

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A/Mira de Béjaïa
Faculté des Sciences Exactes
Département d'Informatique

Mémoire de fin de cycle

Master 2 Professionnel en Informatique

Option

Administration et Sécurité des Réseaux

Thème

Conception et réalisation d'un système
d'information de gestion du service stages et
formations pour l'entreprise BMT (Bejaïa
Mediterranean Terminal)

Réalisé par :

Mlle ABID Sabrina.
Mlle KHELFAOUI Aïda.

présenté devant le jury :

Président	M ^r BAADACHE Abdelrahmane	U. A/Mira Béjaïa.
Rapporteur	M ^r LARBI Ali	U. A/Mira Béjaïa.
Examinatrice	M ^{me} LARBI Wahiba	U. A/Mira Béjaïa.
Examineur	M ^r MIR Foudil	U. A/Mira Béjaïa.

Béjaïa 2013.

**** Remerciements ****

Nous remercions dieu, le tout puissant de nous avoir accordé la volonté et le courage pour accomplir ce travail.

Au terme de cette fructueuse étape de notre formation et fiers de notre acquis, nous pensons à tous les membres du corps scientifique de la faculté des sciences exactes, qu'ils trouvent à travers ce modeste travail, la matérialisation et /ou la concrétisation de cette mosaïque de connaissances qu'ils nous ont transmise durant notre parcours.

Que le Prof LARBI ALI, encadreur daigne trouver à travers ces quelques lignes, l'expression de notre profonde gratitude pour avoir accepté de diriger ce travail ainsi que pour l'aide compétente qu'il nous a apportée. Son œil critique nous a été très précieux pour structurer le travail et pour améliorer la qualité des différentes sections.

Nous remercions tout particulièrement le chef de service informatique de BMT, Mr BOUMER-ZOUG Moussa pour avoir mis personnellement à notre disposition les moyens humains et matériels nécessaires au bon déroulement de notre stage.

Nous ne manquerons pas non plus de dire un grand merci aux membres du jury qui ont accepté d'évaluer ce mémoire à sa juste valeur, et de nous faire part de leurs remarques sûrement pertinentes qui contribueront, sans nul doute, au perfectionnement du présent travail.

Que tous ceux et celles qui ont contribué de près ou de loin à l'accomplissement de ce travail trouvent ici l'expression de nos remerciements les plus chaleureux.

*** *Dédicaces* ***

À la mémoire de mes grands-parents, qui ont toujours aimés de partager mes succès que leurs âmes reposent en paix.

À mes très chers parents.

Vous vous êtes dépensés pour moi sans compter. En reconnaissance de tous les sacrifices consentis par vous, pour me permettre d'atteindre cette étape de ma vie, vous avez toute ma tendresse.

À mon adorable frère Yacine et petite sœur Hanane pour votre profonde passion et soutien tout au long de mes études. Affectueuse reconnaissance.

ma sœur Salima et son marie ainsi que ma petite nièce Anaïs

À mes oncles, tantes, cousins et cousines. Vous avez de près ou de loin contribué à ma formation.

A mes amis (es) ceux avec qui j'ai parcouru un long chemin avec tant de peine et de joie, inquiétude et espoir.

À tous ceux dont les noms n'y figurent pas pour une raison ou une autre trouve l'expression de ma profonde gratitude.

Je dédie ce travail.

Aïda

*** *Dédicaces* ***

Je dédie ce modeste travail aux deux personnes les plus chères pour moi dans ce monde : mes deux parents qui n'ont pas cessé d'être à mes côtés dans toutes les étapes que j'ai franchies et qui m'ont aidé pour arriver là où je souhaitais.

À mes frères Yacine, Nassim

À mes oncles, tantes, cousins et cousines, sans oublier mes grands parents.

À mes amis (es) ceux avec qui j'ai parcouru un long chemin avec tant de peine et de joie, inquiétude et espoir.

À tous ceux dont les noms n'y figurent pas pour une raison ou une autre trouve l'expression de ma profonde gratitude.

Sabrina

Table des matières

Table des matières	i
Liste des Figures	vii
Liste des tableaux	viii
Liste des abréviations	ix
Introduction générale	1
I Généralités et présentation de l'organisme d'accueil	4
1 Généralités sur les systèmes d'information	5
Introduction	6
I Les systèmes d'information	6
I.1 Définition d'un système d'information	6
I.2 Les fonctions d'un système d'information	7
I.3 Les firmes à système d'information	7
I.3.1 Les systèmes d'information formels	7
I.3.2 Les systèmes d'information informels	8
I.4 La sécurité des systèmes d'information	8
I.5 Les composantes d'un système d'information	8
I.5.1 Les ressources humaines	8
I.5.2 Les composantes matérielles	9
I.5.3 Les composantes immatérielles	9
I.6 Les bases de données	10

I.6.1	Définition d'une base de données (Database)	10
I.6.2	Application de bases de données (Database application)	10
I.6.3	Système de gestion de bases de données (SGBD)	10
I.6.4	Types de bases de données	12
I.6.5	Intérêt d'utilisation d'un SGBD	13
II	Les Réseaux Informatiques	13
II.1	Définition	13
II.2	Catégorie des réseaux	13
II.3	Topologies des réseaux	14
II.4	Les protocoles réseaux	17
II.4.1	Le modèle OSI(Open System Interconnection)	17
II.4.2	Le modèle TCP/IP(Transmission Control Protocol/Internet Protocol)	19
II.5	Internet et intranet	20
III	L'architecture Client/Serveur	20
III.1	Concept de Client/Serveur	20
III.2	Les trois niveaux d'abstraction d'une application :	21
III.2.1	L'architecture un tier	22
III.2.2	L'architecture deux tiers	22
III.2.3	L'architecture trois-tiers	23
III.2.4	L'architecture n-tiers	24
	Conclusion	24
2	Présentation de l'organisme d'accueil	25
	Introduction	26
I	Présentation Générale De La BMT	26
I.1	Création (Joint venture)	26
I.2	Présentation de la BMT	26
I.2.1	La structure de l'entreprise	27
I.3	Situation géographique	30
I.3.1	Situation du terminal a conteneurs	30
II	Les activités principales de la BMT et ses mission	30

II.1	Les missions de la BMT	30
II.2	Opérations du terminal	31
II.3	Les objectifs de la BMT	32
III	Présentation du champ d'étude	32
IV	Problématique	32
	Conclusion	33
II	Conception et mise en oeuvre	34
3	Démarche adoptée	35
	Introduction	36
I	La méthode 2TUP	36
I.1	Le processus unifié (UP)	36
I.2	Le processus 2TUP	37
I.3	Processus de modélisation avec UML	39
	Conclusion	40
4	Etude préliminaire	41
	Introduction	42
I	Recueil des besoins fonctionnels	42
II	Déscription du contexte	43
II.1	Identification des acteurs	43
II.2	Identification des messages	43
II.3	Modélisation du contexte	44
	Conclusion	45
5	Capture des besoins fonctionnels	46
	Introduction	47
I	Identification des cas d'utilisations	47
I.1	Définition	47
I.2	Les cas d'utilisation correspondant à chaque acteur	48
II	Description des cas d'utilisation	53

Conclusion	57
6 Capture des besoins techniques	58
Introduction	59
I Capture des spécifications matérielles	59
I.1 Architecture client/serveur	59
I.2 Architecture Oracle	59
I.2.1 Connexion utilisateur	60
I.2.2 Le fichier de paramètres	61
I.2.3 La base de données	62
II Capture des spécifications logicielles	62
II.1 Les cas d'utilisations techniques	63
II.1.1 Identification et description des cas d'utilisation techniques	63
Conclusion	66
7 Analyse	67
Introduction	68
I Développement du modèle statique	68
I.1 Diagramme de classe	68
I.2 Le diagramme de classe de notre système	69
II Développement du modèle dynamique	69
II.1 Formaliser les scénarios	70
II.1.1 Diagramme de séquence	70
II.2 Les diagrammes de séquence de notre système	71
II.2.1 Diagramme de séquence de cas d'utilisation "Authentification"	71
II.2.2 Diagramme de séquence de cas d'utilisation "Ajouter une formation"	72
II.2.3 Diagramme de séquence de cas d'utilisation "Modifier une formation"	74
II.2.4 Diagramme de séquence de cas d'utilisation "Chercher une formation"	76
II.2.5 Diagramme de séquence de cas d'utilisation "Ajouter un stagiaire"	78
II.2.6 Diagramme de séquence de cas d'utilisation "Modifier un stagiaire"	78
II.2.7 Diagramme de séquence de cas d'utilisation "Supprimer un stagiaire"	79

II.2.8	Diagramme de séquence de cas d'utilisation "Effectuer une demande de formation"	80
	Conclusion	82
8	Conception	83
	Introduction	84
I	Dictionnaire de données	84
II	Modèle relationnelle de données	87
II.1	Terminologie de l'approche relationnelle	87
II.2	Règles de passage	88
II.3	Présentation du schéma relationnel	88
	Conclusion	89
9	Réalisation	90
	Introduction	91
I	Langage et outils de développements	91
I.1	Langage d'implémentation (JAVA)	91
I.2	Outils de développement	91
I.2.1	Présentation de la plateforme de développement Netbeans	91
I.2.2	Ireport	91
I.2.3	Présentation du SGBD Oracle 10g	92
II	Présentation de prototype réalisé	92
	Conclusion	96
	Conclusion générale	97
	Bibliographie	98
	Annex	101

Table des figures

1.1	Système d'information.	6
1.2	Les trois niveaux d'abstraction de l'ANSI.	12
1.3	La topologie en bus.	15
1.4	La topologie en étoile.	15
1.5	La topologie en anneau.	16
1.6	La topologie en arbre.	16
1.7	Présentation graphique du modèle OSI.	18
1.8	Présentation graphique du modèle TCP/IP.	20
1.9	Architecture à 2 niveaux.	23
1.10	Architecture à 3 niveaux.	24
2.1	La joint venture de BMT.	27
2.2	Organigramme général de la BMT.	29
2.3	Situation du terminal a conteneurs.	30
3.1	Le système d'information soumis à deux types de contraintes.	37
3.2	Le processus de développement en Y.	38
4.1	Diagramme du contexte dynamique.	45
5.1	Diagramme de cas d'utilisation associé au "Chargé de stage et formation".	50
5.2	Diagramme de cas d'utilisation associés au "responsable de structure et l'employé".	51
5.3	Diagramme de cas d'utilisation "Gérer stage et formation".	52
5.4	Diagramme d'activité gérer les stagiaires	53
5.5	Diagramme d'activité consulter les formations	55
5.6	Diagramme d'activité chercher un employé.	56
5.7	Diagramme d'activité gérer les attestations.	57

6.1	Architecture Oracle.	60
6.2	Modèle de spécification logicielle.	66
7.1	Diagramme de classe.	69
7.2	Diagramme de séquence de cas d'utilisation "Authentification".	72
7.3	Diagramme de séquence de cas d'utilisation "Ajouter une formation".	73
7.4	Diagramme de séquence de cas d'utilisation "Modifier une formation".	75
7.5	Diagramme de séquence de cas d'utilisation "Chercher une formation".	77
7.6	Diagramme de séquence de cas d'utilisation "Ajouter un stagiaire".	78
7.7	Diagramme de séquence de cas d'utilisation "Modifier un stagiaire".	79
7.8	Diagramme de séquence de cas d'utilisation "Supprimer une formation".	80
7.9	Diagramme de séquence de cas d'utilisation "Effectuer une demande de formation".	81
9.1	Interface d'accueil.	93
9.2	Interface d'authentification.	94
9.3	Interface d'espace administrateur.	94
9.4	Interface de gestion des stages.	95
9.5	Interface de gestion des formations.	96

Liste des tableaux

5.1	Les cas d'utilisation du système à réaliser.	49
5.2	Description du cas d'utilisation "Gérer stagiaire".	53
5.3	Description du cas d'utilisation "Consulter formations".	54
5.4	Description du cas d'utilisation "Chercher employé".	55
5.5	Description du cas d'utilisation "Gérer attestations".	56
6.1	Description du cas d'utilisation technique "Gérer utilisateur".	63
6.2	Description du cas d'utilisation technique "Sauvegarder la BDD".	64
6.3	Description du cas d'utilisation technique "Restaurer la BDD".	64
6.4	Description du cas d'utilisation technique "S'authentifier".	64
6.5	Description du cas d'utilisation technique "Changer le mot de passe".	65
6.6	Description du cas d'utilisation technique "Manipuler les objets".	65
6.7	Description du cas d'utilisation technique "Consulter l'aide d'utilisation".	65
8.1	Dictionnaire de données.	87

Liste des abréviations

2TUP	2 Tracks Unified Process
ANSI	American National Standards Institute
ANSI/SPARC	American National Standards Institute/ Scalable Processor ARChitecture
BLOB	Binary Large OBject
BMT	Bejaia Mediterranean Terminal
CPE	Conseil de la Participation de l'Etat
EDI	Environnement de Développement Intégré
ETCD	Equipement Terminal de Circuit de Données
EPB	Entreprise Portuaire de Bejaïa
GSM	Global System for Mobile Communication
IBM	International Business Machines
ICMP	Internet Control Message Protocol
IHM	Interface Homme-Machine
IP	Internet Protocol
ISO	International Organization for Standardization
LAN	Local Area Network
MAN	Metropolitan Area Network
MySQL	My Structured Query Language
OSI	Open System Interconnect
PAN	Personal Area Network
POO	Programmation Orienté Objet
SGBD	Système de Gestion de Bases de Données
SI	Système d'Information
SII	Système d'Information Informatisé
SPARC	Scalable Processor ARChitecture

SQL	Structured Query Language
TCP/IP	Transmission Control Protocol/Internet Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UP	Unified Process
WAN	Wide Area Network

Introduction générale

La rationalisation des choix technologique et la rentabilisation des investissements constituent de véritables défis pour les entreprises. L'arrivée de l'informatique a répondu à leurs besoins stratégiques et technologiques.

Actuellement, l'informatique représente un outil inévitable de gestion et de pilotage des systèmes d'information, surtout au niveau des organismes possédant des systèmes qui reposent sur la gestion des bases de données. Pour cela, il s'est avéré nécessaire d'avoir un environnement de connaissances sur la théorie des réseaux ainsi que les systèmes d'information.

En tenant compte du volume des transactions, des traitements à effectuer et des délais de diffusion de l'information, le système d'information est un composant important pour l'organisation de l'entreprise. Il augmente la capacité de traitement et d'analyse des informations, autrement dit, ces technologies permettent une meilleure transmission de l'information. Ces objectifs essentiels est de maintenir en condition opérationnelle les systèmes et les réseaux d'information et de communication. La réussite d'un système d'information repose sur la conception qui met en jeu la définition du travail humain, l'organisation du processus de production et son contrôle.

C'est dans ce cadre général que s'introduit notre projet de fin d'étude qui consiste à réaliser un système d'information pour le suivi des stages et formations à BMT (Bejaïa Mediterranean Terminal).

Le service des ressources humaines enregistre toutes les informations manuellement sur des supports en papier ce qui engendre beaucoup de problèmes tel que la difficulté de stockage et la perte de temps considérable dans la recherche de ces informations.

Le but de notre travail consiste à doter ce service d'une application client/serveur qui met en œuvre une solution moderne et évolutive permettant d'atteindre les objectifs prévus et répond aux différentes exigences fonctionnelles et techniques, tout en facilitant les différents travaux des acteurs de notre système.

Pour bien structurer notre étude, nous avons adopté une démarche spécialisée dans la conception des systèmes d'information qui nous accompagnera tout au long du projet, celle-ci est basée sur le langage UML et connue sous le nom : processus 2 TUP " 2 Track Unified Process ".

Afin d'atteindre les objectifs sollicités auparavant, nous avons organisé ce mémoire en deux parties :

– **Première partie : *Généralités et présentation de l'organisme d'accueil.***

Cette partie constitue l'aspect théorique du mémoire. Elle inclut deux chapitres :

Chapitre 1 : *Généralités sur les systèmes d'information.*

Ce chapitre est consacré à des généralités et principaux concepts liés aux systèmes d'information et aux réseaux.

Chapitre 2 : *Présentation de l'organisme d'accueil.*

Dans ce chapitre, nous allons présenter l'organisme d'accueil de l'entreprise BMT.

– **Deuxième partie : *Conception et mise en œuvre.***

Cette partie est divisée en 7 chapitres, le premier définit la démarche adoptée alors que les autres chapitres traitent une phase du processus du développement 2TUP que nous avons adopté pour concevoir notre système. Cette partie est organisée comme suit :

Chapitre 3 : *Démarche adoptée.*

Dans ce chapitre, nous présenterons la méthode d'analyse et de conception adoptée.

Chapitre 4 : *Etude préliminaire.*

Ce chapitre constitue une étude préliminaire du système à réaliser.

Chapitre 5 : *Capture des besoins fonctionnels .*

Ce chapitre traite le rôle que tient UML pour compléter la capture des besoins fonctionnels ébauchée durant l'étude préliminaire par la technique des cas d'utilisation.

Chapitre 6 : *Capture des besoins techniques .*

Dans ce chapitre nous arrivons à la capture des spécifications techniques liées à la configuration matérielle et la capture des spécifications logicielles.

Chapitre 7 : *Analyse.*

Dans ce chapitre nous analysons le système, nous procéderons au développement du modèle statique et celui du modèle dynamique.

Chapitre 8 : *Conception.*

Nous arrivons dans ce chapitre à la conception du système qui étudie et documente précisément les classes et les méthodes qui constitue le codage de la solution.

Chapitre 9 : *Réalisation.*

C'est la mise en œuvre de notre application client/serveur. Nous ferons appel à un ensemble d'outils et langages de développement. Ensuite, nous utiliserons des captures d'écrans pour

illustrer les principales fonctionnalités offertes par le système.

Et enfin nous terminerons par une conclusion générale. Deux annexes sont ajoutées à la fin pour éclaircir les notions non approfondies dans les chapitres précédents.

Première partie

Généralités et présentation de
l'organisme d'accueil

Chapitre 1

Généralités sur les systèmes d'information

Introduction

Disposer de la bonne information au bon moment est l'objectif de tous les gestionnaires et décideurs, tant la qualité de l'information et sa disponibilité sont des facteurs de compétitivité pour l'entreprise. Le système d'information est la mémoire de l'organisation, sa qualité est donc un enjeu majeur pour cette dernière. Il comporte un aspect statique : mémoriser les données, et un aspect dynamique : mise à jour des données.

Dans ce chapitre, nous allons présenter les principaux concepts liés aux systèmes d'information en présentant des généralités sur les bases de données et quelques notions à-propos des réseaux informatiques et les modèles de communications OSI (Open System Interconnect) et TCP/IP (Transmission Control Protocol/Internet Protocol).

I Les systèmes d'information

I.1 Définition d'un système d'information

Un système d'information peut être défini comme “la partie du réel constituée d'informations organisées et d'acteurs qui agissent sur ces informations ou à partir de ces informations, selon des processus visant une finalité de gestion et utilisant les technologies de l'information ”. Dans un projet, le maître d'ouvrage est responsable de la définition et la mise en œuvre du système d'information[1].

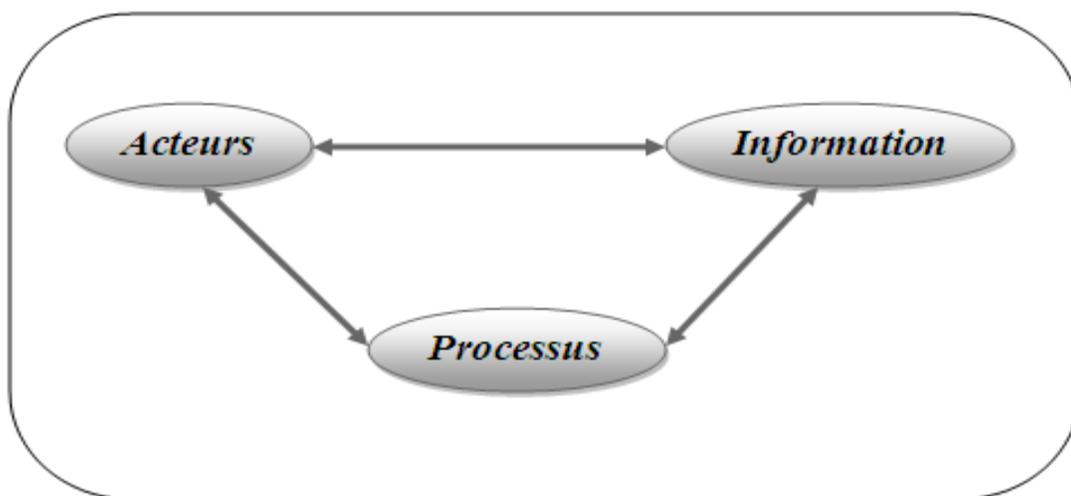


FIG. 1.1 – Système d'information.

La figure 1.1 représente un schéma général d'un système d'information :

- **Les informations** organisées sont celles que l'entreprise choisit de gérer, en général prédéfinies et structurées.
- **Un acteur** est un être humain ou une machine, qui crée, manipule, transforme les informations ou qui est sollicité par la présence ou la valeur de certaines informations.
- **Le processus** est un plan d'ensemble indiquant comment les acteurs collaborent au moyen des informations gérées pour accomplir l'objectif de production [1].

I.2 Les fonctions d'un système d'information

Les SI assurent généralement les rôles de collecte, de traitement et de diffusion de l'information sur l'environnement (constitué notamment par les actionnaires, les clients, les fournisseurs, l'état) ou sur son organisation. Ceci se traduit par trois activités :

- **L'entrée** : il s'agit du regroupement et de la saisie des données brutes provenant soit de l'organisation, soit de l'environnement externe
- **Le traitement** : c'est la transformation des données brutes en synthèses utiles pour les décisions
- **La sortie** : c'est le processus de diffusion de l'information obtenue à la suite du traitement.

Pour avoir un certain équilibre, les SI se basent sur la rétroaction. Cette dernière est la transmission de l'information obtenue en sortie vers l'entrée afin de l'évaluer ou de la corriger [13].

I.3 Les firmes à système d'information

Un système peut être constitué de sous-systèmes. Cette règle s'applique aussi en matière de SI. un SI dans une organisation, une fois isolé peut être formalisé ou non.

I.3.1 Les systèmes d'information formels

Ce sont des systèmes qui fonctionnent avec des règles précises, préalablement définies. Ces SI peuvent être :

- **Automatisés** : on parlera aussi de Système d'Information Informatisé (SII). Ici, on a recours aux procédures informatiques dans l'exécution d'au moins une des activités du SI.
- **Manuels** : ce sont des systèmes pour lesquels (au moins) l'activité de traitement-stockage ne se fait pas par des procédés automatiques.

I.3.2 Les systèmes d'information informels

Il s'agit des SI n'obéissant à aucune règle. Il n'existe aucune procédure relative à chacune des activités [2].

I.4 La sécurité des systèmes d'information

La sécurité des SI est l'ensemble des mesures prises pour protéger les infrastructures du SI et les données stockées. Cela signifie que la sécurité doit être abordée dans un contexte global. Ce qui sous-entend :

- La sensibilisation des utilisateurs aux problèmes de sécurité.
- La sécurité logique, c'est-à-dire la sécurité au niveau des données.
- La sécurité des télécommunications.
- La sécurité des applications : sécurité des différents processus.
- La sécurité physique : la sécurité des infrastructures matérielles.

Les enjeux de la sécurité des SI sont très importants de nos jours. Les notions de virus et de pirates informatiques (hackers) sont devenues monnaie courante. Les risques encourus sont très significatifs [2].

I.5 Les composantes d'un système d'information

Le fonctionnement des SI s'appuie sur des éléments matériels et immatériels qui doivent évoluer dans un environnement entièrement sécurisé.

I.5.1 Les ressources humaines

C'est elles qui animent l'ensemble du système. Elles sont composées essentiellement du personnel. Il s'agit :

- **Des utilisateurs** : qui sont des personnes qui utilisent les SI. Leur intervention peut se faire soit à l'entrée, soit au traitement, soit à la sortie du système. Il peut s'agir de comptables, d'ingénieurs, de cadres, de dirigeants, de clients ou de fournisseurs.
- **Des informaticiens** : qui sont des personnes qui s'occupent de la conception et du déploiement du SI. Ils veilleront aussi au bon état de marche du matériel et des applications. Parmi les informaticiens nous avons notamment : les analystes fonctionnels, les programmeurs, certains cadres et les employés de bureau. Ces personnes peuvent être internes ou externes à

l'entreprise [2].

