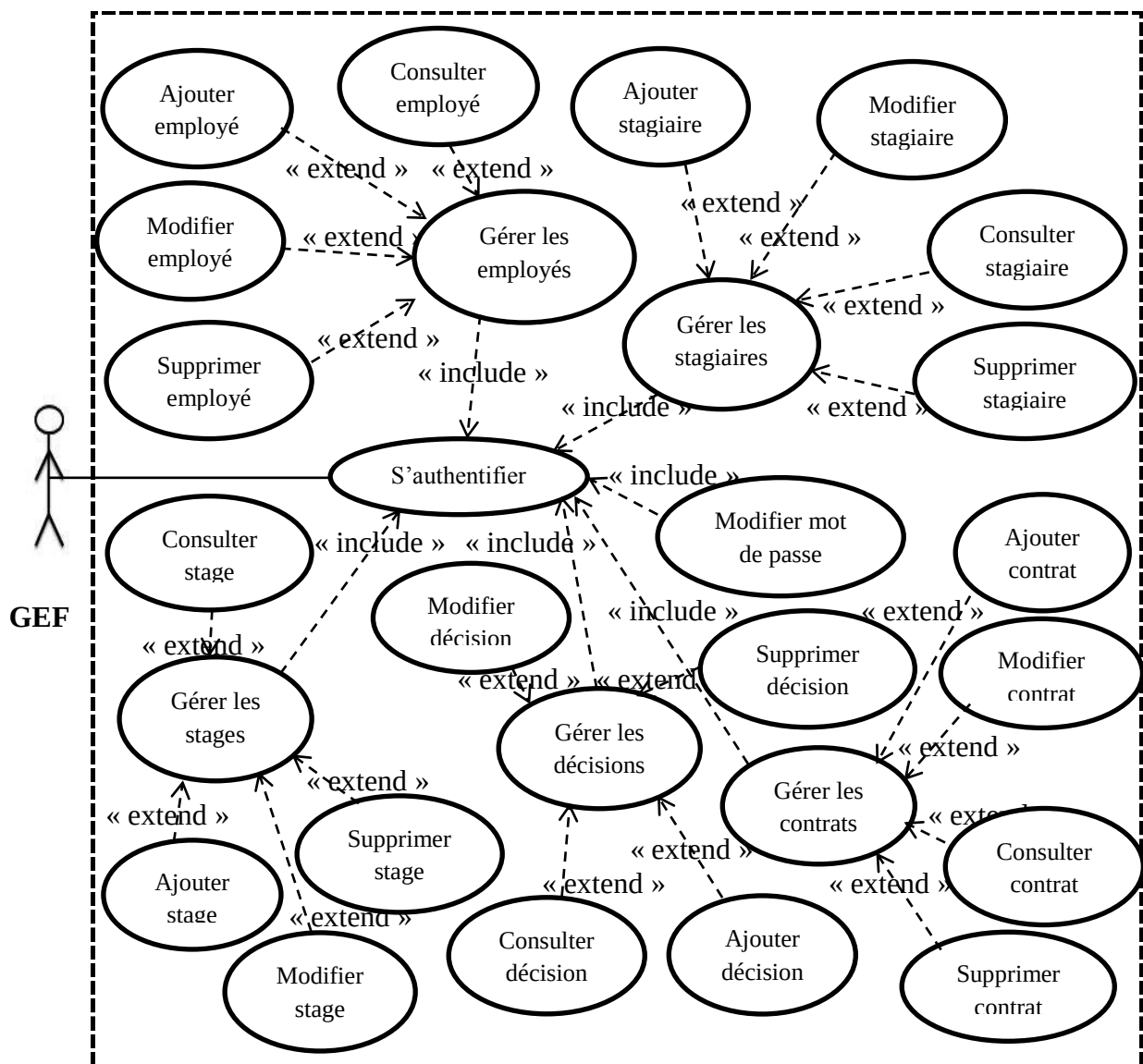


Figure : Diagramme de cas d'utilisation relatif au GEF

1.4. Les scénarios par les diagrammes de séquence et d'activité :

- ✓ Un scénario est une séquence d'évènement qui se produit pendant une exécution du système.
- ✓ Les diagrammes de séquence permettent de représenter les interactions entre objets selon un point de vue temporel. L'accent est mis sur la chronologie des envois de messages.
- ✓ Les diagrammes d'activité décrivent le comportement d'une méthode, le déroulement d'un cas d'utilisation, les enchaînements d'activités.

a. Scénario de cas d'utilisation « modifier mot de passe » :**➤ Scénario « modifier mot de passe » :**

Acteur : tous les acteurs du système.

Pré-condition : l'utilisateur doit être connecté (password, login).

Poste_condition : mot de passe modifié.

Description textuelle de scénario :

L'utilisateur sollicite du système la modification du mot de passe. Le système demande de préciser l'ancien et le nouveau mot de passe puis procède à la recherche de l'existence de l'ancien mot de passe. S'il n'est pas trouvé, le système signalera « mot de passe erroné ». Sinon il fournira un message de confirmation.

Illustration du scénario par un diagramme de séquence :

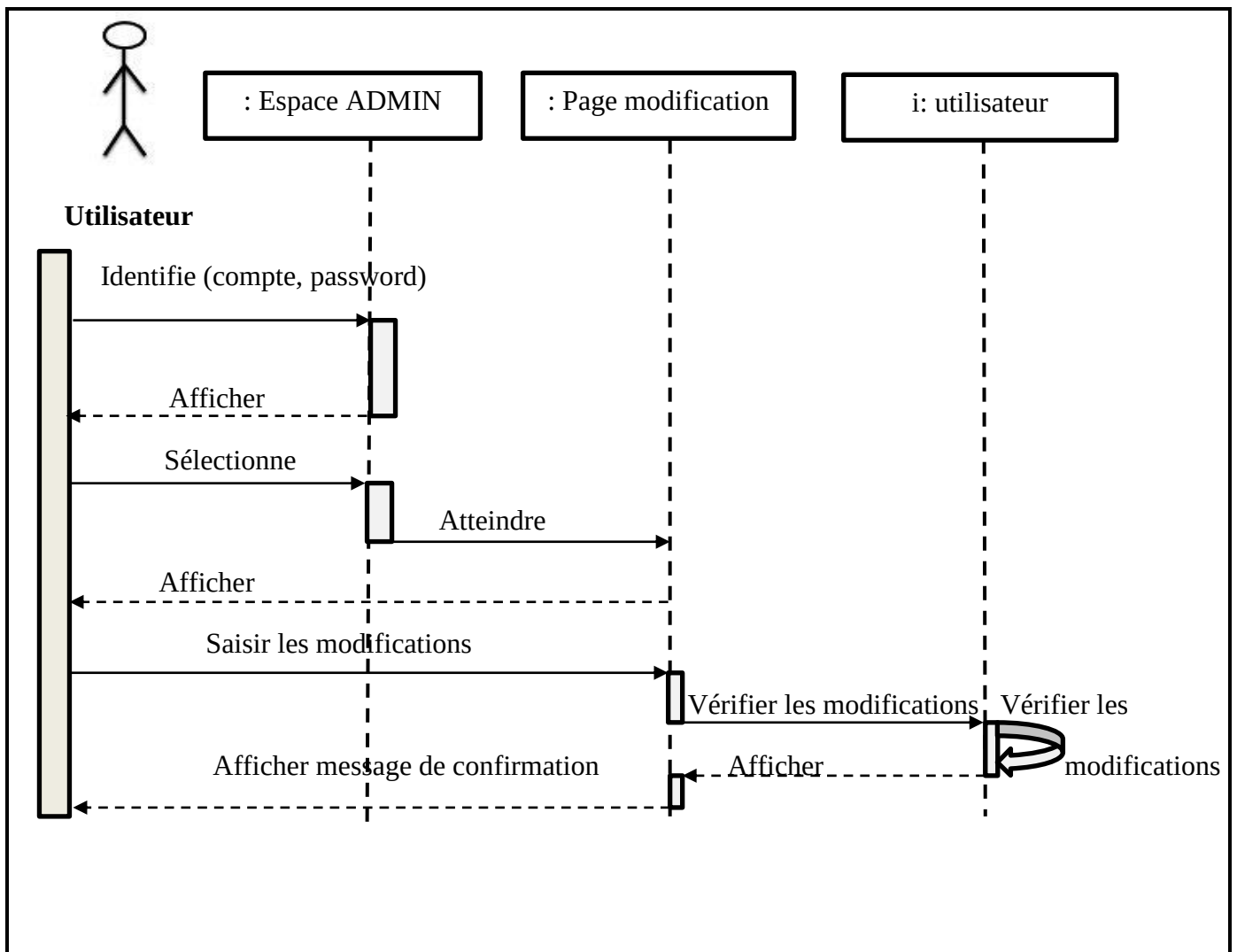


Figure : Diagramme de séquence associé au scénario « modifier mot de passe »

Illustration du scénario par un diagramme d'activité :

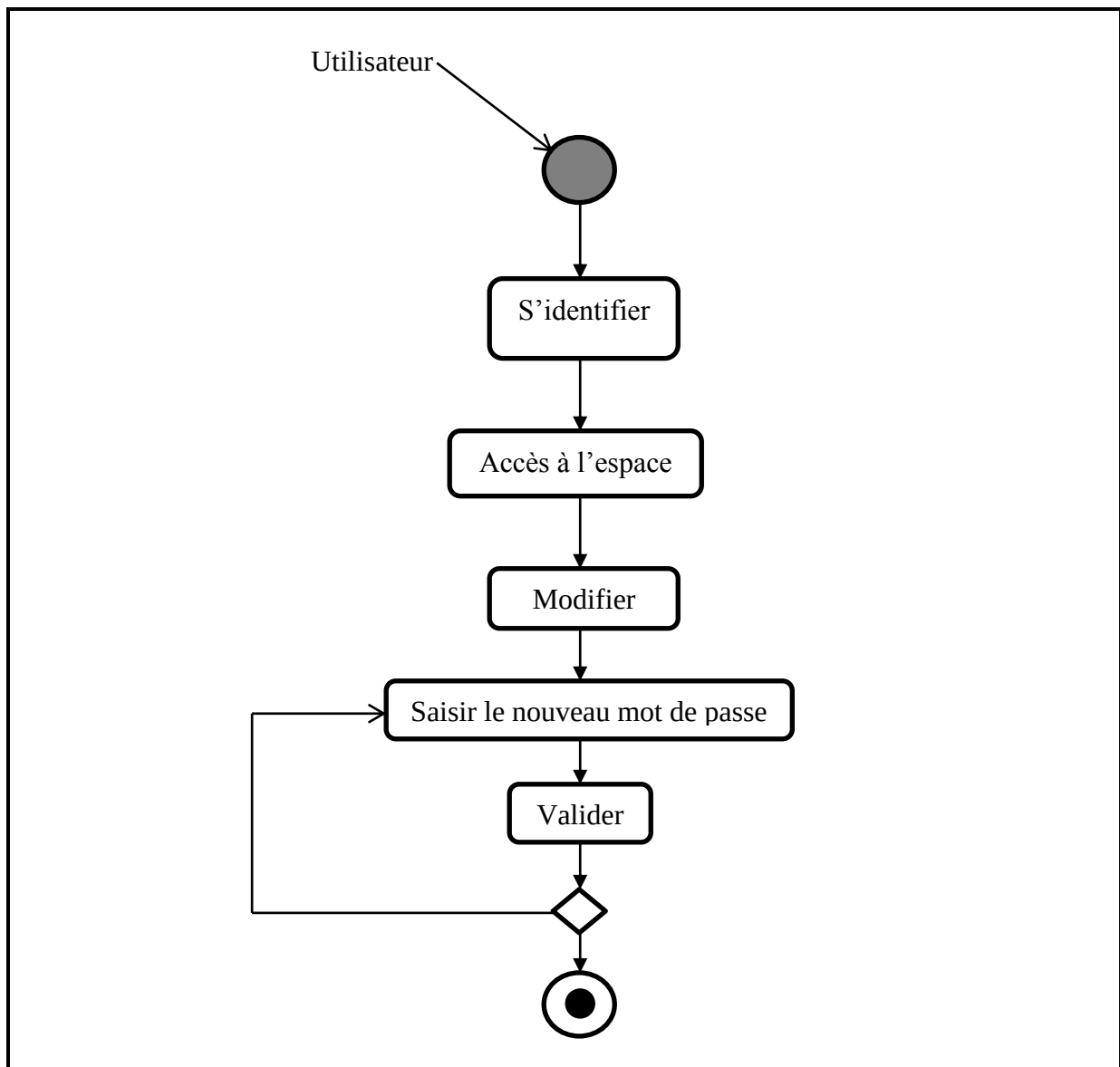


Figure : Diagramme d'activité « modifier mot de passe »

b. Scénario de cas d'utilisation « gérer les employés »

✓ **Scénario « ajouter employé » :**

Acteur : gestionnaire personnel et paie (GPP).

Pré-condition : GPP doit être connecté (password, login).

Poste-condition : l'employé est ajouté.

Description textuelle de scénario :

Le GPP demande au système l'ajout d'un employé via l'interface appropriée. Le système lui demande alors de fournir les informations associées à l'employé. Le système procède alors à la vérification des différentes informations fournies au clavier puis le contrôle. Au cas d'absence d'erreur l'enregistrement sera effectué, sinon toute erreur sera signalée.

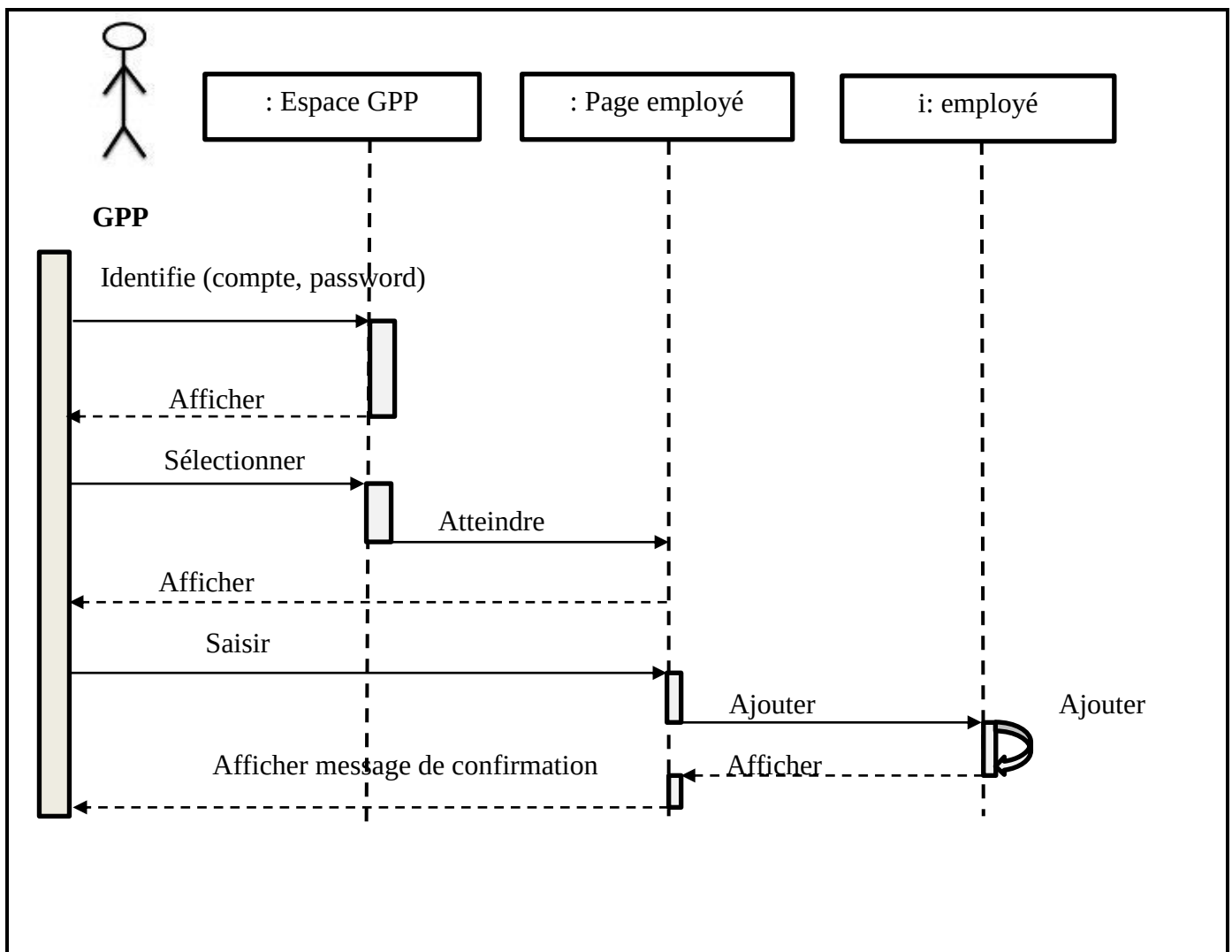
Illustration du scénario avec un diagramme de séquence :

Figure : Diagramme de séquence associé au scénario « ajouter employé »

Illustration du scénario par un diagramme d'activité :

