

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



N° Réf :.....

Centre Universitaire
Abd elhafid Boussof Mila

Institut des sciences et de la technologie

Département de Mathématiques et Informatique

Mémoire préparé En vue de l'obtention du diplôme de Master en :
Filière informatique

Spécialité : Sciences et Technologies de l'Information et de la Communication STIC

Thème

**Conception et réalisation d'une application
client/serveur pour la traçabilité des dossiers
au niveau de la CNR de Mila**

Préparé par : Bourafa Saida
Djaaboub Aziza

Soutenue devant le jury :

Président : M^{eur} Hettab Abdelkamel (MAA)
Examineur : M^{me} Bouzahzah Mounira (MAA)
Encadreur : M^{me} Mme Benabderrahmane Fatiha (MAA)

Année universitaire : 2014/2015



Remerciement



Après avoir terminé ce mémoire de fin d'étude, nous Réserveons ces lignes pour exprimer nos remerciements les plus sincères à Allah.

*Nous tenons à exprimer nos remerciements avec un grand respect à notre encadreuse **Mme Ben Abderrahmane Fatima**, pour son aide et ses encouragements et surtout pour ses précieux conseils qui nous ont assistés pour réaliser ce travail.*

Nous tenons à exprimer nos sincères remerciements à toutes les personnes qui travaillent dans la Caisse Nationale des retraite et qui nous ont offert les meilleures conditions pour suivre notre stage.

Nos vifs remerciements sont également aux membres du jury pour L'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner Notre travail et de l'enrichir par leurs propositions.

*Nous tenons aussi à remercier et à saluer **Mr Youcef** et les ingénieurs et tous les travailleurs De la bibliothèque pour l'aide et l'assistance qu'ils nous ont apportée.*

Merci à tous



Bourafa saida

Djaaboub Aziza



DEDICACE



A l'occasion de ce travail fait par moi-même je le dédie sous peine à :

*La chandelle de ma vie et source d'amour et d'affection, celle qui a su être patiente avec moi et qui n'a ménagé aucun effort pour m'aider et me soutenir dans les moments difficiles, celle qui m'a mis au monde, ma très chère mère **Zelikhia**.*

*Celui qui fait de moi que ce je suis, mon cher père **Douadi** pour son assistance et encouragement durant toutes mes études.*

*L'amour et le soutien de ma chère sœur : **Leila** et mes chers frères : **Farid, Yazid, Djamel et Tarek**,*

*Toute ma grande famille et tous mes amis, surtout **Amel, Farida, Sara, Iman, Amina, Samia, Farida** et mon binôme **Saida**.*

*Ainsi que tous ceux qui connaissent et aiment **Aziza** et à ceux que j'aime.*

Aziza



DEDICACE



À l'occasion de ce travail fait par moi-même je le dédie sous peine à :

*À mon cher père **Brahim** qui ma donné tout ces pouvoir et tout son temps et tous mon parcours d'étude, il ma donné le courage, la volonté, la confiance en moi-même.*

*À ma très chère mère **Saliha** la plus chère personne pour moi, la meilleure femme du monde, elle a était comme une bougie qui illumine ma vie et brule pour l'épanouissement de ses fils et sa famille, A celle qui m'a soutenu durant toutes mes années d'étude, qui mérité mon amour éternel pour ses conseils spécieux, sa tendresse, sa patience.*

*À mes sœur **Yasmina** et la petite princesse **Hiba**. A tous mes frères **Yahya, Nadir** et **Moussa**.*

*À mes chers amies **Iman, Sara, Amina, Farida, Amal, Farida, Samia**, ma très chère binome **Aziza**.*

*À mon fiancé **Mourad** avec un grand remerciement pour leur soutien et toute son famille.*

À Tous mes amis et mes collègues de centre université, surtout filière Informatique.

A tous ceux que j'aime tant et que je n'ai pas cités.

Saida

Table des matières

Résumé.....	13
ملخص.....	14

Introduction générale

1. Motivation.....	16
2. Problématique	17
3. Organisation du mémoire	17

Partie 01: Présentation du domaine d'étude

Chapitre 1:Présentation de l'organisation : (CNR)

Introduction	21
1. La définition de CNR	21
1.1. Statut Juridique	21
1.2. L'établissement de la Caisse.....	21
1.3. Fonctionnement Administratif.....	22
1.3.1. Le Conseil d'Administration.....	22
1.3.2. Le Directeur général	22
1.3.3. Le Siège de la Caisse	22
1.4. Les Missions de la caisse nationale de retraite	22
1.5. L'organigramme de la CNR	24
2. Le système de retraite.....	26
2.1. La pension de retraite	27
2.1.1. Conditions d'octroi.....	27
2.1.2. La retraite proportionnelle et sans condition d'âge	28
2.2. L'allocation de retraite	28
Conclusion.....	28

Chapitre 2: Etude de l'existant

Introduction	30
1. Circuit de traitement d'un dossier de retraite.....	30
2. Problématique et objectif du système	36

Chapitre 3 : Méthodologie 2TUP

Introduction au processus unifié.....	38
---------------------------------------	----

1.	Définition d'un processus de développement logiciel.....	38
1.1.	Processus unifié (Unified Process).....	38
1.2.	Le Processus 2TUP	39
	Conclusion.....	41

Chapitre 4: Architecture client/serveur

	Introduction	43
1.	Définition	43
2.	Caractéristiques des systèmes client serveur	44
3.	Concepts de base d'une architecture Client/serveur	45
3.1.	Définition d'un serveur.....	45
3.2.	Définition d'un client	45
3.3.	Définition d'un middleware.....	45
4.	Types des architectures Client/serveur	46
4.1.	Architecture à 2 niveaux.....	46
4.2.	Architecture à 3 niveaux.....	46
4.3.	L'architecture multi niveaux.....	46
5.	Fonctionnement d'un réseau client/serveur	47
6.	Avantages de l'architecture client/serveur.....	47
7.	Inconvénients de l'architecture client/serveur par rapport à une architecture pair à pair	48
	Conclusion.....	48

Partie 02 : Etude de cas

Chapitre1:Etude préliminaire

	Introduction	51
1.	Cahier des charges.....	51
1.1.	Présentation de projet	51
1.2.	Les grands choix techniques	52
1.3.	Recueil des besoins fonctionnels	52
1.3.1.	La codification des dossiers	52
1.3.2.	La Saisie alpha.....	53
1.3.3.	L'envoi des dossiers.....	55
1.3.4.	La réception des dossiers	55
1.3.5.	La localisation des dossiers	55
1.3.6.	La trace des dossiers	55
1.3.7.	Changer état dossier.....	55
1.3.8.	L'établissement des statistiques.	56
1.4.	Recueil des besoins techniques	56

2.	Identification des acteurs	57
2.1.	L'administrateur.....	57
2.2.	L'agent de saisie.....	57
2.3.	L'employé.....	57
3.	Identification des messages	58
4.	Modélisation de contexte.....	59
4.1.	Diagramme de contexte dynamique.....	60
4.2.	Signification des messages	60
	Conclusion.....	61

Chapitre 2 : Capture de besoins fonctionnels

	Introduction	63
1.	Déterminer les cas d'utilisations	63
1.1.	Liste préliminaire des cas d'utilisation.....	63
1.2.	Diagramme de cas d'utilisation.....	65
2.	Description détaillée des cas d'utilisations.....	66
2.1.	Codifier dossier	66
2.2.	Envoyer dossiers	68
2.3.	Réceptionner dossiers	70
2.4.	Saisir alpha.....	72
2.5.	Changer état dossier.....	74
2.6.	Tracer dossier	76
2.7.	Localiser dossier	77
2.8.	Etablir statistiques dossiers	79
2.8.1.	Etablir statistiques dossiers par utilisateur.....	79
2.8.2.	Etablir statistiques dossiers par service.....	80
	Conclusion.....	82

Chapitre 3 : Capture des besoins technique

	Introduction	84
1.	Capture des spécifications matérielles.....	84
1.1.	Configuration matérielle.....	84
1.2.	Spécification du style d'architecture 2 niveaux	85
2.	Capture des spécifications logicielles.....	85
2.1.	Identification des exploitants du système	86
2.2.	Identification des cas d'utilisation technique.....	86
2.3.	Description des cas d'utilisation technique	86
2.3.1.	Cas d'utilisation « Gérer la sécurité ».....	86

2.3.2. Cas d'utilisation « Gérer l'intégrité des données ».....	92
2.4. Organisation du modèle de spécification logicielle	93
Conclusion.....	94

Chapitre 4 : Analyse

Introduction	96
1. Le découpage en catégories	96
1.1. La répartition des classes candidates en catégories.....	97
1.2. Elaboration des diagrammes de classes préliminaires par catégorie	97
1.3. Dépendance entre catégories.....	98
2. Développement du modèle statique	98
3. Développement du modèle dynamique	100
1.1. Diagrammes d'interaction	100
1.1.1. Diagramme d'interaction de cas « Codifier dossier ».....	100
1.1.2. Diagramme d'interaction de cas « Saisir alpha »	101
1.1.3. Diagramme d'interaction de cas « Changer état ».....	102
1.1.4. Diagramme d'interaction de cas « Localiser dossier »	103
1.1.5. Diagramme d'interaction de cas « Tracer dossier »	104
1.1.6. Diagramme d'interaction de cas « Envoyer dossier »	105
1.2. Diagramme d'état transition	106
Conclusion.....	107

Chapitre 5: Conception préliminaire

Introduction	109
1. Développement modèle de déploiement.....	109
1.1. Architecture adoptée.....	109
1.2. Déploiement du modèle d'exploitation	110
2. Définition des interfaces	111
Conclusion.....	111

Chapitre 6: Conception détaillée

Introduction	113
1. Dictionnaire des données	113
1.1. Les classes et les attributs	113
1.2. les classes association.....	114
1.3. Les opérations	114
.2 Diagramme de classe.....	115
.3 Passage vers le model relationnel.....	115

3.1. Les règles de passage	115
3.2. Les tables de la base de données	116
Conclusion.....	116

Chapitre 7 : Dossier technique

Introduction	118
1. Environnement de travail.....	118
1.1. Le langage de programmation Java.....	118
1.2. IDE NetBeans	119
1.3. Implémentation de la base de données	119
2. Quelques interfaces de notre système.....	120
2.1. Interface authentification	120
2.2. Interface Accueil	120
2.3. Interface Codification.....	121
2.4. Interface Envoi dossier	121
2.5. Interface Réception dossiers	122
2.6. Interface Localisation dossier	122
2.7. Interface Trace dossier	123
Conclusion.....	123

Conclusion générale

Conclusion générale.....	125
Références Bibliographiques.....	126

Liste des figures

Figure 1.1.1: Organigramme de la CNR.	24
Figure 1.2.2 : Diagramme de circuit d'un dossier de retraite directe.	32
Figure 1.2.3:Diagramme de circuit d'un dossier de retraite de réversion.	34
Figure 1.2.4: Diagramme de circuit d'un dossier en révision (ajout).	35
Figure 1.2.5: Diagramme de circuit d'un dossier en révision (renouvellement).	36
Figure 1.3.6: Le système d'information soumis à deux types de contraintes.	39
Figure 1.3.7: le processus de développement en Y.....	40
Figure 1.4.8:Modèle client/serveur.....	43
Figure 1.4.9: Middleware.....	45
Figure 1.4.10: Architecture client/serveur.....	47
Figure 2.1.11: Situation de l'étude préliminaire dans 2TUP.....	51
Figure 2.1.12 : Format de code à barre pour un dossier de retraite direct.	53
Figure 2.1.13 : Diagramme de contexte dynamique.....	60
Figure 2.2.14: Diagramme de cas d'utilisation.....	65
Figure 2.2.15: Diagramme d'activité de cas « Codifier dossier ».	67
Figure 2.2.16: Diagramme de séquence de cas « Codifier dossier ».	67
Figure 2.2.17: Diagramme d'activité de cas « Envoyer dossier ».	69
Figure 2.2.18: Diagramme de séquence de cas « Envoyer dossier ».	70
Figure 2.2.19: Diagramme d'activité de cas « Réceptionner dossier ».	71
Figure 2.2.20:Diagramme de séquence « Réceptionner dossier ».	72
Figure 2.2.21: Diagramme d'activité «Saisir alpha ».....	73
Figure 2.2.22: Diagramme de séquence « Saisir alpha ».	74
Figure 2.2.23 : Diagramme d'activité «Changer état dossier».	75
Figure 2.2.24 : Diagramme de séquence «Changer état dossier».....	75
Figure 2.2.25: Diagramme d'activité « Tracer dossier ».....	76
Figure 2.2.26: Diagramme de séquence « Tracer dossier ».	77
Figure 2.2.27 : Diagramme d'activité de cas «Localiser dossier ».....	78
Figure 2.2.28: Diagramme de séquence « Localiser dossier ».	78
Figure 2.2.29 : Diagramme d'activité « Etablir statistiques dossiers par utilisateur ».	79
Figure 2.2.30 : Diagramme de séquence « Etablir statistiques dossiers par utilisateur ».	80
Figure 2.2.31: Diagramme d'activité « Etablir statistiques dossiers par service».....	81
Figure 2.2.32 : Diagramme de séquence « Etablir statistiques dossiers par service ».	81
Figure 2.3.33 : Situation de la capture des besoins techniques dans 2TUP.....	84
Figure 2.3.34: Architecture 2 niveaux de notre système.....	85
Figure 2.3.35: Modèle de spécification logicielle de système.....	86
Figure 2.3.36 : Diagramme d'activité de cas « S'authentifier ».....	87
Figure 2.3.37 : Diagramme de séquence « S'authentifier ».	88
Figure 2.3.38 : Diagramme d'activité de cas « Gérer comptes ».	90
Figure 2.3.39 : Diagramme de séquence « Gérer compte ».	91
Figure 2.3.40 : Diagramme d'activité de cas « Gérer l'intégrité des données ».	92
Figure 2.3.41 : Diagramme de séquence « Gérer l'intégrité des données ».	93
Figure 2.3.42: Organisation du modèle de spécification logicielle.	94