I.5.2 Les composantes matérielles

Ces composantes font référence à tout dispositif physique ou machines servant à la collecte, au traitement, au stockage et à la diffusion de l'information. Il s'agit :

- **Des machines** : qui sont des appareils intervenant dans le cycle des activités du SI. Il s'agit notamment des ordinateurs, des écrans, des scanners, des lecteurs de code barre, des imprimantes.
- **Des supports** : qui sont des outils permettant d'enregistrer les données et les informations. Exemples : disque dur, disquette, carte mémoire, clé USB, CD/DVD ROM(RW)...
- **Des réseaux** : c'est un ensemble d'ordinateurs et de périphériques connectés les uns aux autres. L'objectif étant de transmettre les données ou les informations. Les développements technologiques des dernières décennies ont permis d'étendre les réseaux à l'échelle mondiale (internet). On note d'ailleurs une forte expansion des réseaux sans fil (WI-FI) car ceux-ci offrent une grande flexibilité.

I.5.3 Les composantes immatérielles

Ces composantes ont la particularité d'être impalpables. Il s'agit : des données, des logiciels et des informations.

- **Les données** : ce sont les ressources, les événements et les faits qui proviennent de l'entreprise ou de son environnement. Elles représentent la matière première (inputs) du SI. Elles seront stockées et traitées afin d'obtenir les informations (outputs).
- **Les informations** : l'information peut être perçue comme des données qui sont présentées sous une forme significative et utile pour les êtres humains.
- **Les logiciels** : qui sont des instructions programmées qui permettent de traiter les données ou de gérer le matériel informatique. Il existe un nombre indéterminé de logiciels. Ils peuvent néanmoins être regroupés en quatre (04) catégories majeures :
 - ✓ **Les logiciels de base** : ce sont les programmes tels que les systèmes d'exploitation qui gèrent toutes les opérations de base de l'ordinateur et ses périphériques.
 - ✓ **Les logiciels d'application** : ce sont des programmes spécialisés destinés à effectuer des traitements particuliers. C'est le cas des programmes d'analyse des ventes, des programmes de calcul de la paie et des programmes de traitement de texte.

- ✓ **Les logiciels de communication** : ce sont des programmes qui permettent de transmettre des informations d'un ordinateur à un autre. Exemple : les logiciels de tchat, les logiciels de téléchargement.
- ✓ **Les procédures (ou didacticiels)** : on parle aussi de Procédures Opérationnelles Standards (POS). ce sont des consignes d'exploitation destinées aux utilisateurs d'un SI et qui expliquent, par exemple, comment remplir un formulaire ou utiliser un logiciel [2].

I.6 Les bases de données

I.6.1 Définition d'une base de données (Database)

C'est un ensemble structuré d'éléments d'informations, souvent agencés sous forme de tables, dans lesquels les données sont organisées selon certains critères, en vue de permettre leur exploitation pour répondre aux besoins d'informations d'une organisation[3].

I.6.2 Application de bases de données (Database application)

C'est l'utilisation de moyens informatiques pour répondre à un besoin déterminé en faisant appel de manière importante à une ou plusieurs bases de données à travers un système de gestion de bases de données (SGBD) [3].

I.6.3 Système de gestion de bases de données (SGBD)

Logiciel, le plus souvent produit par un éditeur commercial, qui gère et contrôle l'accès à une base de données, assurant ainsi une interface normalisée entre les applications et les bases de données [3].

Plusieurs SGBD sont des logiciels commerciaux offerts à grand prix, sous forme de licences d'utilisation sur un serveur par des éditeurs de logiciel tels que IBM Oracle Microsoft Sybase pour ne nommer que les plus importants qui proposent des SGBD conçus pour exploiter les bases de données de grande envergure. Ces SGBD sont le DB2 de la société IBM, Oracle10i de la société Oracle ou le Microsoft SQL Server. Certains éditeurs offrent par ailleurs des SGBD bas de gamme peu coûteux destinés aux applications de base de données de petites envergures Microsoft Access est peut être le mieux connu et le plus utilisé des SGBD de cette catégorie, souvent appelé SGBD bureautique. Il existe aussi des SGBD dans le monde du logiciel libre, le plus populaire est sans aucun doute MySQL, qui est opéré sous le système d'exploitation Linux, MySQL n'a rien à envier aux SGBD commerciaux au plan des capacités et des performances[3].

I.6.3.1 Fonctionnalités d'un SGBD : De même que l'ISO a déterminé un modèle théorique en sept couches pour distinguer les applications réseaux et leurs interactions, il existe désormais un modèle théorique en trois couches (trois niveaux d'abstraction) afin de concevoir et d'organiser les fonctionnalités des SGBD. Ce modèle est élaboré par la commission SPARC(Scalable Processor ARChitecture) de l'ANSI (American National Standards Institute) : C'est l'architecture ANSI/SPARC (voir figure 1.2). Cette dernière, qui date de 1975, s'inscrit dans les concepts et théories de la première génération des bases de données, dont l'objectif est d'avoir une indépendance entre les données et les traitements :

- **Niveau interne ou physique :** C'est le niveau le plus " bas ". On décrit les structures de stockage de l'information, ce qui le rend très dépendant du SGBD employé. Il se fonde sur un modèle de données physique.
- **Niveau conceptuel :** Il correspond à l'implémentation du schéma conceptuel de la base de données, que l'on réalise lors de la phase d'analyse. Il est utilisé pour décrire les éléments constitutifs de la base de données et les contraintes qui leur sont associées. Il s'agit d'une certaine façon de la " documentation " de la base de données.
- **Niveau externe :** Le niveau externe sert à décrire les vues des utilisateurs, c'est-à-dire le schéma de visualisation des données qui est différent pour chaque catégorie d'utilisateurs. Un schéma externe permet de masquer la complexité de la base de données complète en fonction des droits ou des besoins des utilisateurs. Cela facilite la lecture et la sécurité de l'information [5].

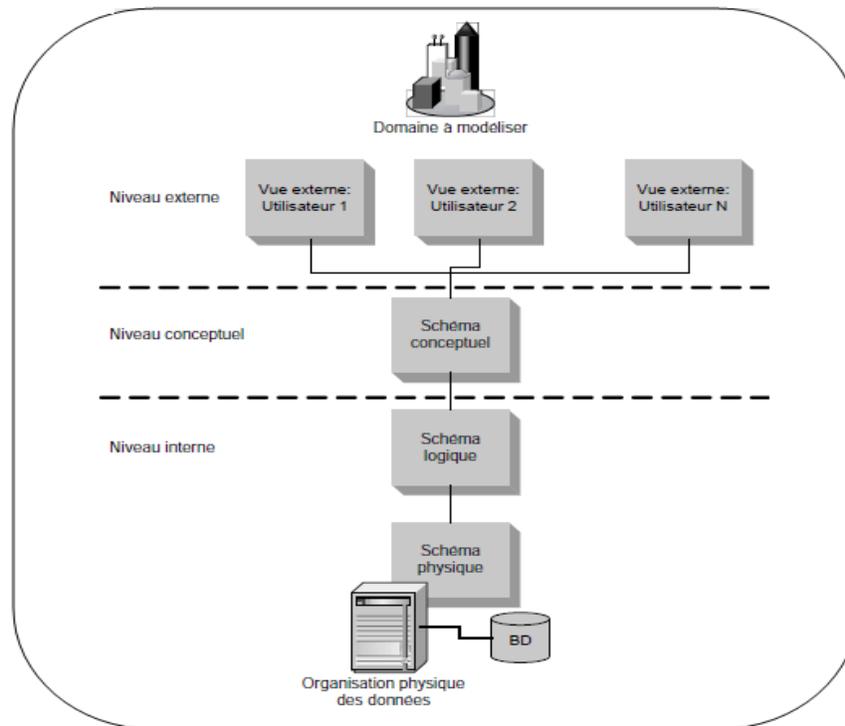


FIG. 1.2 – Les trois niveaux d'abstraction de l'ANSI.

I.6.4 Types de bases de données

I.6.4.1 Les bases de données hiérarchiques et réseaux Le modèle "hiérarchique" propose une classification arborescente des données à la manière d'une classification scientifique. Dans ce type de modèle, chaque enregistrement n'a qu'un seul possesseur par exemple, une commande n'a qu'un seul client. Cependant, notamment à cause de ce type de limitations, ce modèle ne peut pas traduire toutes les réalités de l'information dans les organisations.

Le modèle "réseau" est une extension du modèle précédent : il permet des liaisons transversales, utilise une structure de graphe et lève de nombreuses limitations du modèle hiérarchique.

I.6.4.2 Les bases de données relationnelles Bien que le modèle de données relationnel repose sur des concepts simples (relations, clés et dépendances fonctionnelles), il permet néanmoins de modéliser artificiellement des données complexes. Le modèle relationnel est fondé sur de solides bases théoriques, car il propose des opérateurs issus de la théorie des ensembles. De plus, on peut appliquer des techniques de normalisation. La programmation de ces concepts est assurée par le langage SQL normalisé par l'ISO depuis 1986.

I.6.4.3 Les bases de données orientées objets : Le modèle de données objet-relationnel étend principalement le modèle relationnel à l'utilisation de pointeurs, de collections et de méthodes au niveau des tables et des types. Le développement des SGBD objet a été freiné par des SGBD hybrides incorporant le modèle relationnel et le stockage d'objets peuvent être soit de type structuré comme ceux créés par un langage orienté objet ou de type non structuré telles que des images, de la vidéo, des trames sonores. Ce type est aussi appelé BLOB(Binary Large Object). On donne le nom SGBD relationnel-objet à cette forme hybride car il combine les propriétés du SGBD relationnel et du SGBD objet [4].

I.6.5 Intérêt d'utilisation d'un SGBD

- **Indépendance physique :** un SGBD permet de manipuler des données sans se soucier de la façon dont elles sont stockées physiquement.
- **Indépendance logique :** un SGBD permet de présenter sous différentes vues les mêmes données.
- **Efficacité des accès aux données :** un SGBD repose sur des algorithmes (transparents pour l'utilisateur) performants qui optimisent l'accès aux données.
- **Sécurité des données :** un SGBD permet de gérer différents niveaux d'accessibilité aux données, en fonction des profils d'utilisateurs [6].

II Les Réseaux Informatiques

II.1 Définition

Un réseau de communication est constitué d'un ensemble de liaisons de données et de nœuds. Il constitue l'ensemble des ressources mises à la disposition des équipements terminaux pour échanger des informations [7].

II.2 Catégorie des réseaux

La classification traditionnelle, fondée sur la notion d'étendue géographique, correspond à un ensemble des contraintes que le concepteur devra prendre en compte lors de la réalisation de son réseau. Généralement, On distingue quatre catégories de réseaux informatiques :

- **PAN (Personal Area Network) :** Les réseaux personnels, interconnectent sur quelques mètres des équipements personnels tels que terminaux GSM, portables, organisateurs, etc,

d'un même utilisateur.

- **LAN (Local Area Network)** : Réseau local d'étendue limitée à une circonscription géographique réduite (bâtiment...), ces réseaux destinés au partage local de ressources informatiques (matérielles ou logicielles) offrent des débits élevés de 10 Mbit/s à 10 Gbit/s.
- **MAN (Metropolitan Area Network)** : D'une étendue de l'ordre d'une centaine de kilomètres, les MAN sont généralement utilisés pour fédérer les réseaux locaux ou assurer la desserte informatique de circonscriptions géographiques importantes (réseau de campus).
- **WAN (Wide Area Network)** : Ces réseaux assurent généralement le transport d'information sur de grandes distances. Lorsque ces réseaux appartiennent à des opérateurs, les services sont offerts à des abonnés contre une redevance. Les débits offerts sont très variables, de quelques kbit/s à quelques Mbit/s [7].

II.3 Topologies des réseaux

Une topologie caractérise la façon dont les différents équipements réseaux sont positionnés les uns par rapport aux autres.

On distingue la topologie physique, relative au plan du réseau, de la topologie logique, qui précise la façon dont les informations circulent au plus bas niveau. Les interconnexions entre nœuds du réseau s'effectuent en liaison point à point, c'est-à-dire un avec un, ou en multipoints, soit n avec n .

- **Topologie en bus** : La topologie en bus (support linéaire) repose sur un câblage, sur lequel viennent se connecter des nœuds (postes de travail, équipements d'interconnexion, périphériques). Il s'agit d'un support multipoints. Le câble est l'unique élément matériel constituant le réseau et seuls les nœuds génèrent les signaux.

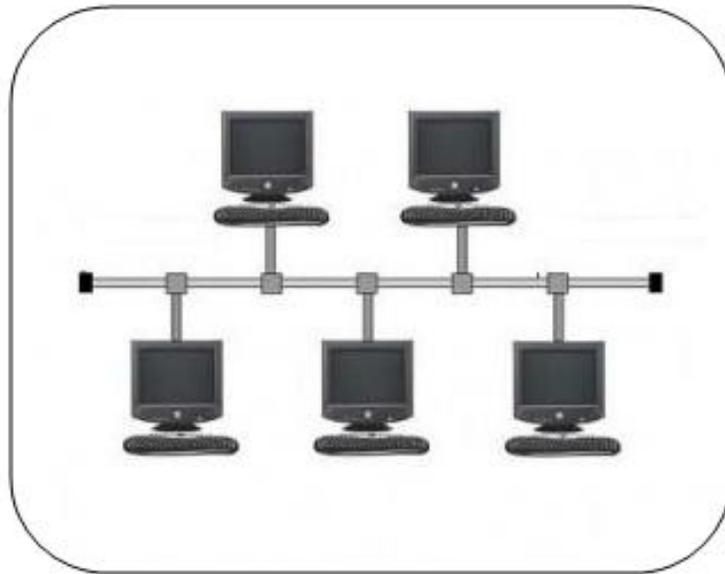


FIG. 1.3 – La topologie en bus.

- **Topologie en étoile :** La topologie en étoile repose, quant à elle, sur des matériels actifs. Un matériel actif remet en forme les signaux et les régénère. Ces points centraux sont appelés des concentrateurs (hubs), switches ou routeurs. Il est possible de créer une structure hiérarchique en constituant un nombre limité de niveaux.

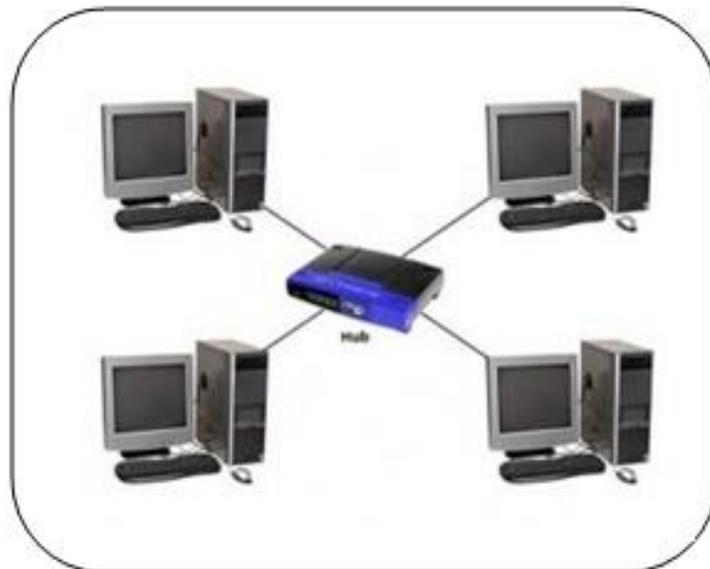


FIG. 1.4 – La topologie en étoile.

- **Topologie en anneau :** Cette topologie repose sur une boucle fermée constitué de liaisons

point à point entre périphériques .

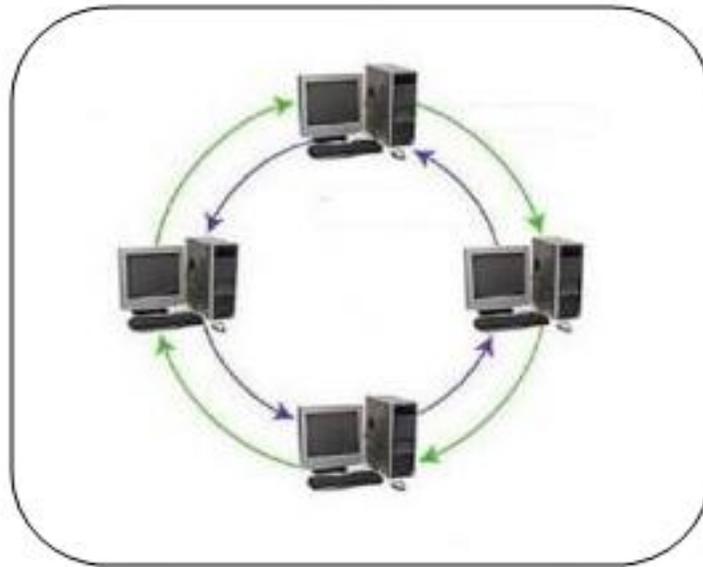


FIG. 1.5 – La topologie en anneau.

- **Topologie en arbre :** Dans l'architecture en arbre, les postes sont reliés entre eux de manière hiérarchique, à l'aide de concentrateurs. Cette connexion doit être croisée [8].

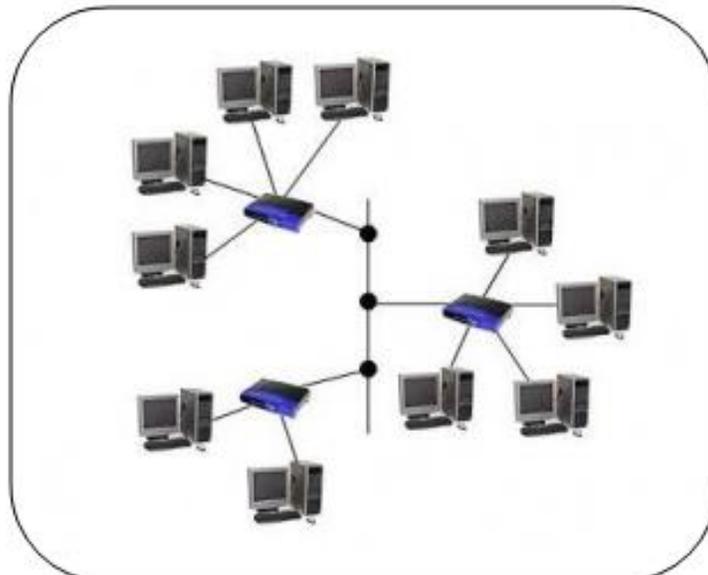


FIG. 1.6 – La topologie en arbre.

II.4 Les protocoles réseaux

Un protocole de communication est un ensemble des règles et des procédures permettant de définir un type de communication particulier. Les protocoles sont hiérarchisés en couches, pour décomposer et ordonner les différentes tâches. Il existe plusieurs familles de protocoles ou de modèles.

II.4.1 Le modèle OSI(Open System Interconnection)

Un aspect important dans l'ouverture des réseaux a été la mise en place d'un modèle de référence, OSI définit un modèle en sept couches réseau, présentes sur chaque station qui désire transmettre des données. Chaque couche dispose de fonctionnalités qui lui sont propres et fournit des services aux couches immédiatement adjacentes.[8]

- **NIVEAU 1 (Couche Physique)** : la couche physique assure un transfert de bits sur le canal physique (support). Cet effet, elle définit les supports et les moyens d'y accéder : spécifications mécaniques (connecteur), spécifications électriques (niveau de tension), spécifications fonctionnelles des éléments de raccordement nécessaires à l'établissement, au maintien et à la libération de la ligne. Elle détermine aussi les moyens d'adaptation (ETCD pour Équipement Terminal de Circuit de Données).
- **NIVEAU 2 (Couche Liaison de données)** : La couche liaison assure, sur la ligne, un service de transfert de blocs de données (trames) entre deux systèmes adjacents, en assurant le contrôle, l'établissement, le maintien et la libération du lien logique entre les entités. Les protocoles de niveau 2 permettent, en outre, de détecter et de corriger les erreurs inhérentes aux supports physiques.
- **NIVEAU 3 (Couche Réseau)** : La couche réseau assure, lors d'un transfert à travers un système relais, l'acheminement des données (paquets) à travers les différents noeuds d'un sous-réseau (routage). Les protocoles de niveau 3 fournissent les moyens d'assurer l'acheminement de l'appel, le routage, le contrôle de congestion, l'adaptation de la taille des blocs de données aux capacités du sous-réseau physique utilisé. Elle offre, en outre, un service de facturation de la prestation fournie par le sous-réseau de transport.
- **NIVEAU 4 (Couche Transport)** : La couche transport est la couche pivot du modèle OSI. Elle assure le contrôle du transfert de bout en bout des informations (messages) entre les deux systèmes d'extrémité. La couche transport est la dernière couche de contrôle des informations,

II.4.2 Le modèle TCP/IP(Transmission Control Protocol/Internet Protocol)

Cette architecture, dite TCP/IP, est à la source du réseau Internet. Elle est aussi adoptée par de nombreux réseaux privés, appelés intranet. Les deux principaux protocoles définis dans cette architecture sont les suivants :

- **IP (Internet Protocol)**, de niveau réseau, qui assure un service sans connexion.
- **TCP (Transmission Control Protocol)**, de niveau transport, qui fournit un service fiable avec connexion.

TCP/IP définit une architecture en couches qui inclut également, sans qu'elle soit définie explicitement, une interface d'accès au réseau. En effet, de nombreux sous-réseaux distincts peuvent être pris en compte dans l'architecture TCP/IP, de type aussi bien local qu'étendu.

- **La couche Accès réseau** : C'est la couche responsable de la transmission physique des données. L'information peut être cheminée sur différents supports avant d'atteindre la destination.
- **La couche Internet** : C'est la couche qui indique où les données doivent être envoyées, sans garantie que la destination sera bien atteinte. Elle peut utiliser les protocoles IP (Internet Protocol) et ICMP (Internet Control Message Protocol). ICMP permet de vérifier que des messages peuvent être échangés et de gérer les erreurs de transmission. IP est utilisé pour la plupart des communications sur Internet.
- **La couche Transport** : Repose sur deux protocoles : TCP et UDP (User Datagram Protocol). TCP s'assure que les paquets sont reçus dans le même ordre qu'ils ont été envoyés et que les paquets perdus sont à nouveau envoyés. TCP est donc un moyen de transmission fiable puisqu'il s'assure que les paquets sont arrivés. UDP est un protocole simplifié. Cela permet de transmettre des informations plus rapidement qu'avec TCP puisqu'il y a finalement moins d'informations échangées.
- **La couche Application** : C'est celle qui permet aux utilisateurs finaux de communiquer sur internet avec des protocoles tels que Telnet, pour agir sur un serveur [9].

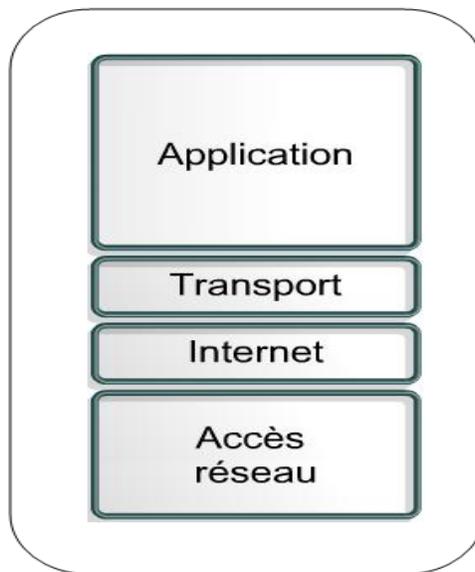


FIG. 1.8 – Présentation graphique du modèle TCP/IP.

II.5 Internet et intranet

1. **Internet** : L'internet est par définition un ensemble de réseaux d'ordinateurs interconnectés, utilisant le protocole TCP/IP .

C'est un service donnant l'accès à un réseau mondial mettant en contact divers médiums de communication et des serveurs, procurant aux utilisateurs une possibilité de partage d'informations, de recherches sur des sujets, d'échange de messages et dossiers à l'aide des courriers électroniques.

2. **Intranet** :Le terme intranet est souvent utilisé pour faire référence à une connexion privée de réseaux locaux et étendus qui appartiennent à une organisation. Cette connexion a été conçue de manière à être uniquement accessible aux membres, aux employés de l'organisation ou à d'autres personnes disposant d'une autorisation [10].

III L'architecture Client/Serveur

III.1 Concept de Client/Serveur

Le concept de base utilisé pour arranger la coopération entre les applications sur un réseau est celui du client/serveur. Chaque application est composée de logiciel serveur et logiciel client.[11]

Un logiciel serveur, peut correspondre à plusieurs logiciels clients développés dans différents

environnements. Afin de bien comprendre l'organisation client/serveur, nous définissons les notions suivantes :[13]

- **Service** :C'est un ensemble de fonctions qui gère des ressources mises à la disposition des programmes exploité par des clients.
- **Serveur** :C'est un programme qui offre un service à m client sur le réseau. Le serveur est utilisé pour le stockage et le partage d'informations. Elle sont stockées dans une base de données.
- **Client** :Contacte un serveur qui lui fournit des services.