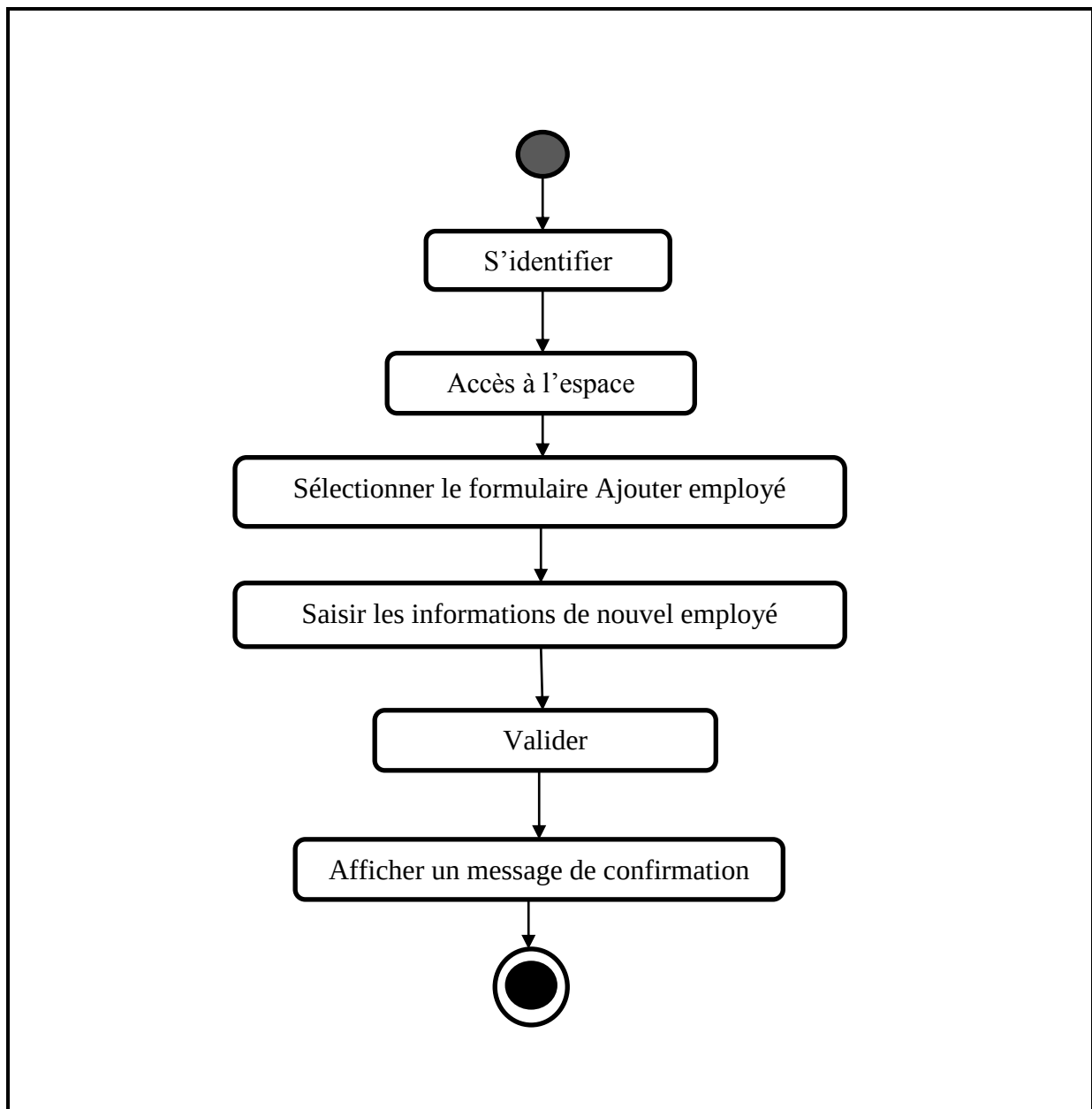


Figure : Diagramme d'activité « ajouter employé »

c. Scénario de cas d'utilisation « gérer les enfants »

✓ Scénario « consulter enfants »

Acteur : gestionnaire social (GS).

Pré-condition : GS doit être connecté (password, login).

Poste-condition : liste des enfants établis.

Description textuelle du scénario :

GS demande la consultation de la liste des enfants. Le système déclenche alors le processus de recherche puis établira la liste des enfants.

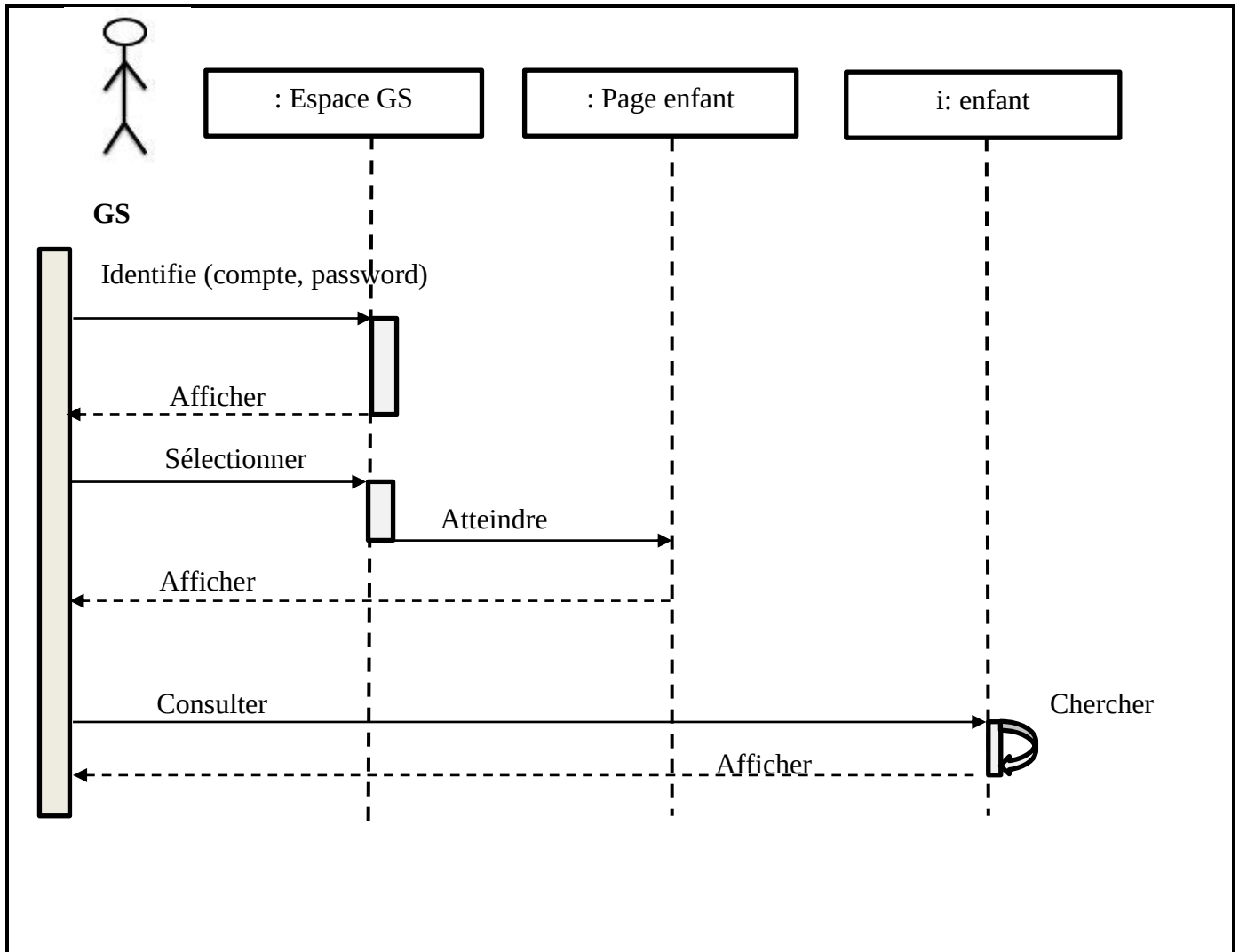
Illustration du scénario par un diagramme de séquence :

Figure : Diagramme de séquence associé au scénario « consulter enfant »

Illustration du scénario par un diagramme d'activité :

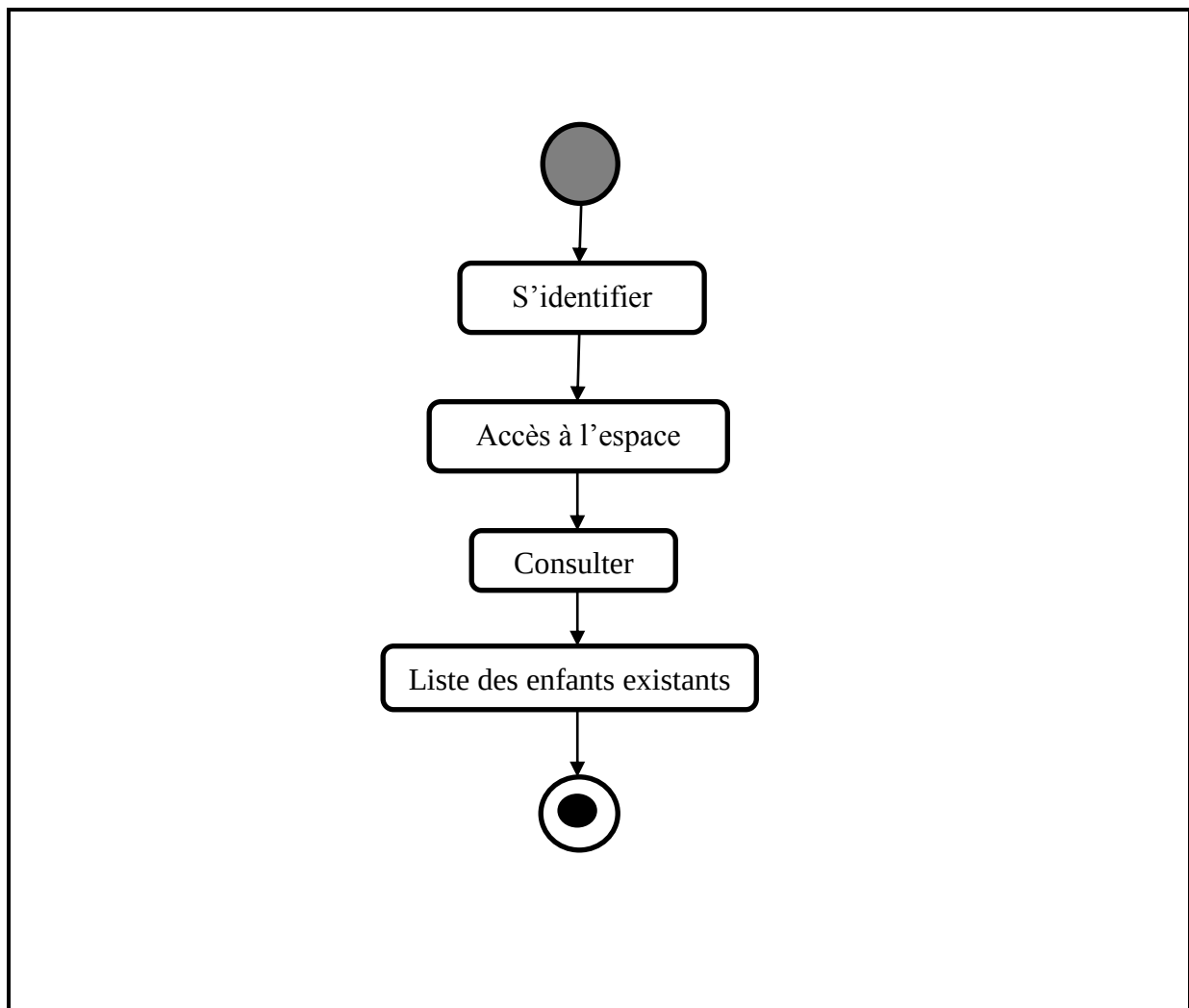


Figure : Diagramme d'activité « consulter enfant »

1.1.1. Structuration des cas d'utilisations dans des packages

Cette phase va permettre de structurer les cas d'utilisations en groupes fortement cohérents, ceci afin de préparer le terrain pour la prochaine phase qui est le « découpage en catégories ».

Définition : un package représente un espace de nommage qui peut contenir :

1. Des éléments d'un modèle.
2. Des diagrammes qui représentent les éléments du modèle.
3. D'autres packages.

La structuration des cas d'utilisations se fait par domaine d'expertise métier c.à.d. les éléments contenus dans un package doivent représenter un ensemble fortement cohérent et sont généralement de même nature et de même niveau sémantique.

1.1.2. Identification des classes candidates

Cette phase va préparer la modélisation orientée objet en aidant à trouver les classes principales du futur modèle statique d'analyse.

La technique utilisée pour identifier les classes candidates est la suivante :

- Chercher les noms communs importants dans les descriptions textuelles des cas d'utilisation.
- Vérifier les propriétés « objet » de chaque concept (identité, propriétés, comportement), puis définir ses responsabilités.

Définition : une responsabilité est une sorte de contrat, ou d'obligation, pour une classe. Elle se place à un niveau d'abstraction plus élevé que les attributs ou les opérations. On formalise ensuite ces concepts métier sous forme de classes et d'associations rassemblées dans un diagramme statique pour chaque cas d'utilisation. Ces diagrammes préliminaires sont appelés « *diagramme des classes participantes* », n'ont pas pour objectif d'être complet. Ils servent uniquement à démarrer la découverte des classes du modèle d'analyse.

Exemple :

- **Diagramme des classes participantes du cas d'utilisation « Gestion des profils »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci, le cas Gestion des profils contient trois classes candidates, il a pour mission de spécifier le rôle (profil) pour chaque utilisateur.

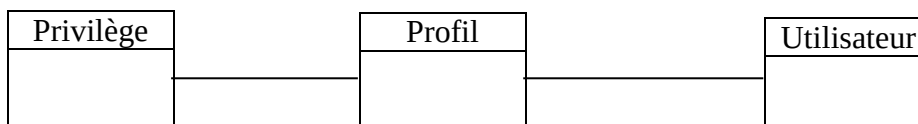


Figure : Diagramme des classe participantes « gestion des droits d'accès »

- **Diagramme des classes participantes du cas d'utilisation « Gestion des employés »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci :

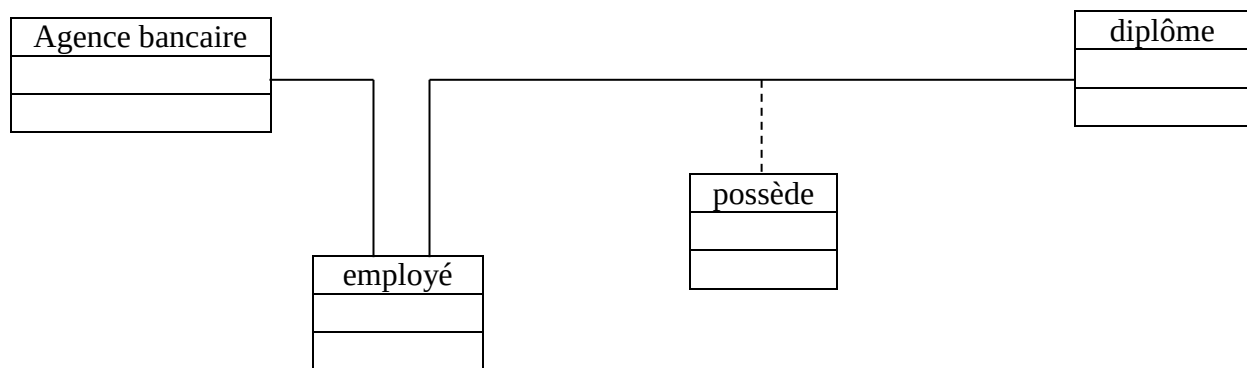


Figure : Diagramme des classes participantes « gérer les employés »

- **Diagramme des classes participantes du cas d'utilisation « Gérer les absences »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci :

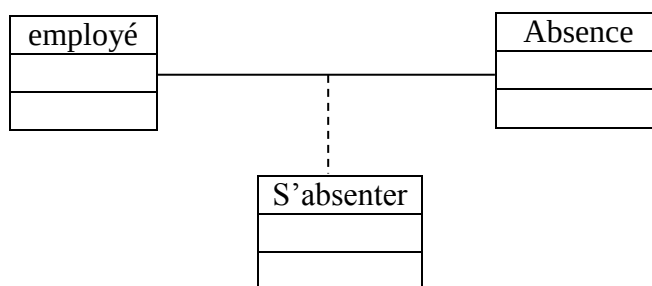


Figure : Diagramme des classes « gérer les absences »

- **Diagramme des classes participantes du cas d'utilisation « Gérer les départs »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci :

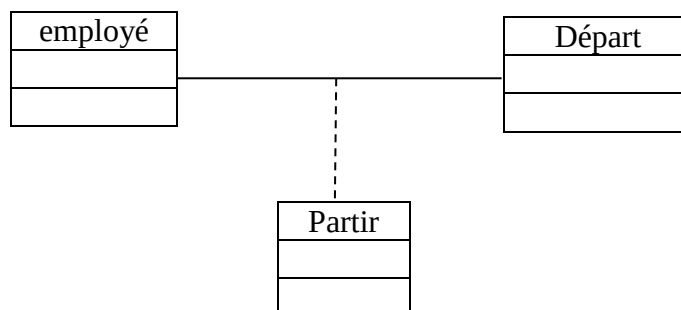


Figure : Diagramme des classes participantes « Gérer les départs »