Figure 2.4.43: Situation de l'étape d'analyse dans le 2TUP.	96
Figure 2.4.44: Découpage en catégories.	97
Figure 2.4.45 : Découpage des classes de chaque catégorie.	98
Figure 2.4.46 : Dépendance entre catégories.....	98
Figure 2.4.47 : Diagramme de classe de catégorie traçabilité dossier	99
Figure 2.4.48 : Diagramme de classe de catégorie service.	99
Figure 2.4.49: Diagramme d'interaction de cas d'utilisation « Codifier dossier ».....	100
Figure 2.4.50 : Diagramme d'interaction de cas d'utilisation« Saisir alpha ».....	101
Figure 2.4.51: Diagramme d'interaction de cas d'utilisation« Changer état »	102
Figure 2.4.52: Diagramme d'interaction de cas d'utilisation« Localiser dossier ».....	103
Figure 2.4.53: Diagramme d'interaction de cas d'utilisation« Tracer dossier»	104
Figure 2.4.54 : Diagramme d'interaction de cas d'utilisation« Envoyer dossiers ».....	105
Figure 2.4.55 : Diagramme d'état transition « dossier »	106
Figure 2.5.56 : Situation de la conception préliminaire dans 2TUP.....	109
Figure 2.5.57 : Schéma du modèle de déploiement de notre système.	110
Figure 2.5.58 : Définition des applications dans le modèle d'exploitation.	110
Figure 2.6.59 : Diagramme de classe	115
Figure 2.7.60: Logo du langage de programmation Java.	118
Figure 2.7.61: Capture écran de l'EDI NetBeans.	119
Figure 2.7.62: Capture fenêtre authentification.	120
Figure 2.7.63: Capture interface Accueil	120
Figure 2.7.64 : Capture fenêtre codification.....	121
Figure 2.7.65: Capture fenêtre Envoi dossier	121
Figure 2.7.66: Capture fenêtre Réception dossiers	122
Figure 2.7.67: Capture fenêtre Localisation dossier	122
Figure 2.7.68: Capture fenêtre Trace dossier	123

Liste des tableaux

Tableau 1.2.1 : fiche d'accompagnement des opérations.	33
Tableau 2.1. 2: Les caractères du champ régime d'affiliation et leur signification.....	54
Tableau 2.1.3: Les caractères du champ régime de liquidation et leur signification.....	54
Tableau 2.1.4: Les caractères du champ avantage des dossiers directes et leur signification..	54
Tableau 2.1.5: Les caractères du champ avantage des dossiers de réversion et leur signification.	55
Tableau 2.1.6: Nomenclature des utilisateurs dans chaque service.....	58
Tableau 2.1.7 : Modélisation des contextes.	59
Tableau 2.1.8: Signification des messages.....	61
Tableau 2.2.9 : Les cas d'utilisations.	64
Tableau 2.2.10 : Fiche descriptive textuelle du cas «Codifier dossiers».....	66
Tableau 2.2.11: Fiche descriptive textuelle du cas «Envoyer dossiers».....	68
Tableau 2.2.12: Fiche descriptive textuelle du cas « Réceptionner dossiers».....	70
Tableau 2.2.13: Fiche descriptive textuelle du cas « Saisir alpha ».....	72
Tableau 2.2.14: Fiche descriptive textuelle du cas « Changer état dossier ».....	74
Tableau 2.2.15: Fiche descriptive textuelle du cas « Tracer dossier ».....	76
Tableau 2.2.16: Fiche descriptive textuelle du cas « Localiser dossier ».....	77
Tableau 2.2.17: Fiche descriptive textuelle du cas « Etablir statistiques dossiers par utilisateur ».....	79
Tableau 2.2.18: Fiche descriptive textuelle du cas « Etablir statistiques dossiers par service ».....	80
Tableau 2.3.19: Fiche description textuelle du cas « S'authentifier».....	87
Tableau 2.3.20: Fiche descriptive textuelle du cas « Gérer comptes».....	90
Tableau 2.3.21: Fiche descriptive textuelle « Gérer l'intégrité des données ».....	92
Tableau 2.3.22 : les interfaces de notre système	111
Tableau 2.3.23: Tableau des classes et des attribues.	113
Tableau 2.3.24: Tableau des classes association et des attribues.	114
Tableau 2.3.25 : Tableau des opérations.....	114
Tableau 2.3.26: les tables de la base de données.....	116

Résumé

La Caisse Nationale de Retraite de Mila cherche toujours à adopter une meilleure gestion de ses dossiers en introduisant de nouvelles techniques permettant de rendre le suivi des dossiers plus facile et plus sûr.

Le présent mémoire synthétise le travail effectué dans le cadre d'un stage effectué au sein de la Caisse Nationale de Retraite. Le but de cette étude est la traçabilité des dossiers à travers les différents services de l'organisation.

Ainsi nous avons mis en place une application client/serveur mettant en œuvre la traçabilité des dossiers des retraites depuis le dépôt du dossier jusqu'au mandatement. Notamment, la codification, la localisation, la trace, La gestion des flux des dossiers.etc.Ceci permettra de diagnostiquer rapidement les facteurs internes ou externes liés aux retards accusés lors du transit des dossiers à travers les services et d'y remédier. Nous avons adopté pour cela la méthode 2TUP comme méthode de développement, UML comme langage de modélisation, JAVA comme langage de programmation et le SGBD Oracle pour l'implémentation de la base de données.

Mot clé: Traçabilité, 2TUP, UML , JAVA.

ملخص

الصندوق الوطني للتقاعد ميلة يسعى دائما إلى اعتماد إدارة أفضل السجلات من خلال إدخال تقنيات جديدة لجعل مراقبة الملفات أسهل وأكثر أمانا. هذه المذكرة تلخص العمل المنجز كجزء من التربص داخل الوكالة الوطنية للتقاعد. والهدف من هذه الدراسة هو تتبع الملف بين مختلف مصالح المؤسسة. لذلك قمنا بتطوير نظام معلومات آلي يسمح بتتبع ملفات المتقاعدين. خاصة، تدوين، المكان الحالي، وتسيير تبادل الملفات، الخ. هذا سيسمح بتحديد العوامل الداخلية أو الخارجية المتسببة في تأخير إرسالها بين مختلف المصالح من أجل معالجتها. لقد اعتمدنا على أسلوب 2TUP كوسيلة للتصميم، UML كلغة للنمذجة، JAVA كلغة برمجة و SGBD Oracle لتنفيذ قاعدة البيانات.

الكلمات المفتاحية: 2TUP, UML, JAVA، أثر.

INTRODUCTION

GENERALE

Préambule

L'informatique a connu durant cette dernière décennie une avancée technologique fulgurante et spectaculaire surtout dans le développement. En effet, l'implémentation des applications professionnelles est devenue plus simple et plus pratique. Mais malgré cela, la plupart des administrations et entreprises algériennes ne disposent pas de systèmes informatiques qui facilitent leur travail.

La caisse nationale de retraite (CNR) ne fait pas exception. En effet, malgré la présence d'un système informatique et d'un réseau informatique, elle souffre jusqu'à aujourd'hui d'un manque dans ce domaine.

Dans le cadre de ce thème qui est l'étude de traçabilité des dossiers de retraite, nous allons essayer de mettre en place un système informatique permettant le suivi des flux des dossiers entre les différents services de la CNR.

1. Motivation

Au terme du stage que nous avons effectué au sein la caisse nationale de retraite, nous avons trouvé que la CNR utilise un système informatique pour la gestion des dossiers de la retraite.

La quantité des dossiers traités et échangés est énorme, Malheureusement, la direction des pensions ne dispose d'aucune application informatique permettant la traçabilité des dossiers, ce qui rend le processus de localisation et trace des dossiers difficiles. Ceci est dû au fait que durant la période séparant la date de dépôt du dossier et le mandatement, le dossier transite à travers différents services où il sensé subir plusieurs traitements. Le dossier peut être sujet à des retards de traitements dûs à plusieurs facteurs peut être internes ou externes à la CNR. Ces retards quelque fois exagérés, provoquent des réclamations de la part des retraités. Pour appréhender ce problème, la direction de la CNR nous a proposé de mettre en place une application informatique permettant d'enregistrer le transit des dossiers au cours de leur étude technique et leur liquidation.

Ainsi, l'objectif de ce projet est la conception et l'implémentation d'une application client/serveur qui permet la traçabilité des dossiers de retraite, notamment :

- ❖ La codification des dossiers par code à barre pour simplifier l'échange des dossiers entre différents services, vu leur nombre important.
- ❖ La gestion des flux des dossiers.
- ❖ La trace et la localisation des dossiers.

2. Problématique

Au cours de notre stage au niveau de la direction des pensions de la caisse nationale de retraite de la wilaya de Mila, nous avons remarqué que la stratégie de recherche d'un dossier prend beaucoup de temps et d'efforts vu la quantité énorme des dossiers traités.

Notre stage pratique a porté sur le suivi de la traçabilité des dossiers de retraite depuis la réception jusqu'au mandatement.

Malgré la présence d'un système de gestion des dossiers de retraite (STAR), certains problèmes restent toujours posés et peuvent être résumés comme suit :

- ☞ Impossibilité de savoir la localisation exacte d'un dossier de retraite.
- ☞ Impossibilité de faire des statistiques par utilisateur sur les dossiers.
- ☞ Indisponibilité des informations concernant les causes provoquant les retards de traitement des dossiers et quels postes en sont responsables.

Vu le nombre très important des dossiers de retraite à traiter au niveau de la CNR et compte tenu des retards très fréquents accusés lors de leur transit à travers les services, le système que nous allons concevoir intervient pour établir la trace des dossiers et rendre de leurs états à tous moments, afin de détecter le nombre des dossiers en instance, liquidés, etc. Ainsi, on diagnostiquera automatiquement les facteurs causant ces retards pour tenter d'y remédier.

3. Organisation du mémoire

Comme nous avons adopté la méthode 2TUP pour l'analyse et la conception de notre système, notre mémoire est structuré selon les phases de cette méthode.

L'introduction générale explique les motivations de ce projet, et en on présente les grandes fonctionnalités attendues.

La première partie : Présentation du domaine d'étude

Cette partie présente le domaine d'étude qui est en fait une synthèse de la documentation faite autour de la CNR. Elle s'articule autour de quatre chapitres:

Le Chapitre 01 présente l'entreprise (CNR) de la wilaya de Mila où nous avons effectué notre stage.

Le Chapitre 02 explique les notions fondamentales liées au dossier de retraite et le circuit de ce dossier au niveau des services de la CNR.

Le Chapitre 03 présente brièvement les concepts fondamentaux du processus de développement que nous avons adopté pour aboutir à notre système.

Le Chapitre 04 décrit l'architecture client/serveur.

La deuxième partie : Etude de cas

Cette partie est constituée de l'essentiel du travail d'ingénierie de système d'information que nous avons effectué, elle s'articule autour des phases essentielles de la méthode 2TUP, et qui sont:

➤ **Le Chapitre 01: L'étude préliminaire**

Dans cette phase, nous dressons une ébauche du cahier des charges qui contient les activités de capture des besoins fonctionnels et de capture des besoins techniques.

➤ **Le Chapitre 02: La capture des besoins fonctionnels**

Dans cette phase, nous complétons le recueil des besoins décrits durant l'étude préliminaire. Cette étape représente la branche gauche du cycle en Y puisqu'elle décrit les différentes fonctionnalités du système et la façon de les utiliser.

➤ **Le Chapitre 03: La capture des besoins techniques**

Dans cette phase nous citons les spécifications techniques du système (l'architecture physique ainsi que les cas d'utilisation techniques).

➤ **Le Chapitre 04: La phase d'analyse**

Dans cette phase, nous découpons notre système en paquetages et définissons les classes d'analyse du système, le modèle statique et le modèle dynamique.

➤ **Le Chapitre 05: La conception préliminaire**

Dans cette phase, nous fusionnons les résultats des études fonctionnelles et techniques pour présenter le modèle de conception.

➤ **Le Chapitre 06: La conception détaillée**

Dans cette phase, nous présentons la conception détaillée des classes et leurs associations.

➤ **Le Chapitre 07: Le dossier technique**

Dans ce chapitre, nous allons présenter les outils de développement que nous avons utilisés et exposons quelques interfaces de l'application.

Nous terminons notre mémoire par une conclusion générale

Partie

Présentation

du

domaine d'étude

1

Chapitre 01 :

PRESENTATION
DE

L'ORGANISATION : (CNR)

Introduction

Dans ce chapitre, on va essayer de présenter l'organisme d'accueil et la structure de La Caisse Nationale des Retraites, puis nous enchainons sur la description du transit des dossiers à travers les services.

1. La définition de CNR

L'agence de la Caisse Nationale de Retraite(CNR): est une agence de wilaya d'une qualité nationale dans chaque wilaya, elle a été créée par le décret N° : 85-223 du 20 août 1985 abrogé et remplacé par le décret N° :92-07 du 04 janvier 1992 portant statut juridique des Caisses de sécurité sociale et organisation administrative et financière de la Sécurité Sociale.

1.1.Statut Juridique

La Caisse Nationale des Retraites est un établissement public à gestion spécifique régi par les Lois applicables en la matière. Le décret N°92-07 du 04 Janvier 1992 précise dans son article 2 que la caisse est dotée de la personnalité morale et de l'autonomie financière.

1.2.L'établissement de la Caisse

La caisse nationale de retraite a été établie en l'obligation du décret numéro 85/223 daté le 20 aout 1985 abrogé et remplacé par le décret numéro 7-92 daté le 4 janvier 1992 qui contient le code principal des caisses de protection social et l'organisation administratif.

Il faut rappeler que la caisse nationale des retraites est un résultat du ralliement de 8 caisses en 1985. Et qu'elle a été chargée de « manager » les différents systèmes de retraite trouvés avant l'établissement du système de retraite national unifié en l'année 1983. Parmi les caisses :

- La caisse nationale de sécurité sociale des non-salariés (CASNOS) chargée de manager les bénéficiaires des pensions des non-salariés.
- La caisse générale des retraités algériens (CGRA) chargée de manager les bénéficiaires de pensions qui s'engagent dans le système des employés.
- La caisse nationale les abonnements agriculture(CNMA) chargée de manager les bénéficiaires de pension du système agri culturelles.