III.2 Les trois niveaux d'abstraction d'une application :

En règle générale, une application informatique peut être découpée en trois niveaux d'abstraction distincts :[12]

- **La couche de présentation**, encore appelée IHM(Interface Homme-Machine), permet, l'interaction de l'application avec l'utilisateur. Cette couche gère les saisies au clavier, à la souris et la présentation des informations à l'écran. Dans la mesure du possible, elle doit être conviviale et ergonomique.
- **La logique applicative, les traitements**, décrivant les travaux à réaliser par l'application. Ils peuvent être découpés en deux familles :
 - ✓ Les traitements locaux, regroupant les contrôles effectués au niveau du dialogue avec l'IHM, visant essentiellement le contrôle et l'aide à la saisie ;
 - ✓ Les traitements globaux, constituant l'application elle-même. Cette couche, appelée Business Logic ou couche métier, contient les règles internes qui régissent une entreprise donnée.
- **Les données**, ou plus exactement l'accès aux données, regroupant l'ensemble des mécanismes permettant la gestion des informations stockées par l'application.

Ces trois niveaux peuvent être imbriqués ou répartis de différentes manières entre plusieurs machines physiques.

Le noyau de l'application est composé de la logique de l'affichage et la logique des traitements. Le découpage et la répartition de ce noyau permettent de distinguer les architectures applicatives suivantes :

- L'architecture 1-tier ;
- L'architecture 2-tiers ;

- L'architecture 3-tiers ;
- Les architectures n-tiers.

III.2.1 L'architecture un tiers

Dans une application un tiers, les trois couches applicatives sont intimement liées et s'exécutent sur le même ordinateur. On ne parle pas ici d'architecture client-serveur, mais d'informatique centralisée.

Une application web désigne à l'origine une application pouvant être exécutée par le biais d'un navigateur Internet. Un webmail, un service bancaire en ligne ou un moteur de recherche sont donc des applications web .

Dans un contexte multi-utilisateurs, on peut rencontrer deux types d'architecture mettant en oeuvre des applications un tiers :

- Des applications sur site central.
- Des applications réparties sur des machines indépendantes communiquant par partage de fichiers [12].

III.2.2 L'architecture deux tiers

L'architecture à deux niveaux 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 [12].

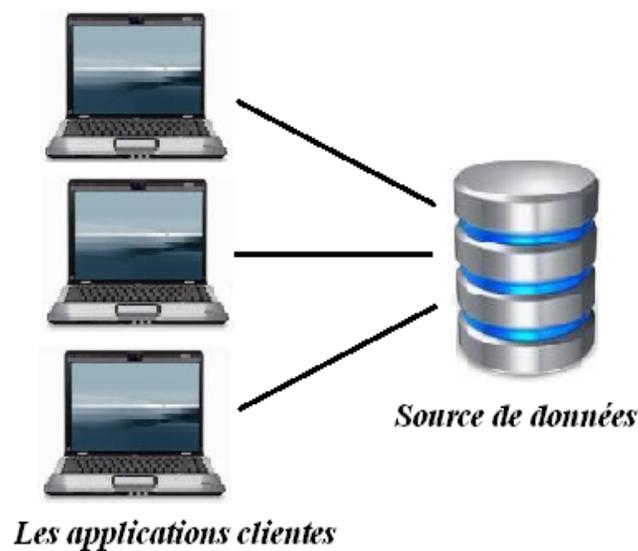


FIG. 1.9 – Architecture à 2 niveaux.

Le modèle client/serveur est particulièrement recommandé pour des réseaux nécessitant un grand niveau de fiabilité, ses principaux atouts sont :[12]

- **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 les problèmes de redondance et de contradiction.
- **Meilleure sécurité** : Lors d'une connexion, un PC client ne voit que le serveur, et non les autres PC clients. De même, les serveurs sont en général très sécurisés contre les attaques de pirates.
- **Meilleure fiabilité** : En cas de panne, seul le serveur fait l'objet d'une réparation, et non le PC client.
- **Facilité d'évolution** : Une architecture client/serveur est évolutive car il est très facile de rajouter ou d'enlever des clients, et même des serveurs.

III.2.3 L'architecture trois-tiers

Cette architecture, également appelée client-serveur de deuxième génération ou client-serveur distribué sépare l'application en 3 niveaux de services distincts :[12]

- **Premier niveau** : L'affichage et les traitements locaux (contrôles de saisie, mise en forme de données...) sont pris en charge par le poste client.

- **Deuxième niveau** : Les traitements applicatifs globaux sont pris en charge par le service applicatif.
- **Troisième niveau** : Les services de base de données sont pris en charge par un SGBD.

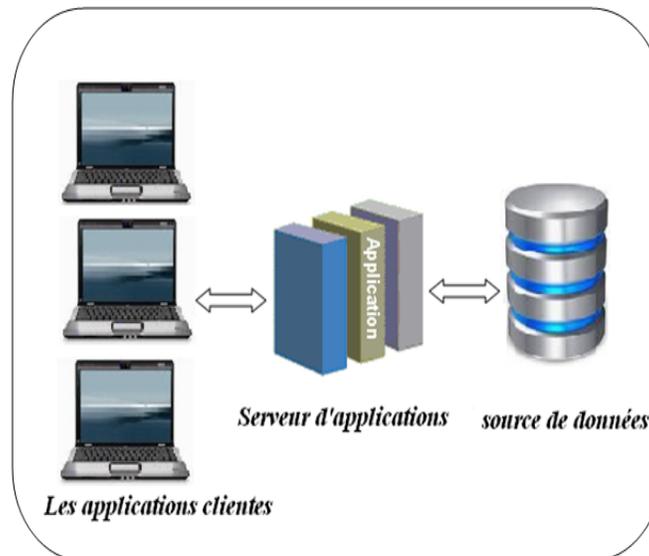


FIG. 1.10 – Architecture à 3 niveaux.

III.2.4 L'architecture n-tiers

L'architecture n-tiers a été pensée pour pallier aux limitations des architectures trois tiers et concevoir des applications puissantes et simples à maintenir. Ce type d'architecture permet de distribuer plus librement la logique applicative, ce qui facilite la répartition de la charge entre tous les niveaux. Cette évolution des architectures trois tiers met en oeuvre une approche objet pour offrir une plus grande souplesse d'implémentation et faciliter la réutilisation des développements [12].

Conclusion

A l'issue de ce chapitre, nous avons présenté les notions de base des systèmes d'informations, les réseaux, ainsi que l'architecture client/serveur. Dans le prochain chapitre, nous entamerons la présentation de l'organisme d'accueil et l'élaboration d'un cahier des charges.

Chapitre 2

Présentation de l'organisme d'accueil

Introduction

I Présentation Générale De La BMT

Pour faire connaître l'entreprise prestataire de services nous allons évoquer tout d'abord, l'historique de BMT, en suite, la situation géographique de BMT et enfin, la présentation des différentes structures de BMT et leurs activités.

I.1 Création (Joint venture)

Dans son plan de développement 2004-2006, l'entreprise portuaire de Béjaia avait inscrit à l'ordre du jour le besoin d'établir un partenariat pour la conception, le financement, l'exploitation et l'entretien d'un terminal à conteneurs au port de Béjaia. Dès lors l'EPB s'est lancée dans la tâche d'identifier les partenaires potentiels et a arrêté son choix sur le groupe PORTEK qui est spécialisé dans le domaine de la gestion des terminaux à conteneurs. Le projet à été présenté au conseil de la participation de l'état (CPE) en février 2004, le CPE a donné son accord au projet en mai 2004.

Sur accord du gouvernement "BMT" Bejaia Mediterranean Terminal a vu le jour avec la jointe venture de l'entreprise portuaire de Bejaia (EPB) à 51% et PORTEK une société singapourienne à 49%, PORTEK est un opérateur de terminaux spécialisé dans les équipements portuaires, il est présent dans plusieurs ports dans le monde. Aujourd'hui Bejaia Mediterranean Terminal " BMT " est une spa au capital de 180 000 000 00 da.

I.2 Présentation de la BMT

BMT est créée comme une société par actions, c'est une entreprise prestataire de services spécialisés dans le fonctionnement, l'exploitation, et la gestion du terminal à conteneurs. Pour atteindre son objectif, elle s'est dotée d'un personnel compétant particulièrement formé dans les opérations de gestion du terminal. Elle dispose d'équipements d'exploitation des plus perfectionnés pour les opérations de manutention et d'acconage afin d'offrir des prestations de services de qualité, d'efficacité et de fiabilité en des temps records et à des coûts compétitifs. BMT offre ses prestations sur la base de 24h/7j. Le niveau de la technologie mis en place et la qualité des infrastructures et équipements performants (portiques de quai, portiques gerbeurs) font aujourd'hui du port de

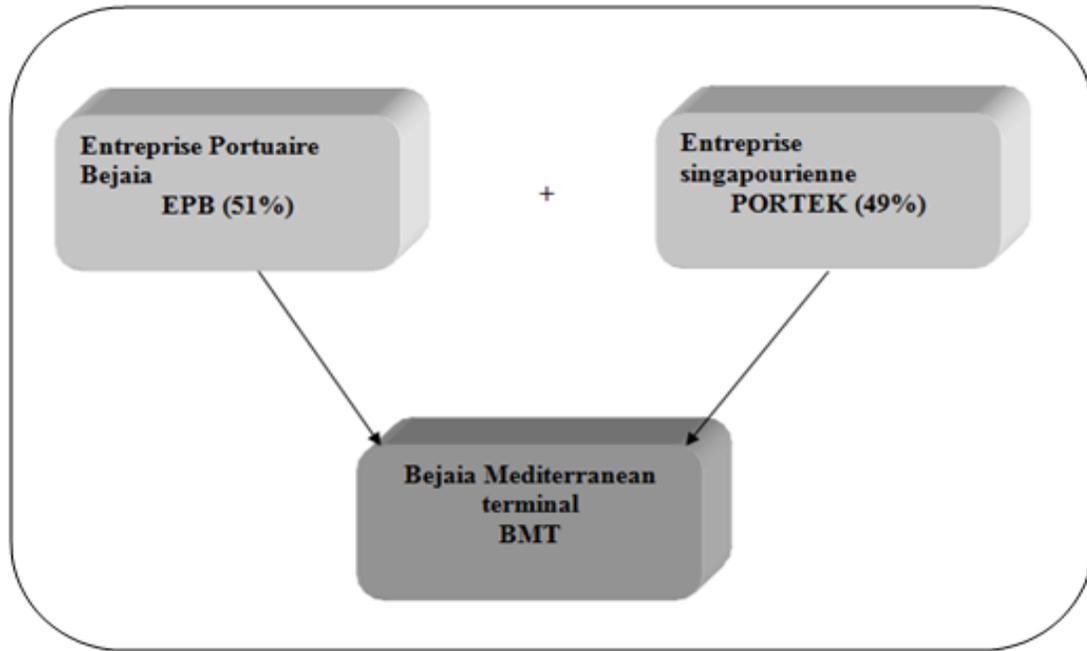


FIG. 2.1 – La joint venture de BMT.

Bejaia et de BMT, le premier terminal moderne d'Algérie avec une plate-forme portuaire très performante.

I.2.1 La structure de l'entreprise

1. **Direction générale (DG)** A sa tête le directeur général qui gère l'entreprise, a le pouvoir de décision, administrer l'entreprise, assigner des directives pour les différentes structures et faire entre les directions de l'entreprise.
2. **Direction des ressources humaines (DRH)**
 - **Service ressources humaines** : Mettre en œuvre des systèmes de gestion intégré à la stratégie de l'entreprise et qui traduise une adéquation entre les impératifs économiques et les attentes du personnel. Pour cela la véritable importance de cette structure réside dans la recherche de meilleur potentiel, le conservé on lui offrant les meilleurs conditions (salaire, climat de travail et environnement, formation).
 - **Service des moyens généraux** : Chargé des achats et de la gestion des stocks de l'entreprise.
 - **Service hygiène et sécurité** : Assure la sécurité de la marchandise, du parc à conteneurs et la propreté de l'entreprise et de son environnement.

3. Direction des opérations (DO)

Assure la planification des escales, de parc à conteneurs et la planification des ressources (humaines et matériels). Elle prend en charge les opérations de manutentions, comme la réception des navires porte conteneurs et leurs chargement et déchargement, comme elle suit les opérations de l'acconage tel que : le suivi des livraisons, dépotages, mise a disposition des conteneurs vides, traitement des conteneurs frigorifiques.

4. Direction marketing (DM)

Veille à la marque de l'entreprise en se préoccupant en permanence d'entretenir des relations avec les clients. Elle vise à faire connaître ses missions, ses programmes, ses orientations et ses performances auprès de ses clients. Elle amène son environnement externe à prendre conscience de l'importance des démarches qu'elle entreprend dans le développement et l'amélioration de la qualité des services.

- **Service commercial** : Suit la facturation, la gestion de portefeuille client et le recouvrement des créances.
- **Département informatique** : Assure le bon fonctionnement du CTMS, la maintenance du parc informatique de l'entreprise et le développement de nouvelles applications aux différentes structures.

5. Direction des finances et de comptabilité (DFC)

Procède à l'enregistrement de toutes les opérations effectuées par l'entreprise au cours de l'année .Elle est constituée de deux services :

- **Service de comptabilité** : Procède au contrôle et l'enregistrement de toutes les factures d'achat, de présentation et d'investissement.
- **Service des finances** : Procède au règlement de toutes les factures d'un coté et de l'autre à l'encaissement de toutes les créances de l'entreprise émis à la banque.

6. Direction technique (DT)

Assure une maintenance préventive et curative des engins du parc à conteneurs.

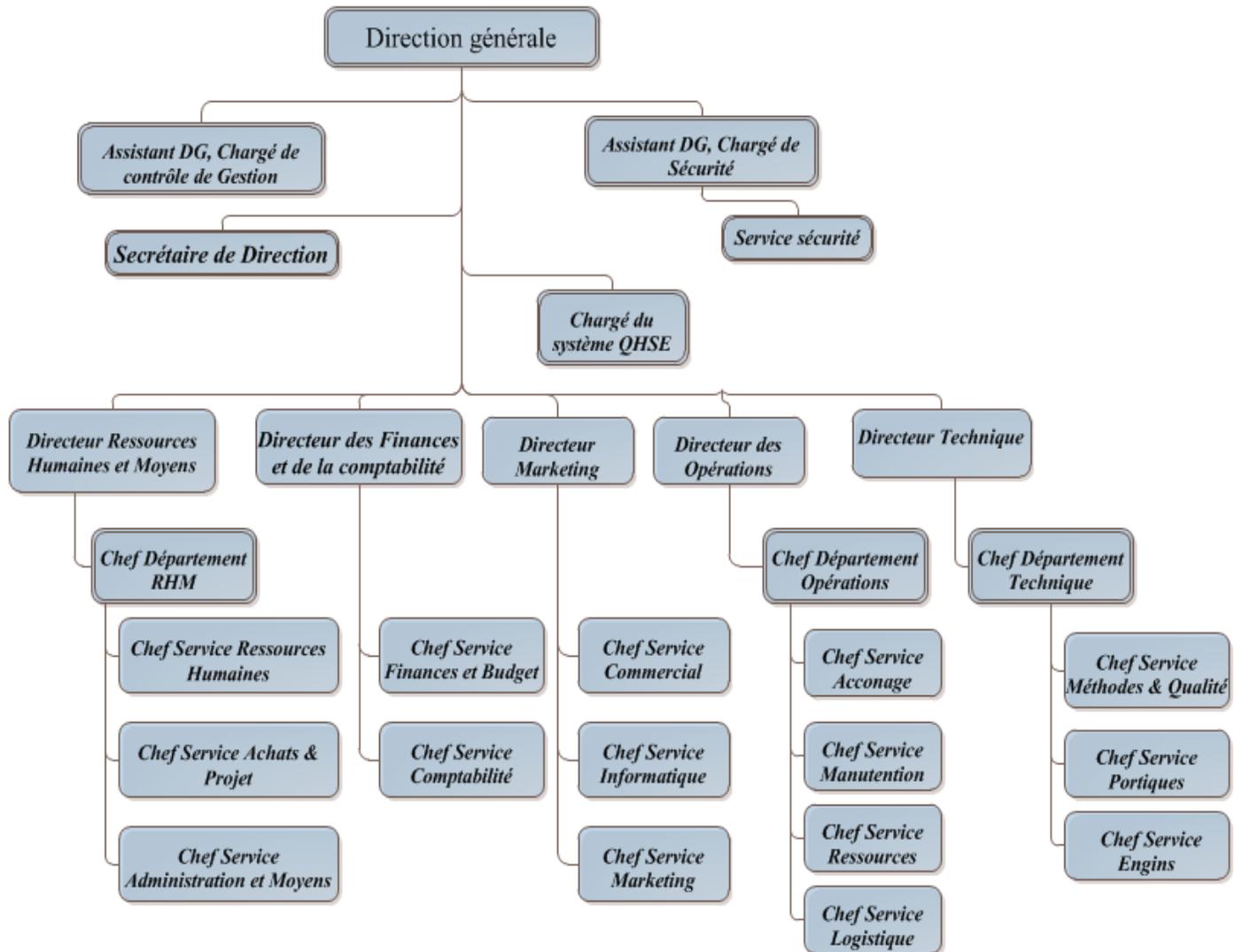


FIG. 2.2 – Organigramme général de la BMT.

I.3 Situation géographique

BMT se situe au niveau du port de Béjaïa, ce dernier est implanté au centre du pays et jouit d'une situation géographique stratégique. Elle se trouve à proximité de la gare ferroviaire, à quelques minutes de l'aéroport de Béjaïa et reliée au réseau routier national qui facilite le transport des marchandises conteneurisées de toute nature vers l'arrière pays et vers d'autres destinations telles que la banlieue d'Alger.

Position GPS : l'altitude nord : $36^{\circ} 45' 14''$

Longitude est : $05^{\circ} 05' 50''$

I.3.1 Situation du terminal à conteneurs



FIG. 2.3 – Situation du terminal à conteneurs.

II Les activités principales de la BMT et ses missions

II.1 Les missions de la BMT

L'activité principale de la BMT est le suivi, la gestion et l'exploitation du terminal à conteneur. BMT a pour mission principale de :

- Traiter dans les meilleures conditions de délais, de coûts et de sécurité, l'ensemble des navires porte conteneurs et des conteneurs.

- La manutention sur navire aussi bien le chargement et le déchargement des conteneurs et leurs entreposages dans les zones de stockage.
- Le service d'aconage sur les aires spécialisées ainsi que leurs livraisons.
- Le déchargement des céréales selon les capacités de la BMT.

Pour se faire, elle est dotée d'équipements performants et de système informatisé (CTMS) liés à la logistique pour pouvoir à la fois offrir des services de qualité, avec efficacité et fiabilité, ainsi que de satisfaire les différents besoins des clients.

II.2 Opérations du terminal

Bejaia Mediterranean Terminal reçoit annuellement un grand nombre de navires pour les quels elle assure les opérations de planification, de manutention et d'aconage avec un suivi et une traçabilité des opérations.

1. *Opération de planification :*

- Planification des escales : Programmation des accostages et des postes a quai.
- Planification déchargement/chargement.
- Planification du parc a conteneurs (visite, dépotage, enlèvement et restitution des conteneurs vides au parc).
- Planification des ressources : Equipes et moyens matériels.

2. *Opération de manutention :*

Elle comprend les opérations :

- D'embarquements, de débarquements des conteneurs.
- La réception des navires porte conteneurs.

Elle est opérationnelle de jour comme de nuit, répartie en deux shifts de 07h à 13h et de 13h à 19h avec un troisième shift over-time optionnel qui s'étaler jusqu'à 07h du matin.

3. *Opération d'aconage :*

- Transfert des conteneurs vers les zones d'entreposage.
- Transfert des conteneurs frigorifiques vers les zones "reefers".
- Suivi des visites du conteneur par les services concernés.
- Changement de position des conteneurs.
- Suivi des livraisons et des dépotages.
- Suivi des restitutions et des mises à quai.

- Mise à disposition des conteneurs vides pour empotage.

II.3 Les objectifs de la BMT

- Faire du terminal a conteneurs de BMT une infrastructure moderne, de répondre aux exigences les plus sévères en matière de qualité dans le traitement des conteneurs.
- La mise a disposition d'une nouvelle technologie dans le traitement des conteneurs pour :
 - ✓ Un gain de productivité.
 - ✓ Une réduction des coûts d'escale.
 - ✓ Une fiabilité de l'information.
 - ✓ Un meilleur service.
 - ✓ Sauvegarder la marchandise des clients.
 - ✓ Faire face a les concurrences nationales et internationales.
 - ✓ Propulser le terminal au stade international
 - ✓ Gagner des parts du marché.
- Pour les conteneurs le passage de 20 à 30 conteneurs l'heure.
- La réalisation de 150.000 EVP à l'horizon de 2015
- La création et la gestion d'un centre de formation.

III Présentation du champ d'étude

Le service de stage et formation de BMT est l'un des plus importants services administratifs de l'entreprise. Le but de notre projet est de l'automatisé, pour ce faire nous allons concevoir une application de gestion de stage et formation pour se service.

IV Problématique

Pour détecter les problèmes existants, nous avons interviewé le personnel du service stages et formations, qui nous a cité les différentes insuffisances relative à la gestion actuelle dont nous mentionnons les suivantes :

- Volume important des informations traitées manuellement au niveau de ce service, ce qui provoque parfois des erreurs dans l'établissement des documents.
- Accès non sécurisé aux informations manipulées.
- Possibilité d'erreur dans le remplissage des différents documents

- La sauvegarde de certaines données, et l'élaboration de la fiche suivie, ainsi que tous les calculs se font avec l'outil Excel se qui entraîne l'augmentation considérable de taux d'erreur et le risque de perte des données.
- La lenteur occasionnée par l'exécution des tâches manuellement.
- Manque du personnel au niveau de ce service.
- Nombre important des archives qui engendre d'une part une difficulté de stockage et d'autre part la complexité de recherche de données.

Conclusion

Dans ce chapitre nous avons présenté la structure de l'entreprise qui nous a permis d'appriivoiser avec l'environnement autour duquel nous allons développer notre projet, ainsi de recenser les principaux dysfonctionnements existants, à fin d'apporter une meilleure solution.

Deuxième partie

Conception et mise en oeuvre

Chapitre 3

Démarche adoptée

Introduction

Cette section porte sur la démarche de formulation des besoins des utilisateurs en regard d'une application de base de données. La phase d'analyse représente un travail systématique de recherche de documentation d'organisation et de suivi des besoins des parties intéressées, sachant que les souhaits et les besoins exprimés ne sont pas toujours explicites et peuvent changer dans le temps.

L'objectif premier de la phase d'analyse des besoins est de définir les besoins des utilisateurs en matière de données, cette tâche consiste à identifier les données présentes dans une organisation ou un secteur de l'organisation, dont il est nécessaire d'assurer la persistance, et de les modéliser sous forme de diagramme entité-association. Pour concevoir notre système, nous avons opté pour l'approche objet.

I La méthode 2TUP

I.1 Le processus unifié (UP)

Un processus unifié est un processus de développement logiciel construit sur UML, il est itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques. La gestion d'un tel processus est organisée d'après les 4 phases suivantes :

- **Préétude (initialisation)** : Cette phase conduit à définir la “ vision ” du projet, sa portée, sa faisabilité, son business case, afin de pouvoir décider au mieux de sa poursuite ou de son arrêt.
- **Élaboration** : Sert à confirmer l'adéquation du système aux besoins des utilisateurs et à livrer l'architecture de base.
- **Construction** : Consiste à concevoir et implémenter l'ensemble des éléments opérationnels (autres que ceux de l'architecture de base). C'est la phase la plus consommatrice en ressources et en effort.
- **Transition** : Permet de faire passer le système informatique des mains des développeurs à celles des utilisateurs finaux.

Ces activités de développement sont définies par six disciplines fondamentales, qui décrivent la modélisation métier, la capture des besoins, l'analyse et la conception, l'implémentation, le test et le déploiement.[14]

I.2 Le processus 2TUP

Le processus 2TUP apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise. Ce processus suit deux chemins :

- Architecture fonctionnel.
- Architecture technique.