- **Diagramme des classes participantes du cas d'utilisation « Gérer les enfant »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci :

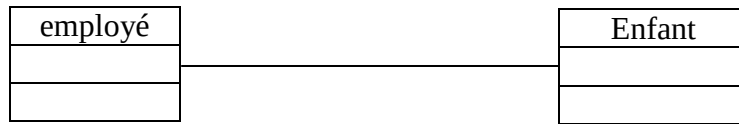


Figure : Diagrammes des classes participantes « Gérer les enfants »

- **Diagramme des classes participantes du cas d'utilisation « Gérer les promotions »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci :

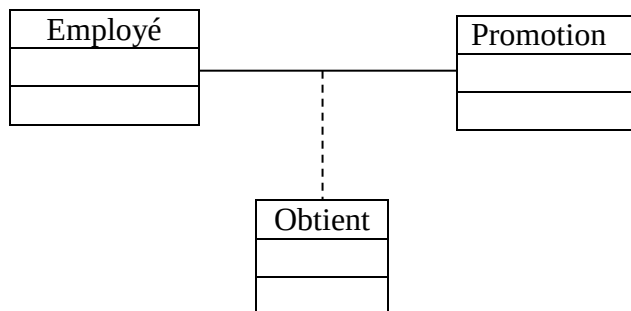


Figure : Diagramme des classes participantes « Gérer les promotions »

- **Diagramme des classes participantes du cas d'utilisation « Gérer les contrats »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci :

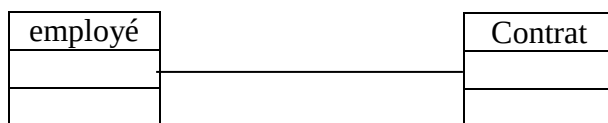


Figure : Diagramme des classes participantes « Gérer les contrats »

- **Diagramme des classes participantes du cas d'utilisation « Gérer les décisions »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci :

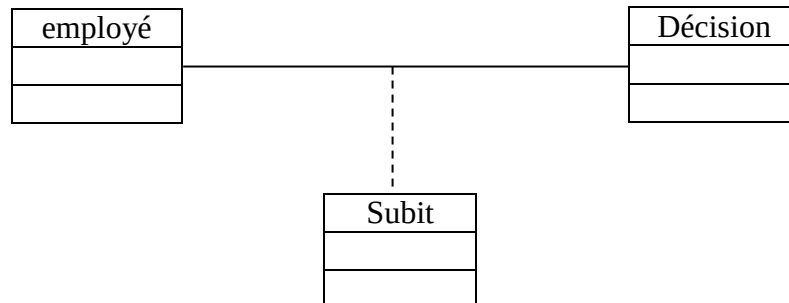


Figure : Diagramme des classes participantes « Gérer les décisions »

- **Diagramme des classes participantes du cas d'utilisation « Gérer les congés »**

Les classes candidates sont tirées de la description textuelle du cas d'utilisation et des diagrammes dynamiques représentant celui-ci :

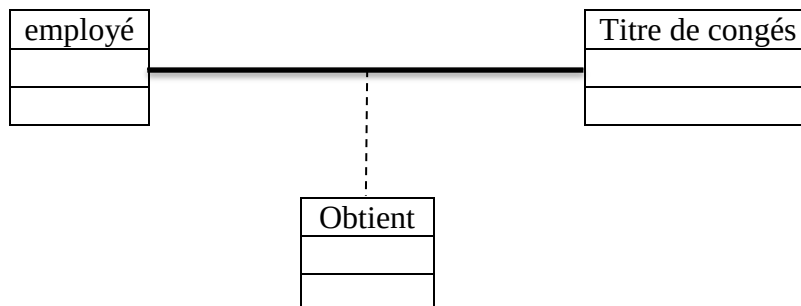


Figure : Diagramme des classes participantes « Gérer les congés »

DEVELOPPEMENT DE MODELE STATIQUE

1. Analyse statique :

Cette étape nous permettra d'illustrer les principales constructions du diagramme de classes. Ces diagrammes de classes sont sommairement élaborés dans l'étape d'analyse des besoins fonctionnels seront détaillés, complétés et optimisés.

2. Dictionnaire de données :

Attribut	Désignation	Type	Taille
MAT_EMP	Matricule de l'employé	AN	9
NOM_EMP	Nom de l'employé	AN	20
PREN_EMP	Prénom de l'employé	AN	20
SEX_EMP	Sexe de l'employé	AN	1
DAT_NAIS_EMP	Date de naissance de l'employé	Date	
ADR_EMP	Adresse de l'employé	AN	40
SIT_F_EMP	Situation familiale de l'employé	AN	18
DAT_RECRUT	Date de recrutement de l'employé	Date	
NUM_SS	Numéro de sécurité sociale	N	10
COD_AGEN_BANC	Code agence bancaire	N	3
NBRE_ECHELON	Nombre échelon	N	2
NBRE_KM_PARCOURUS	Nombre de Km parcourus	N	3
COD_ABS	Code absence	AN	4
DES_ABS	Désignation absence	AN	30
DAT_ABS	Date absence	Date	
NBRE_H_ABS	Nombre heures absentes	N	3
DES_AGEN_BA NC	Désignation agence bancaire	AN	45
ADR_AGEN_BANC	Adresse agence bancaire	AN	40
COD_CONGE	Code congé	AN	3
DES_CONGE	Désignation congé	AN	40
COD_CONJ	Code conjoint	N	7
NOM_CONJ	Nom conjoint	AN	20
PRE_CONJ	Prénom conjoint	AN	20
DAT_NAIS_CONJ	Date de naissance conjoint	Date	
SIT_PROF_CONJ	Situation professionnelle conjoint	N	1
COD_CONTRAT	Code contrat	N	6
DES_CONTRAT	Désignation contrat	AN	30
DUR_CONTRAT	Durée contrat	AN	7
DATE_CONTRAT	Date contrat	Date	
COD_DECISION	Code décision	N	4
LIB_DECISION	Libellé décision	AN	20
DAT_DECISION	Date décision	Date	
MOTIF_DEG	Motif dégradation	AN	10
DAT_DEG	Date dégradation	Date	
COD_DEMANDE	Code demande	N	4
OBJET_DEMANDE	Objet demande	AN	30
DAT_DEMANDE	Date demande	Date	
DATE_ARRET	Date de l'arrêt	Date	
NBR_H_SUPPL	Nombre heures supplémentaires	N	3
COD_ENF	Code enfant	N	8

PRE_ENF	Prénom enfant	AN	20
DAT_NAIS_ENF	Date naissance enfant	Date	
SEX_ENF	Sexe enfant	AN	1
SIT_SCOLAIRE	Situation scolaire de l'enfant	AN	10
TAUX_RUB_PAIE	Taux rubrique paie	AN	6
TOTAL_GAIN	Total gain	N	8
TOTAL_RETENIR	Total à retenir	N	8
NET_PAYER	Net à payer	N	8
MOIS_ANN	MM\AAAA	Date	
COD_H_SUPP	Code heure supplémentaire	N	3
DES_H_SUPP	Désignation heure supplémentaire	AN	50
DAT_INTERIM	Date intérim	Date	
COD_MOD_REG	Code mode règlement	AN	4
DES_MOD_REG	Désignation mode règlement	AN	30
COD_PERIOD_ESSAI	Code période d'essai	AN	3
APPRECIA_FINALE	Appréciation finale	AN	35
DAT_CONGE	Date du congé	Date	
NBR_JOUR_CONGE	Nombre de jour congé	N	3
DAT_PROMOTION	Date promotion	Date	
MOTIF_REINTEG	Motif réintégration	AN	30
DAT_REINTEG	Date réintégration	Date	
COD_RUB_PAIE	Code rubrique paie	N	3
DES_RUB_PAIE	Désignation rubrique paie	AN	50
COD_SANCT	Code sanction	AN	4
DES_SANCT	Désignation sanction	AN	40
DAT_DEBUT	Date début de la période d'essai	Date	
DUREE_P_ESSAI	Durée de la période d'essai	N	3
DAT_SANCT	Date sanction	Date	
MOTIF_SANCT	Motif sanction	AN	40
COMPTE_USER	Compte utilisateur	AN	10
PWD_USER	Mot de passe utilisateur	AN	10
SECTION_USER	Section utilisateur	AN	10
COD_DEPART	Code départ	AN	3
NAT_DEPART	Nature départ	AN	30
DAT_DEPART	Date départ	Date	
DAT_CESSA_PAIE	Date de cessation paie	Date	
MOTIF_DEPART	Motif départ	AN	30
NUM_COMPTE	Numéro de compte	AN	10
COD_ARRET	Code arrêt du travail	AN	5
DES_ARRET	Désignation de l'arrêt	AN	30
COD_PROMOTION	Code promotion	AN	4
COD_INTERIM	Code intérim	AN	7
COD_REINTEG	Code réintégration	AN	7
COD_DIPL	Code diplôme	AN	6
DES_DIPL	Désignation du diplôme	AN	20
DAT_OBT_DIPL	Date obtention du diplôme	Date	
DURE_ARRET	Durée arrêt	N	5
DAT_H_SUPPL	Date heure supplémentaire	Date	

DES_INTERIM	Désignation intérim	AN	30
DES_PROMOTION	Désignation promotion	AN	30
GRADE_EMP	Grade de l'employé	N	2
GENRE_RUB_PAIE	Genre rubrique paie	A	10
TYPE_RUB_PAIE	Type rubrique paie	A	1
SIGNE_RUB_PAIE	Signe rubrique paie	A	1
ID_USER	Identificateur de l'utilisateur	A	4
LOGIN_USER	Login de l'utilisateur	AN	30
PASSWORD_USER	Mot de passe de l'utilisateur	AN	30
ETAT_USER	Etat de l'utilisateur (actif ou inactif)	A	10
DATE_CRE_USER	Date de création de l'utilisateur	Date	
NOM_USER	Le nom de l'utilisateur	A	30
PRE_USER	Le prénom de l'utilisateur	A	30
DATE_NAIS_USER	La date de naissance de l'utilisateur	Date	
LIEU_NAIS_USER	Lieu de naissance de l'utilisateur	A	50
ADR_USER	Adresse de l'utilisateur	A	50
TEL_USER	Numéro téléphone de l'utilisateur	N	13
EMAIL_USER	Adresse email de l'utilisateur	AN	50
ID_PRIVILEGE	Identificateur du privilège	N	4
NOM_PRIVILEGE	Nom du privilège	A	30
VAL_PRIVILEGE	Valeur du privilège	Booléen	
ID_ROLE	Identificateur du rôle de l'utilisateur	N	2
NOM_ROLE	Nom du rôle de l'utilisateur	A	50

Tableau : Tableau du dictionnaire de données

3. Liste des classes :

Classe	Attributs
Employé	mat_emp, nom_emp, pren_emp, num_compte, sexe_emp, adr_emp, dat_nais_emp, sit_f_emp, dat_recrut, nbre_echelon, num_ss, grade_emp, nb_km_parcourus, poste_emp, num_tel_emp.
Utilisateur	Id_user, login_user, password_user, etat_user, date_cre_user, nom_user, pre_user, email_user, lieu_naiss_user, adr_user, tel_user, date_nais_user.
Rôle	Id_role, nom_role.
Privilège	Id_privilege, val_privilege, nom_privilege
Enfant	Cod_enf, pre_enf, dat_nais_enf, sex_enf, cod_scol_enf
Conjoint	Cod_conj, nom_conj, pre_conj, dat_nais_conj, sit_prof_conj
Absence	Cod_abs, des_abs
Agence Bancaire	Cod_agen_banc, des_agen_banc, adr_agen_banc
Contrat	Cod_contrat, des_contrat, dur_contrat, dat_contrat
Congé	Cod_conge, des_conge
Demande	Cod_demande, objet_demande, dat_demande

Départ	Cod_depart, nat_depart, dat_cessa_paie, motif_depart
Décision	Cod_decision, lib_decision, dat_decision
Rubrique paie	Cod_rub_paie, des_rub_paie, genre_rub_paie, type_rub_paie, signe_rub_paie
Sanction	Cod_sanct, des_sanct
Mode règlement	Cod_mod_reg, des_mod_reg
Période d'essai	Cod_period_essai, apprecia_finale
Heure supplémentaire	Cod_h_suppl, des_h_suppl
Arrêt	Cod_arret, lib_arret
Diplôme	Cod_dip, des_dip
Intérim	Cod_interim, des_interim, dat_interim
Réintégré	Cod_reinteg, motif_reinteg, dat_reinteg
Promotion	Cod_promotion, des_promotion, dat_promotion

Tab : tableau des classes

4. Liste des classes d'association :

En modélisation UML, toute association porteuse d'attributs doit être conçue et représentée en classe d'association. Le tableau suivant explique les classes d'association énumérées dans cette étude avec leurs principaux attributs.

Classe	Attributs
Effectuer	Date_h_suppl, nbr_h_suppl
Est-payé	Total_gain, total_retenir, nbre_rub_paie, taux_rub_paie
Est-en	Dure_arret, date_arret
Partir	Dat_depart
Soumettre	Date_début, durée_p_essai
Subir	Dat_sanct, motif_sanct
Prendre	Nbre_jour_conge, dat_conge
S'absente	Dat_abs, nbrs_h_abs
Possède	Date_obt_dip
Est-inscrit	
Faire	
Correspond	
Concerner	
Avoir	
Lier	
Etre	
Obtenir	
Est-en	

Tab : tableau des classes « association »

Nous avons vu dans la capture des besoins fonctionnels qu'un cas d'utilisation décrit un ensemble de scénarios. Lors de l'étape de la description des cas d'utilisation, un scénario représente une séquence d'interactions entre le système et ses acteurs. Le système est alors considéré comme une boîte noire. Maintenant que nous avons développé le modèle statique d'analyse, nous allons remplacer le système par une collaboration d'objets dans chaque scénario. Pour notre cas nous allons détailler chaque cas d'utilisation par un diagramme de collaboration et diagramme de séquence.

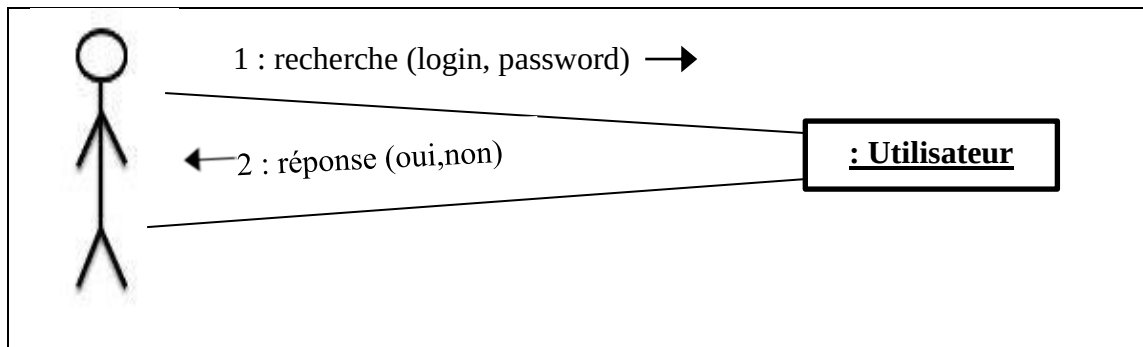


Figure : diagramme de collaboration « identification des utilisateurs » ;

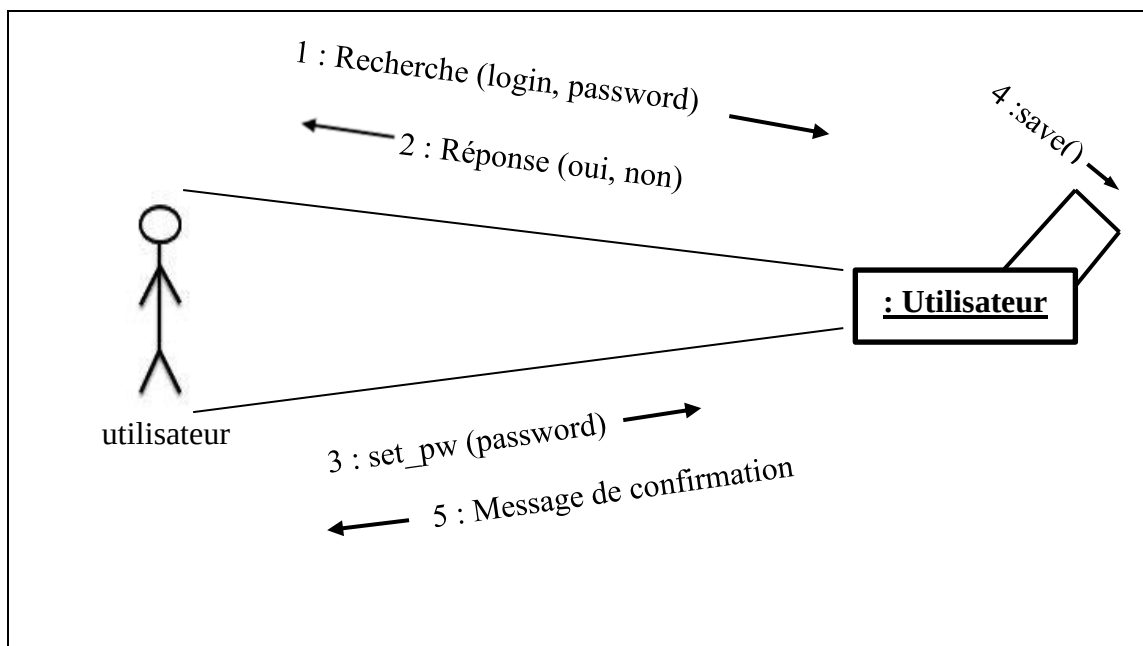


Figure : diagramme de collaboration « modifier mot de passe » ;