1.3.Fonctionnement Administratif

Les organes essentiels chargés d'assurer le fonctionnement de la caisse sont :

- Le conseil d'administration.
- Le directeur général.

1.3.1. Le Conseil d'Administration

Il administre, contrôle et anime la Caisse. Il est composé de 29 membres répartis comme suit :

- 18 représentants des travailleurs par les organisations syndicales les plus représentatives.
- 9 représentants des employeurs dont 2 représentants de la fonction publique.
- 2 représentants du personnel de la Caisse.

1.3.2. Le Directeur général

Le directeur général dirige la Caisse et assure son fonctionnement sous le contrôle du conseil d'administration.

1.3.3. Le Siège de la Caisse

Le siège de la caisse est chargé notamment :

- D'organiser, de planifier, de coordonner et de contrôler: Les activités des agences de wilaya et d'antennes d'administration ou d'entreprise, La gestion des équipements et des moyens humains et matériels de la caisse; De gérer le budget de la caisse, de coordonner les opérations financières et de centraliser la comptabilité générale.
- De coordonner le recouvrement des cotisations de retraite.
- De gérer et de reconstituer les carrières des assurés sociaux.
- D'organiser l'information des assurés sociaux et des employeurs.
- De suivre l'application des conventions et accords en matière de retraite.

1.4.Les Missions de la caisse nationale de retraite

Elles sont fixées par l'Article 9 du décret N° : 92-07 du 04 janvier 1992. Elles sont les suivantes :

- Gérer les pensions et allocations de retraite, ainsi que les pensions et allocations des ayant-droit.
- Gérer jusqu'à extinction des droits, les pensions et allocations servies au titre de la législation antérieure au 1er Janvier 1984.

- Assurer le recouvrement, le contrôle et le contentieux du recouvrement des cotisations destinées au financement des prestations de retraite.
- Mettre en application les dispositions relatives prévues par les conventions nettes accords internationaux de sécurité sociale.
- Assurer l'information des bénéficiaires et des employeurs.
- Gérer le fonds d'aide et de secours en application de l'article 52 de la Loi N° : 83-12 du 02 Juillet 1983 relative à la retraite.

1.5.L'organigramme de la CNR

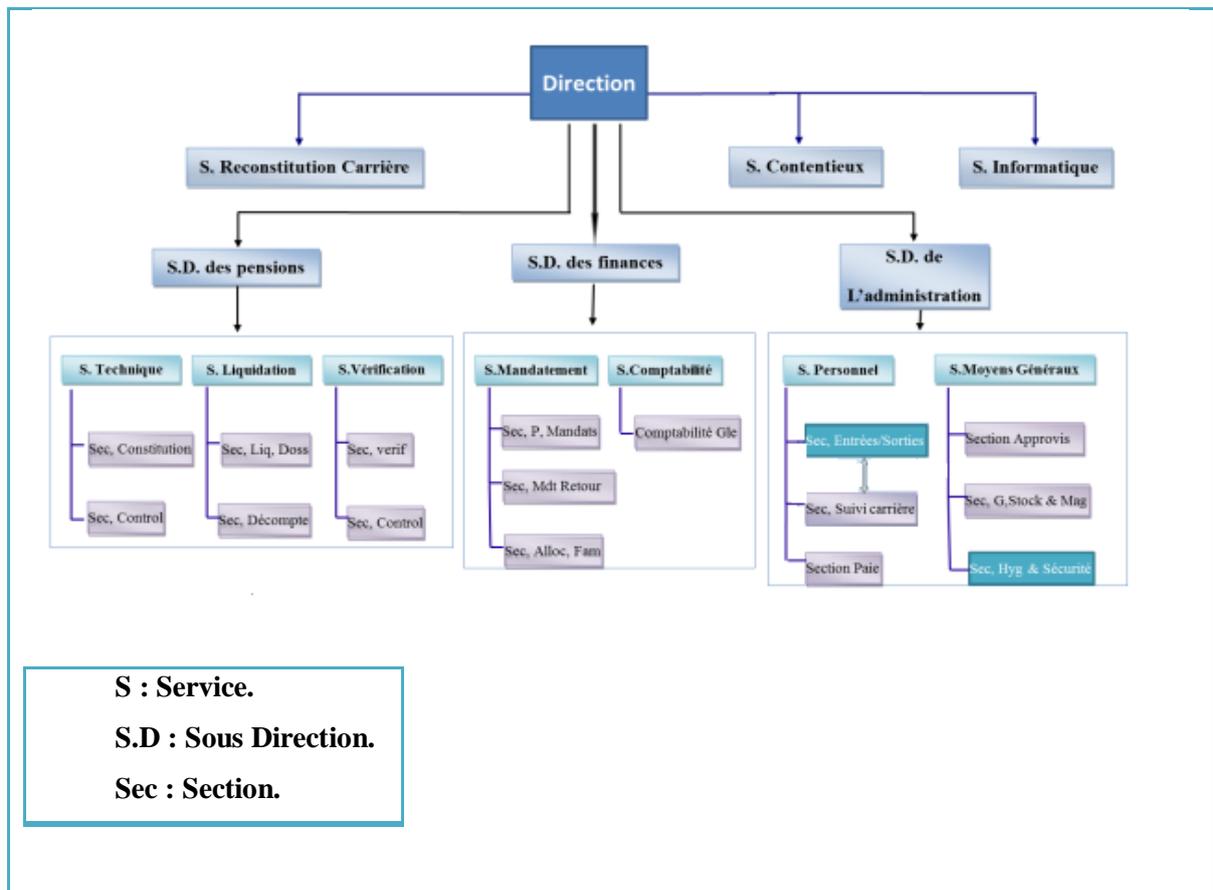


Figure 1.1.1: Organigramme de la CNR.

La caisse nationale de retraite consiste en:

✓ **La Direction d'agence**

C'est la structure qui coordonne et suit les différentes sous-directions et services et également la liaison entre la direction générale et l'agence.

✓ **Le service contentieux**

C'est la structure responsable du suivi des fichiers des retraites, pour des causes juridiques ou administratives. Elle joue aussi le rôle de l'intermédiaire juridique entre l'assuré socialement (la retraite) et l'administration (la délégation de la direction des pensions).

Ses fonctions sont :

- Suivre les thèses au niveau des tribunaux.
- Probité de la commission locale qualifiée pour le recours préconçu.
- Suivre les encaissés de l'excès (surplus) pour les retraités défunts.
- Recevoir les recours des retraités plus les plaintes.

✓ **Le service Informatique**

C'est un service chargé du traitement des données et l'envoi de tous les imprimés indiqués pour les pensions. Il est également chargé de concevoir les programmes nécessaires à la gestion de la CNR.

Parmi ses fonctions:

- Aider les utilisateurs de l'informatique et leurs accompagnants
- Préparer et copier les mandats et les envoyés au service de poste et les donner aux personnes bénéficiaires de retraite.
- Élaborer et préparer des programmes spéciaux pour les besoins locaux.
- Réparation et entretien de matériel et de logiciel etc....

✓ **Sous direction de l'administration**

➤ **Le service personnel :**

Ce service est responsable de l'organisation du travail, du cheminement de carrière des travailleurs. Il est aussi le responsable de toutes les possessions de l'agence.

Parmi ses fonctions:

- Affecter les employés aux services.
- Enregistrer les absences et présence des employés.
- Recevoir les demandes de travail.

- Manager le temps.

➤ **Le service des moyens généraux**

Ce service est responsable de l'approvisionnement de l'équipement du matériel et du mobilier, Il effectue le suivi du stock, le bercail des voitures et l'inventaire annuel. Il contient quatre bureaux qui sont : le bureau de moyens généraux, la sécurité, le stock, la propreté.

Parmi ses fonctions :

- L'accomplissement des opérations d'approvisionnement en matériel, les meubles et les outils.
- Manager et suivre les projets de l'accomplissement des établissements de la caisse.
Poser l'inventaire des possessions transporté et foncier de la caisse.

✓ **La Sous direction des pensions**

Elle contient quatre services:

- **Le service de suivi de carrière:** est considéré comme un point de liens entre l'extérieur et l'intérieur car il vérifie la situation et le parcours de travail concernant le salarié pour confirmer les informations du demandeur de retraite.
- **Le service de liquidation:** est un service responsable du contrôle préliminaire du dossier de l'étude et du calcul des pensions ainsi que l'établissement d'une fiche de rachat et une fiche de liquidation.
- **Le service vérification:** Le vérificateur contrôle minutieusement l'ensemble des éléments de dossier ainsi que le contenu de la fiche de liquidation, notamment l'identité de l'assuré, la validation, Le salaire de référence, et signe le dossier et appose sa griffe.
- **Le service technique:** il est chargé de la réception des dossiers, la vérification de toutes les pièces qui les constituent puis la saisie des données concernent chaque dossier.

2. Le système de retraite

Le système de retraite en Algérie, applicable à partir du 01 janvier 1984 se présente comme un système assurantiel, contributif, caractérisé par une solidarité entre les actifs et les retraités, fonctionnant donc sur le principe de la répartition. Ce système se caractérise par le champ de protection qu'il couvre, c'est-à-dire les salariés concernés par les dispositions de cette loi.

En cas de décès du retraité, une pension de réversion est servie en faveur:

- Du conjoint survivant.
- Des orphelins.
- Des ascendants.

Un retraité peut bénéficier d'une pension ou d'une allocation de retraite.

2.1.La pension de retraite

En vertu de la législation en vigueur, les bénéficiaires d'une pension de retraite sont les personnes salariées, travaillant sur le territoire national à quelque titre ou quelque lieu que ce soit, pour un ou plusieurs employeurs ou organismes employeurs.

2.1.1. Conditions d'octroi

a. La Condition d'âge

L'âge légal pour pouvoir bénéficier d'une pension de retraite est fixé à :

- 60 ans pour l'homme.
- 55 ans pour la femme.

Aucune condition d'âge n'est requise pour l'octroi d'une pension de retraite dans les deux cas ci-après :

- Lorsque le travailleur est atteint d'une incapacité totale et définitive.
- Lorsque le travailleur a la qualité de moudjahid.

b. La Condition de travail

Pour pouvoir bénéficier d'une pension de retraite, le travailleur doit avoir travaillé pendant au moins quinze (15) années.

Cette durée minimale doit avoir donné lieu à un travail effectif et à versement de cotisations pendant une période égale à au moins sept ans et six mois.

La validation des années de travail s'effectue sur les bases suivantes :

- Est considérée et compte comme une année d'assurance, l'année ayant donné lieu à au moins 180 jours ou 1440 heures de travail.
- Est validée pour un trimestre, le trimestre au cours duquel ont été accomplis au moins 45 jours ou 360 heures de travail.

2.1.2. La retraite proportionnelle et sans condition d'âge

Avec la parution de l'ordonnance 97-13 du 31 mai 1997, qui assouplit les conditions d'octroi de la retraite, le bénéfice de la pension peut être accordé avec jouissance immédiate, avant l'âge prévu à l'article 6 de la loi 83-12 du 02 juillet 1983, dans les cas et selon les modalités ci-après :

- Sans aucune condition d'âge lorsque le travailleur salarié a accompli une durée de travail effectif ayant donné lieu à versement de cotisations égales à trente deux (32) ans au moins.
- A partir de l'âge de cinquante (50) ans, le travailleur salarié qui réunit une durée de travail effectif ayant donné lieu à un versement de cotisations égales à vingt (20) ans au moins, peut demander le bénéfice d'une pension de retraite proportionnelle.

2.2.L'allocation de retraite

Les travailleurs âgés d'au moins 60 ans, ne justifiant pas à ce âge de la condition de travail requise (15 ans) mais pouvant valider au moins cinq (05) années ou vingt (20) trimestres ont droit à une allocation de retraite.

Le montant de cette allocation est proportionnel au nombre d'années d'activité et n'est pas porté au minimum des pensions de retraite.

Conclusion

Dans ce chapitre, nous avons présenté l'organisme d'accueil avec ses différents services. Dans le prochain chapitre nous allons présenter le circuit des dossiers des retraités au niveau de CNR.

Chapitre 02 :

ETUDE DE L'EXISTANT

Introduction

Parmi les fonctions principales des services de sous direction des pensions, le traitement des dossiers des retraités afin d'offrir un service de qualité. Donc, on va concentrer dans ce chapitre sur le processus de traitement des différents types de dossiers au niveau de ces services en en soulignant particulièrement le chemin parcouru.