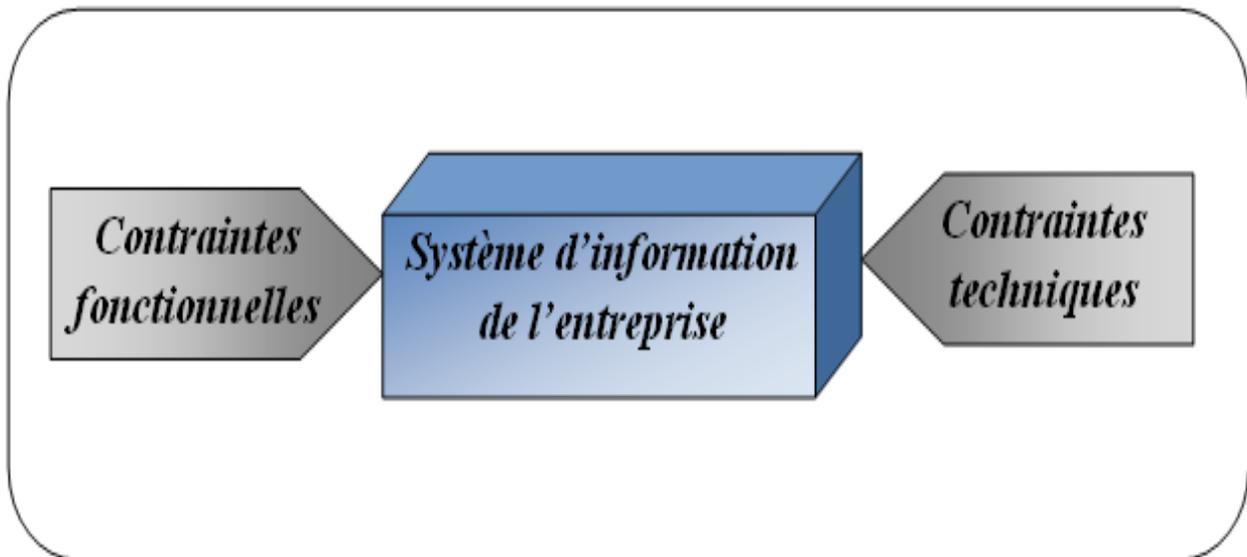


FIG. 3.1 – Le système d'information soumis à deux types de contraintes.

1. Branche fonctionnel (gauche) :

- Capture des besoins fonctionnels, qui produit le modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie, au plus tôt le risque de produire un système inadapté aux utilisateurs.
- L'analyse, qui consiste à étudier précisément la spécification fonctionnelle de manière à obtenir une idée de ce que va réaliser le système en terme de métier.

2. Branche architecture technique (droite) :

- La capture des besoins techniques, qui recense toutes les contraintes sur les choix de dimension et la conception du système. Les outils et le matériels sélectionnés ainsi que la prise en compte des contraintes d'intégration avec l'existant (pré requis d'architecture technique).
- La conception générique, qui définit ensuite les composants nécessaires à la construction de l'architecture technique. Cette conception est complètement indépendante des aspect fonctionnels. Elle a pour objectif d'uniformiser et de réutiliser les mêmes mécanismes pour tous

les systèmes. L'architecture technique construit le squelette du système, son importance est telle qu'il est conseillé de réaliser un prototype.

3. Branche conception (milieu) :

- La conception préliminaire, qui représente une étape délicate, car elle intègre le modèle d'analyse fonctionnelle dans l'architecture technique de manière à tracer la cartographie des composants du système à développer.
- La conception détaillée, qui étudie ensuite comment réaliser chaque composant.
- L'étape de codage, qui produit ses composants et teste au fur et à mesure les unités de code réalisées.
- L'étape de recette, qui consiste enfin à valider les fonctionnalités du système développé[14].

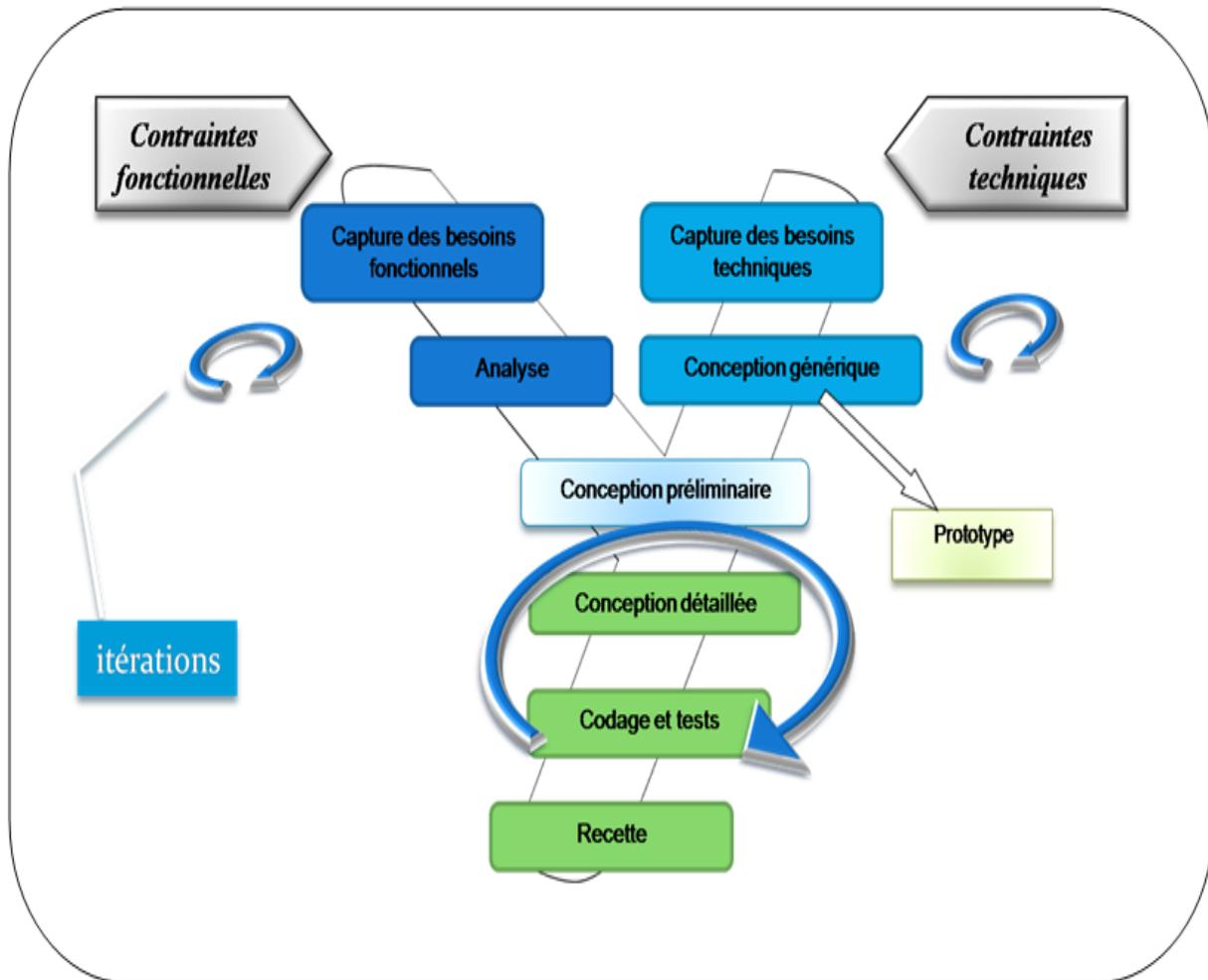


FIG. 3.2 – Le processus de développement en Y.

I.3 Processus de modélisation avec UML

UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. UML unifie à la fois les notations et les concepts orientés objets. 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 l'expression de besoin jusqu'au codage[15].

UML 2 s'articule autour de treize types de diagrammes, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Ces types de diagrammes sont répartis en deux groupes :

1. Six diagrammes structurels :

- **Diagramme de classes** : Il montre les briques de base statiques : classes, associations, interfaces, attributs, opérations, généralisations, etc.
- **Diagramme d'objets** : Il montre les instances des éléments structurels et leurs liens à l'exécution.
- **Diagramme de packages** : Il montre l'organisation logique du modèle et les relations entre packages.
- **Diagramme de structure composite** : Il montre l'organisation interne d'un élément statique complexe.
- **Diagramme de composants** : Il montre des structures complexes, avec leurs interfaces fournies et requises.
- **Diagramme de déploiement** : Il montre le déploiement physique des " artefacts " sur les ressources matérielles.

2. Sept diagrammes comportementaux :

- **Diagramme de cas d'utilisation** : Il montre les interactions fonctionnelles entre les acteurs et le système à étudier.
- **Diagramme de vue d'ensemble des interactions** : Il fusionne les diagrammes d'activité et de séquence pour combiner des fragments d'interaction avec des décisions et des flots.
- **Diagramme de séquence** : Il montre la séquence verticale des messages passés entre objets au sein d'une interaction.

- **Diagramme de communication** : Il montre la communication entre objets dans le plan au sein d'une interaction.
- **Diagramme de temps** : Il fusionne les diagrammes d'états et de séquence pour montrer l'évolution de l'état d'un objet au cours du temps.
- **Diagramme d'activité** : Il montre l'enchaînement des actions et décisions au sein d'une activité.
- **Diagramme d'états** : Il montre les différents états et transitions possibles des objets d'une classe[15].

Conclusion

Dans ce chapitre, nous avons présenté la démarche adoptée dans notre projet qui se résume en un processus unifié de développement logiciel construit autour d'UML. L'axiome fondateur de ce processus consiste à constater que toute évolution imposée aux système d'information peut se décomposer et se traiter parallèlement, suivant un axe fonctionnel et un axe technique qu'on va développer dans le chapitre suivant.

Chapitre 4

Etude préliminaire

Introduction

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

I Recueil des besoins fonctionnels

Il s'agit des fonctionnalités du système. Ce sont les besoins spécifiant un comportement d'entrée/sortie du système. Les besoins fonctionnels déduits à partir de notre étude se résument ci-dessous :

a) Gestion des formations : A travers notre application, nous devrons garder trace des employés et la gestion de leurs formation, qui est un des moyens de développer leurs compétences, pour que chacun enrichie au maximum son potentiel professionnel la BMT. Cette demande sera récupérée et validée par le responsable de structure.

La gestion des formations est caractérisée par les scénarios suivants :

- Mettre à jour les renseignements de formations effectuées et/ou celles validées par le responsable de structure .
- Mettre à jour les formations(Modification de la date et/ou lieu de formation, ajout, annulation d'une formation. . .) et les établissements.
- Déterminer les coûts généraux des formations (le budget prévisionnel).
- Méttre à jour les données individuelles, et collectives du personnel de l'entreprise.
- Editer les statistiques des formations effectuées.

b) Gestion des stagiaires : Le chargé des stages et formations reçoit une demande de stage. Le stagiaire est identifié par une matricule, nom, prénom, année d'étude, intitulé de thème, la durée du stage. La gestion des stages est caractérisée par les scénarios suivants :

- La mise à jour des informations des stagiaires
- Création des attestations.
- Mettre à jour un thème (modifier, supprimer, créer un thème).
- Chercher un thème .
- Editer les statistiques des stages.

II Description du contexte

II.1 Identification des acteurs

Un acteur est un type stéréotypé représentant une abstraction qui réside juste en dehors du système à modéliser. Un acteur représente un rôle joué par une personne ou une chose qui interagit avec le système. (la même personne physique peut donc être représentée par plusieurs acteurs en fonction des rôles qu'elle joue)[20].

1. **L'employé** : Remplir une demande de formation afin d'exprimer ses besoins.
2. **Le responsable de structure** : Son rôle consiste à Consulter les formations et le budget réservé à son service.
3. **Chargé de satge et formation** :
 - a) **Gestion des formations** : Son rôle consiste à consulter les informations concernant les employés, mise à jour des formations, des écoles et des employés (ajouts, suppression, modification), gérer des coûts de formations(modifier, créer), et enfin définir le type et le lieu de formations à suivre (interne, externe).
 - b) **Gestion des stages** : Consulter et mettre à jour les informations des stagiaires, suivi des stagiaires (récupération des rapports de stage, remises des attestassions de stage), gestion des thèmes et encadreurs .

II.2 Identification des messages

- **Le système reçoit les messages suivants** :
 - 1) Créer, enregistrer, imprimer une attestation de stage.
 - 2) Chercher une établissement .
 - 3) Chercher une formation.
 - 4) Créer, enregistrer, imprimer une demande de formation.
 - 5) Consulter les statistiques.
 - 6) Mise à jour d'un budget de formation.
 - 7) Consulter le budget des formations par direction.
 - 8) Mise à jour d'une formation.
 - 9) Mise à jour des informations d'un stagiaire.
 - 10) Mise à jour des informations d'un employé.

- 11) Editer les statistique des formations, des stages et budgets.
- 12) Mise à jour des établissements .
- ***Le système répond par les messages suivants :***
 - 13) Mise à jour des établissements.
 - 14) La demande à remplir.
 - 15) Afficher les formations disponible.
 - 16) Mise à jour d'une formation.
 - 17) Les budgets des formations pour chaque direction.
 - 18) Les messages de confirmations.
 - 29) Mise à jour des informations d'un stagiaire.
 - 20) Les statistiques des stages, formations et budget.
 - 21) Afficher les établissements disponible.
 - 22) Les attestation de stage.
 - 23) Mise à jour d'un budget.
 - 24) Mise à jour des informations d'un employé.
 - 25) Mise à jour des informations d'un stagiaire.

II.3 Modélisation du contexte

Le diagramme de contexte dynamique signifie la représentation superficielle de l'activité métier du système projeté. Après avoir recueilli les besoins fonctionnels et identifié les acteurs candidats du système, le condidats du système , le contexte dynamique sera finalement ainsi représenté :

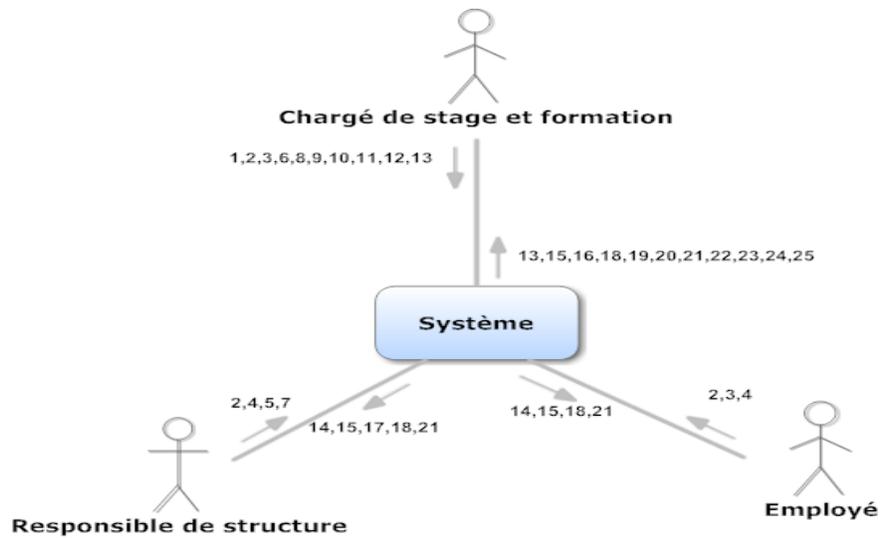


FIG. 4.1 – Diagramme du contexte dynamique.

Conclusion

Dans ce chapitre, nous avons spécifié les besoins fonctionnels. Par la suite, nous avons modélisé le contexte général du système, considéré comme étant une boîte noire, en identifiant les différents acteurs qui interagissent directement avec le système étudié, ainsi que les messages échangés.

Dans le chapitre suivant nous allons identifier les cas d'utilisations correspondant à chacun des acteurs de notre système.

Chapitre 5

Capture des besoins fonctionnels

Introduction

Ce chapitre traite du rôle que tient UML pour compléter la capture des besoins fonctionnels qui est la première étape de la branche gauche du cycle en Y, elle formalise et détaille ce qui a été ébauché au cours de l'étude préliminaire.

La technique des cas d'utilisation est la pierre angulaire de cette étape. Elle va nous permettre de préciser l'étude du contexte fonctionnel du système, en décrivant les différentes façons qu'auront les acteurs d'utiliser le futur système.

I Identification des cas d'utilisations

I.1 Définition

Un cas d'utilisation (use case) représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier. Un cas d'utilisation modélise un service rendu par le système. Il exprime les interactions acteurs/système et apporte une valeur ajoutée “notable ” à l'acteur concerné [15].

I.2 Les cas d'utilisation correspondant à chaque acteur

<i>N°</i>	<i>Cas d'utilisation</i>	<i>Acteur</i>
01	Authentification	Tous les acteurs du système
02	Gérer une formation	Ajouter
		Modifier
		Supprimer
		Chercher
03	Editer les statistiques	Statistique de formation
		Statistique de stage
		Statistique de budget
04	Gérer un stagiaire	Ajouter
		Modifier
		Supprimer
		Chercher
05	Gérer un employé	Ajouter
		Modifier
		Supprimer
		Chercher
06	Gérer une établissement	Ajouter
		Modifier
		Supprimer
07	Gérer un budget	Définir
		Modifier
		Supprimer
08	Gérer un thème	Ajouter
		Modifier
		Supprimer
		Chercher
09	Gérer une attestation	Créer
		Enregistrer
		Imprimer
10	Gérer un apprenti	Ajouter
		Modifier
		Supprimer
		Chercher
11	Consulter les statistiques de formations	Résponsable de structure Employé

12	Chercher une formation		Chargé de stage et formation
13	Chercher une établissement		Employé
14	Consulter un budget		Responsable de structure
15	Consulter une demande d'une formation valider		
16	Gérer une demande de formation	Créer	Employé
		Enregistrer	
		Imprimer	

TAB. 5.1 – Les cas d'utilisation du système à réaliser.

- *Diagramme de cas d'utilisation associé au "Chargé de stage et formation"*



FIG. 5.1 – Diagramme de cas d’utilisation associé au “Chargé de stage et formation”.

- *Diagramme de cas d'utilisation associé aux "Responsable de structure et l'employé"*

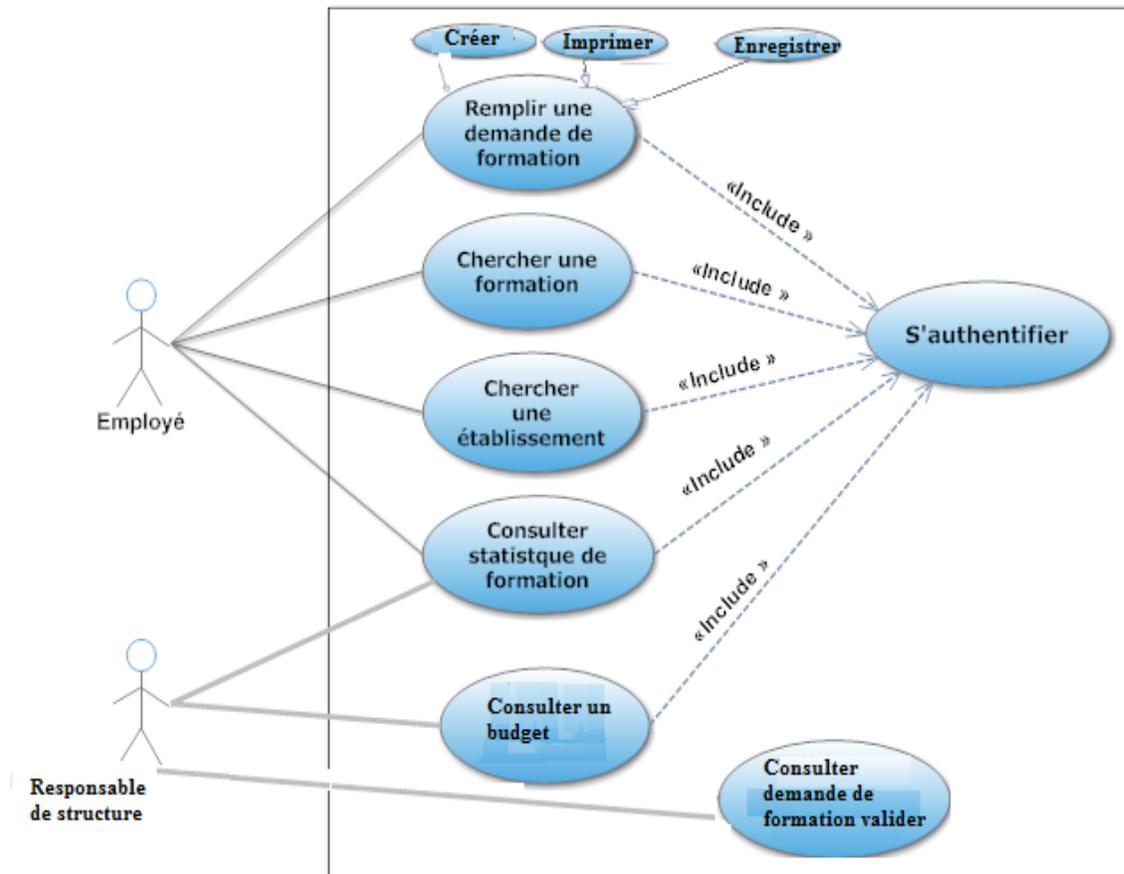


FIG. 5.2 – Diagramme de cas d'utilisation associés au "responsable de structure et l'employé".

- *Diagramme de cas d'utilisation générale*

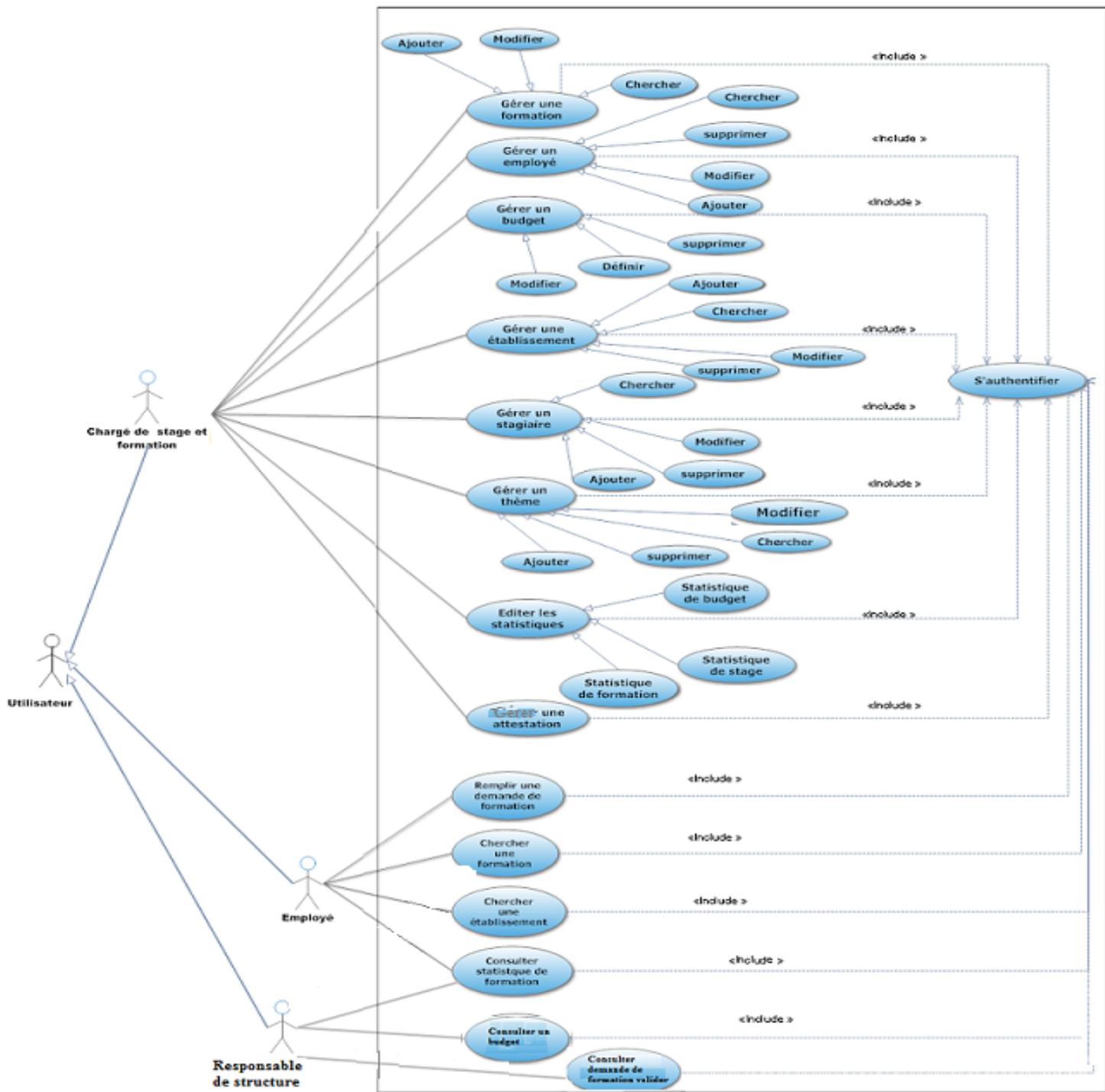


FIG. 5.3 – Diagramme de cas d'utilisation “Gérer stage et formation”.

II Description des cas d'utilisation

1. Cas d'utilisation "Gérer stagiaires" :

<p>Identification Titre :Gérer stagiaires. But :Gestion des stagiaires (Ajouter, modifier, supprimer) Acteur :Chef de service stage et formation.</p>
<p>Séquencement Pré-conditions : Chargé de stage et formation doit s'authentifier Enchainement nominal :</p> <ul style="list-style-type: none"> • Chargé de stage et formation sélectionne la rubrique stagiaire. • Saisie les informations relatives au nouveau stagiaire. • Dans le cas de modification, le chef de service choisit le stagiaire à modifier, puis effectuer les modifications souhaitées. • Dans le cas de suppression, Chargé de stage et formation choisit le stagiaire a supprimer puis valide <p>Post-conditions : Mise à jour de la base de données.</p>

TAB. 5.2 – Description du cas d'utilisation "Gérer stagiaire".