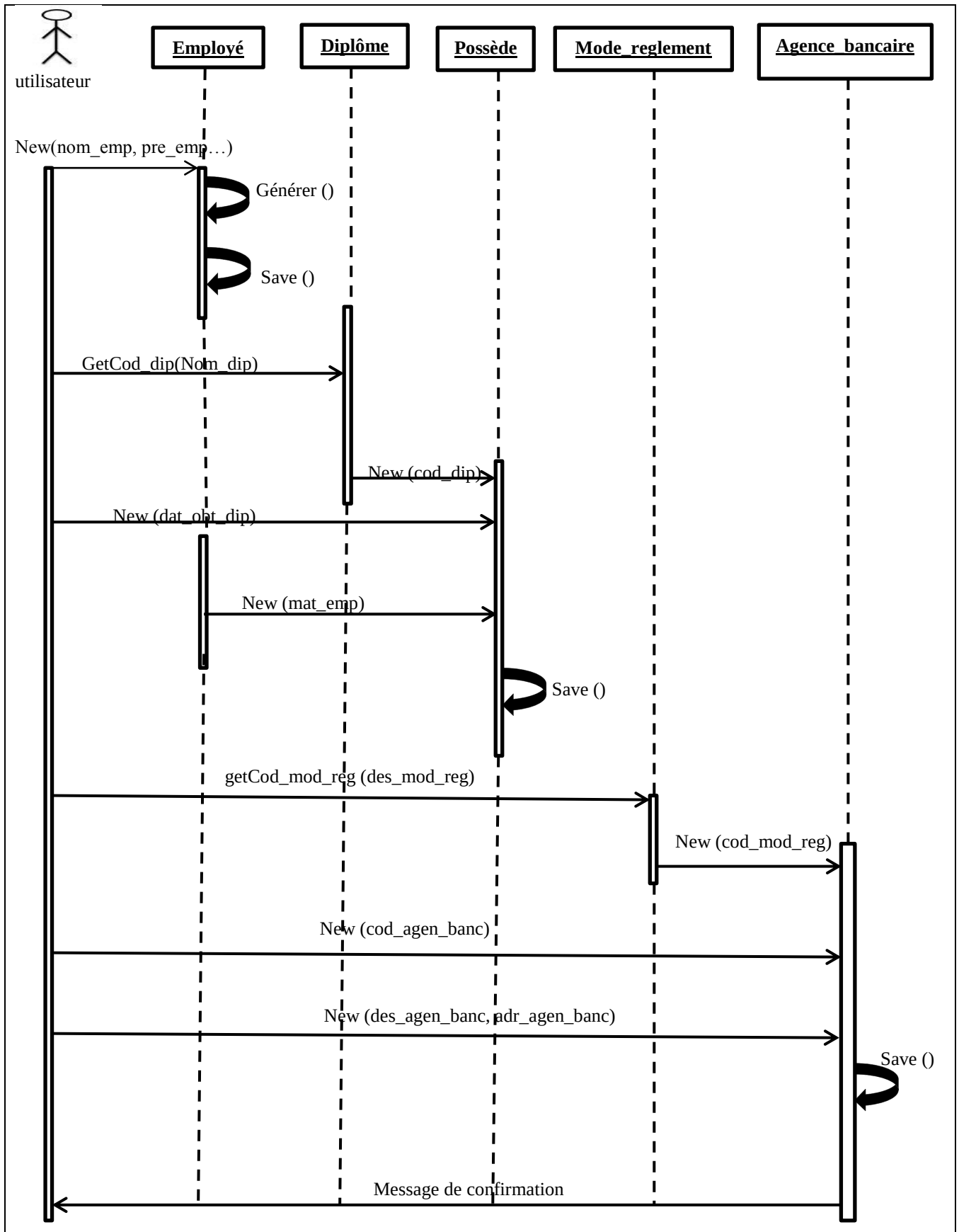


Figure : Diagramme de séquence « ajouter employé ».

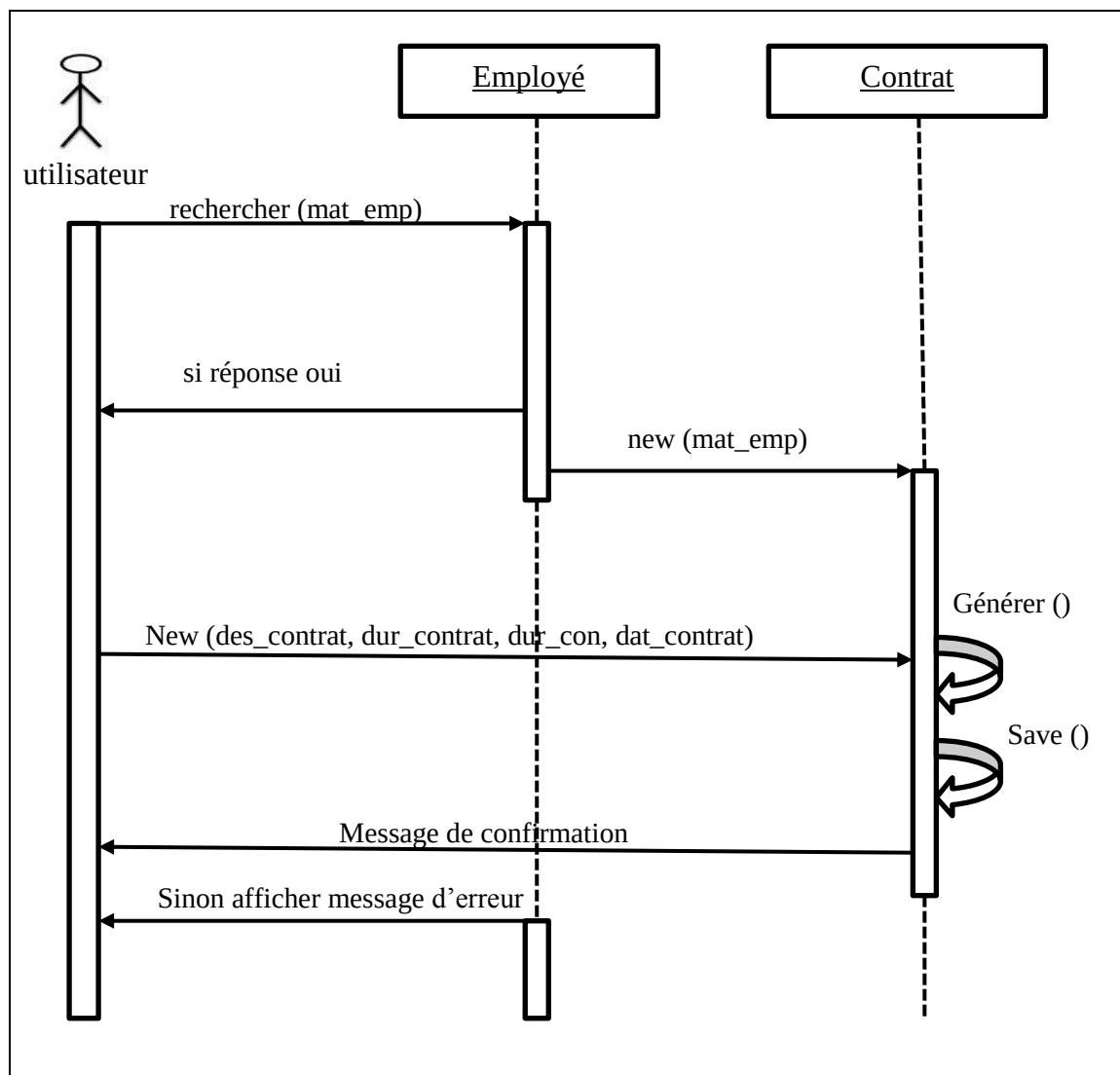


Figure : Diagramme de séquence « Ajouter contrat »

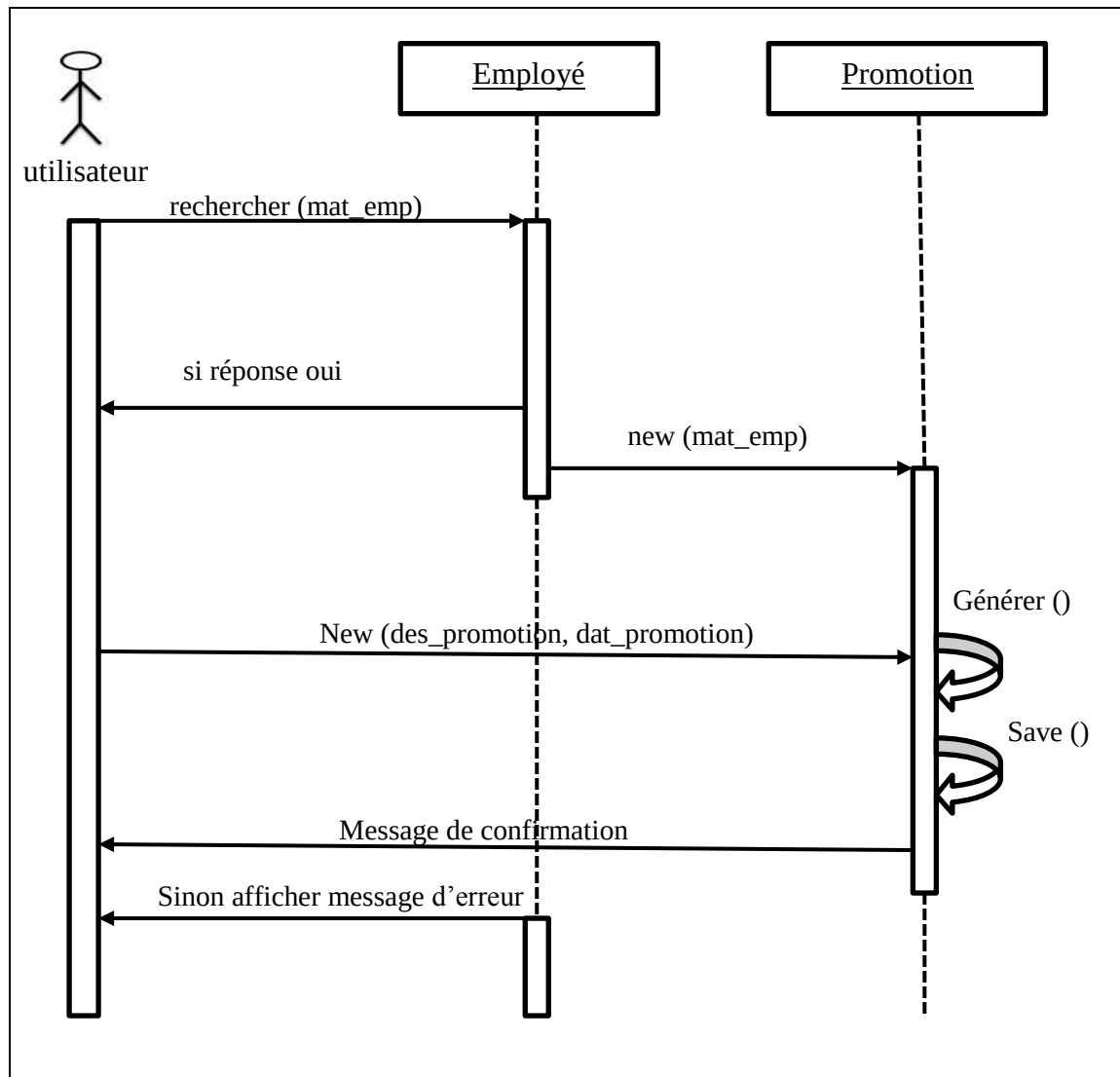


Figure : Diagramme de séquence « Ajouter promotion »

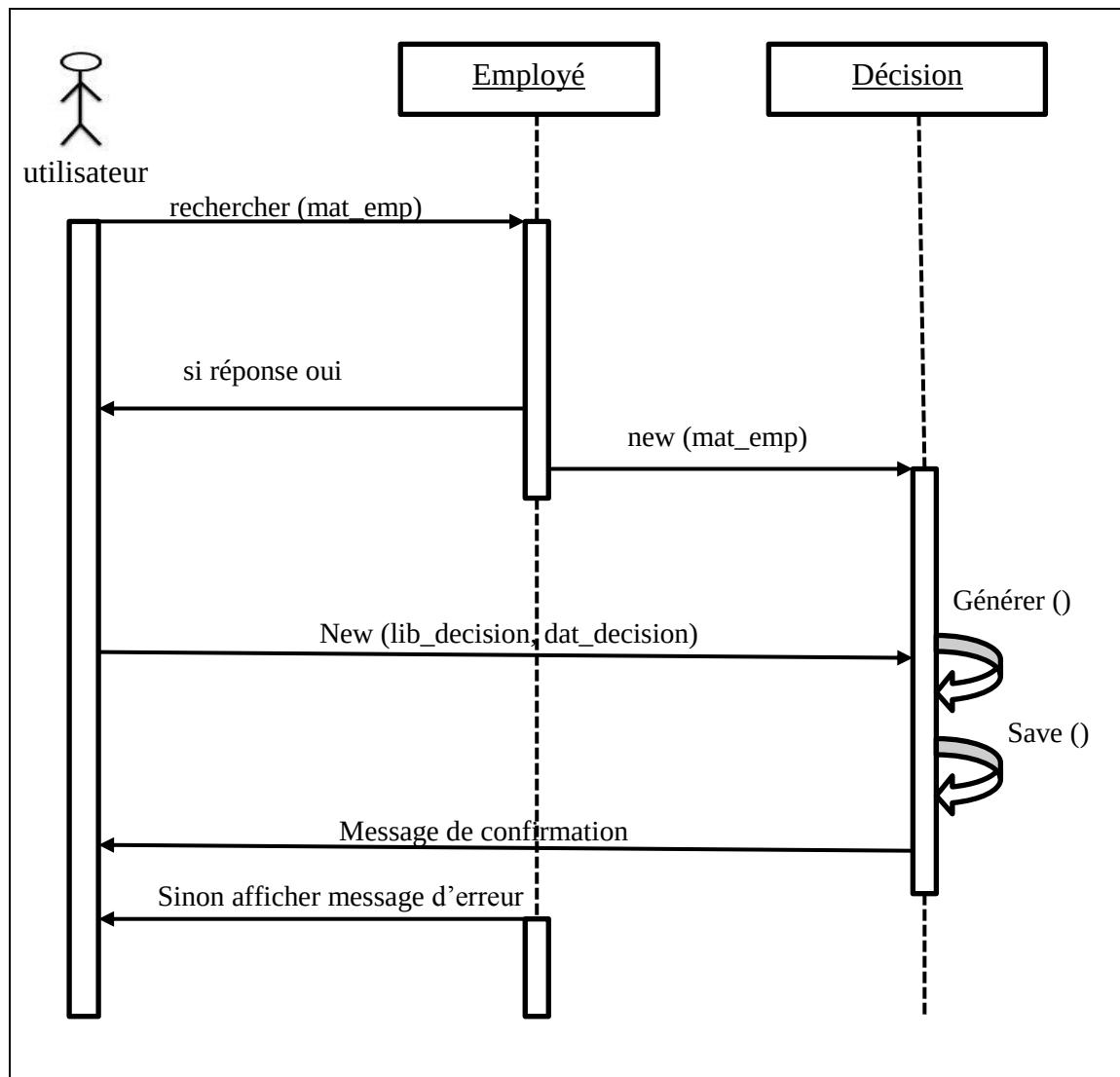


Figure: Diagramme de séquence « Ajouter décision »

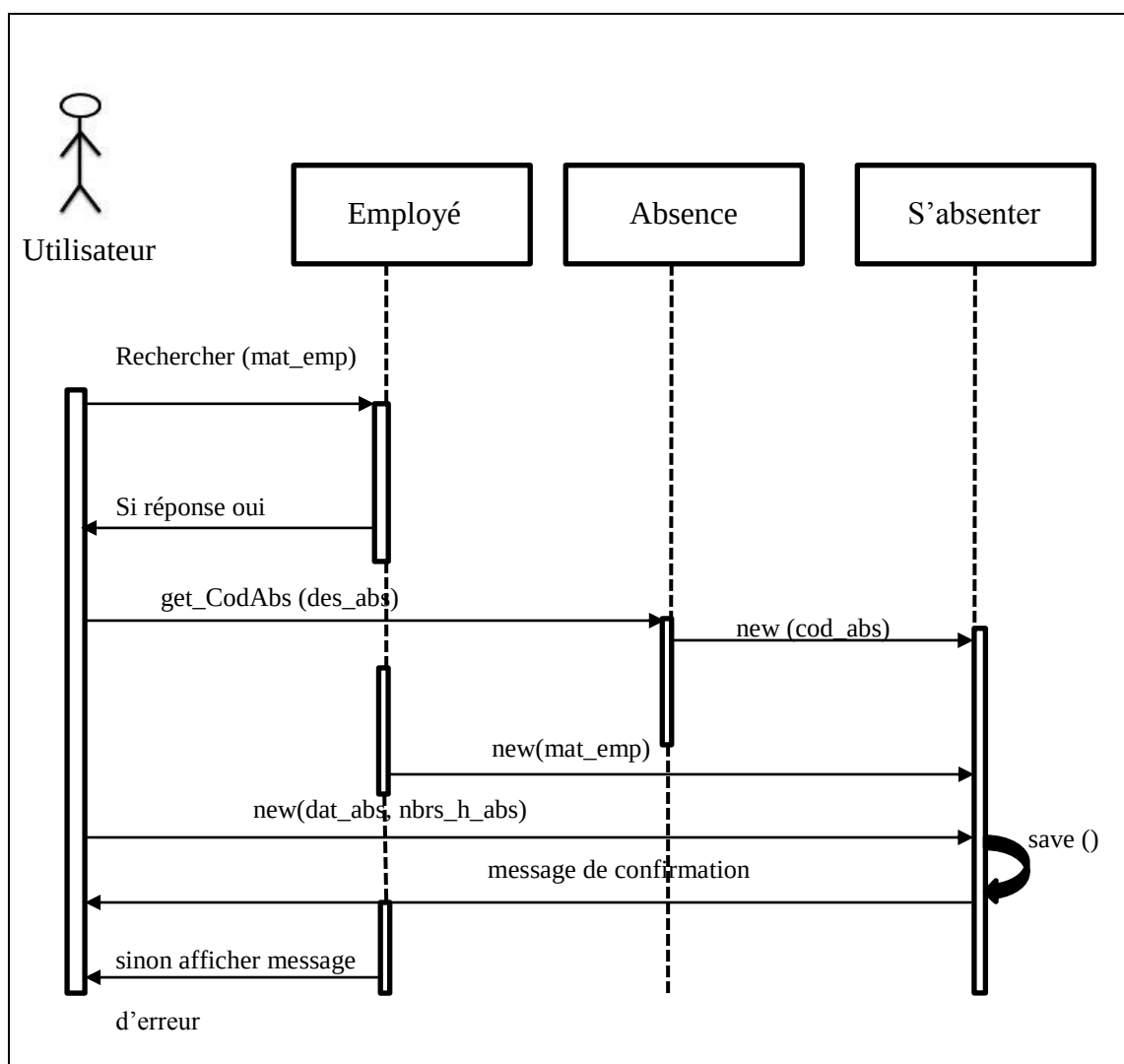


Figure: Diagramme de séquence « Ajouter absence ».