1. Circuit de traitement d'un dossier de retraite



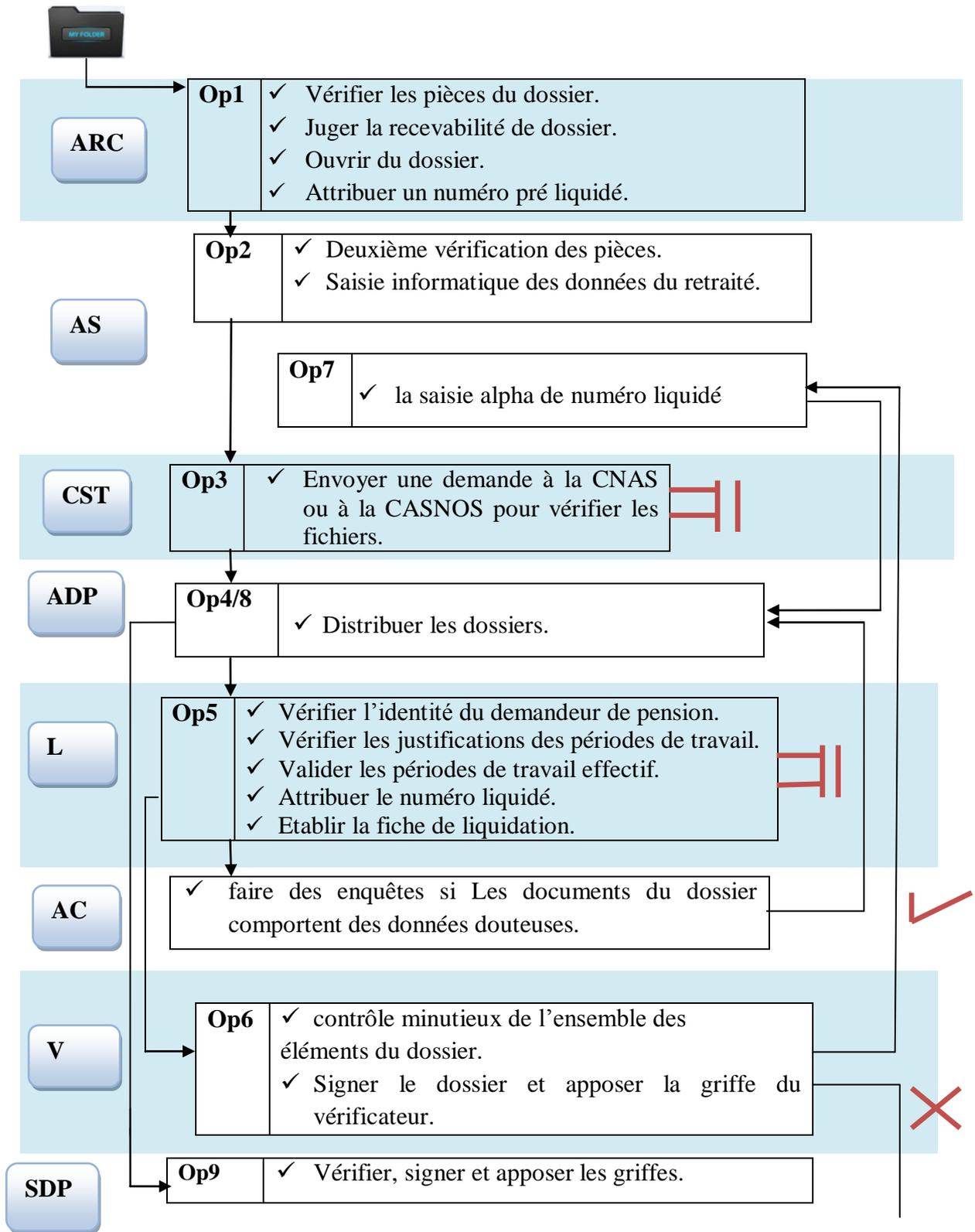
La retraite directe

Une fois un nouveau dossier de demande de pension est déposé par:

1. L'employeur.
2. L'intéressé.
3. Ou transmis par voie postale.

Il passe par les postes de travail suivants:

- ⇒ **ARC**: L'agent de réception et constitution.
- ⇒ **AS**: L'agent de saisie.
- ⇒ **CST** : Le chef de service technique.
- ⇒ **ADP**: L'agent de divisions des pensions.
- ⇒ **L**: Le liquidateur.
- ⇒ **V**: Le vérificateur.
- ⇒ **SDP**: Le sous-directeur des pensions.
- ⇒ **DA**: Le directeur d'agence.
- ⇒ **SDF**: Le sous-directeur de finances.
- ⇒ **AC**: L'agent de contrôle.
- ⇒ **A**: L'archiviste.
- ⇒ **ASC**: L'agent de suivi et contrôle.



- : Dossier en instance
- : Dossier liquidé
- : Dossier rejeté

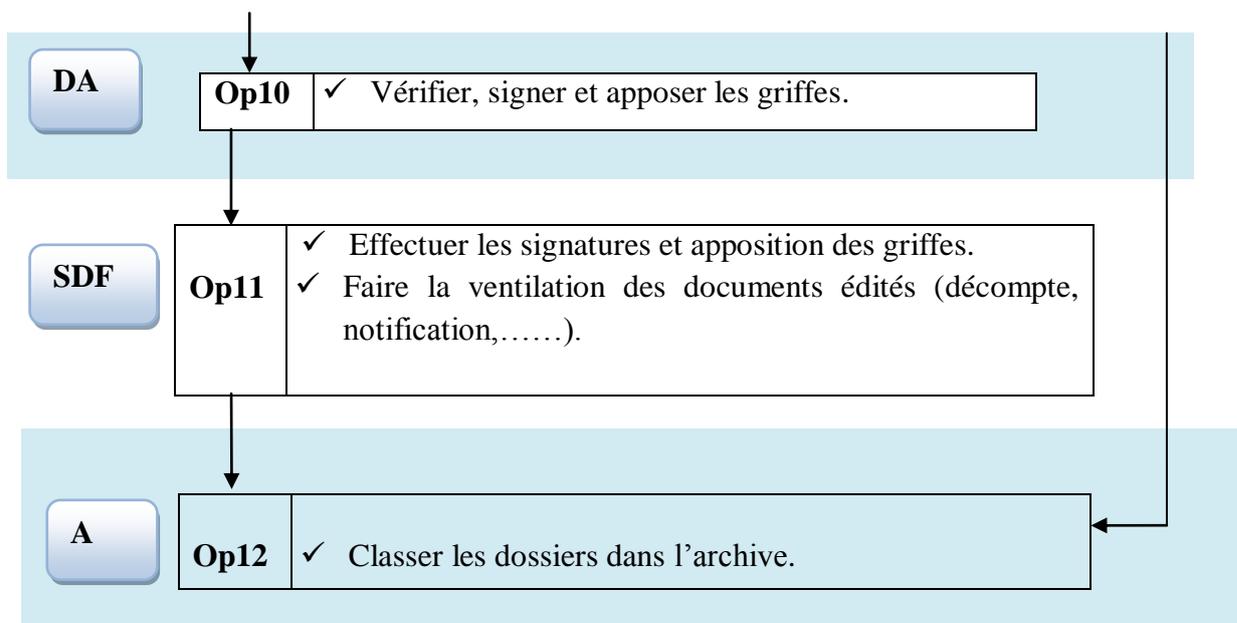


Figure 1.2.2 : Diagramme de circuit d'un dossier de retraite directe.

🕒 Description détaillée des quelques opérations :

N° Op	Description de l'opération
Op1	<p>Vérification des pièces composant le dossier: état civil, salariat et autre.</p> <ul style="list-style-type: none"> • S'assurer que le demandeur réunit les conditions d'âge et de durée d'activité. • Contrôle de l'imprimé de demande. • Contrôle de l'identité de demandeur. <p>Juger la recevabilité de dossier par:</p> <ul style="list-style-type: none"> • Le refus de sa réception en justifiant la ou les raisons : (manque document, divergence de documents, condition d'âge non remplie, etc.), et porter à la connaissance de l'intéressé, par écrit, le manque de ces pièces, ou tout autre motif légal de refus. • Sa réception et, dans ce cas, le compostage de l'ensemble des pièces. • Remise d'un accusé de réception. <p>Attribution du numéro pré liquidé:</p> <p>Un numéro de pré liquidé permet une identification provisoire du dossier de demande de retraite en attendant l'attribution d'un numéro de liquidation. Il est porté sur l'accusé de réception.</p>

	<p>Ouverture de dossier:</p> <ul style="list-style-type: none"> • Classer le dossier dans une chemise appropriée. • Inscrire Nom, Prénom, date de naissance, N°PL sur la chemise. • Apposer les cachets.
Op2	<p>Deuxième vérification des pièces</p> <p>Saisie informatique:</p> <p>Renseigne la base de données, en saisissant toutes les informations concernant le demandeur avec N°PL.</p>
Op3	<p>Envoyer une demande à la CNAS ou à la CASNOS pour vérifier les fichiers.</p>
Op4 /8	<p>Distribution des dossiers: sur les liquidateurs puisque les dossiers sont très nombreux.</p>
Op5	<p>Vérification de l'identité du demandeur de pension:</p> <p>Vérifier l'identité de l'assuré sur la demande, les pièces d'état civil et les documents de salariat.</p> <p>Vérification des justifications des périodes de travail :</p> <p>Vérifier si toutes les périodes signalées dans la demande, sont justifiées par des certificats, etc.</p> <p>Etablissement de la fiche de liquidation :</p> <ul style="list-style-type: none"> • Contient : L'identité de l'assuré, La validation, Le salaire de référence. • signée et portant la griffe du liquidateur. <p>Demander de l'agent de contrôle la corroboration des justificatifs:</p> <p>du secteur privé, du secteur agricole, des périodes de stage, et des documents comportant des données douteuses.</p>

Tableau 1.2.1 : fiche d'accompagnement des opérations.



La retraite de réversion

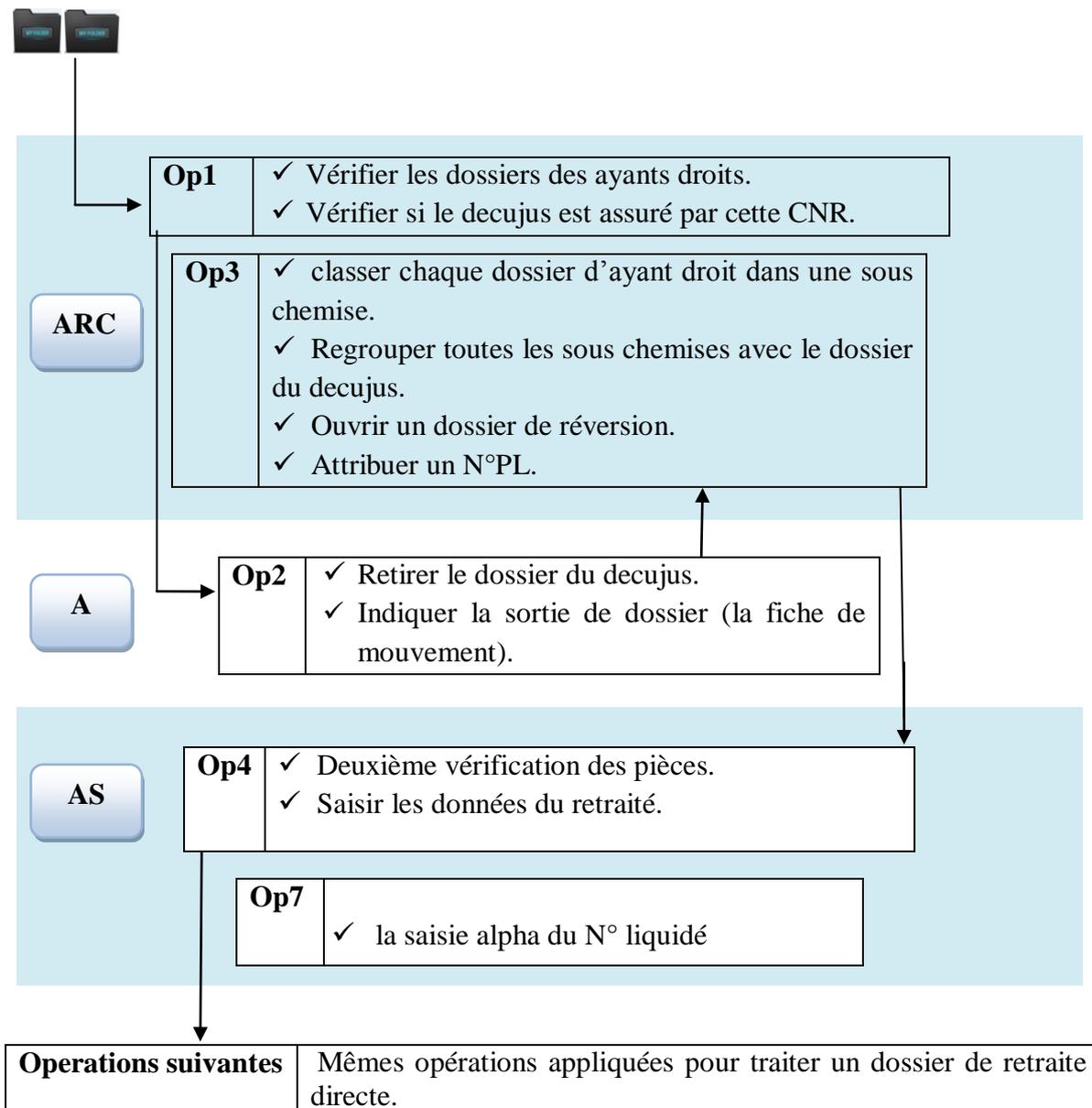


Figure 1.2.3: Diagramme de circuit d'un dossier de retraite de réversion.



Révision

Un dossier de retraite mis en paiement, peut, à tout moment faire l'objet d'un réexamen, à la demande de son titulaire, et, si les raisons invoquées sont fondées et justifiées, le dossier est révisé. Plusieurs motifs peuvent être à l'origine d'une révision, les plus fréquents sont:

- ✓ L'ajout d'un nouvel élément au dossier (changement de salaire,.....),
- ✓ Le renouvellement des pièces de dossier direct, ou de réversion.