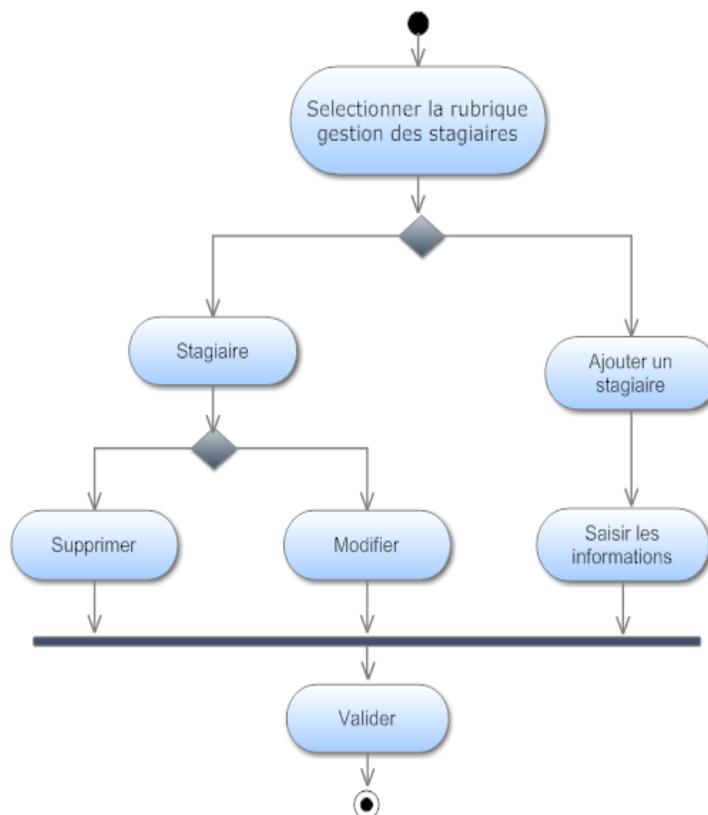


FIG. 5.4 – Diagramme d'activité gérer les stagiaires .

2. *Cas d'utilisation "Consulter formations" :*

Identification Titre : Consulter formations. But : Consulter les détails concernant une formation. Acteur : Chargé de stage et formation, responsable de structure, employé
Séquencement Pré-conditions : Les acteurs doivent s'authentifier. Enchaînement nominal : <ul style="list-style-type: none">● L'utilisateur sélectionne la rubrique consultation et choisit d'afficher la liste des formations (ou des écoles).● La liste des formations (ou des écoles) s'affiche.● L'utilisateur sélectionne la formation (ou l'école) dont il veut consulter les informations.● Les informations concernant la formation (ou l'école) s'affichent. Exception : Le nombre des formations affichés est trop important alors l'utilisateur filtre la liste par structure, service et/ou école. Post-conditions

TAB. 5.3 – Description du cas d'utilisation "Consulter formations".

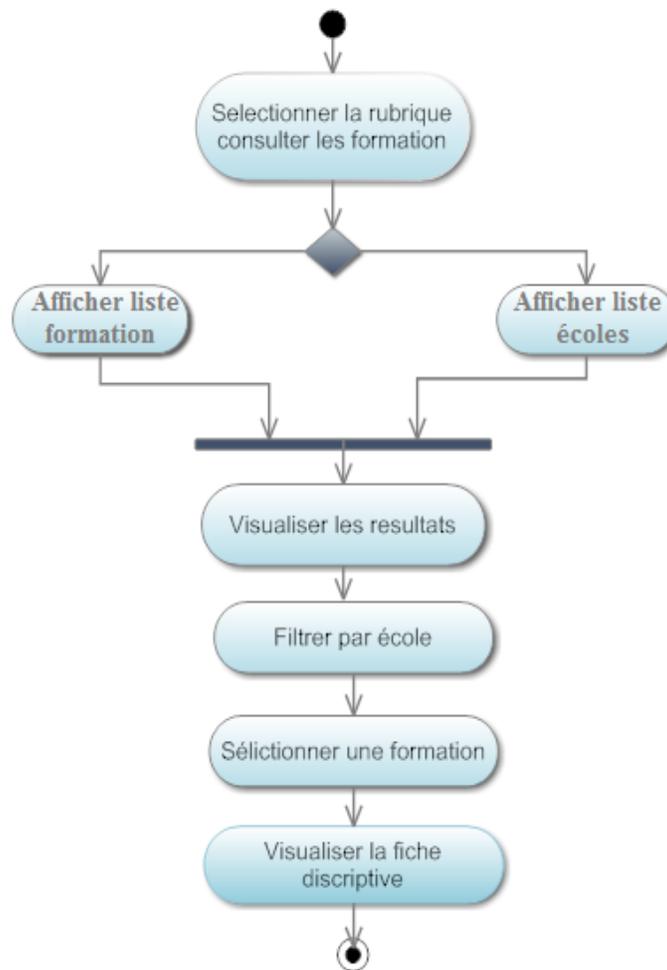


FIG. 5.5 – Diagramme d’activité consulter les formations .

3. *Cas d’utilisation “Chercher employé” :*

<p>Identification Titre : Chercher employé. But : Chercher un employé à partir de son identifiant Acteur : Chargé de stage et formation</p>
<p>Séquencement Pré-conditions : Le chargé de stage et formation doit s’authentifier. Enchaînement nominal :</p> <ul style="list-style-type: none"> • L’utilisateur lance une recherche à partir du mot clé ” numéro d’identification ”. • Le système affiche une page de résultat. • L’utilisateur sélectionne un employé, alors le système lui présente sa fiche descriptive détaillée. <p>Post-conditions : L’utilisateur a trouvé l’employé qu’il cherchait, ou un employé qui l’intéresse, voir plusieurs.</p>

TAB. 5.4 – Description du cas d’utilisation “Chercher employé”.

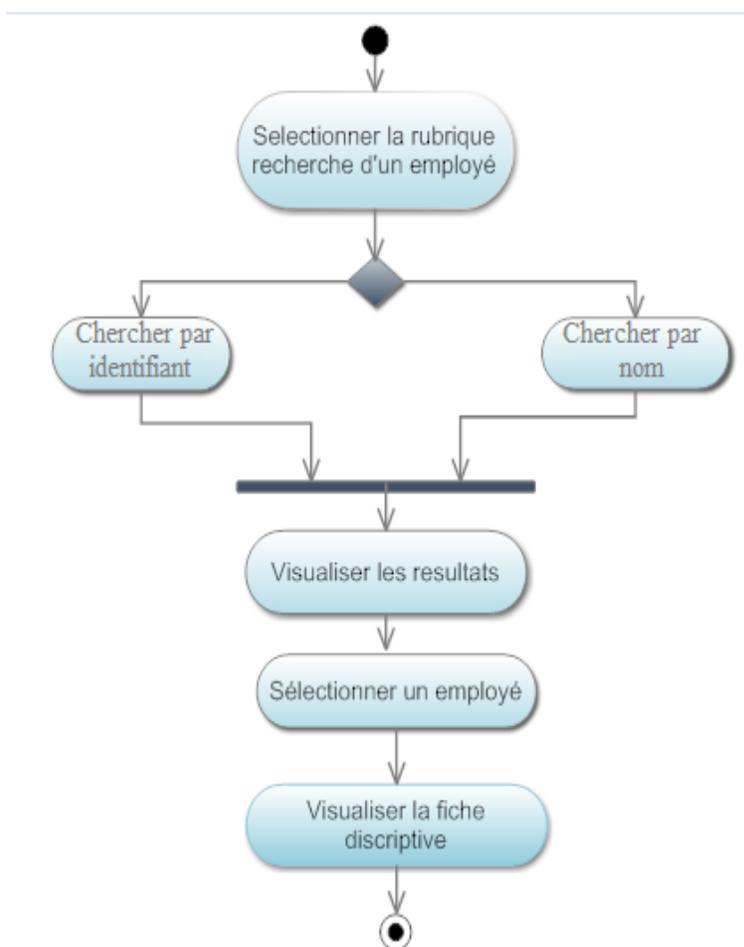


FIG. 5.6 – Diagramme d’activité chercher un employé.

4. *Cas d’utilisation “Gérer attestations” :*

<p>Identification Titre :Gérer attestations. But :Gestion des attestations (Ajouter, modifier, supprimer, chercher) Acteur :Chargé de stage et formation.</p>
<p>Séquencement Pré-conditions : Le chargé de stage et formation doit s’authentifier. Enchaînement nominal :</p> <ul style="list-style-type: none"> ● Chargé de stage et formation sélectionne la rubrique attestation de stage. ● Saisie les informations relatives a la nouvelle attestation, en cas de création. ● Dans le cas de modification, l’utilisateur choisit l’attestation à modifier, puis effectuée les modification souhaitées sur la fiche de ce bloc. ● Dans le cas de suppression, l’utilisateur choisit le bloc à supprimer puis valide. <p>Post-conditions : Mise à jour de la base de données.</p>

TAB. 5.5 – Description du cas d’utilisation “Gérer attestations”.

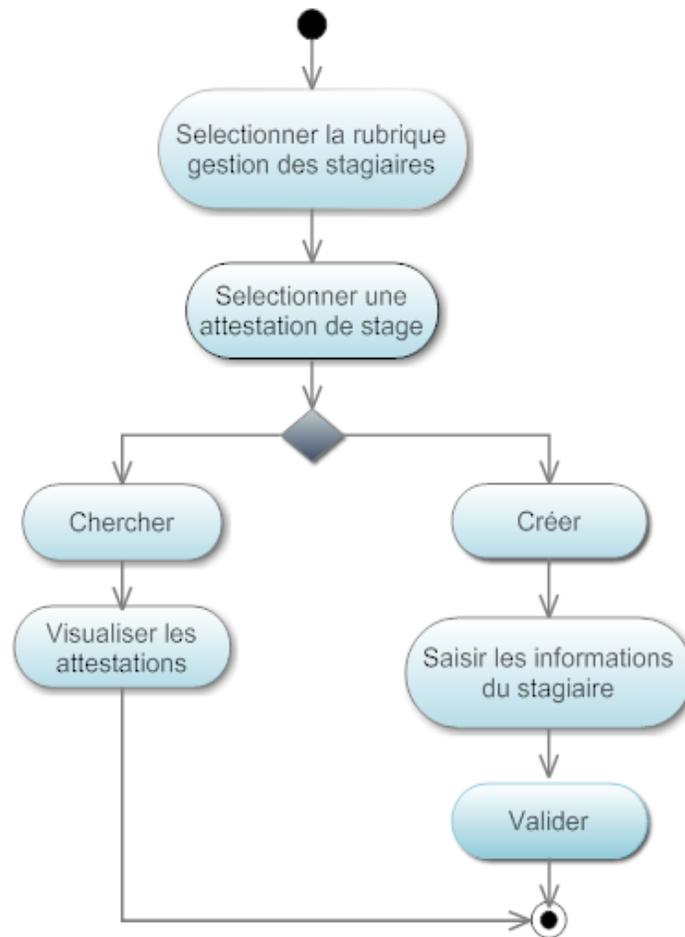


FIG. 5.7 – Diagramme d'activité gérer les attestations.

Conclusion

Dans ce chapitre nous avons décrit les différents cas d'utilisations ainsi que leurs diagrammes d'activités associés au projet. Dans le chapitre qui suit, nous allons spécifier les besoins techniques (Configuration matérielles, spécification logicielle).

Chapitre 6

Capture des besoins techniques

Introduction

La spécification technique est une activité de la branche droite du model en Y ; elle est primordiale pour la conception d'architecture. Cette phase couvre, en complément et en parallèle à la capture des besoins fonctionnels, toutes les contraintes et les choix dimensionnant la conception du système.

La capture des besoins techniques se présente comme suit :

- Capture des spécifications liées à la configuration matérielle.
- Capture des spécifications logicielles.

I Capture des spécifications matérielles

Le point de vue matériel développe la structure physique des machines et des réseaux sur lequel repose le système informatique.

I.1 Architecture client/serveur

L'architecture à deux niveaux met en œuvre un environnement de travail de niveau départemental et local. Un tel système répond généralement à la demande d'un métier particulier dans l'entreprise.

Etant donné que notre système est une architecture client/serveur, il possède :

- **Machines clients** : ce sont des ordinateurs de bureau ou toutes sortes de machine ayant un navigateur web qui permet d'accéder à internet.
- **Machine serveur (BDD)** : comporte une importante capacité de stockage, doit être disponible afin qu'on puisse y accéder à tout moment, et doit avoir une puissante capacité de traitement dans le cas où plusieurs clients y accèdent en même temps [12].

I.2 Architecture Oracle

L'architecture oracle est constituée d'une instance et d'une base de données appelée database. Une instance est constituée :

- D'une zone de mémoire partagée appelée System Global Area (SGA).
- D'un ensemble de processus d'arrière plan (PMON, SMON, logwriter, DBwriter) ayant chacun un rôle bien précis.
- D'un ensemble de processus serveur chargé de traiter les requêtes des utilisateurs.

- D'un fichier de contrôle, contenant les informations sur tous les autres fichiers de la base (nom, emplacement, taille).
- Des fichiers de Redo Log, contenant l'activité des sessions connectées à la base. Ce sont des journaux de transactions de la base. Ils sont organisés en groupe possédant le même nombre de membres. Et éventuellement, de fichiers de Redo Log archivés contenant les archives d'anciens fichiers de Redo Log.
- D'un ou plusieurs fichiers de données qui contiennent les données des tables de la base.

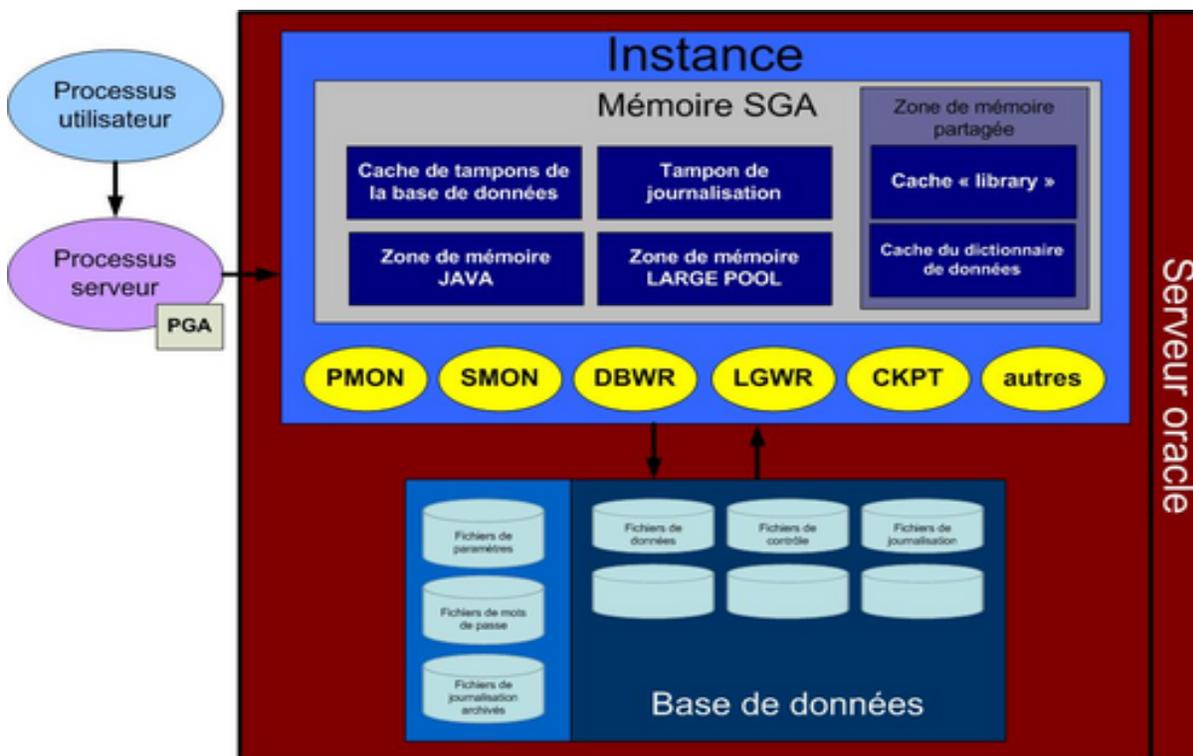


FIG. 6.1 – Architecture Oracle.

Une instance est l'ensemble des processus d'arrière-plan (background process) et de zones mémoire qui sont allouées au démarrage de la base de données, pour permettre l'exploitation des données.

I.2.1 Connexion utilisateur

Lorsqu'un utilisateur se connecte à la base de données, il ouvre une session. Les processus utilisateur sont alors pris en charge par les processus serveur qui sont chargés de traiter les requêtes des utilisateurs, notamment de charger dans le Database Buffer Cache les données nécessaires. Le

processus serveur communique (localement ou à travers le réseau) avec un processus utilisateur correspondant à l'application de l'utilisateur.

I.2.1.1 La PGA (Program Global Area) Mémoire privée des différents processus distribuée au moment de la connexion d'un client. Pour un processus serveur, la PGA contient :

- Une zone de tri (allouée dynamiquement lors d'un tri).
- Des informations sur la session.
- Des informations sur le traitement des requêtes de la session.
- Les variables de session.

Dans une configuration multithreaded, une partie de la PGA est en fait stockée dans la SGA (dans la Large Pool ou éventuellement dans la Shared Pool).

I.2.1.2 La SGA (System Global Area) Cette zone de mémoire partagée par les différents processus de l'instance est allouée au démarrage de l'instance et est libérée lors de l'arrêt de celle-ci.

Les principaux composants de la SGA sont :

- SPA : Shared Pool Area
- Database Buffer Cache
- Redo Log Buffer
- Large Pool
- Java Pool
- Streams Pool
- Reserved Area

Ces différentes zones mémoires sont configurées à l'aide du paramètre contenu dans le fichier de paramètres SPFILE. En dehors de la SGA, chaque processus possède une zone de mémoire privée appelée PGA (Program Global Area).

I.2.2 Le fichier de paramètres

Au démarrage, l'instance lit un fichier de paramètres qui contient des paramètres d'initialisation. Ce fichier est géré par le DBA. Les paramètres d'initialisation permettent notamment à l'instance :

- D'allouer la mémoire souhaitée aux différentes structures de la SGA
- De trouver le nom et l'emplacement des fichiers de contrôle de la base

Il existe 2 types de paramètres, les paramètres dynamiques et les paramètres statiques. Les paramètres dynamiques sont modifiables sans avoir besoin d'arrêter la base de données.

Les vues V\$SYSTEM_PARAMETER et V\$SYSTEM_PARAMETER2(idem V\$SYSTEM_PARAMETER avec une mise en forme des paramètres) permettent de connaître la valeur des paramètres de l'instance en cours de fonctionnement.

Règles :

- Les paramètres sont spécifiés sous la forme nom_paramètre = valeur.
- Tous les paramètres sont optionnels et ont une valeur par défaut.
- Des commentaires peuvent être inclus et commencent par le caractère #
- La valeur peut être spécifiée entre des guillemets doubles si elle contient des caractères spéciaux (égal, espace...)
- Les valeurs multiples sont spécifiées entre parenthèses, séparées par des virgules

I.2.3 La base de données

La base de données est l'ensemble des fichiers qui permettent de gérer les données stockées.

Une base de données est constituée :

- D'un fichier de contrôle.
- Des fichiers de Redo Log.
- D'un ou plusieurs fichiers de données.
- D'un fichier de paramètres binaire SPFILE<SID>.ORA.
- D'un fichier de mots de passe.
- Des utilisateurs et connexion à la base de données : A la création d'une base de données un ensemble d'utilisateurs sont créés, dont SYSTEM et SYS [17].

Schéma : Le terme schéma désigne l'ensemble des objets qui appartiennent à un utilisateur, ces objets sont préfixés par le nom de l'utilisateur qui les a créés [19].

II Capture des spécifications logicielles

Une fois les spécifications matérielles déterminées, on passe aux spécifications non fonctionnelles autrement dit techniques ou logicielles. C'est des fonctionnalités techniques que le système va assurer à l'utilisateur indépendamment du terme métier ou fonctionnel.

Dans ce qui suit, nous allons énumérer une liste de fonctionnalités techniques que notre système va pouvoir assurer et offrir aux utilisateurs.

- **Les exploitants** : Appelés aussi “ acteurs techniques ” du système, ce sont les acteurs qui bénéficient des fonctionnalités techniques du système :
 - ✓ Utilisateur : C’est les personnes qui utilisent d’une manière ou d’une autre les fonctionnalités du système (La majorité des acteurs de la branche fonctionnelle).
 - ✓ Administrateur : C’est la personne chargée de gérer toute la gestion de la plateforme ainsi que de celle des utilisateurs [18].

II.1 Les cas d’utilisations techniques

Un cas d’utilisation technique est destiné à l’exploitant. C’est une séquence d’actions produisant une valeur ajoutée opérationnelle ou purement technique [18].

II.1.1 Identification et description des cas d’utilisation techniques

1. Cas d’utilisation “Gérer un utilisateur ” : L’administrateur peut ajouter, supprimer, modifier un utilisateur.

<p>Identification Titre :Gérer Utilisateur. Acteur :Administrateur.</p>
<p>Pré-conditions : L’administrateur doit s’authentifier. Scénario nominal :</p> <ul style="list-style-type: none"> • A l’ajout, on affecte à chaque utilisateur un rôle, un mot de passe . • Une fois l’utilisateur authentifié par son compte, notre système le redirige selon son rôle vers son espace personnelle. • L’administrateur peut supprimer, migrer un utilisateur d’un rôle à un autre.

TAB. 6.1 – Description du cas d’utilisation technique “Gérer utilisateur”.

2. Cas d’utilisation “Gérer la sécurité ” :
 - ✓ Cas d’utilisation “Sauvegarder la BDD ”

<p>Identification Titre :Sauvegarder la BDD. Acteur :Tous les utilisateurs.</p>
<p>Pré-conditions : L'utilisateur doit être authentifié. Scénario nominal :</p> <ul style="list-style-type: none"> ● Il lance le script de sauvegarde. ● Le système sauvegarde la BDD dans un fichier. dmp et le stocke dans un répertoire spécifié.

TAB. 6.2 – Description du cas d'utilisation technique "Sauvegarder la BDD".

✓ Cas d'utilisation "Restaurer la BDD"

<p>Identification Titre :Restaurer la BDD. Acteur :Administrateur.</p>
<p>Pré-conditions : L'administrateur doit être authentifié. Scénario nominal :</p> <ul style="list-style-type: none"> ● Il lance le script de restauration. ● Le système importe le fichier .dmp sauvegardé auparavant et l'injecte dans la BDD en écrasant les données existantes.

TAB. 6.3 – Description du cas d'utilisation technique "Restaurer la BDD".

✓ Cas d'utilisation "S'authentifier"

<p>Identification Titre :S'authentifier. Acteur :Tous les utilisateurs.</p>
<p>Pré-conditions : L'utilisateur est inscrit dans le système. Scénario nominal :</p> <ul style="list-style-type: none"> ● L'utilisateur saisi ses identifiants dans la page d'authentification. ● Le système le dirige vers l'espace qui lui est approprié.

TAB. 6.4 – Description du cas d'utilisation technique "S'authentifier".

✓ Cas d'utilisation "Changer le mot de passe"

<p>Identification Titre :Changer le mot de passe Acteur :Tous les utilisateurs.</p>
<p>Pré-conditions : L'utilisateur doit s'authentifier. Scénario nominal :</p> <ul style="list-style-type: none"> • L'utilisateur ouvre sa page. • S'il a oublié le mot de passe il doit contacter l'administrateur. • Sinon il peut accéder après authentification sur son espace perso à l'option de changement de mot de passe. • Il doit saisir son ancien et nouveau mot de passe afin de valider le changement.

TAB. 6.5 – Description du cas d'utilisation technique “Changer le mot de passe”.

3. Cas d'utilisation “Manipuler les objets”

<p>Identification Titre :Manipuler les objets. Acteur :Tous les utilisateurs.</p>
<p>Pré-conditions : L'utilisateur doit s'authentifier. Scénario nominal :</p> <ul style="list-style-type: none"> • l'utilisateur désire agir sur le cycle de vie d'un ou plusieurs objets. • Il peut créer, modifier, supprimer un objet.

TAB. 6.6 – Description du cas d'utilisation technique “Manipuler les objets”.

4. Cas d'utilisation : “Consulter l'aide d'utilisation”

<p>Identification Titre :Consulter l'aide d'utilisation Acteur :Tous les utilisateurs.</p>
<p>Pré-conditions : L'utilisateur doit s'authentifier. Scénario nominal :</p> <ul style="list-style-type: none"> • Il consulte l'aide disponible sur la plateforme, afin de connaître les différentes fonctionnalités de l'application.