CONCEPTION DETAILLÉE

1. Conception :

La phase de conception a comme objectif de s'accorder sur le comment mais pas sur le pourquoi, en d'autre terme, elle vise à trouver des solutions informatiques et techniques pour mettre en œuvre et construire le système analysé au cours des phases précédentes. Elle doit permettre d'élaborer les différentes couches de système analysé au cours de différentes couches du système et leurs interactions, d'abord à un niveau plus général puis à un niveau plus détaillé. Cette phase succède à la phase d'analyse, qui se base sur les différents modèles de l'analyse (modèle statique et le modèle dynamique).

➤ Affinage de diagrammes des classes :

Dans cette étape on va procéder à l'affinage des classes, il se fait principalement par l'ajout des méthodes à chaque classe.

➤ Conception des méthodes :

Les méthodes d'une classe sont déduites du modèle d'analyse dynamique, chaque méthode est présentée dans le diagramme de collaboration et le diagramme de séquence par message provenant à cette classe. Le tableau suivant représente la liste des méthodes de chaque une des classes :

➤ Les méthodes des classes :

Classe	Méthodes
Employé	New(nom_emp, pren_emp, sex_emp, dat_nais_emp, adr_emp, sit_f_emp, nbre_echelon, dat_recrut, num_ss, num_compte, cod_agen_banc, nbre_km_parcourus, grade_emp, num_tel_emp), Générer (), save(), getCod_dip(nom_dip) , set_inf(nom_emp, pren_emp, sex_emp, dat_nais_emp, adr_emp, sit_f_emp, nbre_echelon, dat_recrut, num_ss, num_compte, cod_agen_banc, nbre_km_parcourus, grade_emp, num_tel_emp), set_inf(grade_emp), rechercher(mat_emp, nom_emp, pren_emp, sex_emp, dat_nais_emp, sit_f_emp, nbre_echelon, dat_recrut, nbre_km_parcourus, grade_emp), recherche(mat_emp), display(mat_emp, nom_emp , pren_emp, sex_emp, dat_nais_emp, adr_emp, sit_f_emp, nbre_echelon, dat_recrut, num_ss, num_compte, cod_agen_banc, nbre_km_parcourus, grade_emp, num_tel_emp), delete(mat_emp, nom_emp, pren_emp, sex_emp, dat_nais_emp, adr_emp, sit_f_emp, nbre_echelon, dat_recrut, num_ss, num_compte, cod_agen_banc, nbre_km_parcourus, grade_emp, num_tel_emp).
Enfant	New (mat_emp), new (pre_enf, dat_nai_enf, sex_enf, cod_scol_enf), save (), set_inf (pre_enf, dat_nais_enf, sex_enf, cod_scol_enf), set_inf (mat_emp), display (mat_emp, pre_enf, dat_nais_enf, sex_enf, cod_scol_enf), delete (cod_enf, mat_emp, pre_enf, dat_nais_anf, sex_enf, cod_scol_enf)
Conjoint	New (mat_emp), new (nom_conj, pren-conj, dat_nais_conj, sit_prof_conj), save (), set_inf(nom_conj, pren_conj, dat_nais_conj, sit_prof_conj), display(mat_emp, nom_conj, pre_conj, dat_nais_conj, sit_prof_conj), delete(cod_conj, mat_emp, nom_conj,

	pre_conj, dat_nais_conj, sit_prof_conj), set_inf(mat_emp)
Absence	getCod_abs(des_abs), afficher(), display(des_abs), getDes_abs(cod_abs)
Agence Bancaire	new(cod_agen_banc), save(), set_inf(cod_agen_banc), new(des_agen_banc, adr_agen_banc, cod_mod_reg), set_inf(des_agen_banc, adr_agen_banc, cod_mod_reg), save(), display(cod_agen_banc, des_agen_banc, adr_agen_banc), delete(cod_agen_banc, des_agen_banc, adr_agen_banc)
Contrat	new (mat_emp), new (des_contrat, dur_contrat, dur_con, dat_contrat) , générer (), set_inf (des_contrat, dur_contrat, dur_con, dat_contrat), set_inf(mat_emp), rechercher (cod_contrat), save(), display(mat_emp, cod_contrat, des_contrat, dur_contrat, dur_con, dat_contrat), delete(cod_contrat, mat_emp, des_contrat, dur_contrat, dur_con, dat_contrat)
Congé	getCod_conge(des_conge), afficher(), display(des_conge), getDes_conge(cod_conge)
Demande	New (mat_emp), new(objet_demande, dat_demande), générer(), save(), set_inf(objet_demande, dat_demande, mat_emp), rechercher(cod_demande), display(cod_demande, mat_emp, objet_demande, dat_demande), delete(cod_demande, mat_emp, objet_demande, dat_demande)
Départ	New (mat_emp), new(nat_depart, dat_cessa_paie, motif_depart, dat_depart), générer(), set_inf(nat_depart, dat_cessa_paie, motif_depart, dat_depart), rechercher (cod_depart), save(), display(cod_depart, mat_emp, nat_depart, dat_cessa_paie, motif_depart, dat_depart), delete (cod_depart, mat_emp, nat_depart, dat_cessa_paie, motif_depart, dat_depart)
Décision	new (mat_emp) ; New (lib_decision, dat_decision), Générer (), set_inf (lib_decision, dat_decision, mat_emp), save (), rechercher (cod_decision), display(mat_emp, lib_decision, dat_decision), delete (mat_emp, cod_decision, mat_emp, lib_decision, dat_decision)
Rubrique paie	
Sanction	get_CodSanct (des_sanct), afficher (), display (des_sanct)
Mode règlement	getCod_mod_reg (des_mod_reg), afficher (), getDes_mod_reg(cod_mod_reg), display(des_mod_reg), getCod_mod_reg(des_mod_reg)
Période d'essai	getCod_period_essai(apprecia_finale), afficher(), display (apprecia_finale), getApprecia_finale(cod_period_essai)
Heure supplémentaire	getCod_h_suppl (des_h_suppl), afficher (), display (des_h_suppl), getDes_h_suppl (cod_h_suppl)
Arrêt	getCod_arret (des_arret), afficher (), display (des_arret), getDes_arret (cod_arret)
Diplôme	Afficher (), get_CodDip (des_dip), display (des_dip)
Intérim	new(mat_emp), new(des_interim, dat_interim), set_inf(mat_emp), Générer(), set_inf(des_interim, dat_interim), Rechercher (cod_interim), save(), display(mat_emp, des_interim, dat_interim), delete(cod_interim, mat_emp, des_interim, dat_interim)
Réintégré	New (mat_emp), new (motif_reinteg, dat_reinteg, mat_emp), générer (), recherché (Cod_reinteg), save (), set_inf (motif_reinteg, dat_reinteg), display (cod_reinteg, mat_emp, motif_reinteg, dat_reinteg), delete (cod_reinteg, mat_emp, motif_reinteg, dat_reinteg)

Promotion	New (mat_emp), new (des_promotion, dat_promotion), Générer (), Save(), rechercher (cod-promotion), display (cod_promotion, mat_emp, des_promotion, dat_promotion), set_inf(des_promotion, dat_promotion), delete(cod_promotion, mat_emp, des_promotion, dat_promotion).
-----------	---

Tableau : les méthodes des classes d'objets.

➤ **Les méthodes des classes d'association :**

Classe	Méthode
Effectuer	New(cod_h_suppl), new(mat_emp), set_inf(cod_h_suppl), new(date_h_suppl, nbre_h_suppl), save(), set_inf(mat_emp); set_inf(date_h_suppl, nbr_h_suppl), delete(cod_h_suppl, mat_emp, dat_h_suppl, nbre_h_suppl)
Est paye	
Est-en	New(cod_arret), new(mat_emp), set_inf(cod_arret), new (date_arret, dure_arret), save(), set_inf(mat_emp), set_inf(date_arret, dure_arret), delete(cod_arret, mat_emp, date_arret, dure_arret)
Soumettre	New(cod_period_essai), new(mat_emp), set_inf(cod_period_essai), new(date_debut, duree_p_essai), save(), set_inf(mat_emp), set_inf(date_debut, duree_p_essai), delete(cod_period_essai, mat_emp, date_debut, duree_p_essai)
Subir	New(cod_sanct), new(mat_emp), set_inf(cod_sanct), new(dat_sanct, motif_sanct), save(), set_inf(mat_emp), set_inf(dat_sanct, motif_sanct), delete(cod_sanct, mat_emp, dat_sanct, motif_sanct)
prendre	New (cod_conge), new (mat_emp), set_inf (cod_abs), new (nbre_jour_conge, dat_conge), save (), set_inf (mat_emp), set_inf (nbre_jour_conge, dat_conge), delete (cod_conge, mat_emp, nbre_jour_conge, dat_conge)
s'absenter	New(cod_abs), new(mat_emp), new (dat_abs, nbre_h_abs), save(), set_inf(cod_abs), set_inf(mat_emp), set_inf(dat_abs, nbrs_h_abs), get_CodAbs(mat_emp), delete(cod_abs, mat_emp, dat_abs, nbrs_h_abs)
possède	New(cod_dip), New(dat_obt_dip), New(mat_emp), save(), set_inf(cod_dip), set_inf(dat_obt_dip), get_CodDip(mat_emp), delete(mat_emp, cod_dip, dat_obt_dip)

Est inscrit	
Faire	
Correspond	
Concerner	
Avoir	
Lier	
Etre	
Obtenir	
Partir	

Tableau : les méthodes des classes d'association

➤ **Le diagramme des classes :**

Après l'ajout des méthodes aux classes, on sort avec des classes complètes ; chaque classe contient des attributs et des méthodes.

Le tableau suivant donne la liste des classes de notre modèle :

Classe	Attributs	Méthodes
Employé	Mat_emp, nom_emp, pren_emp, num_compte, sex_emp, adr_emp, dat_nais_emp, sit_f_emp, dat_recrut, nbre_echelon, num_ss, grade_emp, nbre_km_parcourus, poste_emp, num_tel_emp,	New(nom_emp, pren_emp, sex_emp, dat_nais_emp, adr_emp, sit_f_emp, nbre_echelon, dat_recrut, num_ss, num_compte, cod_agen_banc, nbre_km_parcourus, grade_emp, num_tel_emp),.....
Utilisateur	Id_user, Login_user, password_user, etat_user, Date_cre_user, nom_user, Pre_user, Email_user, Lieu_naiss_user, Adr_user, Tel_user, Date_naiss_user	
Rôle	Id_role, Nom_role	
Privilège	Id_privilege, val_privilege, nom_privilege	
Enfant	Cod_enf, Pre_enf, Dat_nais_enf, Sex_enf, Cod_scol_enf	New (mat_emp), set_inf(mat_emp), ...etc
Conjoint	Cod_conj, Nom_conj, pren_conj, Dat_nais_conj, sit_prof_conj	New (mat_emp), set_inf(mat_emp), ...etc
Absence	Cod_abs, Des_abs	getCod_abs(des_abs), afficher(),.....etc
Agence Bancaire	Cod_agen_banc, Des_agen_banc, Adr_agen_banc	New(cod_agen_banc), save(), set_inf(),.....etc
Contrat	Cod_contrat, Des_contrat, Dur_contrat, Dat_contrat	New (mat_emp), Générer (), ...etc

Congé	Cod_conge, Des_conge	Afficher (), display (des_conge),.....etc
Demande	Cod_demande, Dat_demande, Objet_demande	New (mat_emp), générer (), Save (),.....etc
Départ	Cod_depart, Nat_depart, Dat_cessa_paie, Motif_depart, Dat_depart	New (mat_emp), générer(), Save(),.....etc
Décision	Cod_decision, Lib_decision, Dat_decision	New (mat_emp), New(lib_decision, dat_decision), Générer (),.....etc
Rubrique paie	Cod_rub_paie, Des_rub_paie, Genre_rub_paie, Type_rube_paie, Signe_rub_paie	
sanction	Cod_sanct, Des_sanct	get_CodSanct (des_sanct), afficher (),.....etc
Mode règlement	Cod_mod_reg, Des_mod_reg	Afficher (), display (des_mod_reg),.....etc
Période d'essai	Cod_period_essai, Apprecia_finale	Display (apprecia_finale), afficher (),.....etc
Heure supplémentaire	Cod_h_suppl, des_h_suppl, nbre_h_suppl	Afficher (), display (des_h_suppl),.....etc
Arrêt	Cod_arret, Lib_arret	Display (des_arret), afficher (),.....etc
Diplôme	Cod_dip, Des_dip	get_CodDip (des_dip), display (des_dip).....etc
Intérim	Cod_interim, Des_interim, Dat_interim	New (mat_emp), New (des_interim, dat_interim),....etc
Réintégré	Cod_reinteg, mtif_reinteg, Dat_reinteg	Save (), set_inf (motif_reinteg, dat_reinteg)...etc
Promotion	Cod_promotion, Des_promotion, Dat_promotion	Rechercher (cod_promotion), Générer (), save(),.....etc

Tableau : Les méthodes et les attributs des classes

➤ Liste des classes d'association :

Classe	Attributs	Méthodes
Effectuer	Date_h_suppl	Save (), set_inf (mat_emp), ...etc
Est paye	Total_gain, Mois_ann, Total_retenir, Net_payer, Nbre_rub_paie, Taux_rub_paie	
Est_en	Dure_arret, Dat_arret	New (cod_arret), new (mat_emp), set_inf (cod_arret),.....etc
Partir		

Soumettre	Date_début, Durée	New (cod_period_essai), new (mat_emp),.....etc
Subir	Dat_sanct, Motif_sanct	New (cod_sanct), new (mat_emp),.....etc
Prendre	Nbre_jour_conge, Dat_conge	New (cod_conge), new (mat_emp), set_inf (cod_abs),.....etc
S'absente	Dat_abs, nbre_h_abs	New (cod_abs), new (mat_emp),.....etc
Possède	Date_obt_dip	New (mat_emp), Save (),etc
Est inscrit		
Faire		
Correspond		
Concerner		
Avoir		
Lier		
Etre		
Obtenir		

Tableau : les méthodes et les attributs des associations

➤ **Le passage vers le modèle relationnel :**

Pour passer du model classe objet vers le modèle relationnel, on va utiliser les règles suivantes :

Modèle objet	Modèle relationnel
Classe	Table
Attribut	Colonne
Méthode	Ne figure pas
Association à ordre un à plusieurs	Clé étrangère
Association à ordre plusieurs à plusieurs	Table

Tableau : la méthode de passage vers le modèle relationnel

➤ **Le modèle relationnel :**

Employé (Mat_emp, Nom_emp, Pren_emp, Sex_emp, Dat_nais_emp, adr_emp, sit_f_emp, Dat_recrut, Num_ss, Num_compte, Cod_agen_banc*, Nbre_km_parcourus, Nbre_echelon, Grade_emp, Poste_emp, Num_tel_emp).