● L'ajout d'un nouvel élément au dossier

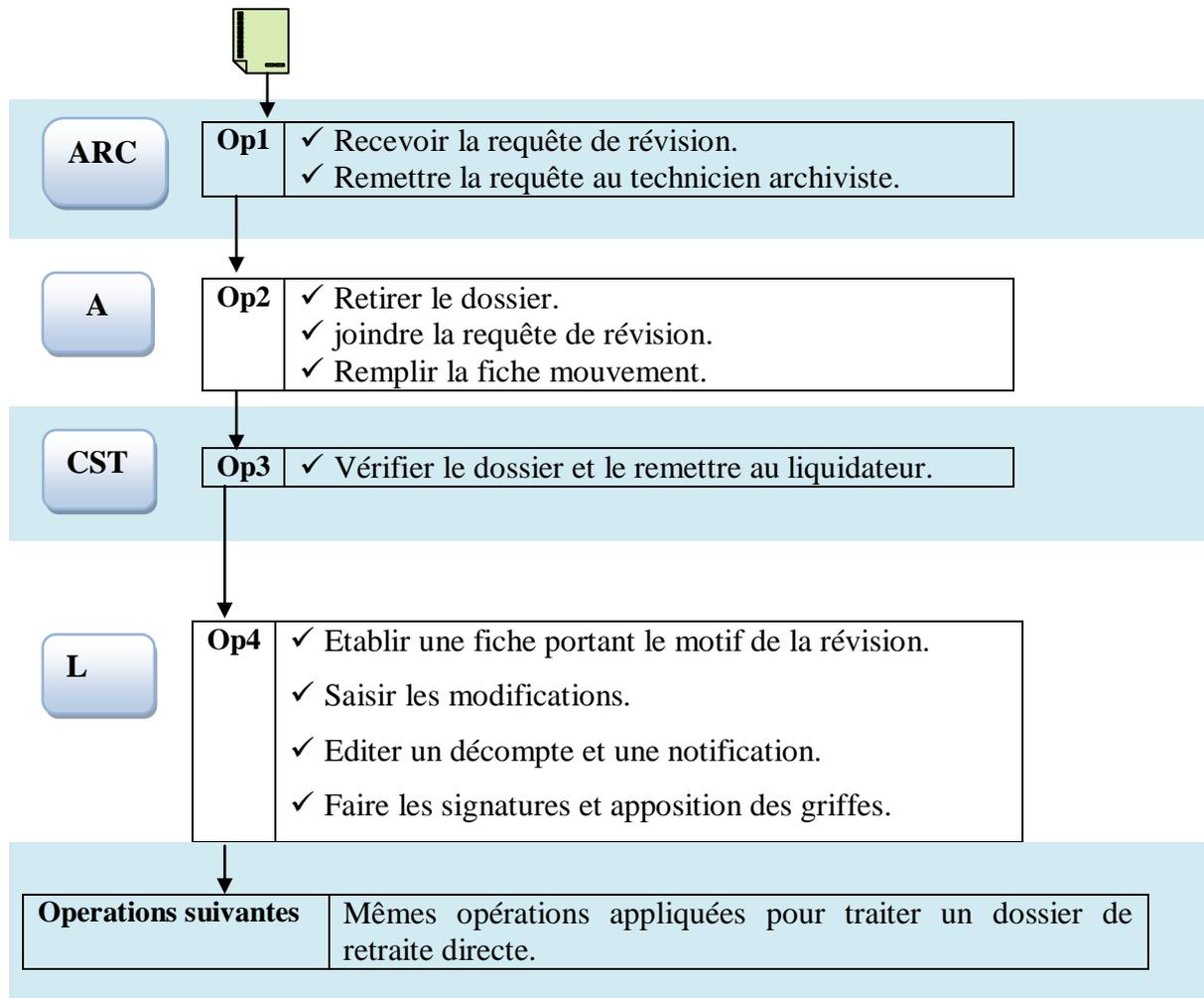
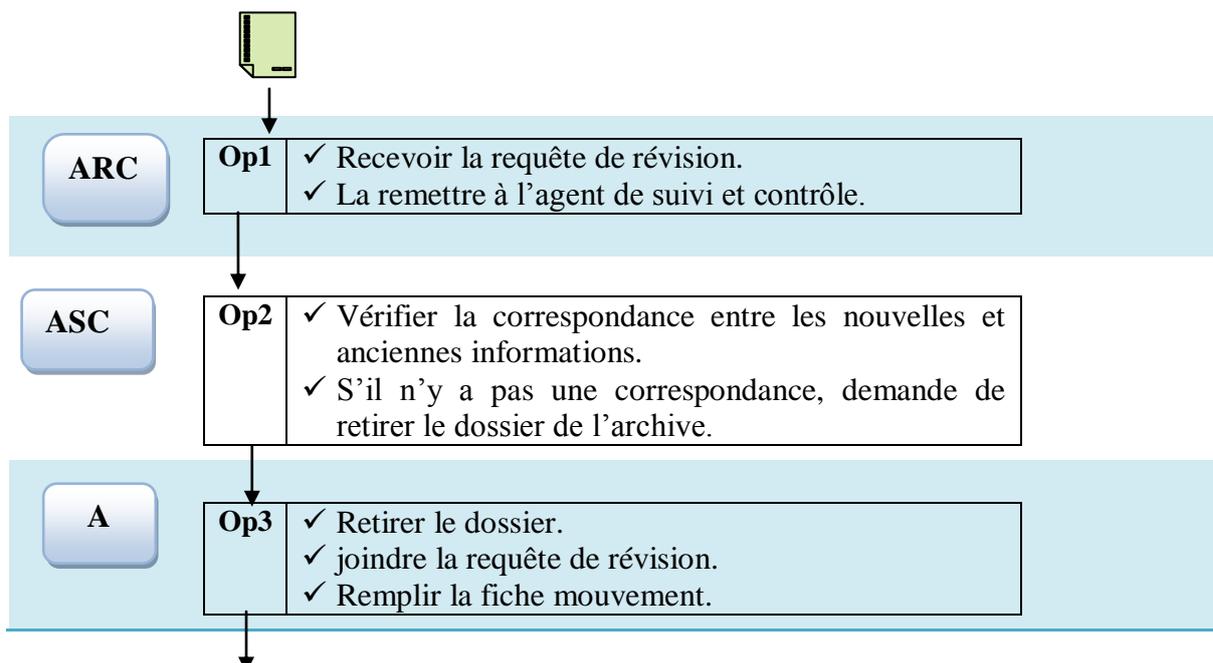


Figure 1.2.4: Diagramme de circuit d'un dossier en révision (ajout).

● Le renouvellement des pièces de dossier



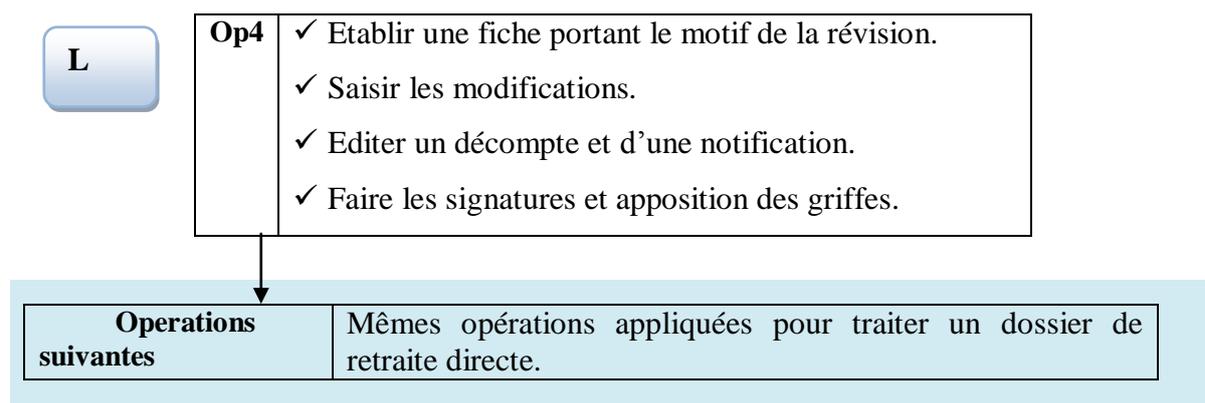


Figure 1.2.5: Diagramme de circuit d'un dossier en révision (renouvellement).

2. Problématique et objectif du système

Un système d'information est un ensemble organisé d'éléments qui permet de regrouper, de classer et de diffuser l'information sur un phénomène donné.

Notre stage pratique a porté sur le suivi de la traçabilité des dossiers de retraite depuis la réception jusqu'au mandatement.

Malgré la présence d'un système de gestion des dossiers de retraite (STAR), certains problèmes restent toujours posés et peuvent être résumés comme suit :

- ☞ Impossibilité de savoir la localisation exacte d'un dossier de retraite.
- ☞ Impossibilité de faire des statistiques par utilisateur sur les dossiers.
- ☞ Indisponibilité des informations concernant les causes provoquant les retards de traitement des dossiers et quels postes en sont responsables.

Vu le nombre très important des dossiers de retraite à traiter au niveau de la CNR et compte tenu des retards très fréquents accusés lors de leur transit à travers les services, le système que nous allons concevoir intervient pour établir la trace des dossiers et rendre de leurs états à tous moments, afin de détecter le nombre des dossiers en instance, liquidé, etc. Ainsi, on diagnostiquera automatiquement les facteurs causant ces retards pour tenter d'y remédier.

Chapitre 03 :

METHODOLOGIE 2TUP

Introduction au processus unifié

Le succès du projet dépend de l'adéquation du processus de développement qui est une étape décisive pour l'élaboration d'une application indépendante de toute plateforme d'exécution et de tout langage de programmation, En effet, le processus de développement est constitué d'une succession de phases (spécification, conception et réalisation).

Il existe plusieurs processus de développement qui sont issus du processus générique UP, on cite notamment 2TUP, RUP, XP, AUP et Open UP. Notre choix d'un processus de développement pour notre application s'est porté sur la variante 2TUP. Ce processus se base lui-même sur le Processus Unifié (Unified Process) qui est devenu un standard générique réunissant les meilleures pratiques de développement.

Cette méthode ne se base aucunement sur un processus linéaire mais bien, sur un développement itératif et incrémental. Mais étant donné la taille de notre projet qui est relativement petite, nous avons élaboré un seul cycle de développement.

1. Définition d'un processus de développement logiciel

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

L'objet d'un processus de développement est de produire des logiciels de qualité qui répondent aux besoins de leurs utilisateurs. [1]

1.1.Processus unifié (Unified Process)

Un processus unifié est une méthode de développement logiciel construite en se basant sur UML, il est itératif, incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques [1].

- **Itératif et incrémental:** la méthode est itérative dans le sens où elle propose de faire des itérations lors de ses différentes phases, ceci garantit que le modèle construit à chaque phase ou étape soit affiné et amélioré. Chaque itération peut servir aussi à ajouter de nouveaux incréments.
- **Conduit par les cas d'utilisation:** la méthode est orientée utilisateur pour répondre aux besoins de celui-ci.
- **Centré sur l'architecture:** les modèles définis tout au long du processus de développement vont contribuer à établir une architecture cohérente et solide.
- **Piloté par les risques:** en définissant des priorités pour chaque fonctionnalité, on peut minimiser les risques d'échec du projet.

La gestion d'un tel processus est organisée d'autour des 4 phases suivantes:

- **La pré-étude:** c'est ici qu'on évalue la valeur ajoutée du développement et la capacité technique à le réaliser (étude de faisabilité).
- **L'élaboration:** sert à confirmer l'adéquation du système aux besoins des utilisateurs et à livrer l'architecture de base.
- **La construction:** sert à livrer progressivement toutes les fonctions du système.
- **La transition:** consiste à déployer le système sur des sites opérationnels [2].

1.2. Le Processus 2TUP

Signifie « **2 Track Unified Process** » . Le processus 2TUP apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise. En ce sens, il renforce le contrôle sur les capacités d'évolution et de correction de tels systèmes.

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



Figure 1.3.6: Le système d'information soumis à deux types de contraintes.

A l'issue des évolutions du modèle fonctionnel et de l'architecture technique, la réalisation du système consiste à fusionner les résultats des deux branches. Cette fusion conduit à l'obtention d'un processus de développement en forme de Y, comme illustré par la figure 1.3.7.

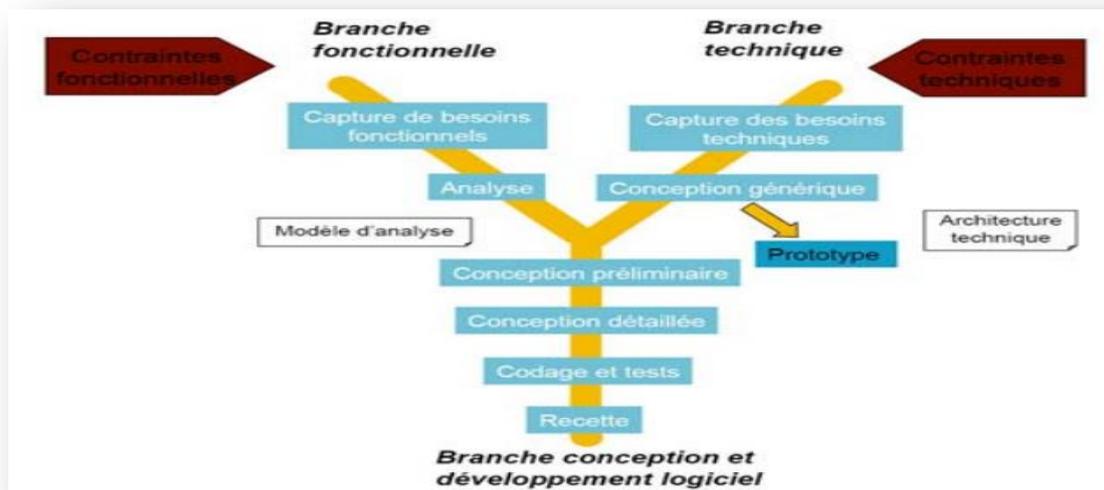


Figure 1.3.7: le processus de développement en Y

✚ **La branche gauche (Architecture fonctionnelle):** capitalise la connaissance du métier de l'entreprise. Elle constitue un investissement pour le moyen et le long terme. Les fonctions du système d'information sont en effet indépendantes des technologies utilisées. Cette branche comporte les étapes suivantes :

- **Capture des besoins fonctionnels:** qui produit le modèle des besoins focalisés 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 termes de métier.

✚ **La branche droite (Architecture technique) :** capitalise un savoir-faire technique. Elle constitue un investissement pour le court et moyen terme. Les techniques développées pour le système peuvent l'être en effet indépendamment des fonctions à réaliser.

Cette branche comporte les étapes suivantes :

- **La capture des besoins techniques:** qui recense toutes les contraintes sur les choix de dimensionnement et la conception du système. Les outils et les 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 aspects fonctionnels. Elle a pour objectif d'uniformiser et de réutiliser les mêmes mécanismes pour tout un système. L'architecture technique construit le squelette du système, son importance est telle qu'il est conseillé de réaliser un prototype.

✚ **Branche conception (milieu):** à l'issue des évolutions du modèle fonctionnel et de l'architecture technique, la réalisation du système consiste à fusionner les résultats de 2 branches. Cette fusion conduit à l'obtention d'un processus en forme de Y.

Cette branche comporte les étapes suivantes:

- **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 ces 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. [1]

Conclusion

Nous nous sommes attelés à étudier à partir de [1] la méthode 2TUP, en particulier ses différents phases. Ce qui va nous permettre dans la partie 2 concernant notre étude de cas de l'utiliser pour l'analyse et la conception de notre application.