TAB. 6.7 – Description du cas d'utilisation technique “Consulter l'aide d'utilisation”.

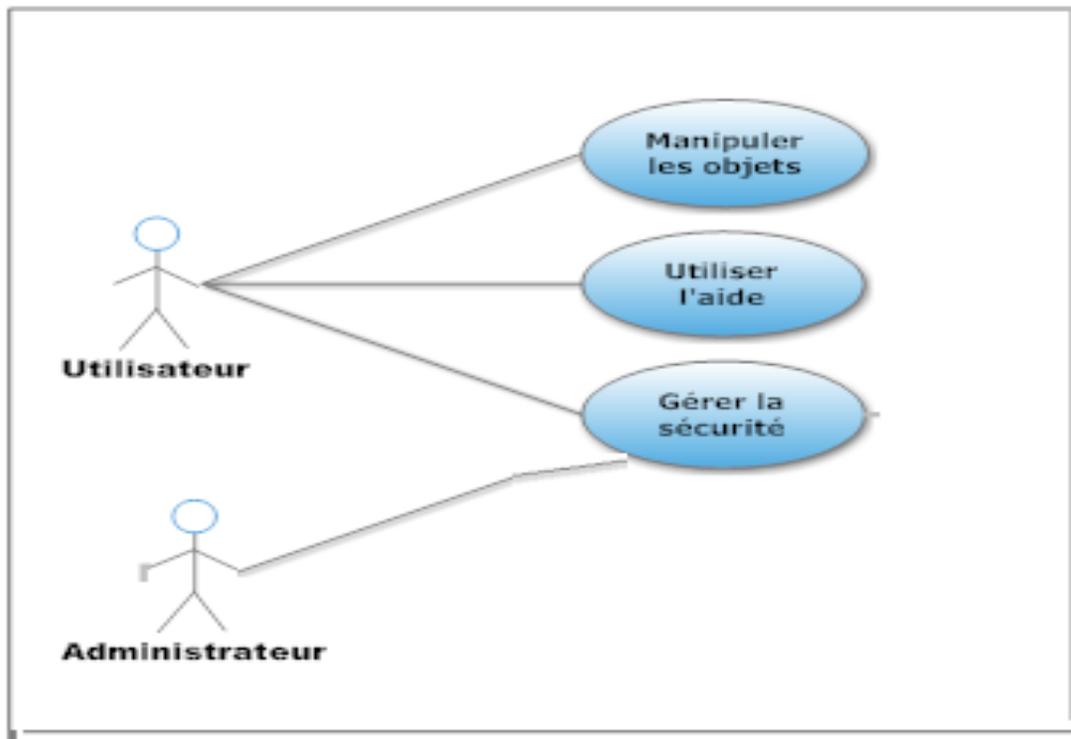


FIG. 6.2 – Modèle de spécification logicielle.

Conclusion

Dans la capture des besoins techniques, nous avons identifié les cas d'utilisation techniques relative a notre projet, présenté l'architecture client/serveur ainsi que les notions de bases liées à l'architecture oracle. La phase qui suit sera consacrée à l'analyse de notre système en se basant sur les principes de l'approche Orientée Objet .

Chapitre 7

Analyse

Introduction

Cette phase marque le démarrage de l'analyse objet du système à réaliser, le modèle commence à représenter le système vu de l'intérieur. Il se compose d'objets représentant une abstraction des concepts manipulés par les utilisateurs. Le modèle comprend par ailleurs deux points de vue, la structure statique et le comportement dynamique.

I Développement du modèle statique

Le développement du modèle statique constitue la deuxième activité de l'étape d'analyse. Elle se situe sur la branche gauche du cycle en Y. Il s'agit d'une activité itérative, fortement couplée avec la modélisation dynamique[16].

I.1 Diagramme de classe

Le diagramme de classes représente la structure statique du système en termes de classes et de relations entre elle. Son intérêt est de modéliser les entités du système d'information .

Le diagramme de classe permet de représenter l'ensemble des informations finalisées qui sont gérées par le domaine. Il met en évidence d'éventuelles relations entre ces classes[20].

Le diagramme de classes comporte six concepts :

- **Une classe d'entité** : Une classe représente la description abstraite d'un ensemble d'objets possédant les mêmes caractéristiques.
- **Attribut** : Un attribut représente un type d'information contenu dans une classe.
- **Association** : Une association représente une relation sémantique durable entre deux classes[20] .
- **Héritage** : L'héritage permet à une sous-classe de disposer des attributs et opérations de la classe dont elle dépend [16].
- **Méthodes** : Représentent l'ensemble des actions, procédures, fonctions ou opérations que l'on peut associer à une classe [20].
- **Composition** : La composition est une association qui permet de représenter un lien de type ensemble comprenant des éléments. Il s'agit d'une relation entre une classe représentant le niveau ensemble et 1 à n classes de niveau éléments. L'agrégation représente un lien structurel entre une classe et une ou plusieurs autres classes[20].

I.2 Le diagramme de classe de notre système

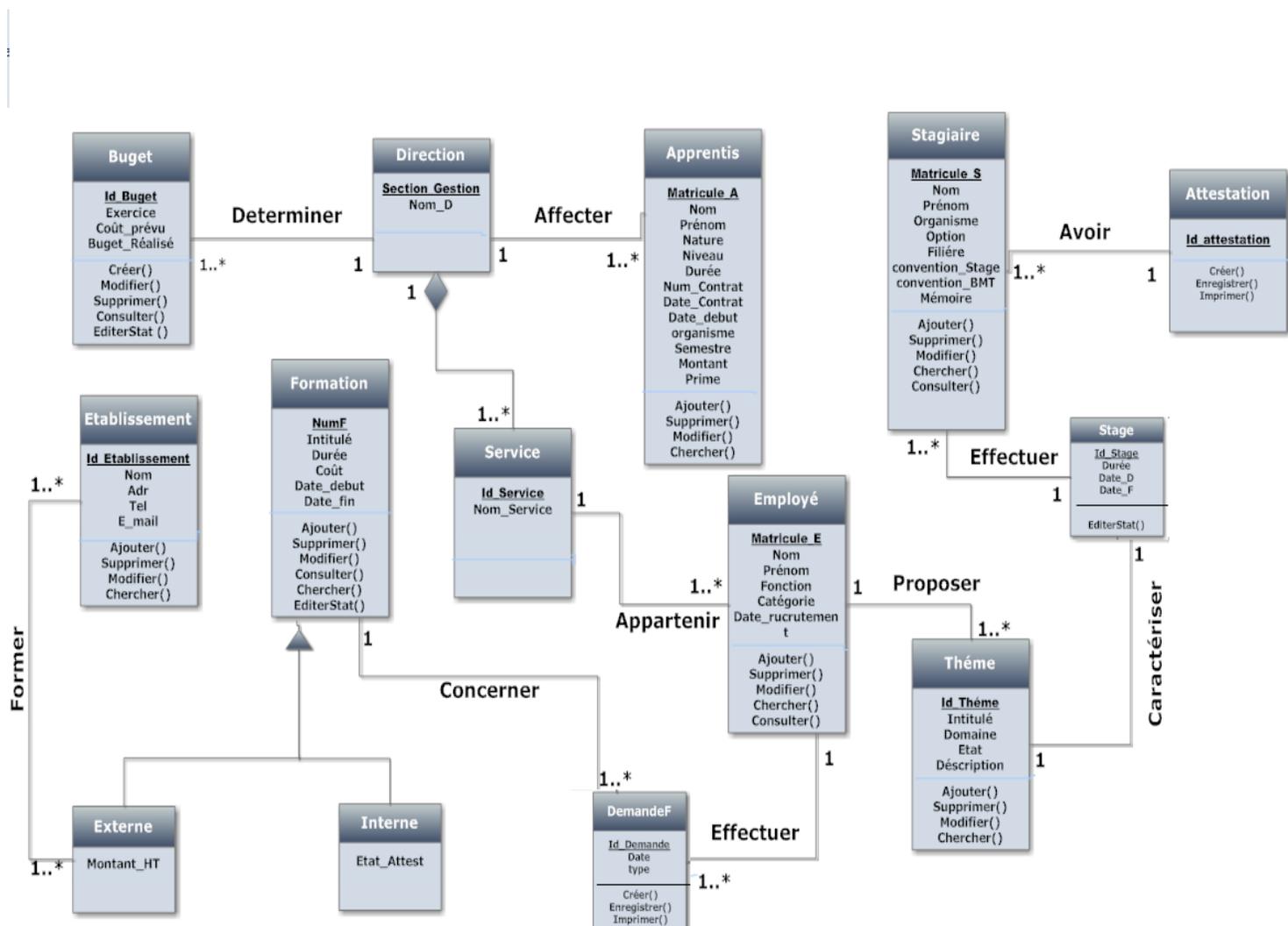


FIG. 7.1 – Diagramme de classe.

II Développement du modèle dynamique

Le développement du modèle dynamique constitue la troisième activité de l'étape d'analyse. Elle se situe sur la branche gauche du cycle en Y. Il s'agit d'une activité itérative, fortement couplée avec l'activité de modélisation statique[15].

II.1 Formaliser les scénarios

Un scénario est une suite spécifique d'interactions entre les acteurs et le système à l'étude. On peut dire que c'est une " instance " du cas d'utilisation, un chemin particulier dans sa combinatoire[16].

II.1.1 Diagramme de séquence

Le diagramme de séquence décrivant la dynamique d'un système, celle-ci contient fréquemment des créations et des destructions d'objets. La création d'objet est représentée par un message spécifique qui donne lieu au début de la ligne de vie du nouvel objet. La destruction d'objet est un message envoyé à un objet existant et qui donne lieu à la fin de sa ligne de vie. Il est représenté par une croix. L'objectif du diagramme de séquence est de représenter les interactions entre objets en indiquant la chronologie des échanges. Cette représentation peut se réaliser par cas d'utilisation en considérant les différents scénarios associés[21].

Les concepts de base utilisés dans ce diagramme sont les suivants :

- **Objet** : Un objet est une entité aux frontières bien définies, possédant une identité et encapsulant un état et un comportement. Un objet est une instance (ou occurrence) d'une classe[20].
- **Ligne de vie** : Une ligne de vie représente l'ensemble des opérations exécutées par un objet. Un message reçu par un objet déclenche l'exécution d'une opération. Le retour d'information peut être implicite (cas général) ou explicite à l'aide d'un message de retour.
- **Message synchrone et asynchrone** : Dans un diagramme de séquence, deux types de messages peuvent être distingués :
 - ✓ Message synchrone : Dans ce cas l'émetteur reste en attente de la réponse à son message avant de poursuivre ses actions. La flèche avec extrémité pleine symbolise ce type de message. Le message retour peut ne pas être représenté car il est inclus dans la fin d'exécution de l'opération de l'objet destinataire du message.
 - ✓ Message asynchrone : Dans ce cas, l'émetteur n'attend pas la réponse à son message, il poursuit l'exécution de ses opérations. C'est une flèche avec une extrémité non pleine qui symbolise ce type de message.
- **L'opérateur "alt"** : L'opérateur alt correspond à une instruction de test avec une ou plusieurs alternatives possibles. Il est aussi permis d'utiliser les clauses de type sinon.

- **L'opérateur "Loop"** : Loop correspond à une instruction de boucle qui permet d'exécuter une séquence d'interaction tant qu'une condition est satisfaite. Il est possible aussi d'utiliser une condition portant sur un nombre minimum et maximum d'exécution de la boucle en écrivant : loop min, max. Dans ce cas, la boucle s'exécutera au minimum min fois et au maximum max fois. Il est aussi possible de combiner l'option min/max avec la condition associée à la boucle.
- **L'opérateur "Ref"** L'opérateur ref permet d'appeler une séquence d'interactions décrite par ailleurs constituant ainsi une sorte de sous-diagramme de séquence.
- **L'opérateur "Opt"** L'opérateur opt (optional) correspond à une instruction de test sans alternative (sinon)[22].

II.2 Les diagrammes de séquence de notre système

II.2.1 Diagramme de séquence de cas d'utilisation "Authentification"

Tous les acteurs interagissent de la même façon avec le système pour le cas d'utilisation authentification. De ce fait, on crée un acteur généralisé nommé utilisateur.

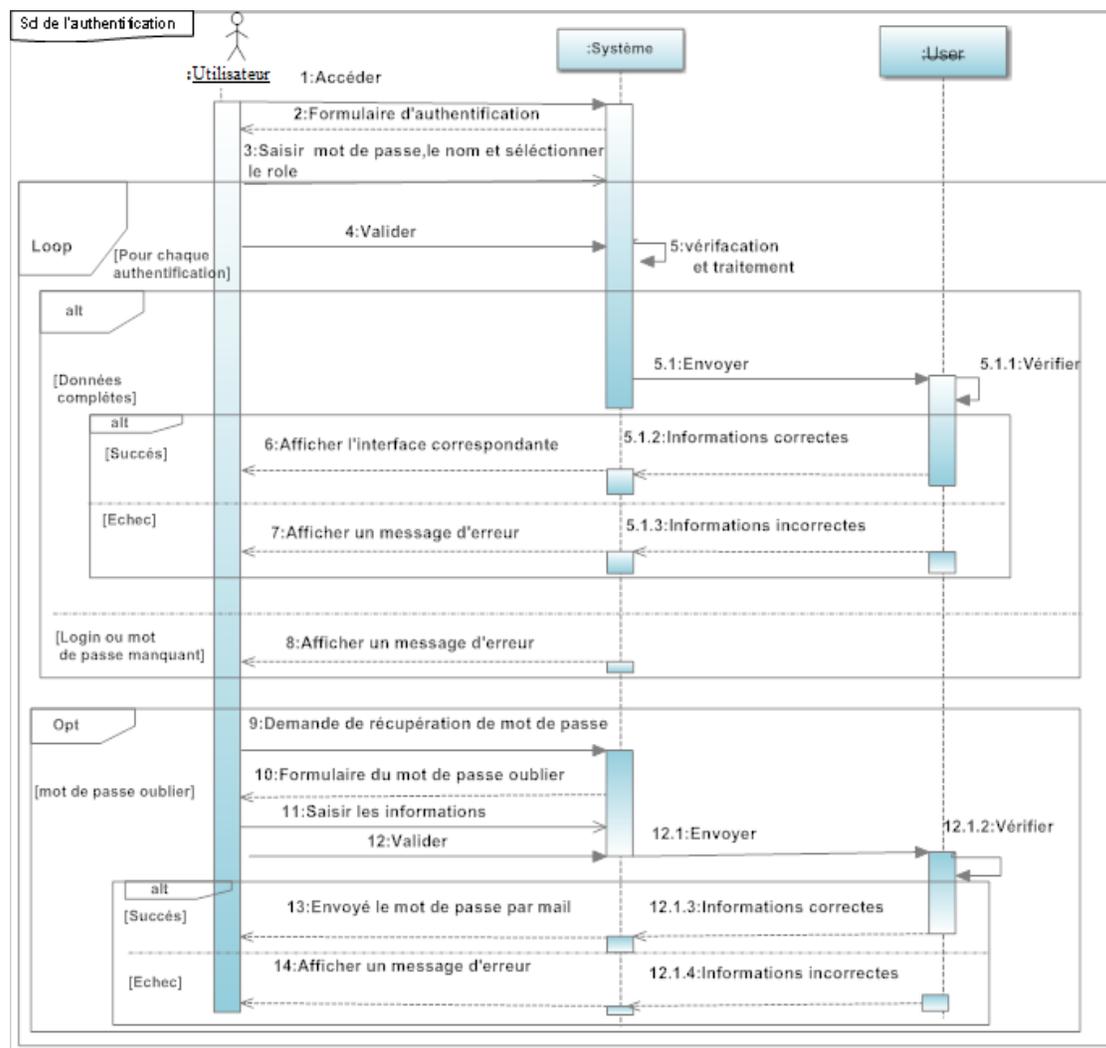


FIG. 7.2 – Diagramme de séquence de cas d'utilisation “Authentification”.

II.2.2 Diagramme de séquence de cas d'utilisation “Ajouter une formation”

Après l'authentification, le chargé de stage et formation presse sur le bouton “Ajout”, le système lui affiche l'interface correspondante, le chargé de stage et formation remplit le formulaire d'ajout, le système vérifie ces champs, à cet effet, si les informations sont bien saisies elles seront automatiquement stocker dans la base de données, sinon un message d'erreur s'affiche .

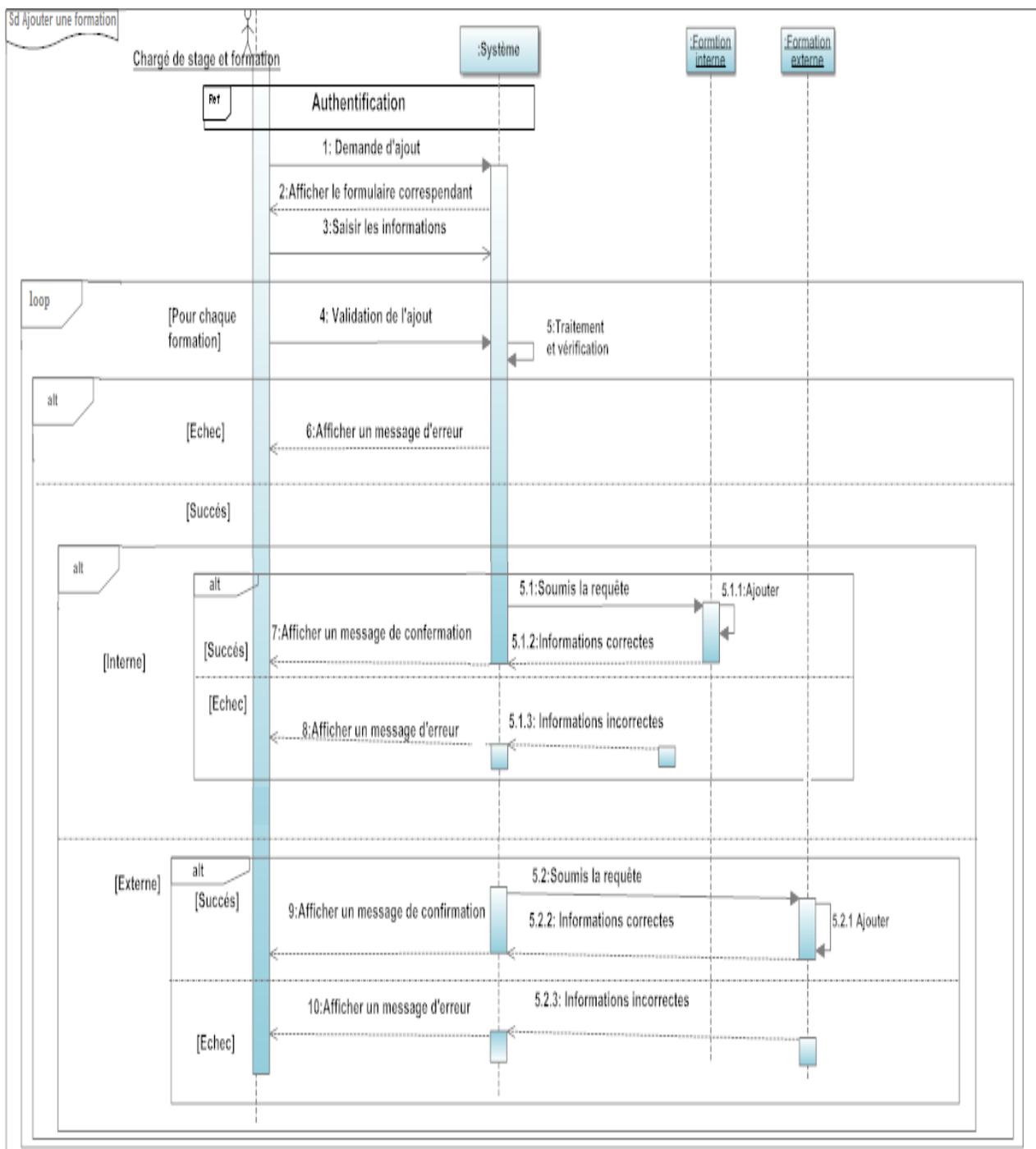


FIG. 7.3 – Diagramme de séquence de cas d’utilisation “Ajouter une formation”.

II.2.3 Diagramme de séquence de cas d'utilisation “Modifier une formation”

Pour effectuer une modification, le chargé de stage et formation s'authentifie, puis clique sur le bouton “Modification”, le système lui demande de saisir le type puis le matricule de formation, le chargé de stage et formation saisi ces derniers, ensuite le système récupère à partir de l'objet les informations correspondantes, le chargé de stage et formation effectue et valide les modifications.

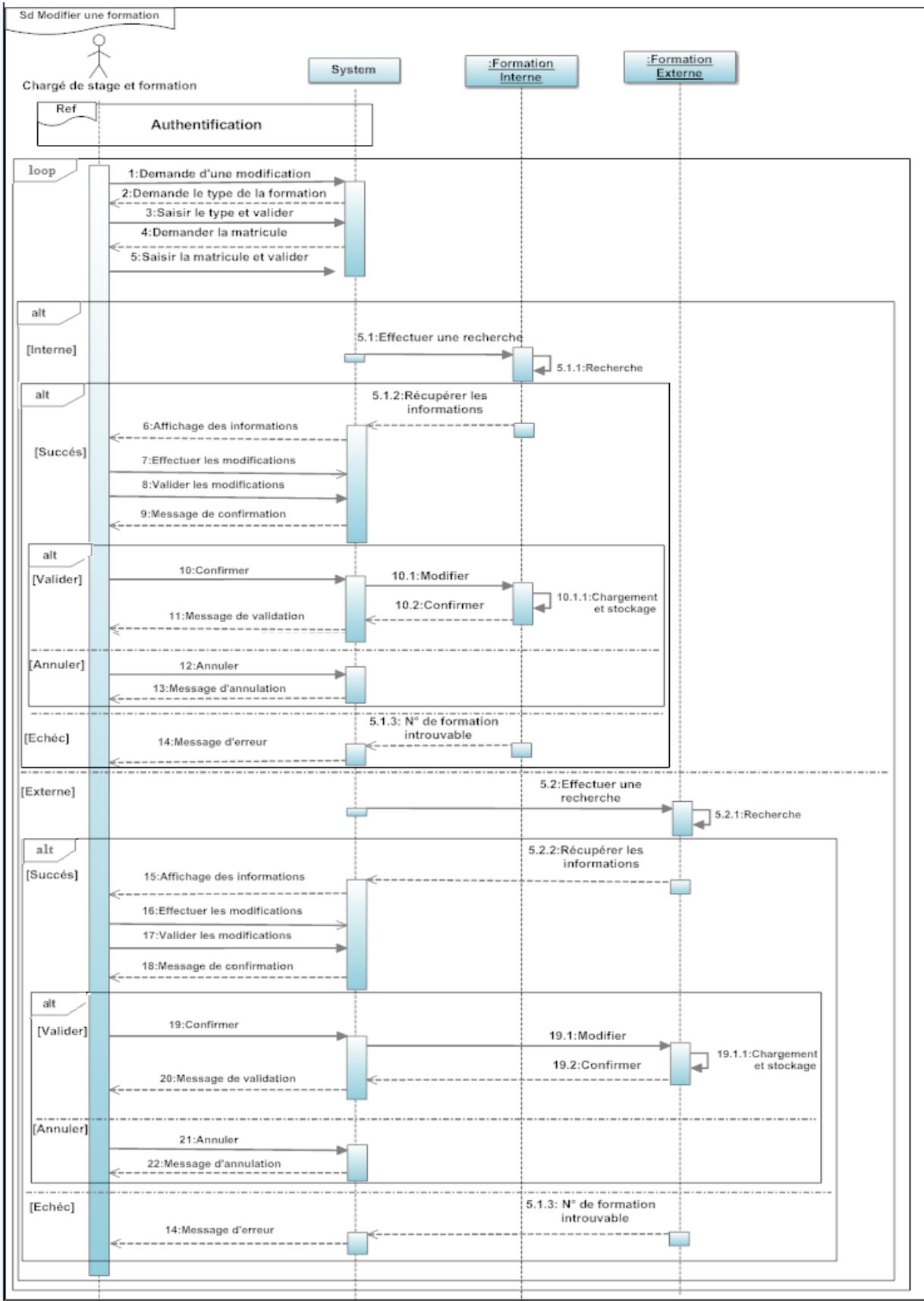


FIG. 7.4 – Diagramme de séquence de cas d’utilisation “Modifier une formation”.

II.2.4 Diagramme de séquence de cas d'utilisation “Chercher une formation”

Après l’authentification, le chargé de stage et formation saisi le matricule, sélectionne le type de formation à visualiser puis presse sur le bouton “ Recherche”, le système lui répond par l’affichage des caractéristiques de celle-ci en lui permettant de voir les détails.