Utilisateur (Id_user, Login_user, Password_user, Etat_user, Date_cre_user, Nom_user, Pre_user, Email_user, Lieu_naiss_user, Adr_user, Tel_user, Date_naiss_user, id_role*)

Rôle (Id_role, Nom_role)

Privilege (Id_privilege, Val_privilege, Nom_privilege, Id_role*)

Enfant (Cod_enf, pre_enf, Dat_nai_enf, Sex_enf, Cod_scol_enf, Mat_emp*)

Conjoint (Cod_conj, Nom_conj, Pre_conj, Dat_nais_conj, Sit_prof_conj, Mat_emp*)

Absence (Cod_abs, Des_abs)

Agence Bancaire (Cod_agen_banc, Des_agen_banc, Cod_mod_reg*, Adr_agen_banc)

Contrat (Cod_contrat, Des_contrat, Dur_contrat, Dat_contrat, Mat_emp*)

Congé (Cod_conge, Des_conge)

Demande (Cod_demande, Objet_demande, dat_demande, Mat_emp*)

Départ (Cod_depart, Nat_depart, Dat_cessa_paie, Motif_depart, Dat_depart, Mat_emp*)

Décision (Cod_decision, Lib_decision, Dat_decision, Mat_emp*)

Rubrique paie (Cod_rub_paie, Des_rub_paie, Genre_rub_paie, Type_rub_paie, Signe_rub_paie)

Sanction (Cod_sanct, Des_sanct)

Mode règlement (Cod_mod_reg, Des_mod_reg)

Période d'essai (Cod_sanct, Des_sanct)

Heure supplémentaire (Cod_h_suppl, Des_h_suppl)

Arrêt (Cod_arret, Lib_arret).

Diplôme (Cod_dip, Des_dip)

Intérim (Cod_interim, Mat_emp, Dat_interim, Des_interim)

Réintégration (Cod_reinteg, Mat_emp*, Motif_Reinteg, Dat_reinteg)

Promotion (Cod_promotion, Mat_emp*, Des_promotion, Dat_promotion)

Effectuer (Cod_h_suppl*, Mat_emp*, Date_h_suppl, Nbre_h_suppl).

Est paye (Cod_rub_paie*, Mat_emp, Total_gain, Total_retenir, Nbre_rub_paie, Taux_rub_paie, Mois_ann, Net_payer)

Est_en (Mat_emp*, Cod_arret*, Dure_arret, Date_arret)

Soumettre (Cod_period_essai*, Mat_emp*, Date_début, Durée_p_essai)

Subir (Cod_sanct*, Mat_emp*, Dat_sanct, Motif_Sanct)

Prendre (Cod_conge*, Mat_emp*, Nbre_jour_conge, Dat_conge)

S'absente (Cod_abs*, Mat_emp*, Dat_abs, nbrs_h_abs).

Possède (Cod_dip*, Mat_emp*, Date_obt_dip)

Remarque :

- Attribut représente la clé primaire.
- Attribut* représente la clé étrangère.

Conclusion

Dans cette partie nous avons réalisé la conception de notre application, nous avons commencé par faire l'étude préliminaire et l'étude fonctionnelle, ce qui nous a permis de faire la description de notre système, ensuite on est passé à l'analyse statique et dynamique, et à la fin on a fait la conception pour pouvoir mettre en œuvre notre base de données.

Dans la partie suivante nous allons passer à la partie réalisation, pour faire le développement de notre application.

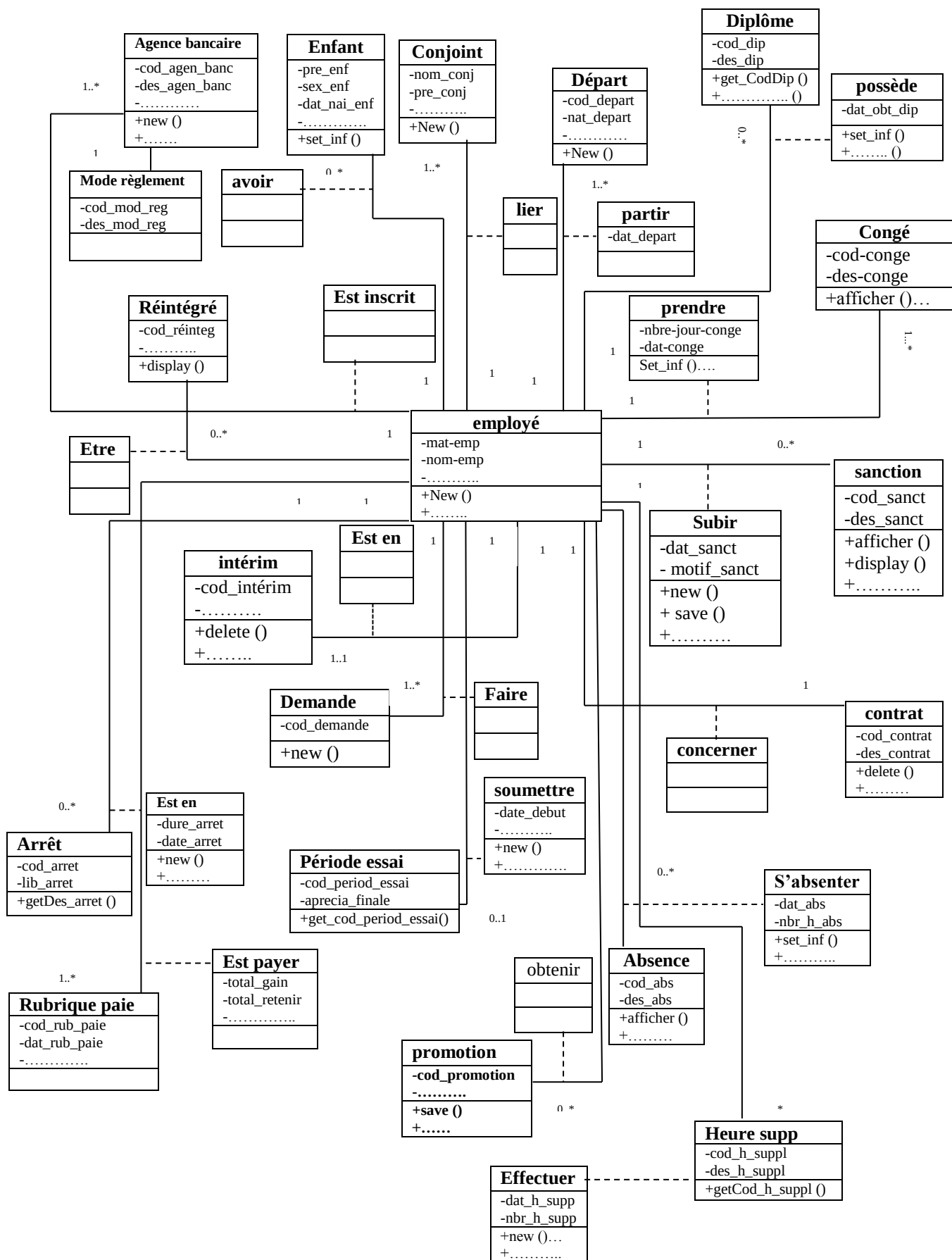


Figure : diagramme des classes

Introduction :

Nous allons présenter dans cette partie le développement de notre travail, on va présenter la plateforme de développement de notre travail ensuite nous allons présenter quelques interfaces de notre application.

1. Environnement de programmation :**1.1. Langage de programmation Java :**

Apparu fin 1995 début 1996 et développé par Sun Microsystems Java s'est très rapidement taillé une place importante en particulier dans le domaine de l'internet et des applications client-serveur.

Destiné au départ à la programmation de centraux téléphoniques sous l'appellation de langage "oak", la société Sun a eu l'idée de le recentrer sur les applications de l'internet et des réseaux. C'est un langage en évolution permanente Java 2 est la version stabilisée de java fondée sur la version initiale 1.2.2 du JDK (Java Development Kit de Sun : <http://java.sun.com>)

Les objectifs de java sont d'être multiplateformes et d'assurer la sécurité aussi bien pendant le développement que pendant l'utilisation d'un programme java. Il est en passe de détrôner le langage C++ dont il hérite partiellement la syntaxe mais non ses défauts. Comme C++ et Delphi, java est algorithmique et orienté objet à ce titre il peut effectuer comme ses compagnons, toutes les tâches d'un tel langage (bureautiques, graphiques, multimédias, bases de données, environnement de développement, etc...). Son point fort qui le démarque des autres est sa portabilité due (en théorie) à ses bibliothèques de classes indépendantes de la plate-forme, ce qui est le point essentiel de la programmation sur internet ou plusieurs machines dissemblables sont interconnectées. La réalisation multiplateforme dépend en fait du système d'exploitation et de sa capacité à posséder des outils de compilation et d'interprétation de la machine virtuelle Java. Actuellement ceci est totalement réalisé d'une manière correcte sur les plates-formes Windows et Solaris, un peu moins bien sur les autres semble-t-il.

En Java l'on développe plusieurs genres de programmes :

- Les applications qui sont des logiciels classiques s'exécutant directement sur une plate-forme spécifique soit à travers une machine virtuelle java soit directement en code exécutable par le système d'exploitation. (Code natif).
- les applets qui sont des programmes java insérés dans un document HTML s'exécutant à travers la machine virtuelle java du navigateur lisant le document HTML.
- J2SE ou Java 2 Standard Édition : permet de développer des applications dites "client".
- J2EE ou Java 2 Enterprise Édition : permet de développer des applications web en Java.
- J2ME ou Java 2 Micro Édition : permet de développer des applications pour appareil portable, comme des téléphones portables, des PDA...etc.

Les outils de développement : pour satisfaire les besoins de l'application à réaliser nous avons utilisé les outils suivants :

1.2. L'IDE NETBEANS :

NetBeans est un environnement de développement intégré, Il est placé en open source depuis juin 2000 par Sun. La plateforme NetBeans est un outil très puissant pour la réalisation d'applications swing java.

➤ **Les principaux langages que NetBeans supporte sont :**

- ✓ [Java](#) (Java SE, Java ME, Java FX, Java EE), [Javadoc](#) ;
- ✓ [Ruby](#)
- ✓ [Groovy](#)
- ✓ [PHP](#)
- ✓ [JavaScript](#)
- ✓ [C](#), [C++](#)
- ✓ [Python](#)
- ✓ [HTML](#)
- ✓ [XML](#)
- ✓ [DTD](#)
- ✓ [CSS](#)
- ✓ [JSP](#), [JSF](#)

➤ **NetBeans et les Bases de données :**

NetBeans comprend un explorateur de bases de données qui supporte toutes les bases relationnelles pour lesquelles un connecteur JDBC existe (selon les versions des gestionnaires de bases de données): Java DB (Derby) MySQL, PostgreSQL, Oracle,...etc. L'explorateur comprend un éditeur de requêtes, un gestionnaire intégré de bases de données MySQL.

➤ **NetBeans et applications sur serveurs :**

NetBeans supporte une importante variété d'environnements pour l'exécution d'applications web et Java EE : Java Server Pages (JSP), Java Server Faces(JSF), Enterprise JavaBeans (EJB 2.1, EJB 3, EJB 3.1), Apache Struts, Spring Web MVC, Hibernate... Il supporte les standards Java EE 6, Java EE 5, J2EE 1.4, Java Persistence API (JPA 2.0), Java Servlet API.

Il permet le déploiement d'applications Java et d'Applets via Java Web Start (JNLP).

NetBeans intègre les fonctions de Hudson (Maven et projets Java SE pour Ant) et de Maven (créer, exécuter, tester et déboguer des projets Maven Apache).

Comme le montre la figure suivante on peut voir un projet J2EE sur l'IDE NetBeans :

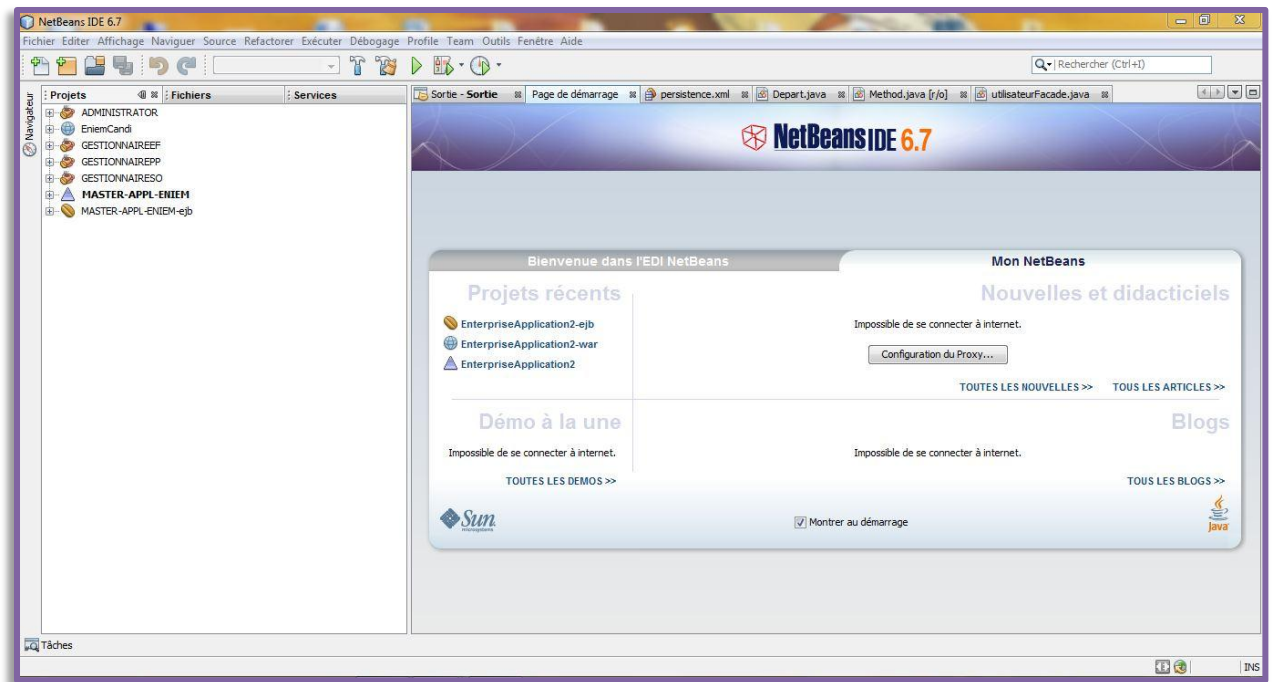


Figure : fenêtre d'accueil de l'IDE NetBeans

2. Réalisation

2.1. Présentation de la base de données :

Les tables de la base de données ont été créées à l'aide de l'outil PostgreSQL (Annexe B). Après la création d'un espace base de données, on est passé à la création des objets de la base et la gestion des contraintes d'intégrité.

La figure suivante montre l'interface graphique de pgAdmin dans laquelle sont montrées les tables ainsi que le schéma de leur appartenance.

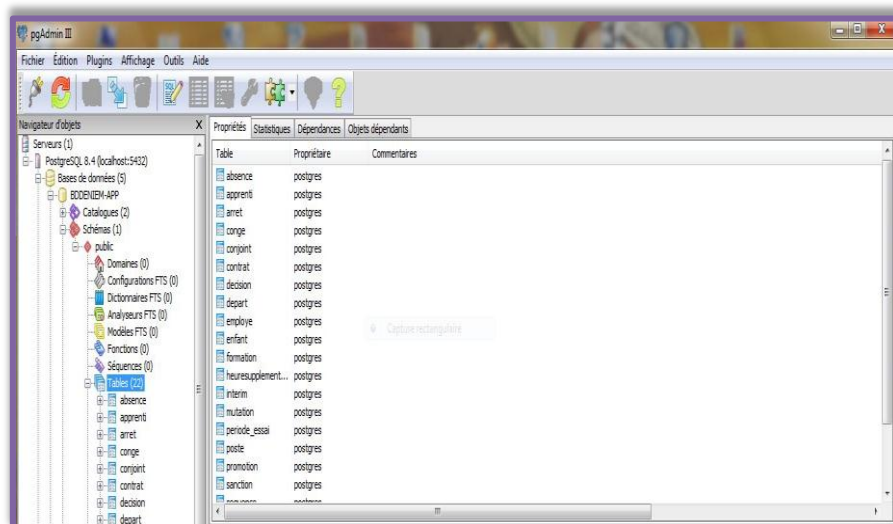


Figure : présentation de la base de données

2.2. Fonctionnement de l'application :

2.2.1. Page d'accueil Administrateur :

L'administrateur saisit son login ainsi que son mot de passe pour s'identifier, une fois que qu'il est connecté il accède à la page d'accueil administrateur, qui est composé de différents onglets (ajouter utilisateur, modifier utilisateur, consulter utilisateur, supprimer utilisateur), et une table ou il peut visualiser tous les utilisateurs. Comme le montre la figure suivante :



Figure : page d'accueil Administrateur.

2.2.2. Connexion du gestionnaire emploi et formation :

Le gestionnaire se rend à l'espace emploi et formation, la fenêtre d'authentification suivante s'affiche :

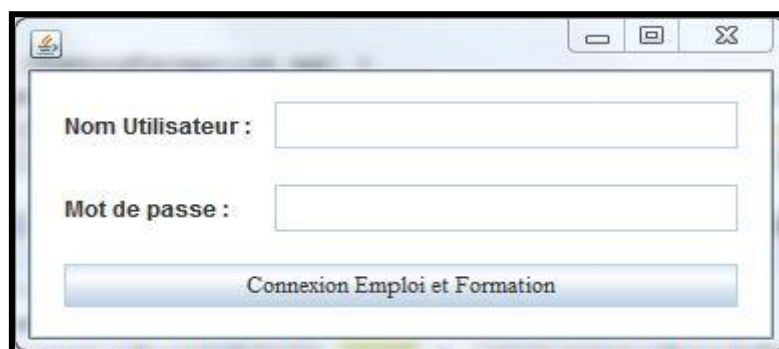


Figure : fenêtre de connexion à l'espace emploi et formation.