Chapitre 04 :

ARCHITECTURE
CLIENT/SERVEUR

Introduction

Dans l'informatique moderne, de nombreuses applications fonctionnent selon un environnement client-serveur; cette dénomination signifie que des machines clientes (faisant partie du réseau) contactent un serveur - une machine généralement très puissante en termes de capacités d'entrées-sorties - qui leur fournit des services. Cette technologie permet d'exploiter au mieux les réseaux, et permet un haut niveau de coopération entre différentes machines sans que l'utilisateur ne se préoccupe des détails de compatibilité. [3]

1. Définition

L'architecture client/serveur est un modèle de fonctionnement logiciel qui se réalise surtout type d'architecture matérielle (petites ou grosses machines), à partir du moment où ces architectures peuvent être interconnectées. On parle de fonctionnement logiciel dans la mesure où cette architecture est basée sur l'utilisation de deux types de logiciels, à savoir un logiciel serveur et un logiciel client s'exécutant normalement sur deux machines différentes. L'élément important dans cette architecture est l'utilisation de mécanismes de communication entre les deux applications.

Le dialogue entre les applications peut se résumer par :

- Le client demande un service au serveur.
- Le serveur réalise ce service et renvoie le résultat au client. [4]

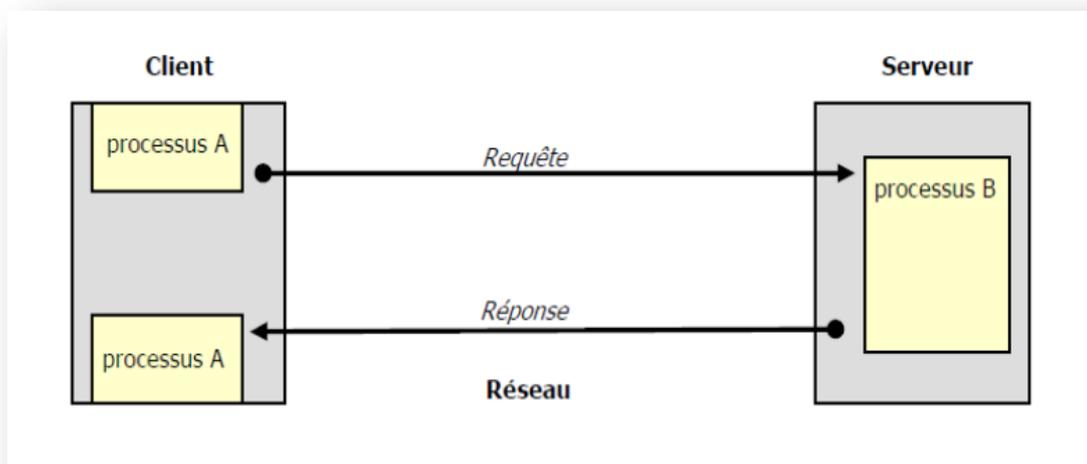


Figure 1.4.8: Modèle client/serveur

Un des principes fondamentaux est que le serveur réalise un traitement pour le client. [4]

2. Caractéristiques des systèmes client serveur

Les éléments qui caractérisent une architecture client serveur sont :

- **Le service**

Le modèle client serveur est s'articule entre des processus qui tournent sur des machines séparées. Le serveur est un fournisseur de services. Le client est un consommateur de services.

- **Le partage de ressources**

Un serveur traite plusieurs clients et contrôle leurs accès aux ressources

- **Un protocole asymétrique**

Conséquence du partage de ressources, le protocole de communication est asymétrique le client déclenche le dialogue, le serveur attend les requêtes des clients.

- **La transparence de la localisation**

L'architecture client serveur doit masquer au client la localisation du serveur (que le service soit sur la même machine ou accessible par le réseau),ce qui assure la transparence par rapport aux systèmes d'exploitation et aux plates-formes matérielles. Idéalement, le logiciel client/serveur doit être indépendant de ces deux éléments.

- **Le message**

Les messages sont les moyens d'échanges entre client et serveur.

- **L'encapsulation des services**

Un client demande un service. Le serveur décide de la façon de le rendre. Une mise à niveau du logiciel serveur doit être sans conséquence pour le client tant que l'interface message est identique.

- **L'évolution**

Une architecture client/serveur doit pouvoir évoluer horizontalement (évolution du nombre de clients) et verticalement (évolution du nombre et des caractéristiques des serveurs). [3]

- **La souplesse et l'adaptabilité**

On peut modifier le module serveur sans toucher au module client. La réciproque est vraie. Si une station est remplacée par un modèle plus récent, on modifie le module client (en améliorant l'interface, par exemple) sans modifier le module serveur. [4]

3. Concepts de base d'une architecture Client/serveur

3.1. Définition d'un serveur

On appelle un logiciel serveur un programme qui offre un service sur le réseau, un serveur accepte une requête, la traite et renvoie le résultat au demandeur. Le terme serveur s'applique à la machine sur laquelle s'exécute le logiciel serveur. Pour pouvoir offrir ces services en permanence, le serveur doit être sur un site avec accès permanent et s'exécute en permanence. [5]

3.2. Définition d'un client

On appelle logiciel client un programme qui utilise le service offert par un serveur. Le client envoie une requête et reçoit la réponse. Le client peut être raccordé par une liaison temporaire. [5]

3.3. Définition d'un middleware

Un Middleware est un logiciel de communication qui permet à plusieurs processus s'exécutant sur une ou plusieurs machines d'interagir à travers un réseau. [6]

Ce terme désigne globalement tous les logiciels distribués nécessaires à l'interaction entre client et serveur. Nous appelons middleware la barre de fraction (/) du terme client serveur. C'est le ciment qui permet d'obtenir un service auprès d'un serveur. [7]

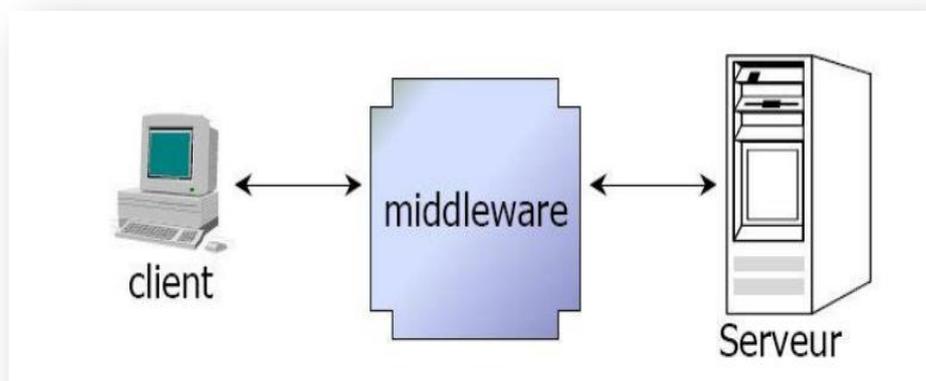


Figure 1.4.9: Middleware

Un middleware est susceptible de rendre les services suivants :

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

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

On distingue aujourd'hui quatre grandes classes de Middleware. [4]

- L'exécution de transaction (Transactions Processing) qui est une classe de logiciel plutôt orienté «base de données».
- Les RPC (Remote Procedure Calls) qui distribue l'exécution de routines sur un réseau.
- Les MOM (Message Oriented Middleware) qui permettent l'échange de données entre applications.
- Les ORB (Object Request Broker) qui permettent la distribution d'objets sur un réseau de machines. [6]

4. Types des architectures Client/serveur

4.1. Architecture à 2 niveaux

L'architecture à deux niveaux (aussi appelée architecture 2-tiers, (tiers signifiant rangé en anglais) 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.

4.2. Architecture à 3 niveaux

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

- Le client: le demandeur de ressources.
- Le serveur d'Application (appelé aussi middleware): le serveur chargé de fournir la ressource mais faisant appel à un autre serveur.
- Le serveur secondaire (généralement un serveur de base de données), fournissant un service au premier serveur.

4.3. L'architecture multi niveaux

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

serveurs afin de fournir son propre service. Par conséquent, l'architecture à trois niveaux est potentiellement une architecture à N niveaux. [8]

5. Fonctionnement d'un réseau client/serveur

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

On parle de fonctionnement logiciel dans la mesure où cette architecture est basée sur l'utilisation de deux types de logiciels, à savoir un logiciel serveur et un logiciel client s'exécutant normalement sur 2 machines différentes. L'élément important dans cette architecture est l'utilisation de mécanismes de communication entre les 2 applications (voir figure 1.4.10). [7]

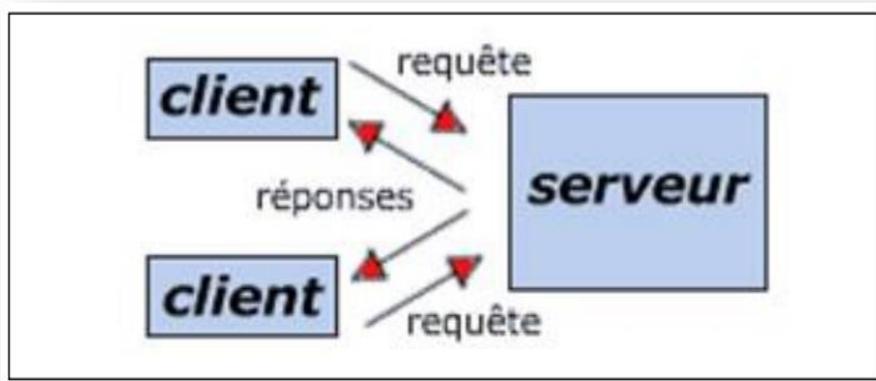


Figure 1.4.10: Architecture client/serveur

6. Avantages de l'architecture client/serveur

- Toutes les données sont centralisées sur un seul serveur, ce qui simplifie les contrôles de sécurité, l'administration, la mise à jour des données et des logiciels.
- La complexité du traitement et la puissance de calculs sont à la charge du ou des serveurs, les utilisateurs utilisant simplement un client léger sur un ordinateur terminal qui peut être simplifié au maximum.
- Recherche d'information : les serveurs étant centralisés, cette architecture est particulièrement adaptée et vélocité pour retrouver et comparer de vastes quantités d'informations (moteur de recherche sur le Web). [7]

7. Inconvénients de l'architecture client/serveur par rapport à une architecture pair à pair

- Si trop de clients veulent communiquer avec le serveur au même moment, ce dernier risque de ne pas supporter la charge (alors que les réseaux pair-à-pair fonctionnent mieux en ajoutant de nouveaux participants).
- Si le serveur n'est plus disponible, plus aucun des clients ne fonctionne (le réseau pair-à-pair continue à fonctionner, même si plusieurs participants quittent le réseau).
- Les coûts de mise en place et de maintenance peuvent être élevés.
- En aucun cas, les clients ne peuvent communiquer entre eux, entraînant une asymétrie de l'information au profit des serveurs. [7]

Conclusion

Le modèle client/serveur est la base de tous les services réseaux informatiques. Dans ce chapitre, nous avons présenté cette architecture puisque notre application a besoin d'un déploiement sur plusieurs postes.

Partie

ETUDE

DE

CAS

2

Chapitre 01 :

ETUDE PRELIMINAIRE

Introduction

L'étude préliminaire est la toute première étape de notre processus de développement. Il est important de démarrer l'analyse par le positionnement le plus précis possible du système à étudier. Ainsi, il est opportun de recueillir les besoins des utilisateurs et de situer le contexte du système.

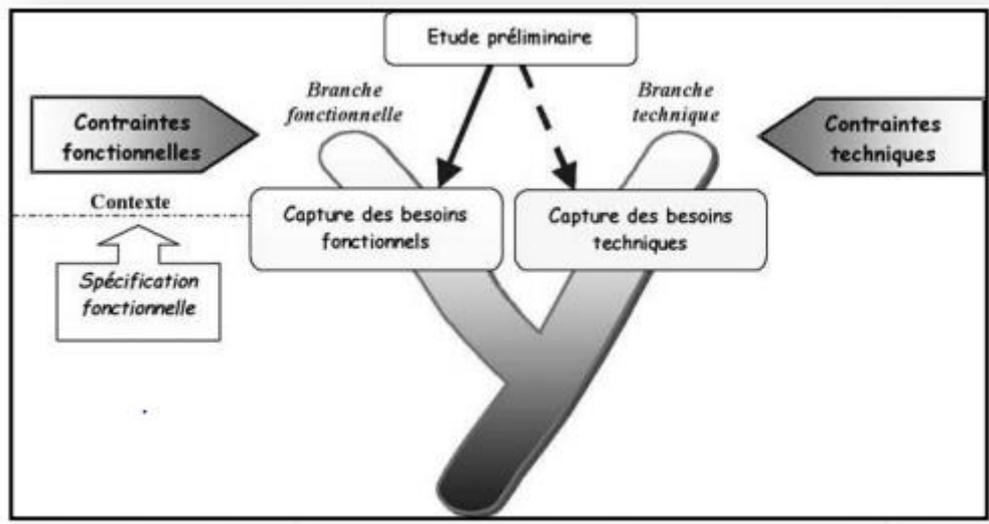


Figure 2.1.11: Situation de l'étude préliminaire dans 2TUP

Dans ce chapitre, il s'agira de décrire les besoins et les acteurs qui vont interagir avec notre système. Ces besoins vont des fonctionnalités jusqu'aux aspects techniques du système.

1. Cahier des charges

1.1.Présentation de projet

Le but de notre mémoire est de concevoir et réaliser une application client/serveur pour l'étude de la traçabilité des dossiers à travers les services de la Caisse Nationale de la Retraite(CNR), au cours du processus de mise en retraite d'un employé assuré, Ce processus va de la réception du dossier jusqu'à son mandatement. A cette fin, nous allons adopter la démarche de développement 2TUP [] qui utilise le langage UML comme langage de modélisation.

1.2. Les grands choix techniques

On souhaite utiliser une approche itérative et incrémentale, fondée sur le processus en Y (méthode 2TUP). Après une première étude, nous avons choisi un certain nombre de techniques clés pour ce projet:

- La modélisation avec UML2.
- L'adoption d'une architecture client/serveur.
- L'utilisation du langage Java pour la programmation de la logique applicative de ce projet sur une plateforme NetBeans.
- Oracle pour le stockage et la gestion des données.

1.3. Recueil des besoins fonctionnels

Nous avons effectué plusieurs interview pour identifier au mieux les besoins de l'application, et ceci afin de répondre aux attentes des travailleurs de la CNR.