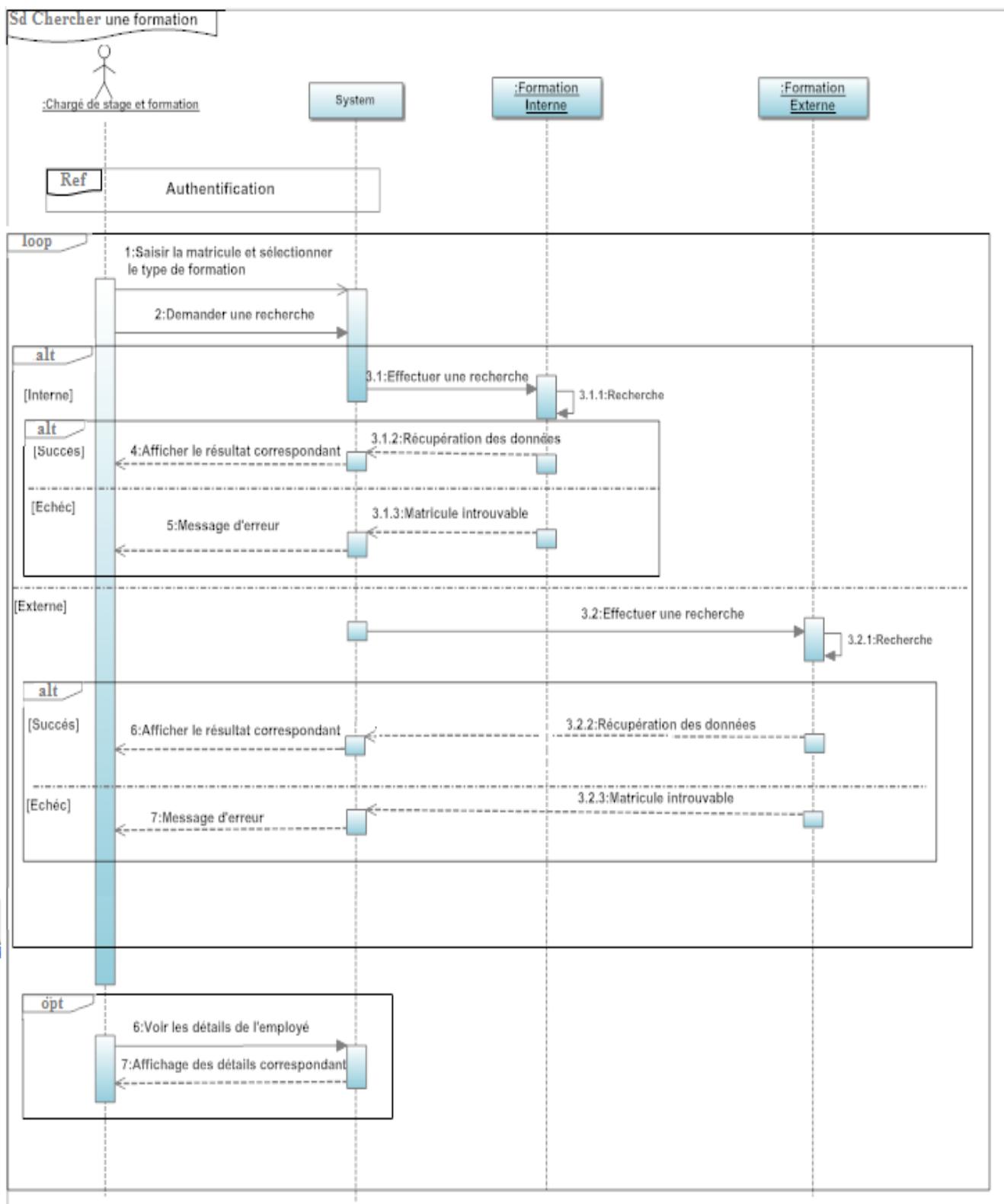


FIG. 7.5 – Diagramme de séquence de cas d'utilisation “Chercher une formation”.

II.2.5 Diagramme de séquence de cas d'utilisation “Ajouter un stagiaire”

Pour ajouter un stagiaire, le chargé de stage et formation doit d’abord s’authentifier ensuite clique sur le bouton “ajout”, le système affiche le formulaire d’ajout, le chargé de stage et formation remplit ce dernier, le système vérifie ces champs, à cet effet si les informations sont valide, elles seront automatiquement stocké dans la base de données sinon un message d’erreur s’affiche.

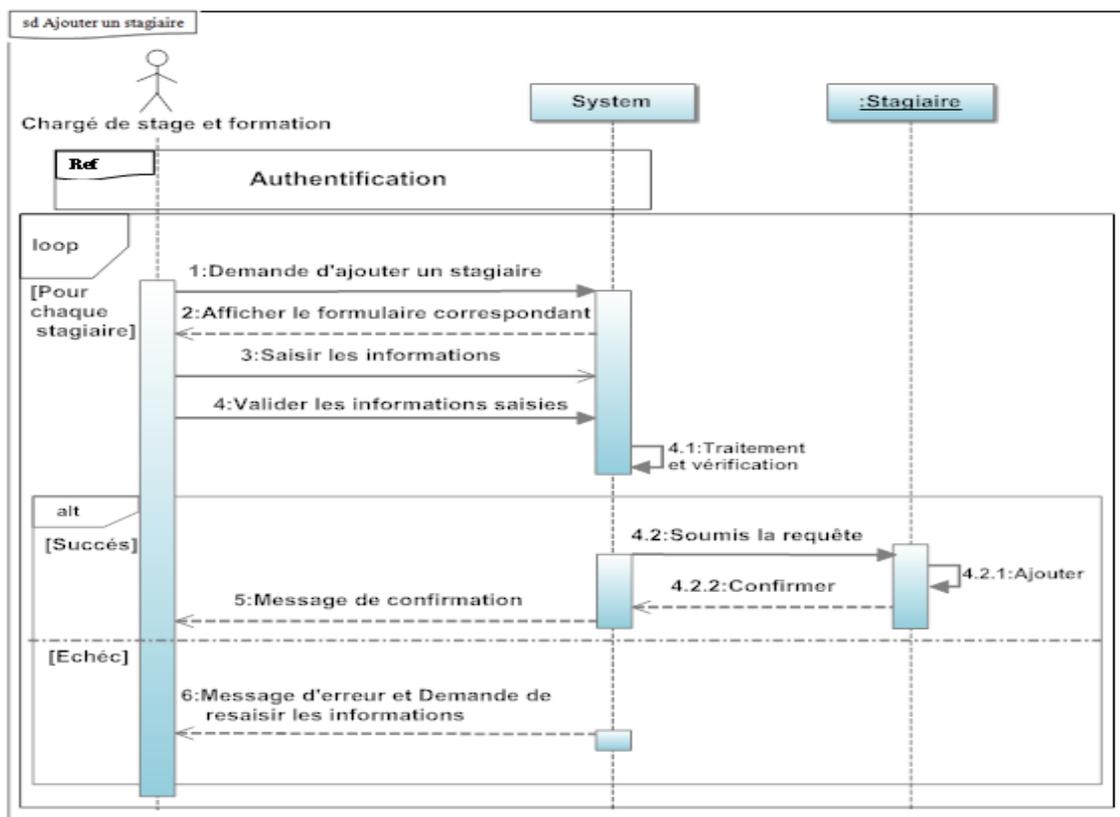


FIG. 7.6 – Diagramme de séquence de cas d’utilisation “Ajouter un stagiaire”.

II.2.6 Diagramme de séquence de cas d’utilisation “Modifier un stagiaire”

Afin d’établir une modification sur les informations d’un stagiaire, le chargé de stage et formation doit d’abord s’authentifier ensuite clique sur le bouton “Modification”, le système lui demande le matricule de stagiaire celle-ci est saisie par le chargé de stage et formation, le système récupère à partir de l’objet les informations correspondante, le chargé de stage et formation effectue et valide les modifications.

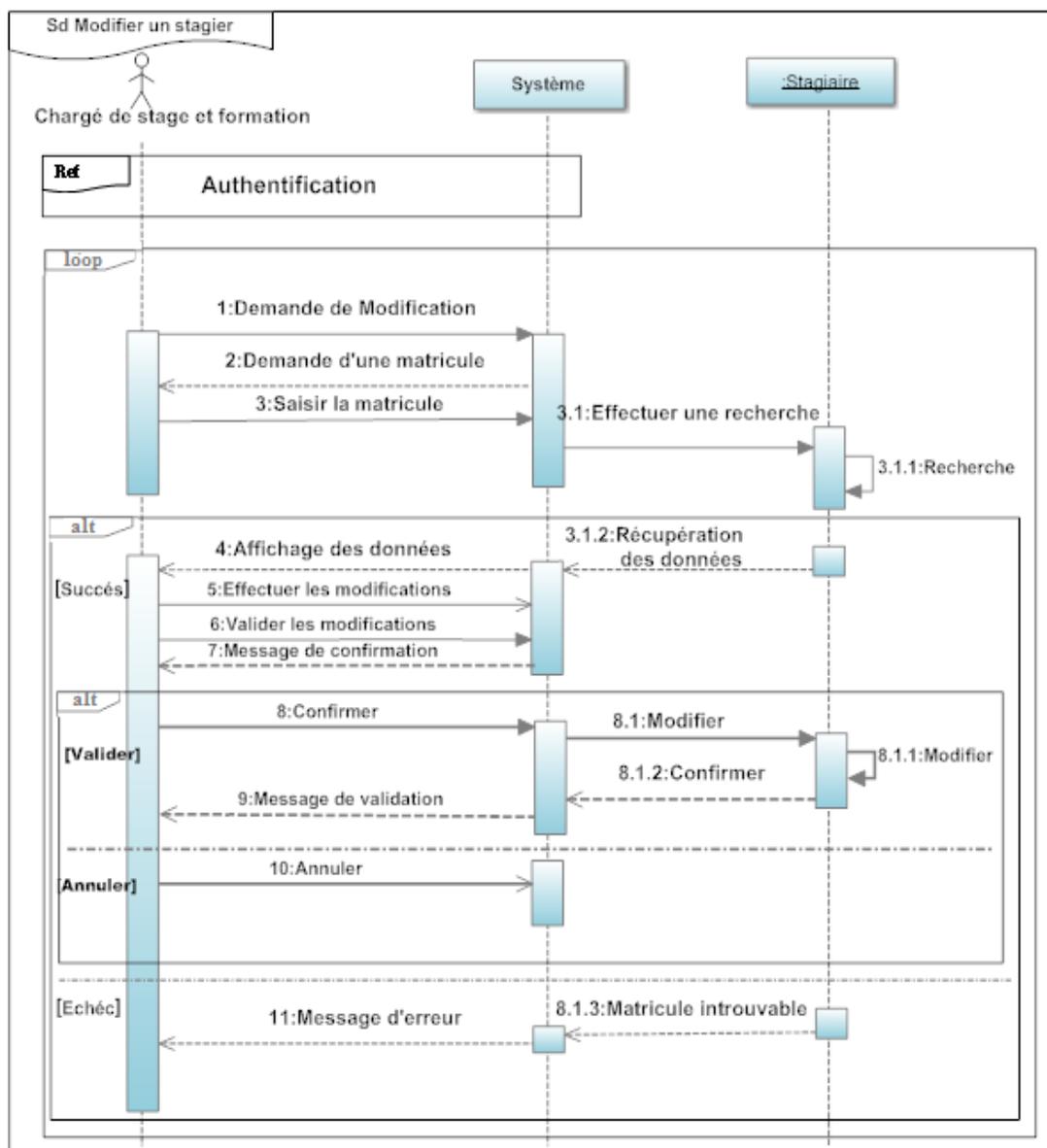


FIG. 7.7 – Diagramme de séquence de cas d'utilisation “Modifier un stagiaire”.

II.2.7 Diagramme de séquence de cas d'utilisation “Supprimer un stagiaire”

Après l’authentification, le chargé de stage et formation saisi le matricule du stagiaire à supprimer, le système lui affiche un message de confirmation à fin de valider la suppression. Même cas pour la suppression d’une formation, un employé, un thème, une établissement, un budget et un apprenti.

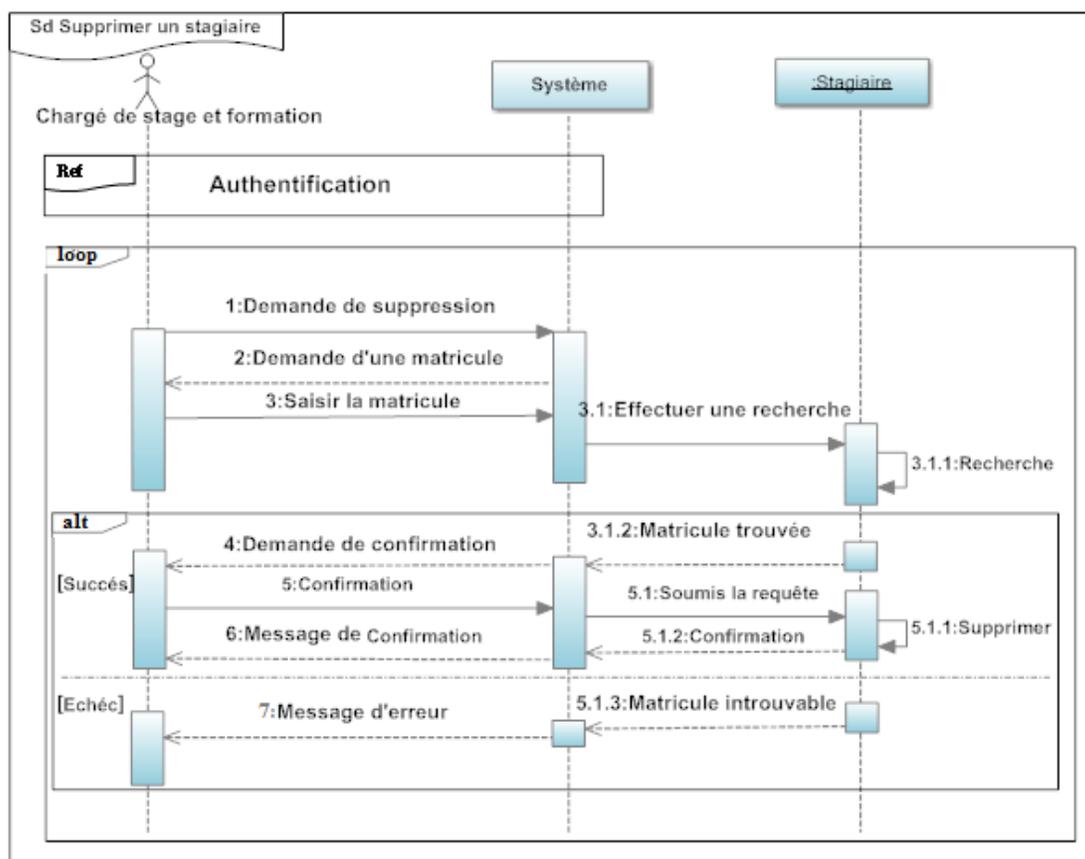


FIG. 7.8 – Diagramme de séquence de cas d'utilisation "Supprimer une formation".

II.2.8 Diagramme de séquence de cas d'utilisation "Effectuer une demande de formation"

Après l'authentification de l'employé, il peut consulter les thèmes et les établissements disponibles pour effectuer une demande de formation.

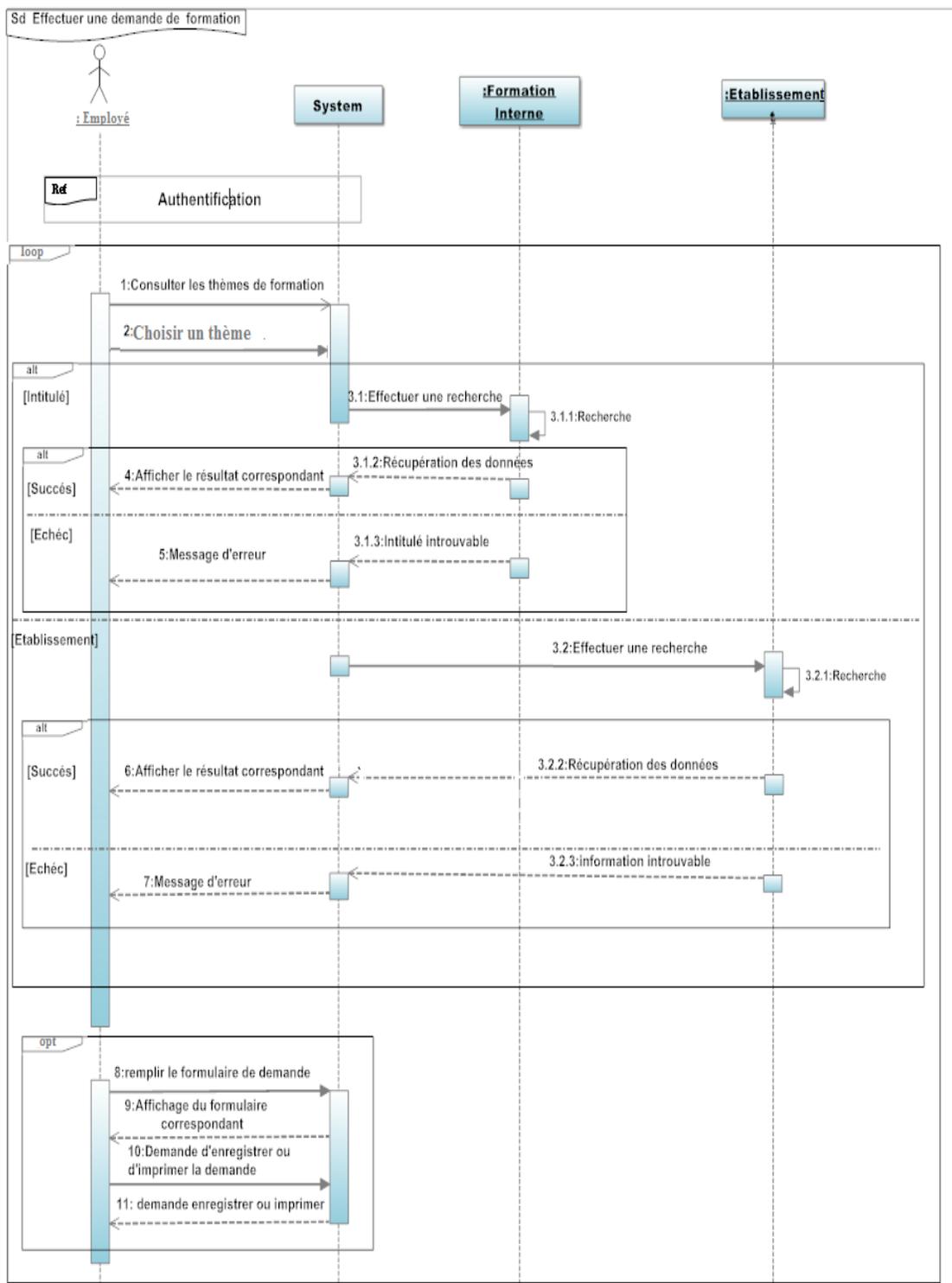


FIG. 7.9 – Diagramme de séquence de cas d’utilisation “Effectuer une demande de formation”.

Conclusion

Dans ce chapitre, nous avons définie le modèle statique, illustrés les principales constructions du diagramme de classes et les concepts dynamiques (diagrammes de séquences) qui sert à définir les enchainements réalisés par l'utilisateur.

Chapitre 8

Conception

Introduction

La conception vise à ajouter aux modèles objets de l'analyse, les détails liés à l'implémentation du système. Elle consiste à concevoir et documenter précisément le code qui va être produit. Dans cette phase, toutes les questions concernant la manière de réaliser le système à développer doivent être élucidées en définissant la structure statique du système sous forme de sous système, classes, associations, attributs.

I Dictionnaire de données

Le tableau suivant représente la signification des attributs de chaque classe de notre système à réalisé :

<i>Classe</i>	<i>Attribut</i>	<i>Type</i>	<i>Désignation</i>	<i>Méthodes</i>
<i>Apprentis</i>	MATRICULE_A	Number	Matricule de l'apprenti	Ajouter()
	NOM	VarChar(30)	Nom de l'apprenti	Modifier()
	PRÉNOM	VarChar(30)	Prénom de l'apprenti	
	NIVEAU	VarChar(10)	Niveau d'étude	Supprimer()
	TYPE	VarChar(30)	Type de stage	
	DURÉEÉÉ	VarChar(20)	Durrée du stage	Chercher()
	NUM_CONTRAT	VarChar(30)	Numéro du contrat	
	DATE_CON	Date	Date du debut du contrat	
	DATE_DEB	Date	Date du debut de stage	
	ORGANISME	VarChar(50)	Etablissement de l'apprenti	
	SEMÉSTRE	VarChar(5)	N° du semestre	
	MONTANT	VarChar(30)	Montant	
	PRIME	VarChar(30)	Prime	
<i>Attestation</i>	Id_Attestation	Number	Numéro de l'attestation	Créer() Enregistrer() Imprimer()
<i>Budget</i>	Id_Budget	Number	Numéro du budget	Créer()
	Exercice	year	Année	Modifier()
	Coût_Prévu	Varchar(50)	Budget prévisionnel pour chaque direction	Supprimer
	Buget_Réalisé	VarChar(50)	Budget consommé	EditerStat()
<i>Demande</i>	Id_Demande	Number	Numéro de la demande de formation	Créer()
	Date	Date	Date de la demande	Enregistrer() Imprimer()
	Type	varchar(20)	type de la formation	

<i>Direction</i>	Section_gestion	VarChar(20)	Numéro de la direction	
	NOM_D	VarChar(30)	Nom de la direction	
<i>Employé</i>	Matricule_E	Number	Matricule de l'employé	Ajouter ()
	Nom	VarChar(30)	Nom de l'employé	Modifier()
	Prénom	VarChar(30)	Prénom de l'employé	
	Fonction	VarChar(30)	Fonction de l'employé	Supprimer()
	Catégorie	VarChar(10)	Catégorie de l'employé	Chercher()
	Date_Rucrutement	Date	Date du recrutement de l'employé	
<i>Etablissement</i>	IdEtablissement	Number	Numéro de l'établissement	Ajouter()
	Nom_E	VarChar(30)	Intitulé de l'établissement	Modifier()
	Tel	Number	Numéro de téléphone	Supprimer()
	Email	VarChar(50)	adresse mail	Chercher()
<i>FormationExterne</i>	Num_F	Number	N° de la formation	
	Intitulé	VarChar(50)	Thème de la formation	Ajouter()
	Durée	VarChar(10)	Durée de la formation	Modifier()
	Coût	VarChar(30)	Coût de la formation	Supprimer()
	Date_Debut	Date	Date de début de la formation	Chercher()
	Date_Fin	Date	Date fin de la formation	
	Montant_HT	VarChar(50)	Prix de la formation en Hors Taxe	EditerStat()
<i>FormationInterne</i>	Etat_Attestation	VarChar(20)	Récupération de l'attestation	Ajouter()
	Nom_formateur	VarChar(30)	Nom du formateur qui va prendre en charge la formation	Modifier() Supprimer() Chercher() EditerStat()
<i>Service</i>	ID_Service	Number	Numéro du service	
	Nom_S	VarChar(30)	Nom du service	
<i>Stage</i>	IdStage	Number	Numéro du stage	
	Durée	VarChar(10)	Durée de stage	EditerStat()
	Date_D	Date	Date du debut de stage	
	Date_F	Date	Date du fin de stage	

<i>Stagiaire</i>	MatriculeS	Number	Matricule du stagiaire	Ajouter()
	Nom	VarChar(30)	Nom du stagiaire	
	Prénom	VarChar(30)	Prénom du stagiaire	Modifier()
	Organisme	VarChar(50)	Etablissement du stagiaire	
	Option	VarChar(50)	spécialité d'étude du stagiaire	Supprimer()
	Convention_Stage	VarChar(10)	Signature de la convention par l'organisme	
	Convention_BMT	VarChar(10)	Signature de la convention par BMT	Chercher()
	Etat_Memo	VarChar(20)	Réception du mémoire réalisé	
<i>Thème</i>	IdThème	Number	Numéro du thème	Ajouter()
	Domaine	VarChar(50)	spécialité du thème	Modifier()
	Intitulé	VarChar(50)	Intitulé du thème	Supprimer()
	Etat	VarChar(20)	Etat de soumission du thème	Chercher()
	Description	Varchar(50)	Explication du thème	

TAB. 8.1 – Dictionnaire de données.

II Modèle relationnelle de données

II.1 Terminologie de l'approche relationnelle

- **Domaine** : Ensemble des valeurs admises pour un attribut, Il établit les valeurs acceptables dans une colonne. Une relation peut faire appel au même domaine pour plusieurs de ses attributs. La notion de domaine est fondamentale en matière de gestion de données. Elle permet d'exprimer des contraintes d'intégrité sémantique très fortes sur les données d'une BD .
- **Tuple** : Une ligne dans une table .
- **Relation** : C'est une table qui contient des données appelées enregistrements (lignes), et de colonnes(champs)[3].
- **Clé étrangère** : Assure les liaisons entre les tables et les contraintes d'intégrités .
- **Clé primaire** :La clé primaire d'une table est l'ensemble minimal de colonnes qui permet d'identifier de manière unique[4] .