2.2.3. Modifier le mot de passe du gestionnaire emploi et formation :

Une fois que le gestionnaire est sur la page d'accueil emploi et formation, il se rend à l'onglet modifier mot de passe, il clique dessus et il obtient la fenêtre montrée ci-dessous ou il remplit les champs avec son nouveau mot de passe.



Figure : page de modification du mot de passe du gestionnaire emploi et formation

2.2.4. Ajouter un contrat :

Le gestionnaire emploi et formation clique sur l'onglet contrat, ce qui l'amène à la page contrat, il clique sur le bouton ajouter contrat et il obtient la fenêtre suivante, il remplit les différents champs et il clique sur ajouter ce qui fait qu'il enregistre le nouveau contrat.

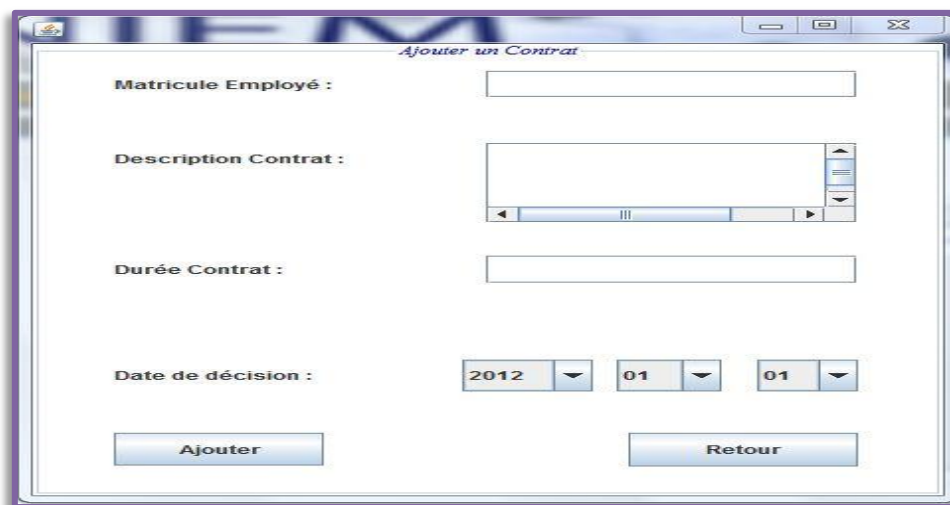


Figure : fenêtre ajouter contrat.

CONCLUSION GENERALE

Conclusion générale

Durant notre stage nous avons pu avoir un réel contact avec le monde du travail, ce qui nous a permis de mettre en pratique nos connaissances théoriques.

Notre travail consistait à réaliser une application 3-tiers J2EE pour la gestion des ressources humaines de l'entreprise ENIEM, ce qui nous a permis : d'acquérir une expérience dans la conception avec la méthode 2TUP et le langage UML.

D'approfondir nos connaissances dans le réseau, la programmation Java, les bases de données, le SGBD PostgreSQL ainsi que dans l'architecture logicielle.

Des connaissances dans le domaine des ressources humaines ainsi que la découverte de nouvelles règles de gestion.

Cependant quoi qu'elle présente différentes fonctionnalités notre application est ouverte à subir des améliorations comme par exemple son extension dans les différents départements, l'ajout d'un site web pour permettre aux candidats à l'emploi et la formation de s'inscrire et de postuler via ce site.

Enfin nous espérons que notre travail sera d'un grand intérêt pour l'ENIEM et un guide efficace pour les nouvelles promotions.

ANNEXE A

2TUP

La méthode 2TUP :

Devant le nombre de méthodes disponibles, le choix parmi elles devient difficile, beaucoup de questions peuvent se poser à un chef de projet lors d'un démarrage de projet :

Comment vais-je organiser les équipes de développement ;

Quelles tâches attribuer à qui ;

Quel temps faudrait-il pour livrer le produit ;

Comment faire participer le client au développement afin de capter les besoins de celui-ci ;

Comment éviter des dérives et de mauvaises estimations qui vont allonger les coûts et le temps de développement.

Comment vais-je procéder pour que le produit soit évolutif et facilement maintenable.

Nous pouvons citer à ce propos les méthodes de développement objet suivantes : 2TUP, RUP, XP, AUP et OpenUP.

Notre choix s'est porté vers la méthode 2TUP, du fait de son approche nouvelle, originale.

Notre projet est basé sur un processus de développement bien défini qui va de la détermination des besoins fonctionnels attendus du système jusqu'à la conception et le codage final.

Ce processus se base lui-même sur le Processus Unifié (Unified Process) qui est devenu un standard général réunissant les meilleures pratiques de développement.

Cette méthode ne se base aucunement sur un processus linéaire mais bien, sur un développement itératif et incrémental.

Nous allons d'abord définir les différents concepts qui vont être utilisés dans ce document.

1. Définition d'un processus de développement logiciel :

Un processus définit une séquence d'étapes, en partie ordonnées, qui concourent à l'obtention d'un système logiciel ou à l'évolution d'un système existant.

L'objet d'un processus de développement est de produire des logiciels de qualité qui répondent aux besoins de leurs utilisateurs dans des temps et des coûts prévisibles. (Rocques & Vallée, 2004)

2. Le processus unifié :

Le Processus Unifié (PU ou UP en anglais pour Unified Process) est une méthode de développement logiciel construite sur UML ; elle est itérative et incrémentale, centrée sur l'architecture, conduite par les cas d'utilisation et pilotée par les risques.

- **itérative et incrémentale** : la méthode est itérative dans le sens où elle propose de faire des itérations lors de ses différentes phases, ceci garantit que le modèle construit à chaque phase ou étape soit affiné et amélioré. Chaque itération peut servir aussi à ajouter de nouveaux incréments.
- **conduite par les cas d'utilisation** : elle est orientée utilisateur pour répondre aux besoins de celui-ci.
- **centrée sur l'architecture** : les modèles définissent tout au long du processus de développement vont contribuer à établir une architecture cohérente et solide.

- **pilotée par les risques** : en définissant des priorités pour chaque fonctionnalité, on peut minimiser les risques d'échec du projet.

La gestion d'un tel processus est organisée d'après les 4 phases suivantes :

1. **Préétude(Inception)** : c'est ici qu'on évalue la valeur ajoutée du développement et la capacité technique à le réaliser (étude de faisabilité).
2. **Elaboration** : sert à confirmer l'adéquation du système aux besoins des utilisateurs et à livrer l'architecture de base.
3. **Construction** : sert à livrer progressivement toutes les fonctions du système.
4. **Transition** : déployer le système sur des sites opérationnels.

Chaque phase est elle-même décomposée séquentiellement en itérations limitées dans le temps (entre 2 et 4 semaines). Le résultat de chacune d'elles est un système testé, intégré et exécutable. L'approche itérative est fondée sur la croissance et l'affinement successifs d'un système par le biais d'itérations multiples. Le système croît avec le temps de façon incrémentale, itération par itération, et c'est pourquoi cette méthode porte également le nom de développement itératif et incrémental. Il s'agit là du principe le plus important du Processus Unifié.

Ces activités de développement sont définies par 6 disciplines fondamentales qui décrivent la capture des besoins, la modélisation métier, l'analyse et la conception, l'implémentation, le test et le déploiement.

Notons que ces différentes étapes (ou disciplines) peuvent se dérouler à travers plusieurs phases.

Le processus unifié doit donc être compris comme une trame commune des meilleures pratiques de développement.

3. Le processus 2TUP

On dit de la méthode UP qu'elle est générique c.à.d. qu'elle définit un certain nombre de critères de développement, que chaque société peut par la suite personnaliser afin de créer son propre processus plus adapté à ses besoins.

C'est dans ce cadre que la société Val Tech a créé la méthode 2TUP.

2TUP signifie « 2 Track Unified Process ». C'est un processus qui répond aux caractéristiques du Processus Unifié. Le processus 2TUP apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise. En ce sens, il renforce le contrôle sur les capacités d'évolution et de correction de tels systèmes.

« 2 Track » signifient littéralement que le processus suit deux chemins. Il s'agit des « chemins fonctionnels » et « d'architecture technique », qui correspondent aux deux axes de changement imposés au système d'information.

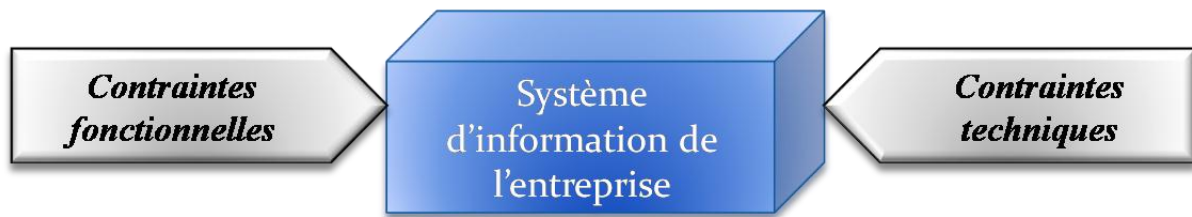


Figure : Le système d'information soumis à deux types de contraintes

✓ **La branche gauche (fonctionnelle) :**

Capitalise la connaissance du métier de l'entreprise. Elle constitue généralement un investissement pour le moyen et le long terme.

Les fonctions du système d'information sont en effet indépendantes des technologies utilisées.

Cette branche comporte les étapes suivantes :

- La capture des besoins fonctionnels, qui produit un modèle des besoins focalisé sur le métier des utilisateurs.
- L'analyse.

✓ **La branche droite (architecture technique) :**

Capitalise un savoir-faire technique. Elle constitue un investissement pour le court et moyen terme. Les techniques développées pour le système peuvent l'être en effet indépendamment des fonctions à réaliser.

Cette branche comporte les étapes suivantes :

- La capture des besoins techniques.
- La conception générique.

✓ **La branche du milieu :**

A l'issue des évolutions du modèle fonctionnel et de l'architecture technique, la réalisation du système consiste à fusionner les résultats des 2 branches. Cette fusion conduit à l'obtention d'un processus en forme de Y.

Cette branche comporte les étapes suivantes :

- La conception préliminaire.
- La conception détaillée.
- Le codage.
- L'intégration.

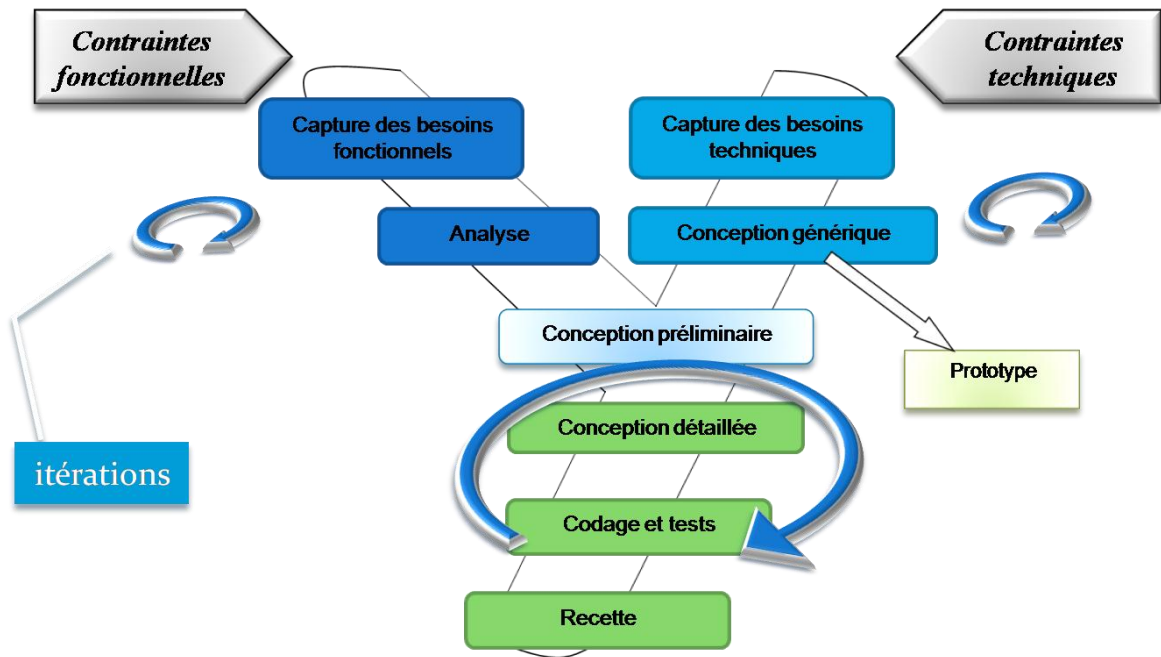


Figure : Le processus de développement en Y .

4. Un processus de modélisation avec UML

Le processus 2TUP s'appuie sur UML tout au long du cycle de développement, car les différents diagrammes de ce dernier permettent par leur facilité et clarté, de bien modéliser le système à chaque étape.

- **« Unified Modeling Language » :** UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier, concevoir des solutions et communiquer des points de vue. (Pitman, 2006)

UML unifie à la fois les notations et les concepts orientés objet. Il ne s'agit pas d'une simple notation, mais les concepts transmis par un diagramme ont une sémantique précise et sont porteurs de sens au même titre que les mots d'un langage, c'est pour ça qu'UML est présenté parfois comme une méthode alors qu'il ne l'est absolument pas.

UML unifie également les notations nécessaires aux différentes activités d'un processus de développement et offre, par ce biais, le moyen d'établir le suivi des décisions prises, depuis la définition des besoins jusqu'au codage. (Roques, 2006)

Voici une présentation rapide des différents diagrammes UML qui vont être utilisés tout au long du projet :

1. **Le diagramme des cas d'utilisation :** représente la structure des fonctionnalités nécessaires aux utilisateurs du système. Il est normalement utilisé lors des étapes de capture des besoins fonctionnels et techniques.

2. **Le diagramme d'activités** : représente les règles d'enchaînement des activités et actions dans le système. Il peut être assimilé comme un algorithme mais schématisé.
3. **Le diagramme de packages** : présent depuis UML 2.0, ce diagramme modélise des catégories cohérentes entre elles, pour un souci de partage des rôles.
Correspond à l'étape de modélisation des différents scénarios d'un cas d'utilisation.
4. **Le diagramme de classes** : sûrement l'un des diagrammes les plus importants dans un développement orienté objet. Sur la branche fonctionnelle, ce diagramme est prévu pour développer la structure des entités manipulées par les utilisateurs.
En conception, le diagramme de classes représente la structure d'un code orienté objet.
5. **Le diagramme de séquence** : représente les échanges de messages entre objets, dans le cadre d'un fonctionnement particulier du système.
6. **Le diagramme d'états** : représente le cycle de vie d'un objet. Il spécifie les états possibles d'une classe et leur enchaînement.
Ce diagramme est utilisé lors des étapes d'analyse et de conception.

ANNEXE B

POSTGRESQL

Présentation de SGBD postgresql

Introduction :

Au cours des dernières années, les bases de données ainsi que les SGBDs ont connu un développement considérable, au point qu'ils jouent un rôle très important dans notre vie quotidienne ainsi que dans les entreprises.

1. Définitions :

1.1. Les bases de données : [GIL 02]

Une base de données représente un ensemble de données d'une entreprise (ou d'une organisation) mémorisé par un ordinateur qui est utilisé par de nombreuses personnes et dont l'organisation est régi par un modèle de données. L'objectif de création d'une base de données est qu'elle doit permettre de retrouver de l'information par son contenu en utilisant les critères de recherches.

1.2. Systèmes de gestion de bases de données (SGBD) :

Un SGBD (système de gestion de base de données) est un logiciel qui permet aux utilisateurs de définir, créer et maintenir une base de données, et qui fournit un accès contrôlé aux données.

1.3. Les types de SGBD :

Aujourd'hui il existe trois catégories principales de SGBD :

- Centralisées : client/serveur.
- Réparties : base de données en réseau.
- Base de données locales ou personnalisées.

1.4. Rôles de SGBD :

Un SGBD permet de gérer les données structurées et persistantes de façon intégré, fiable, concurrente et efficace :

- **Structuration des données** : le SGBD doit offrir un système de typage.
- **Persistance des données** : la durée de vie des données doit être supérieure à la durée des programmes qui les manipulent
- **Intégrité des données** : le SGBD assure une vue cohérente des données, en des accès concurrents.
- **Fiabilité des données** : Les SGBD ont pour rôle de :
 - ✓ Vérifier les contraintes des données (intégrité référentielle, réflexe, etc).
 - ✓ Gérer les transactions (atomicité des transactions), et sécurité (mot de passe, etc.).
- **Accès efficace** : le SGBD intègre des techniques spécifiques pour avoir de bonnes performances :
 - ✓ Index, hash_coding.
 - ✓ Regroupement des bases de données sur disque.