Nous sommes allés chercher les informations au niveau de sous-direction des pensions et l'exploitation informatique de l'organisation. Cette phase correspond à une recherche sur le terrain pour bien définir le cadre de notre système.

A partir de ces informations que nous sommes procurées, nous pouvons établir le cahier des charges préliminaire suivant:

1.3.1. La codification des dossiers

Elle s'effectue en deux étapes:

Premièrement elle permet d'attribuer un code à barre à chaque dossier.

Les codes barres offrent une méthode rapide, facile et précise pour saisir des données. Une utilisation judicieuse des codes barres peut réduire les délais nécessaires aux employés pour effectuer certaines tâches.

Il existe de nombreux types de codes barres, le Code 128 représente la meilleure alternative pour notre système puisqu'il possède une structure alphanumérique extrêmement efficace permettant de produire des codes barres très denses et il accepte les 128 caractères ASCII.

Format de code barre utilisé

- Le premier caractère signifie le type de dossier où «D» pour dossier direct et «R» pour dossier de réversion.
- Les deux caractères suivants signifient l'année de dépôt du dossier.
- Les cinq caractères suivants signifient un numéro séquentiel.



Figure 2.1.12 : Format de code à barre pour un dossier de retraite direct.

Ensuite, elle permet de saisir les autres informations inhérentes aux dossiers à savoir: le nom de retraité, le prénom de retraité, la date de naissance de retraité, le numéro pré liquide (N°PL), le type de dossier, la date dépôt, etc.

Il est à noter que ces informations sont utilisées par une autre application (STAR) existante au niveau de la CNR. Mais, elles sont également indispensables pour les besoins de notre application.

1.3.2. La Saisie alpha

Elle consiste à attribuer le numéro liquidé au dossier (NL) qui constitue de 9 caractères où les 3 premiers chiffres sont variables et les 6 derniers chiffres sont séquentiels.

Chaque valeur de 3 premiers chiffres a une signification normalisée:

- La première valeur signifie le **Régime d'affiliation**.

Régime d'affiliation	
Reg	Intitulé
A	Reg.agricole
B	Reg.general
C	Reg.complement
E	Reg.etranger
F	Reg.fonction
G	Enfant.de.chahid
H	Reg.agr.non.sal
M	Reg.minier
P	Reg.retr.antic
R	Reg.cheminot
S	Reg.sonelgas
T	Reg.traminant
V	Reg.cavnos

Y	Reg.tuteur
Z	Reg.marine

Tableau 2.1. 2: Les caractères du champ régime d'affiliation et leur signification.

- La deuxième valeur signifie le **Régime de liquidation**.

Régime de liquidation	
Reg	Intitulé
1	normal
2	Pens ret/inv.n
3	Pens ret i/moud
6	Reg compl.s/bas
A	Retr.anticipee
C	Pens moudjahed
G	Enfant de chahi
I	Invalide cnr.coordinationcas
L	Liq coordinat
M	Moudj.coordinat
N	Coord.casnos
O	coexistence
P	Casnos moudj
S	Casnos.entrang
T	Casnos.etr.moud
W	Taux cristalise
Z	Liq.provisoire

Tableau 2.1.3: Les caractères du champ régime de liquidation et leur signification.

- La troisième valeur signifie l'**Avantage**.

Avantage	
Directe	
Av	Intitulé
1	Pension directe
4	A-V-T-S
5	Secours viager
7	Allocation directe
9	Retraite complémentaire
G	Retraite invalidité pension direct
W	Retraite proportionnelle
Z	Retraite A 32 ans

Tableau 2.1.4: Les caractères du champ avantage des dossiers directes et leur signification.

Avantage	
Réversion	
Av	intitulé
2	Pension veuve/veuf
3	Pension veuf (ve)
8	Transfert en pivv
D	Allocation enfant handicapé
E	Réversion allocation pens.inv
F	Allocation veuve/veuf
H	Pension handicapé
I	Pension jeune fille majeure
J	Rente invalidité veuf (ves)
K	Allocation enfant mineur
L	Pension enfant mineur
M	Pension ascendant mère
O	Pension j.femme collatérale m
P	Pension ascendant père
Q	Rente invalidité père
R	Allocation ascendant père
S	Allocation ascendant mère
T	Pension rev.enf.mineur sous
U	Rente invalidité mère
V	Allocation jeune femme majeure
X	Rente invalidité enfant mineur
Y	Rente invalidité handicapé

Tableau 2.1.5: Les caractères du champ avantage des dossiers de réversion et leur signification.

1.3.3. L'envoi des dossiers

Il consiste à envoyer un fichier contenant la liste des dossiers traités par un employé vers un autre service ou autre employé en amont de l'envoi physique des dossiers.

1.3.4. La réception des dossiers

Elle consiste à consulter les fichiers reçus et enregistrer la date de réception des dossiers.

1.3.5. La localisation des dossiers

Elle se fait selon plusieurs critères (nom prénom et date de naissance de retraite, numéro pré liquide (N°PL), numéro liquide(NL)).

1.3.6. La trace des dossiers

La trace des dossiers peut être effectuée selon plusieurs critères (nom et prénom, numéro pré liquide (N°PL), numéro liquide(NL)) pour connaître l'historique d'un dossier.

1.3.7. Changer état dossier

Elle consiste à changer l'état du dossier.

1.3.8. L'établissement des statistiques.

Les statistiques constituent un besoin urgent et prépondérant pour l'exploitant du service informatique. En effet, le retard accusé fréquemment dans le traitement de certains dossiers accompagné des réclamations des retraités, impose de diagnostiquer les causes provoquant le rallongement des délais de traitement, de localiser les services qui en sont responsables et de trouver les solutions adéquates. A cette fin, plusieurs statistiques seraient d'une grande utilité. En effet, l'exploitant peut avoir besoin de calculer et d'éditer instantanément l'état global des dossiers à savoir :

- Le nombre de dossiers en instance CNAS ou CASNOS (les dossiers en attente de la réponse de la CNAS ou CASNOS).
- Le nombre de dossiers liquidés.
- Le nombre de dossiers en instance d'instruction.
- Le nombre de dossiers en instance de liquidation.
- Le nombre de dossiers classés.
- Le nombre de dossiers rejetés.

En outre, ces calculs peuvent être effectués selon deux critères: par utilisateur ou par service:

- Le délai moyen de traitement des dossiers.
- Le nombre de dossiers traités dans un délai donné.

1.4. Recueil des besoins techniques

➤ **La sécurité**

La sécurité des données est mise en exergue par un certain nombre de besoins techniques:

✓ **La gestion des utilisateurs**

- L'ajout d'un compte: consiste à ajouter un nouveau compte avec des informations concernant l'employé (numéro utilisateur, nom et prénom d'utilisateur, mot de passe, le poste, le rôle, username, le service).
- La modification des comptes: elle permet de modifier les informations d'un utilisateur.
- La suppression d'un compte: elle permet de supprimer un compte d'employé.

✓ **Authentification**

Chaque utilisateur se connecte au système par son rôle, son username et son mot de passe.

✓ **Gérer l'intégrité des données**

Une contrainte d'intégrité est une clause permettant de contraindre la modification de tables, faite par l'intermédiaire de requêtes d'utilisateurs, afin que les données saisies dans la base soient conformes aux données attendues. Ces contraintes sont:

- Définir un attribut par défaut.
- Forcer la saisie obligatoire des champs.
- Emettre des conditions sur les champs.
- Etc.

2. Identification des acteurs

Un acteur représente l'abstraction d'un rôle joué par des entités externes qui interagissent directement avec le système étudié. Il représente un rôle joué par une entité externe au système, donc peut consulter ou modifier directement l'état du système. Il émet ou reçoit des messages qui sont porteurs de données.

Nous avons identifié les acteurs suivants qui interagissent avec notre système:

2.1.L'administrateur

L'administrateur est l'exploitant dans le service informatique de la CNR. Il effectue la gestion des comptes des utilisateurs, établit au besoin des statistiques, effectue la trace des dossiers et aussi la localisation des dossiers.

2.2.L'agent de saisie

L'agent de saisie effectue la codification, l'envoi et la réception des dossiers, la saisie alpha.

2.3.L'employé

C'est un travailleur qui travaille de la CNR. Il effectue depuis le service auquel il est affecté l'envoi, la réception, la localisation des dossiers de retraites.

Le tableau suivant résume l'utilisation de notre application par les différents utilisateurs.

	Service	Utilisateurs
Sous direction des pensions	Service technique	- L'agent de saisie. - Le chef de service technique.

		- L'archiviste - L'agent de suivi et contrôle
	Service liquidation	- L'agent de divisions des pensions. - Les liquidateurs. - L'agent de contrôle.
	Service vérification	- Les vérificateurs.
		- Sous directeur des pensions
Sous direction de finances		- Le sous directeur de finances
Direction générale		- Directeur
	Service informatique	- L'exploitant informatique

Tableau 2.1.6: Nomenclature des utilisateurs dans chaque service.

3. Identification des messages

Nous détaillerons, ici, les différents messages échangés entre le système et les acteurs précédents.

- ✓ Le système émet les messages suivants :
 - La confirmation d'ajout d'un compte.
 - La confirmation de modification d'un compte.
 - La confirmation de suppression d'un compte.
 - La confirmation de codification un dossier.
 - La confirmation d'envoi d'un dossier.
 - La confirmation de réception d'un dossier.
 - La confirmation de saisie alpha.
 - La confirmation de changement de l'état de dossier.
 - L'affichage des informations de dossier.
 - La trace d'un dossier.
 - Le résultat des statistiques.
 - Le fichier de liste des dossiers envoyés.
- ✓ Le système reçoit les messages de demande suivants :
 - L'ajout d'un compte.
 - La modification d'un compte.
 - La suppression d'un compte.
 - L'envoi d'un message.

- La réception des dossiers.
- La codification un dossier.
- La consultation de la liste des dossiers.
- Le calcul des statistiques.
- La consultation de la trace des dossiers.
- La saisie alpha.
- Le Changement de l'état de dossier.

4. Modélisation de contexte

A partir des informations obtenues dans les étapes précédentes, nous allons modéliser le contexte de notre application. Ceci va nous permettre dans un premier temps, de définir le rôle de chaque acteur dans le système.

Utilisateurs finaux	Description des besoins fonctionnels
Administrateur	L'application doit permettre à l'administrateur de : <ul style="list-style-type: none"> - Gérer comptes. - Effectuer les différentes statistiques. - Localiser dossiers. - Tracer dossier. - Se faire authentifier.
Employé	L'application doit permettre à l'employé de : <ul style="list-style-type: none"> - Envoyer dossiers. - Réceptionner dossier. - Localiser dossiers. - Changer état. - Se faire authentifier.
Agent de saisie	L'application doit permettre à l'agent de saisie de: <ul style="list-style-type: none"> - Codifier dossiers et saisir les informations des retraites. - Saisie alpha. - Envoyer dossier. - Réceptionner dossier. - Changer état. - Se faire authentifier.

Tableau 2.1.7 : Modélisation des contextes.

4.1. Diagramme de contexte dynamique.

Le diagramme de contexte dynamique représente les messages échangés entre l'application et les acteurs identifiés (représentés de façon synthétique sur un diagramme).

Chaque numéro, dans le diagramme, correspond à un message dans la table de messages représentée par le tableau (2.1.13).

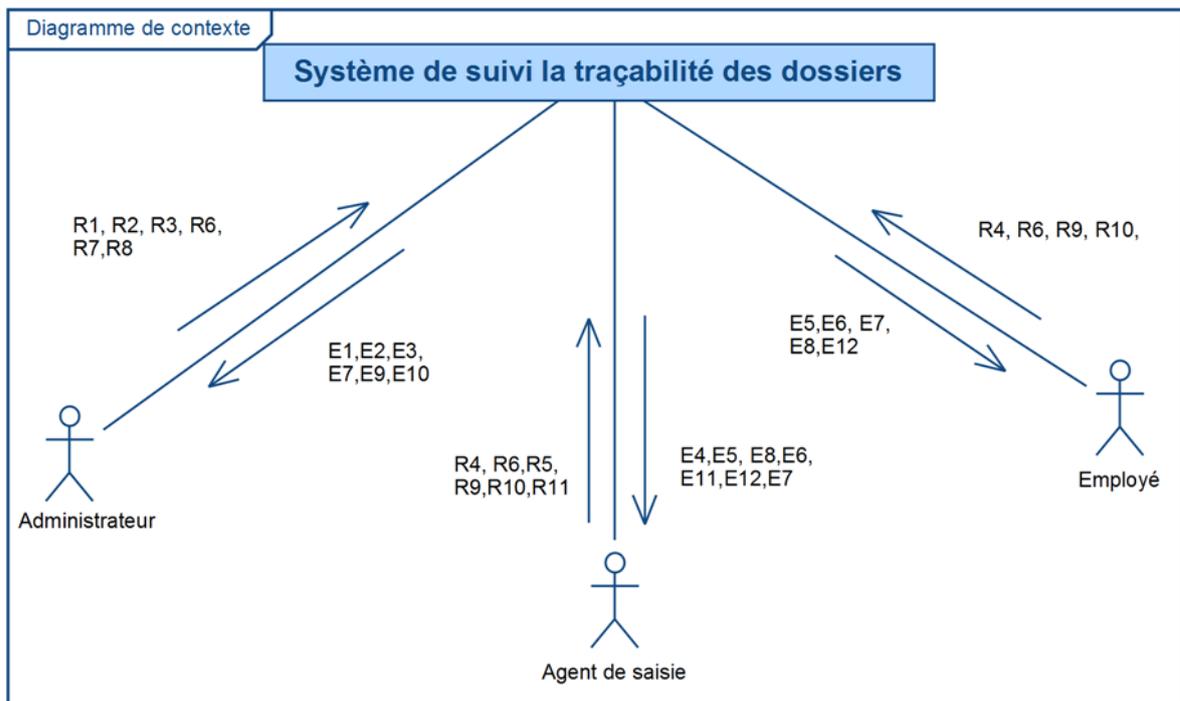


Figure 2.1.13 : Diagramme de contexte dynamique

4.2. Signification des messages

Message émis	Message reçus
E1: La confirmation d'ajout d'un compte	R1: L'Ajout d'un compte.
E2: La confirmation de modification d'un compte.	R2: La modification d'un compte.
E3: La confirmation de suppression d'un compte.	R3: La suppression d'un compte.
E4: La confirmation de codification un dossier.	R4: L'envoi d'un dossier.
E5: La confirmation d'envoi d'un dossier.	R5: La codification un dossier.
E6: La confirmation de réception d'un dossier.	R6: La consultation de la liste des dossiers.
	R7: Le calcul des statistiques.
	R8: La consultation de la trace des dossiers.
	R9: Le changement d'état de dossier.
	R10: La réception des dossiers.