II.2 Règles de passage

- **Règle1 : Transformation des entités/classes**

Chaque entité devient une relation. L'identifiant de l'entité devient clé primaire pour la relation. Chaque classe du diagramme UML devient une relation. Il faut choisir un attribut de la classe pouvant jouer le rôle d'identifiant.

- **Règle2 :Associations un-à-plusieurs**

Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association.

- **Règle3 : L'association (classe-association)**

Elle devient une relation dont la clé primaire est composée par la concaténation des identifiants des entités (classes) connectés à l'association. Chaque attribut devient clé étrangère si l'entité (classe) connectée dont il provient devient une relation en vertu de la règle R1. Les attributs de l'association (classe-association) doivent être ajoutés à la nouvelle relation. Ces attributs ne sont ni clé primaire, ni clé étrangère.

- **Règle4 :Associations un-à-un**

Il faut ajouter un attribut clé étrangère dans la relation dérivée de l'entité ayant la cardinalité minimale égale à zéro. Dans le cas d' UML, il faut ajouter un attribut clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de l'entité (classe) connectée à l'association.

- **Règle5 :Décomposition ascendante (push-up)(héritage)**

Il faut supprimer la (les) relation(s) issue(s) de la (des) sous-classe(s) et faire migrer les attributs dans la relation issue de la sur-classe.

- **Règle6 :Composition**

La clé primaire des relations déduites des classes composantes doit contenir l'identifiant de la classe composite (quelles que soient les multiplicités)[4].

II.3 Présentation du schéma relationnel

- **Budget**(Id_Budget, Exercice, Coût_Prévu, Budget_Réalisé, section_gestion#).
- **Direction** (section_gestion, Nom_D).
- **Etablissement** (Id_Etablissement, Nom_E, Adr, Tel, Email).
- **FormationInterne**(NumF, Intitulé, Durée, Coût, Date_Debut, Date_Fin, Etat_Attestation,

Nom_formateur).

- **FormationExterne** (NumF, Intitulé, Durée, Coût, Date_Debut, Date_Fin, Montant_HT).
- **Service**(Id_Service, section_gestion#, Nom_Service).
- **Apprentis**(MatriculeA, Nom, Prénom, Niveau, Nature, Durée, Num_Contrat, Date_Contrat, Date_Debut, Organisme, Semestre, Montant, Prime, Section_gestion#).
- **Stagiaire**(MatriculeS, Noms, Prénoms, Organismes, Option, Filière_et_Convention, Convention_BMT, Etat_Memo, Matricule_E#, Id_Attestation#, Id_Stage#).
- **Attestation**(IdAttestation).
- **Stage** (Id_Stage, Durée, Date_D, Date_F, Id_Thème#).
- **Employé**(MatriculeE, Nom, Prénom, Fonction, Catégorie, Date_Rucrutement, Id_Service#).
- **Thème**(Id_Thème, Intitulé, Domaine, Etat, Description, Matricule_E#).
- **Demande**(Id_Demande, Date, type, NumF#, Matricule_E#).
- **Former** (Id_Etablissement#, NumF#).

Conclusion

Dans ce chapitre, nous avons présenté l'étude conceptuelle de notre système. Nous avons respecté la modélisation qui a été élaboré dans le chapitre précédant, afin de répondre aux objectifs fixés.

Chapitre 9

Réalisation

Introduction

La réalisation est l'étape qui nous permet de concrétiser les solutions et suggestions que nous avons proposées lors de l'étude de l'existant ainsi que la conception que nous avons mis en place.

Dans ce chapitre nous allons définir les outils et l'environnement de développement de notre application ainsi que son schéma descriptif.

I Langage et outils de développements

I.1 Langage d'implémentation (JAVA)

Java se veut un langage de la première catégorie, autrement dit un pur langage de P.O.O. Par nature, un programme s'y trouvera formé d'une classe ou de la réunion de plusieurs classes et il instanciera des objets. Il sera impossible de créer un programme n'utilisant aucune classe.

Java dispose de types dits primitifs pour représenter les entiers, les flottants, les caractères et les booléens. Les variables correspondantes ne sont pas des objets. Certes, la plupart du temps, ces types primitifs seront utilisés pour définir les champs d'une classe, donc finalement d'un objet, cependant, il y aura des exceptions...[23].

I.2 Outils de développement

I.2.1 Présentation de la plateforme de développement Netbeans

L'EDI(Environnement de Développement Intégré) NetBeans est un environnement de développement un outil pour les programmeurs pour écrire, compiler, déboguer et déployer des programmes. Il est écrit en Java mais peut supporter n'importe quel langage de programmation. Il y a également un grand nombre de modules pour étendre l'EDI NetBeans [24].

I.2.2 Ireport

JasperReports est un outil Open Source de génération d'états /rapports développé en Java, pour des applications Java. Il se présente sous la forme de bibliothèques à intégrer aux applications. Ces bibliothèques permettent la visualisation ou l'export de données vers de multiples formats. il peut :

- Permettre aux utilisateurs eux même de créer leur rapport ou alors des utilisateurs qui connaissent SQL et un peu de Java ...

- Permettre l'export de données à des formats divers pour une application existante (sortie imprimable, mailing ...)[25].

I.2.3 Présentation du SGBD Oracle 10g

Un SGBD est une application qui fonctionne sur un système d'exploitation donné. Par conséquent, il faut se connecter au système avant d'ouvrir une session Oracle. Cette connexion peut être implicite ou explicite.

Oracle 10g est une base de données multiutilisateur autorisant des clients locaux, intranet ou distants à se connecter. Il permet un réel fonctionnement client-serveur, ce qui fait de Windows une plate-forme privilégiée pour le déploiement d'applications de type départemental[26].

I.2.3.1 Caractéristiques d'un SGBD Oracle 10g : Oracle est un SGBD qui permet :[19]

- D'autoriser une souplesse d'architecture, incorporant différentes technologies (client/serveur, Java, etc.), fédérées par une seule base de données.
- De changer le système d'exploitation qui héberge les bases de données sans remettre en cause les applications.
- De prendre en charge des projets allant du plus simple, nécessitant peu de puissance, au plus complexe, qui en requiert beaucoup.
- De restaurer et sauvegarder les données.

II Présentation de prototype réalisé

a)**Interface d'accueil** : C'est l'interface principale de notre application. Elle permet aux acteurs de notre système d'accéder à leurs interfaces après l'authentification.



FIG. 9.1 – Interface d'accueil.

b) Interface d'authentification : Il est nécessaire pour chaque utilisateur (administrateur, chargé de stage et formation, responsable de structure) de passer par un système d'authentification. En effet, l'authentification consiste à introduire le login, le mot de passe, et sélectionner un rôle de l'utilisateur, une fois ses informations sont reconnues, le système affiche l'interface appropriée. Dans le cas contraire, un message d'erreur s'affiche.

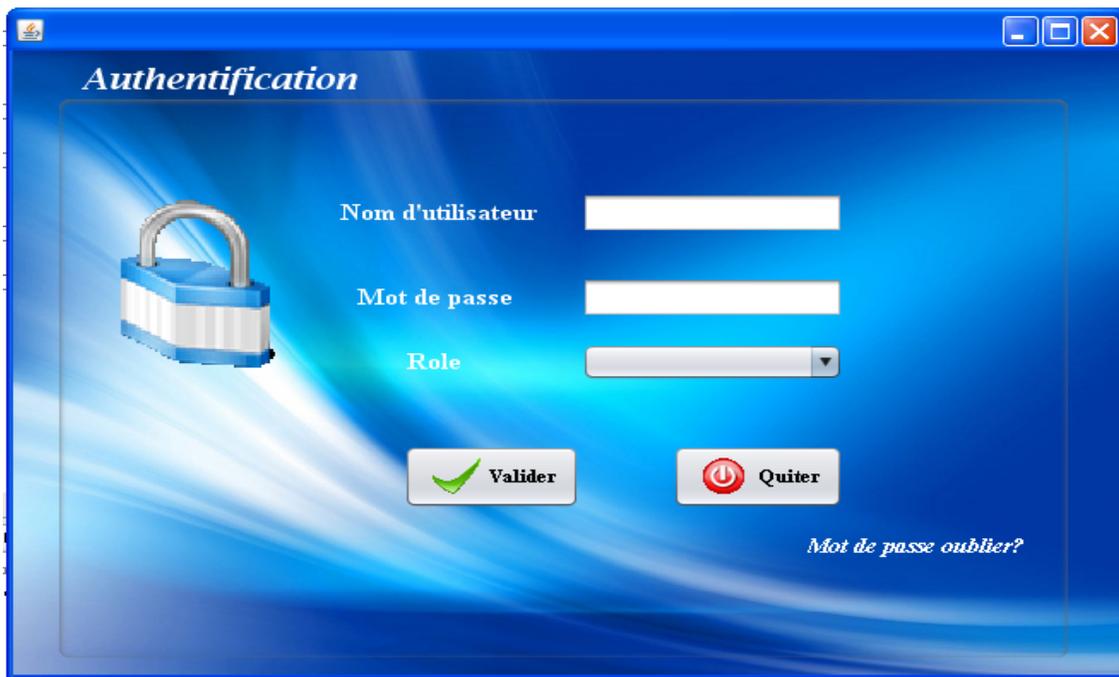


FIG. 9.2 – Interface d'authentification.

c) **Interface d'espace administrateur** : Cette interface correspond à l'espace administrateur après sa connexion il peut ajouter, modifier, supprimer un utilisateur.

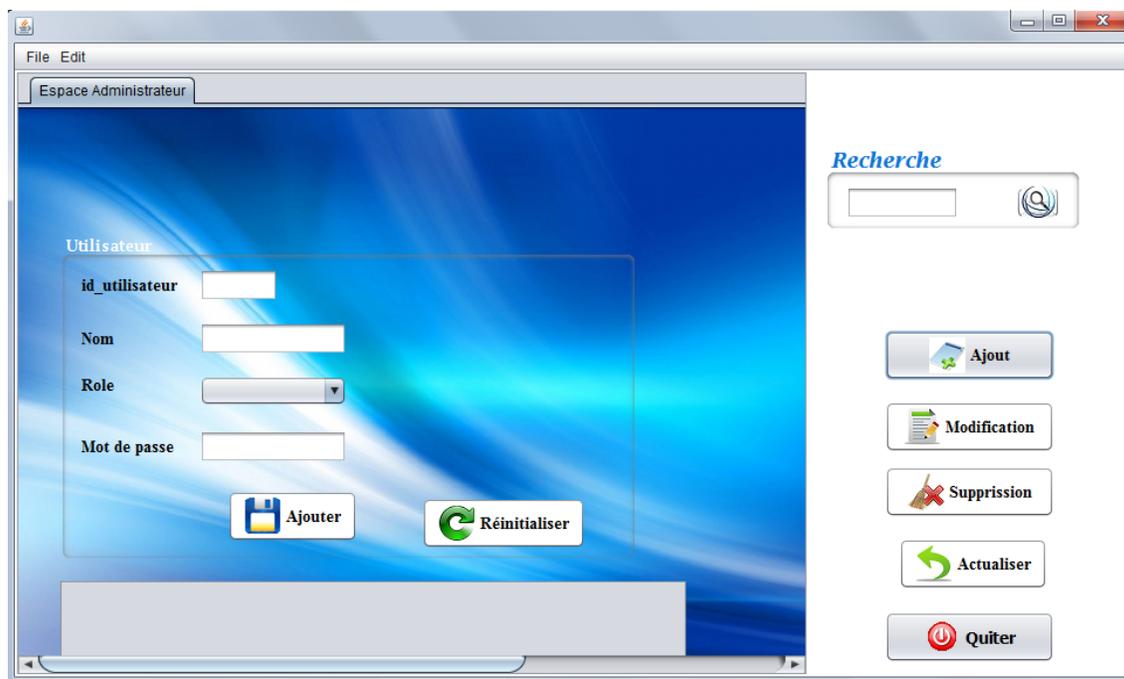


FIG. 9.3 – Interface d'espace administrateur.

d) **Interface de gestion des stages** : Cette interface s'affiche après la réussite de la connexion

du chargé de stage et formation. Il peut effectuer toutes les opérations se rapportant à son activité, il peut donc :

- Ajouter, modifier, supprimer, chercher un stagiaire, un thème et/ ou un apprenti.
- Editer les statistiques de stage.
- Etablir une attestation.



FIG. 9.4 – Interface de gestion des stages.

e) **Interface de gestion des formations** : Cette interface s'affiche après la réussite de la connexion du chargé de stage et formation. Il peut effectuer toutes les opérations se rapportant à son activité, il peut donc :

- Ajouter, modifier, supprimer, chercher une formation, un établissement, employé . . .
- Editer les statistiques de formation et budget.

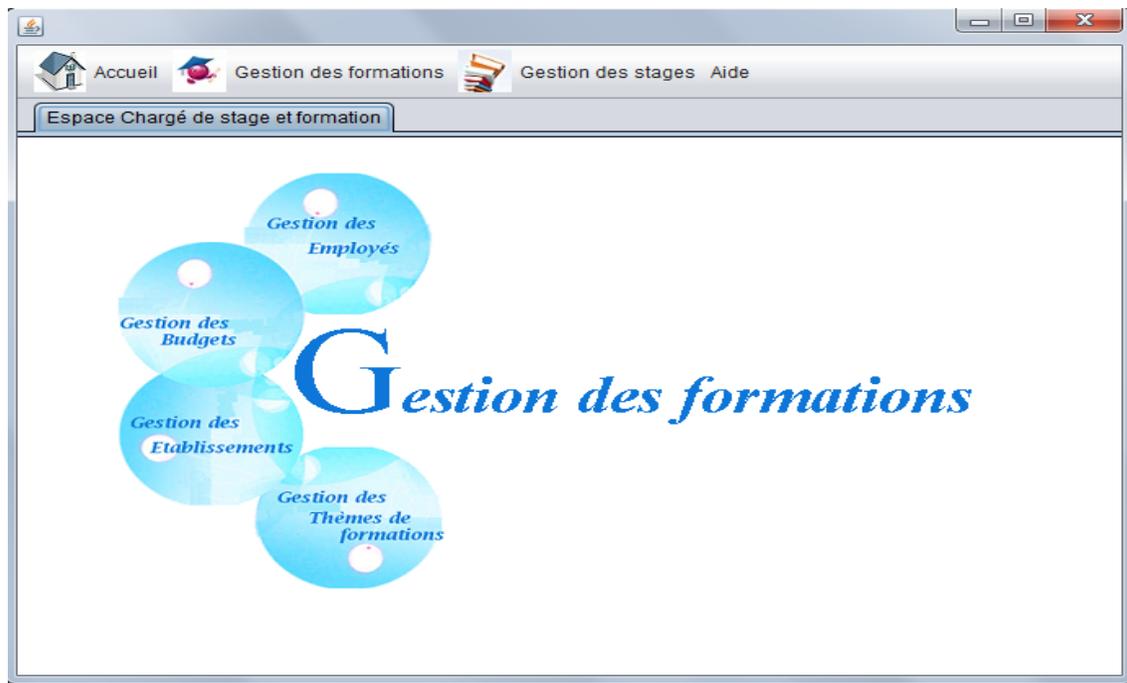


FIG. 9.5 – Interface de gestion des formations.

Conclusion

La phase réalisation est l'étape la plus importante dans le cycle de vie d'une application. Dans cette phase nous avons présenté les outils de développement mis en œuvre pour réaliser notre application ainsi qu'un aperçu de quelques IHM qui la constituent afin d'exposer la manière dont elle fonctionne à travers l'interaction avec les utilisateurs.

Conclusion générale

Le travail que nous avons accompli a pour principale vocation la conception et la réalisation d'un système d'information pour la gestion des stages et formations au sein de BMT. Ce projet nous a permis de mettre en pratique les connaissances acquises durant le cycle de notre formation, de se familiariser avec un environnement dynamique et d'avoir une idée plus profonde et plus pratique sur l'importance des systèmes d'informations dans les entreprises.

Pour mettre en œuvre ce projet, nous avons été amené dans un premier lieu à élargir nos connaissances dans le domaine des applications client/serveur, système d'information, base de donnée, ainsi que les réseaux informatiques puis étudier l'existant au niveau de la direction ressources humaine.

Afin d'accomplir notre travail et d'aboutir au résultat escompté nous avons suivi le processus de développement unifié "2TUP", et opté pour UML comme langage de conception et de modélisation étant le plus utilisé, il s'avère être le plus proche de notre style au niveau de la programmation dans le sens où nous avons adopté une approche orientée objet étant donné l'utilisation de JAVA comme langage de programmation, dans un environnement Netbeans ainsi que le SGBD Oracle qui nous a servis à gérer les bases de données.

La réalisation de ce projet a été bénéfique et fructueuse pour nous dans le sens où elle nous a permis d'approfondir nos connaissances et en acquérir de nouvelles qui seront utiles et déterministes pour nous à l'avenir.

Bibliographie

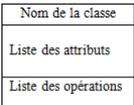
Bibliographie

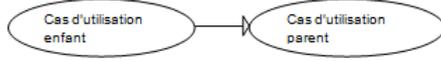
- [1] Yves Gillette, Processus métiers et système d'information gouvernances, management, modélisation de Chantal Morley, Marie Bia-Figueiredo, Yves Gillette 3^{ème} édition Dunod, PARIS, 2011.
- [2] J. O'BRIEN, Les systèmes d'information de gestion, édition renouveau pedagogique, Canada, 1995.
- [3] Gilles Roy, Conception de bases de données avec UML, 2009.
- [4] Christian Soutou, UML 2 pour les bases de données, édition EYROLLES, 2012.
- [5] Nicolas Larrousse CNRS Avec la contribution de Éric Innocenti Université de Corse Pasquale Paoli, Création de base de données, Pearson Education France, 2009.
- [6] Anthony ASSI ,SGBDR et conception d'un système d'information avec MERISE, Séminaires Codes et Travaux @ IRISA 26.
- [7] Danièle Dromard et Dominique Seret , Architecture des réseaux, Pearson Education France, 2009 .
- [8] Philippe Atelin, Réseaux informatiques notion fondamentales, 3^{ème} édition, 2009.
- [9] Jacques Nozick, Les réseaux, édition EYROLLES.
- [10] [http ://www.yakaferci.com](http://www.yakaferci.com).
- [11] Andrew Tanenbaum, Concepts des systèmes d'exploitation, Systèmes centralisés, Système distribués, Interedition, 1994.
- [12] Rémi Leblond, Vers une architecture n-tiers, Oral probatoire, 1999.
- [13] Paul KRZYZANOWSKI, Distributed Systems, Rutgers University, 2003.
- [14] K. C. LAUDON, J. P. LAUDON, Les systèmes d'information de gestion : organisation et réseaux stratégiques, édition VILLAGE MONDIAL, Paris/ Pearson Education France, 2000.
- [15] Pascal Roques et Franck Vallée, UML 2 en action, 4^{ème} édition EYROLLES, Mars 2007.

-
- [16] Eyrolles, Les Cahiers du Programmeur UML2 Modéliser une application web, 4^{ème} édition, Octobre 2008.
 - [17] Clotilde Attouche, Oracle 11g Exploitation, Société TELLORA, Version 1.2, Janvier 2013.
 - [18] Gilles Briard, Oracle10g sous Windows, Avec la collaboration de la société Digora, 2006.
 - [19] Laurent Piechocki, Cours UML.
 - [20] Pierre-Alain Muller, UML2 pour les développeur, Triskell-INRIA Rennes, Décembre 2012.
 - [21] Joseph Gabay, David Gabay, UML 2 Analyse et Conception, 2008.
 - [22] Cloude Delannoy, Programmer en Java, édition EYROLLES, 5^{ème} édition, Avril 2008.
 - [23] http://www.netbeans.org/index/_fr.html.
 - [24] http://www.Ireport/index_fr.html.
 - [25] Clotilde Attouche, Oracle PL/SQL, Version 1.1, 2012.

Annex

Commande	Signification
ALTER DATABASE	Ouvre (open) ou monte (mount) une base. Choisit les modes ARCHIVELOG/NOARCHIVELOG. Restaure à partir de bandes. Agit sur les membres des groupes redolog. Renomme un fichier de la base ou des membres d'un redolog. Sauvegarde le fichier de contrôle courant. Sauvegarde des commandes SQL dans un fichier trace. Ajoute un nouveau fichier à la base de données. Autorise l'extension automatique des fichiers de la base. Positionne un fichier de la base en ligne ou hors ligne. Modifie le nom de la base de données.
ALTER ROLE	Modifie les autorisations regroupées sous le rôle.
ALTER TABLE	Ajoute une colonne ou une contrainte d'intégrité à une table. Modifie une colonne ou les caractéristiques de stockage (storage) d'une table. Active, désactive, supprime des contraintes d'intégrité. Active et désactive les triggers d'une table. Agit sur les extensions (extents) d'une table. Modifie le degré de parallélisme affecté à une table (nombre de processus actifs simultanément sur une table).
ALTER TABLESPACE	Ajoute et renomme des fichiers de la base. Modifie les caractéristiques de stockage (storage). Met en ligne, hors ligne un tablespace. Commence et arrête une sauvegarde (backup) en ligne. Autorise ou interdit les écritures sur un tablespace (cas des disques optiques ou CD-Rom).
ALTER USER	Modifie les caractéristiques d'un utilisateur : mot de passe, tablespace affecté par défaut, tablespace temporaire, quotas, profils ou rôles.
CREATE DATABASE	Crée une nouvelle base de données.
CREATE ROLE	Crée un rôle qui regroupera tout un ensemble de droits d'accès.
CREATE TABLE	Crée une table, ses colonnes, ses caractéristiques de stockage (storage) et ses contraintes d'intégrité.
CREATE TABLESPACE	Crée un emplacement dans la base de données où seront stockés des tables, des index, des rollback segments, des espaces de tri temporaires. Cet espace est lié à un ou plusieurs fichiers répartis sur les disques de l'ordinateur.
CREATE USER	Crée un utilisateur.
DELETE	Supprime des enregistrements d'une table.
DROP	Commande pour la suppression(user,tablespace...)
GRANT	Donne des droits, privilèges, rôles à des utilisateurs ou à des rôles.
INSERT	Ajoute de nouveaux enregistrements à une table.
SELECT	Recherche des enregistrements dans une ou plusieurs tables via un ensemble de critères de recherche, de tri, etc.
UPDATE	Modifie des données d'enregistrements existants

Notation	Signification
	<p>Cette notation est la représentation graphique d'une classe.</p>
	<p>Cette notation graphique appelée stick man en UML désigne un acteur, avec le nom de l'acteur sous le dessin.</p>
	<p>Cette notation graphique représente un cas d'utilisation (ou use case) correspond à un objectif du système, motivé par un besoin d'un ou plusieurs acteurs.</p>
	<p>Notation utilisée pour désigner la relation d'héritage .</p>
	<p>Notation utilisée pour désigner la relation de composition entre deux classes.</p>
	<p>Dans un diagramme de séquence l'acteur appelle le système par cette notation pour les messages asynchrone .</p>
	<p>Cette notation pour les messages synchrone .</p>
	<p>Cette notation est la raiponce du système pour l'acteur.</p>
	<p>Cette notation est appelée ligne de vie de l'opération émet par le système/l'acteur.</p>
	<p>Et celle la est l'activation de l'opération.</p>



Dans une relation de généralisation entre 2 cas d'utilisation, le cas d'utilisation enfant est une spécialisation du cas d'utilisation parent.

Résumé

En générale notre travail consiste à réaliser un système d'information pour le suivi des stages et formations à BMT (Bejaïa Méditerranéen Terminale) d'une manière à doter le service des ressources humaines d'une application Client/serveur qui met en œuvre une solution moderne et évolutive qui répond aux différentes exigences fonctionnelles et techniques , tout en facilitant les travaux des acteurs de notre système. A fin de le réaliser nous avons tout d'abord les rebiques théoriques et conceptuelles servent à définir le travail, ensuite nous avons adopté le processus 2TUP pour concevoir notre système.

Mots-clées :

reseaux informatiques ,2TUP, UML2,Oracle,Client/Serveur.

Abstract

Within a framework general our work consists in carrying out an information system for the follow-up of the training courses and formations with BMT (Bejaïa Méditerranéen Final) in a manner to equip the service with human resources of a Client/serveur application which implements a modern solution and evolutionary which answers the various functional and technical requirements, while facilitating work with the actors with our system. With end to carry it out we have the rebic theoretical ones first of all and conceptual are used to define work, then we adopted process 2TUP to conceive our system.

Keywords :

computer networks ,2TUP, UML2,Oracle,Client/Serveur.