- ✓ Organisation des requêtes.
- ✓ Langage des requêtes comme SQL
- **Indépendance physique/logique :**
 - ✓ **Indépendance physique :** le niveau physique peut être modifié indépendamment du niveau conceptuel, cela signifie que tous les aspects matériel de base de données n'apparaissent pas pour l'utilisateur, il s'agit simplement d'une structure transparentes de représentation des informations.
 - ✓ **Indépendance logique :** le niveau conceptuel doit pouvoir être modifié sans remettre en cause le niveau physique, c.-à-d. que l'administrateur de la base doit pouvoir modifier sans que les programmes d'application soient changés.

1.5. Les modèles de SGBD : [GIL 02]

- **Modèle de SGBD hiérarchique :**

Une base de données hiérarchique est une base de données dont le système de gestion lie les enregistrements dans une structure arborescente ou chaque enregistrement n'a qu'un seul possesseur. Rapide, puisque les liens sont simples, et peu flexible comme modèle reflétant la réalité mais ce genre de BD n'est plus développé, sauf dans des rares occasions.
- **Modèle de SGBD réseau :**

Le modèle réseau est une manière de représenter graphiquement les données dans le cadre d'une base de données. Ce modèle est en mesure de lever de nombreuses difficultés du modèle hiérarchique grâce à la possibilité d'établir des liaisons de type 1-n en définissant des associations entre tous les types d'enregistrements.
- **Modèle de SGBD relationnel :**

Dans le modèle relationnel, les données sont enregistrées dans des tableaux à deux dimensions (lignes appelées enregistrement ou tuples et colonnes appelés champs). La manipulation de ces données se fait selon la théorie mathématique des ensembles. Les SGBD relationnels sont une application directe de ce modèle.
- **Modèle de SGBD orienté objet :**

Dans le modèle orientée objet, les données sont stockées dans des structures appelées objets.

 - ✓ Les objets ont une existence persistante indépendante des données (avec leur identité, leur fonction, leur classe, etc.)
 - ✓ Les données peuvent être accédé uniquement par un objet, selon les méthodes qui lui ont été attribuées.
 - ✓ Une hiérarchie entre objets peut être définie au besoin

1.6. Structure d'un SGBD : un SGBD est composé de trois couches :

- a. **Le système de gestion de fichier :**
 - ✓ Gère les stockages internes des données.
 - ✓ Le SGF est indépendant du matériel utilisé.
- b. **Le SGBD interne :**
 - ✓ Gère le placement et l'assemblage des données (gestion des liens et gestion des accès rapides).
- c. **Le SGBD externe :**
 - ✓ Gère la présentation et la manipulation des données.
 - ✓ Gère le langage des requêtes et des outils de représentation (état, forme,...)

2. PostgreSQL :

2.1.Définition :

PostgreSQL est un système de gestion de base de données relationnelle et objet (SGBDRO). C'est un outil libre disponible selon les termes d'une licence de type BSD. Ce système est concurrent d'autres systèmes de gestion de base de données, qu'ils soient libres (comme MySQL et Firebird), ou propriétaires (comme Oracle, Sybase, DB2 et Microsoft SQL Server). Comme les projets libres Apache et Linux, PostgreSQL n'est pas contrôlé par une seule entreprise, mais est fondé sur une communauté mondiale de développeurs et d'entreprises.

2.2. Bref historique :

Le système de gestion de bases de données relationnel objet PostgreSQL est issu de POSTGRES, programme écrit à l'université de Californie à Berkeley. Après plus d'une vingtaine d'années de développement, PostgreSQL annonce être devenu la base de données Open Source de référence. Le projet POSTGRES mené par le professeur Michael Stonebraker était sponsorisé par le DARPA (Defense Advanced Research Projects Agency), la NSF (National Science Foundation) et ESL, Inc. Le développement de POTGRES a débuté en 1986. Depuis plusieurs versions majeures ont vu le jour (Démo, la version1, la version2...). POSTGRES a aussi été utilisé comme support de formation dans plusieurs universités.

2.3. Installation PostgreSQL

1. extraire les fichiers de postgresql-8.2.3-1.zip dans un répertoire temporaire
2. lancer postgresql-8.2.msi et choisissez le Français
3. sélectionner le support des langues
4. configuration du service : noter le nom du compte de service « postgres » et adjoignez lui un mot de passe à ne pas oublier (utilisateur qui lance le service) !
5. notez le numéro de port 5432 ; acceptez les connexions externes (pas seulement localhost ; il faudra éditer le fichier pg_hba.conf) ; choisissez une locale France, french ; codage LATIN1 ; super utilisateur : c'est le DBA : il peut être le même que celui qui lance le service...

2.4.Les différentes plates-formes supportées par PostgreSQL :PostgreSQL fonctionne sur de nombreuses plates-formes :

- Unix.
- Linux.
- FreeBSD.
- Aix (IBM).
- HP-UX (Hewlett Packard).
- IRIX.

- Solaris (Sun)...

2.5. Caractéristique de postgresQL :

PostgreSQL possède de nombreuses caractéristiques en faisant un SGBDR robuste et puissant digne des SGBD commerciaux

- Des interfaces graphiques (x-window est donc nécessaire) pour gérer les tables.
- Des bibliothèques pour de nombreux langages (appelés *frontaux*) afin d'accéder aux enregistrements à partir de programmes écrits en : java, langage C/C++, Perl, Tcl/Tk.
- PostgreSQL fonctionne selon une architecture client/serveur, il est ainsi constitué :
- ❖ D'une partie serveur, capable de traiter les requêtes des clients, il s'agit dans le cas de PostgreSQL d'un programme résident en mémoire appelé **postmaster**.
- ❖ D'une partie cliente qui interroge le serveur de base de données à l'aide de requêtes SQL.

2.6. Connexion de PostgreSQL sur un environnement de développement :

Pour établir une connexion entre PostgreSQL et un environnement de développement il nous faut un **Driver** qui est un fichier **.jar**, qu'on peut télécharger facilement en effectuant une simple recherche sur **Google**. On choisit la version qu'on veut télécharger une fois que c'est fait on l'intègre dans l'environnement de développement une fois que c'est fait, on peut manipuler les bases de données à partir des programmes.

2.7. Les avantages de PostgreSQL :

La possibilité d'imposer des contraintes à l'insertion des données

- Les contraintes d'intégrités (garantie d'intégrité des données) .
- La possibilité de créer des vues
- La possibilité d'hériter des tables et des types de données
- Les procédures stockées (programmation coté serveur)
- Les transactions (plusieurs étapes en une seule opération)
- Les triggers (gestion des événements).
- La Portabilité
- PostgreSQL est le serveur de bases de données le plus standardisé des logiciels open source.
- Outils pour générer du code SQL portable à partager avec d'autres systèmes compatibles SQL.

2.8. Les inconvénients de PostgreSQL :

- L'utilitaire VACUUM rend PostgreSQL difficile à utiliser en environnement de production 24h/24 et 7j/7.

- Les tables sont obligatoirement transactionnelles.
- Les commandes INSERT, DELETE et UPDATE sont plus lentes.
- PostgreSQL est lent par rapport aux autres SGBDs

Conclusion :

Dans cette annexe nous avons présenté le système de gestion de base de données PostgreSQL, ce qui nous a permis d'avoir des connaissances générales en matière de base de données PostgreSQL.

ANNEXE C

PROCEDURE DES

RESSOURCES HUMAINES

« ENIEM »

Procédures des ressources humaines :

1. Le recrutement :

Le recrutement est une opération par laquelle l'entreprise s'engage à intégrer parmi ses effectifs une personne ayant sollicité un emploi. Pour cela l'entreprise procède comme suite :

1. Faire une annonce dans au moins deux quotidiens nationaux.
2. Recueillir les demandes des candidatures.
3. Convoquer les candidats retenus pour un test de sélection.
4. Etablir un contrat temporaire.
5. Etablir une décision d'affectation.
6. Soumettre le candidat à la période d'essai réglementaire.
7. Faire le nécessaire pour la confirmation, l'infirmité ou prolongation de la période d'essai.
8. Notification par écrit aux candidats non retenus, le motif du rejet de leurs demandes.

Le contrat temporaire à une durée de six mois (cette durée varie selon le besoin) renouvelable à la fin de cette durée.

2. Procédure de promotion :

Tout travailleur promu est soumis à une période d'essai dans son nouveau poste. Si l'employé est confirmé dans ses nouvelles fonctions, le chargé d'étude du personnel établit une décision de confirmation en trois exemplaires.

Dans le cas où l'employé est infirmé il réintègre son poste d'origine ou un poste équivalent.

3. Procédure de mutation :

Une demande de mutation peut provenir de la hiérarchie ou de l'employé lui-même. La demande de mutation doit d'abord passer par la structure (celle d'où vient l'employé) et en suite par la structure prenante (celle où va travailler l'employé).

Elle devient effective qu'après les avis des deux structures et l'avis de la direction.

Il y a infirmité s'il existe un seul avis défavorable et il y a confirmation si tous les avis sont favorables et dans ce cas, le gestionnaire emploi et formation établit une décision d'affectation.

Si la mutation est faite dans la même structure, alors la décision sera établit en trois exemplaires (l'intéressé, gestionnaire du personnel et chargé d'étude).

Si la mutation est faite dans deux structures différentes la décision sera établie en cinq exemplaires.

1. Gestionnaire pp.
2. Structure prenante.
3. L'intéressé.

4. Structure cédante.

4. Procédure de période d'essai :

L'employé est soumis à une période d'essai après le recrutement, mutation ou promotion.

La période d'essai pour la promotion et la mutation étant comprise entre

02 mois pour le personnel exécutif

04 mois pour le personnel de maîtrise

04 mois pour le personnel d'encadrement

04 mois pour le personnel supérieur.

La période d'essai après le recrutement a une durée de 06 mois.

Après l'établissement de la fiche d'évaluation par le chef hiérarchique de l'agent, elle sera transmise au DRH pour avis et de là on établit soit une décision de confirmation ou d'infirmerie.

S'il s'agit de recrutement, l'intéressé a droit à une prolongation (deuxième P.E) s'il n'est pas confirmé après la première période d'essai.

5. Procédure de détachement :

Le détachement est une autorisation temporaire d'absence pour permettre à un travailleur d'accomplir un mandat électif auprès des institutions publiques. Pour cela l'intéressé fait une demande au chef de service qui va l'étudier. Si la demande est acceptée et signée par le DU, elle sera transmise au D.R.H pour décision.

6. Procédure de mise en disponibilité :

La mise en disponibilité est une suspension temporaire de la relation de travail pour une durée supérieure à un mois, prononcée pour une durée maximale d'une année, renouvelable dans la limite de cinq années.

L'intéressé fait une demande de mise en disponibilité accompagnée par un formulaire qui doit être visée par ses responsables hiérarchiques. Ils seront transmis au chef de service pour les traiter. Dans le cas favorable, le C.E établit une décision de mise en disponibilité.

7. Procédure de la réintégration :

A la fin de la mise en disponibilité l'intéressé doit formuler une demande de réintégration au moins trente jours avant la fin de la période prévue adressée au responsable de l'unité exprimant son intention de reprendre le travail à la date de reprise prévue.

8. Procédure d'intérim :

L'intérim est une action par laquelle l'employeur désigne un agent pour occuper temporairement un poste de travail égale ou supérieur à celui qu'il occupe.

Le responsable hiérarchique fait une proposition puis il transmettra au chef de service personnel pour étude. Ce dernier la transmet à son tour au D.U pour signature. Dans le cas favorable le G.P.P établit une décision d'intérim.

Dans le cas où l'intérim est appelé à assurer l'intérim d'un poste de cadre ou cadre supérieur, la proposition sera transmise au D.R.H pour avis.

9. Procédure de formation :

Cette procédure peut se déclencher par une proposition des responsables hiérarchiques ou par une demande de l'employé, en prenant compte du programme de formation élaborée par l'entreprise.

L'intéressé fait une demande ou proposition, le C.E l'étudie selon la nécessité de de besoin de formation. Si elle est acceptée on lui élabore un contrat de formation et une décision de détachement et à la fin de la formation l'employé doit être réintégré à son poste

10. Procédure de congé annuel, exceptionnel :

Le congé annuel survient toujours après la fin du mois de juillet de chaque année. Dans ce cas, une liste ainsi que des titres de congé seront élaborés par le gestionnaire personnel et paie. Le congé exceptionnel survient après demande formulée par l'intéressé qui est destinée au gestionnaire personnel et paie qui procède à l'étude de la demande, dans le cas favorable, un titre de congé sera établi.

11. Procédure de sanction :

En cas de faute passible d'une sanction, le chef hiérarchique directe de l'employé établit une proposition de sanction et un rapport qui seront transmis au chef de service personnel pour spécifier le degré de la faute. S'il s'agit d'une faute de premier ou de deuxième degré il établira alors une décision de sanction. Sinon, on aura recours à la commission de discipline qui établit un procès verbal. Les différents cas de fautes disciplinaires de 1^{er}, 2^{eme} ou 3^{eme} degré donnent lieu à l'application de l'une des sanctions disciplinaires suivantes :

➤ Faute de 1^{er} degré :

1. Rappel à l'ordre/ Avertissement verbal.
2. Avertissement écrit.
3. Blâme.
4. Mise à pied de 1 à 3 jours ouvrables.

➤ Faute de 2^{eme} degré :

Mise à pied de quatre (04) à quinze (15) jours ouvrables.

➤ **Faute de 3^{ème} degré :**

1. Rétrogradation.
2. Licenciement sans indemnités ni préavis.

12. Procédure de licenciement :

Le licenciement est une action par laquelle l'employeur procède au licenciement de l'employé. Pour cela le chef hiérarchique établit un rapport qui sera transmis au chef de service pour étude et établissement d'une convocation qui sera envoyée à l'intéressé, après avoir justifié ce dernier et dans le cas où sa justification n'est pas accordée, le chef de service transmettra son rapport à la commission de discipline pour étude et décision.

13. Procédure de démission :

L'intéressé fait une demande de démission avec justification au GPP. Ce dernier la transmettra au D.U pour avis. Dans le cas favorable, le GPP délivre une fiche de circuit et il effectue ensuite un certificat de travail et un certificat de cessation de paiement et d'activité qui seront transmis à l'intéressé.

14. Procédure de retraite :

La retraite survient après que l'employé aura acquis un minimum de quinze ans d'expérience et que son âge est égal à soixante ans pour l'homme et cinquante-cinq ans pour la femme.

Il existe trois cas de mise en retraite :

1. Retraite d'ancienneté :

Pour que le fonctionnaire ait droit à cette retraite, il faut qu'il remplisse les conditions suivantes :

1. 30 ans de service.
2. Avoir l'âge de 60 ans pour l'homme et de 55 ans pour la femme.

2. Retraite anticipée :

Elle est ouverte à tout employé ne pouvant plus remplir ses obligations de fonction (impossibilité de travailler). A tout employé ayant 32 ans de service.

3. Retraite proportionnelle :

Pour que l'employé ouvre droit à cette retraite, il faut qu'il remplisse les conditions suivantes :

- 60 ans d'âges.
- 15 ans de service.

Le fonctionnaire retraité touchera 50% de son salaire au cours de sa retraite. A cet effet le gestionnaire social procède à l'établissement du dossier de retraite qui sera transmis à la

C.N.R pour étude et établissement d'une notification d'attribution de pension de retraite. Ensuite le GS établit une fiche de circuit. Après l'étude de la fiche de circuit par le GPP, il établit cinq certificats de cessation de paiement et d'activité et deux certificats de travail.

Bibliographie et Webographie

Livres & documents

- [1] Document → les PGI, fichier PDF écrit par Laure Ferbourg
- [2] Document → **La FAQ Java EE : Developer.com**, fichier PDF ; Date de publication 17/05/2003, Dernière mise à jour : 22/12/2009
- [3] Document → le chapitre 12 Client/Serveur, fichier PDF BTS IG 1^{ère} année PLSI
- [4] Document → Sécurité informatique, Fichier PDF
- [5] Livre → **UML2 en action** [de l'analyse des besoins à la conception] écrit par Pascal Roques et Franck Vallée 4^e édition EYROLLES 2009
- [6] Document → **Catalogue de l'entreprise ENIEM**
- [7] Document → **Conception Orientée Objet**, fichier PDF publié par **Alexis Drogoul**
- [8] Livre → **UML pour la modélisation d'un système d'information** [le cahier des charges du maître d'ouvrage] chantal Morlley, Jean Hugues, Bernard Leblanc..... 2^e édition. Mai 2002.
- [9] Document → **L'orienté objet, cours et exercices en UML**, fichier PDF créé par Hugues Bersini, Ivan Wellesz et publié par Groupe Eyrolles. 2002.
- [10] Document → **Méthodologie des systèmes d'information – UML**, fichier PDF créé par DI GALLO Frédéric. 2000-2001
- [11] Xavier PERRAS, Architecte des systèmes informatique, 2000
- [12] Livre → George et Olivier Gardain, le Client/serveur, Eyrolles 2000

Webographie :

- [13] <http://www.javadebutant.org/>
- [14] <http://www.uml.free.fr>
- [15] <http://www.commentcamarche.fr>
- [16] <http://www.PostgreSQL.com>
- [17] <http://www.sun.com>
- [18] <http://www.wikipédia.fr>
- [19] <http://www.site du zéro.fr>
- [20] <http://www.NetBeans.org>
- [21] <http://www.developpez.com>