E7: Affichage des informations de dossier. E8:Le fichier de liste des dossiers envoyés. E9: Résultat de statistiques. E10: La trace des dossiers. E11: La confirmation de saisie alpha. E12: La confirmation de changement d'état	R11: La saisie alpha.
--	-----------------------

Tableau 2.1.8: Signification des messages.

Conclusion

Après avoir dégagé les besoins fonctionnels et opérationnels et tous les critères qu'on doit prendre en considération, dans le prochain chapitre nous allons poursuivre la formalisation de ces besoins.

Chapitre 02 :

CAPTURE
DES
BESOINS FONCTIONNELS

Introduction

Cette phase est la première étape de la branche gauche du cycle en Y. Elle représente un point de vue « fonctionnel » de l'architecture système. Pour ce faire nous utiliserons la notion de cas d'utilisation. Chaque cas d'utilisation sera identifié, décrit, organisé, et classé en fonction de son importance dans le projet.

1. Déterminer les cas d'utilisations

1.1. Liste préliminaire des cas d'utilisation

En considérant l'intention fonctionnelle de l'acteur par rapport au système dans le cadre de l'émission ou de la réception de chaque message, Chaque intention fonctionnelle est modélisée par un cas d'utilisation.

Cas d'utilisation		Les acteurs	Les messages
Codifier dossiers		Agent de saisie	Emet : codifier dossiers. Reçoit : confirmation de codification.
Envoyer dossiers		Employé Agent de saisie	Emet : envoyer dossiers Reçoit : confirmation d'envoi des dossiers.
Réceptionner dossiers		Employé Agent de saisie	Emet : réceptionner dossiers. Reçoit : la liste des dossiers réceptionnés.
Saisir alpha		Agent de saisie	Emet : saisir numéro liquidé. Reçoit : la confirmation de saisie alpha.
Changer état dossier		Employé Agent de saisie	Emet : changer état de dossier. Reçoit : la confirmation de changement d'état.
Tracer dossier	Par numéro pré liquidé (N°PL)	Administrateur	Emet : numéro pré liquidé (N°PL). Reçoit : la trace du dossier (service, utilisateur, état, etc.).
	Par numéro liquidé (NL)		Emet : numéro liquidé (NL). Reçoit : la trace du dossier (service, utilisateur, état, etc.).
	Par nom et prénom et date naissance		Emet : nom et prénom et date naissance. Reçoit : la trace du dossier (service, utilisateur, état, etc.).
Localiser dossier	Par numéro pré liquidé (N°PL)	Administrateur Employé	Emet : numéro pré liquidé (N°PL). Reçoit : la localisation du

			dossier (service, utilisateur, état, etc.).
	Par numéro liquidé (NL)		Emet : numéro liquidé (NL). Reçoit : la localisation du dossier (service, utilisateur, état, etc.).
	Par nom et prénom et date naissance		Emet : nom et prénom et date naissance. Reçoit : la localisation du dossier (service, utilisateur, état, ... etc.).
Etablir Statistiques	Par utilisateur	Administrateur	Emet: nom utilisateur. Date_début, date_fin. Reçoit: détails des statistiques-dossiers par utilisateur.
	Par service		Emet : nom service, date_début, date_fin. Reçoit : détails des statistiques dossiers par service.

Tableau 2.2.9 : Les cas d'utilisations.

1.2. Diagramme de cas d'utilisation

Maintenant que nous avons identifié les cas d'utilisation et leurs acteurs, nous allons les représenter graphiquement sur un diagramme de cas d'utilisation qui représente le fonctionnement du système vis-à-vis de l'utilisateur :

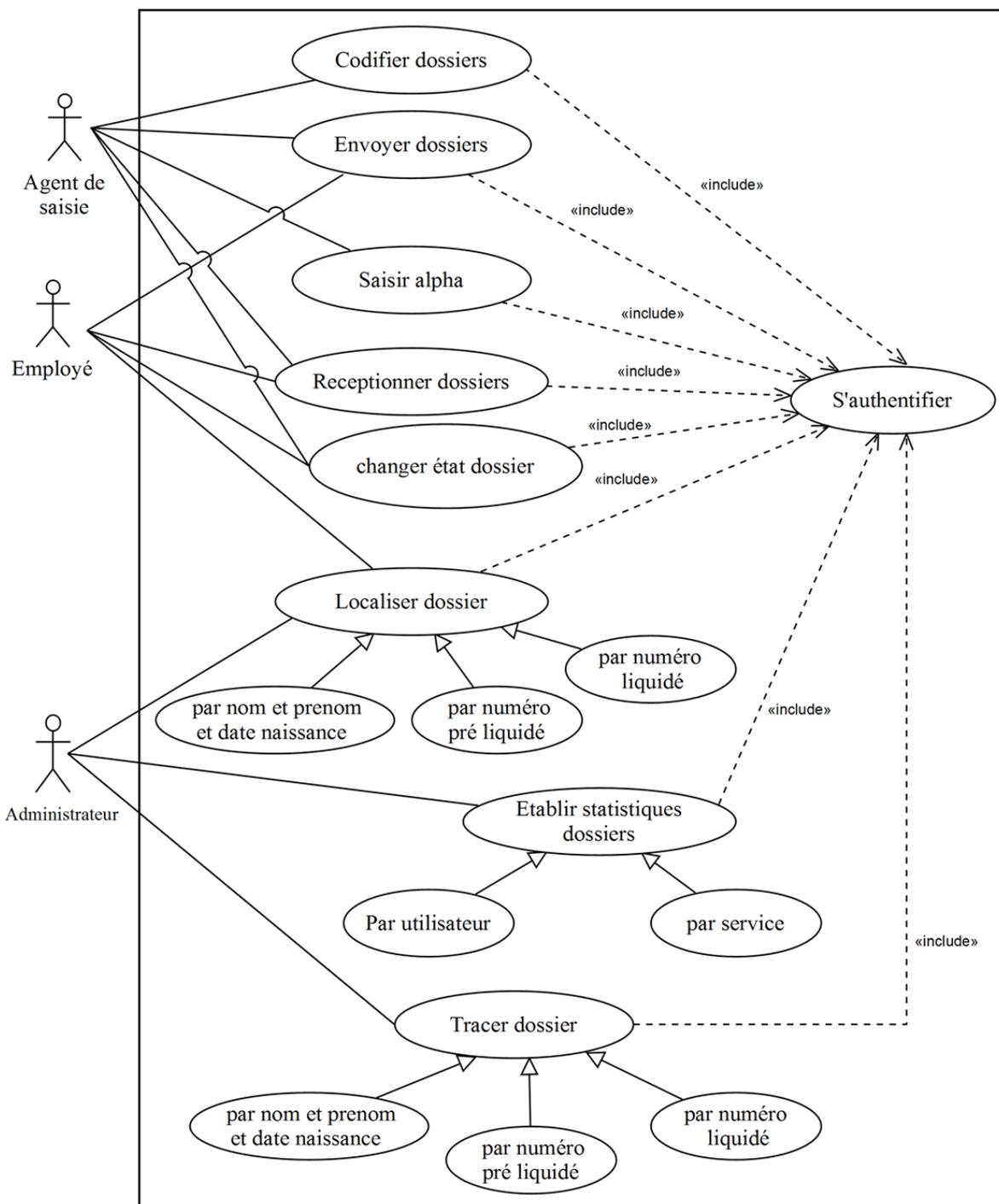


Figure 2.2.14: Diagramme de cas d'utilisation

2. Description détaillée des cas d'utilisations

Nous allons maintenant décrire chaque cas d'utilisation par un scénario nominal, un diagramme d'activité et un diagramme de séquence modélisant la séquence d'actions principales que le système est susceptible d'effectuer.

2.1. Codifier dossier

<p>Titre : Codifier dossier.</p> <p>Acteur : Agent de saisie.</p> <p>But : Codifier un dossier.</p> <p>Pré condition :</p> <ul style="list-style-type: none">- L'agent de saisie s'authentifie.- Au moins un dossier pour l'ajout. <p>Post conditions : La codification est effectuée.</p> <p>Scénario nominal :</p> <ol style="list-style-type: none">1- Le système affiche le formulaire «Codification».2- Le système lit les informations du dossier (code barre, nom retraité, prénom retraité, date de naissance, type de dossier, date dépôt).3- Le système enregistre les informations.4- Le système affiche une notification «La codification est avec succès».5- Le système affiche une notification «Voulez vous ajouter un autre dossier?». si confirmation, reprendre au point 2.
--

Tableau 2.2.10 : Fiche descriptive textuelle du cas «Codifier dossiers».

❖ Diagramme d'activité «Codifier dossier»

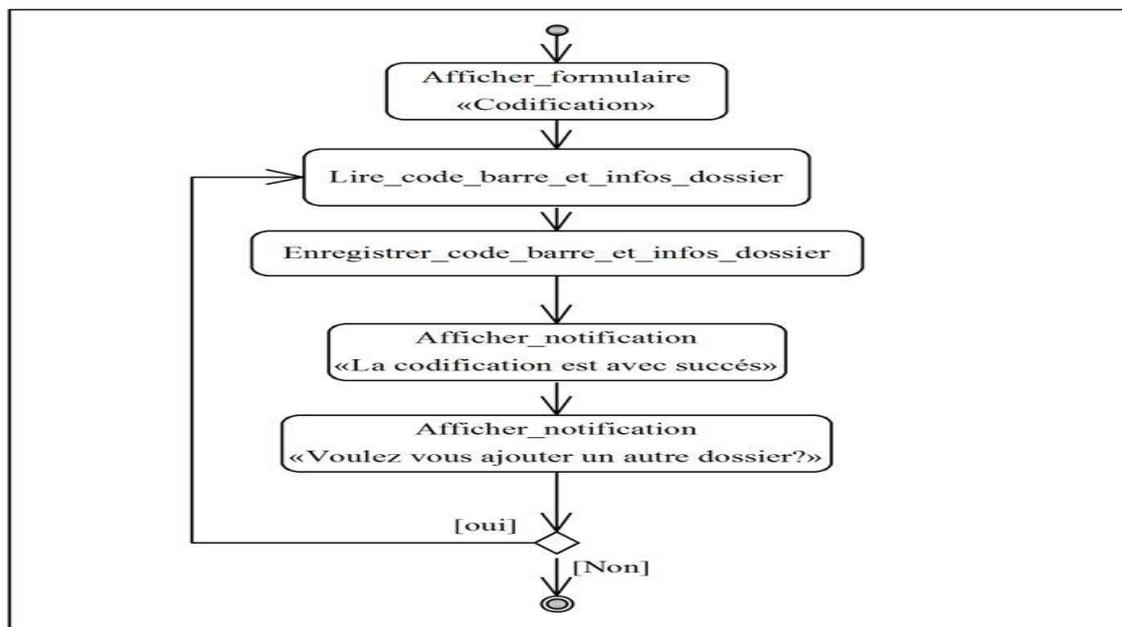


Figure 2.2.15: Diagramme d'activité de cas « Codifier dossier ».

❖ Diagramme de séquence «Codifier dossier »

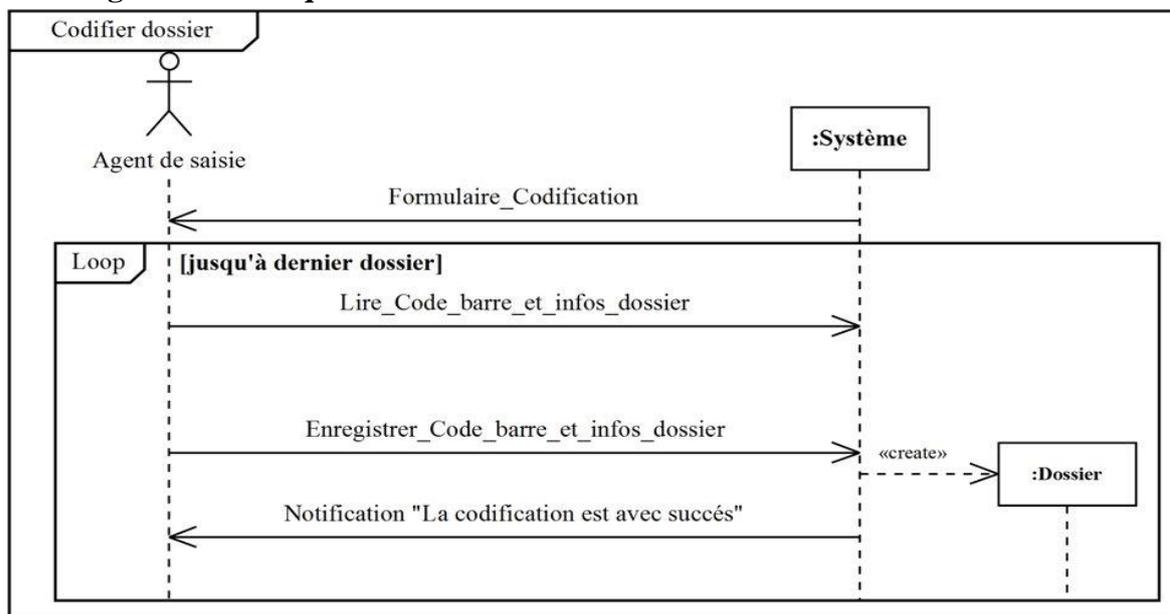


Figure 2.2.16: Diagramme de séquence de cas « Codifier dossier ».

2.2. Envoyer dossiers

<p>Titre : Envoyer dossiers.</p> <p>Acteur : Employé, Agent de saisie.</p> <p>But : Envoyer des dossiers.</p> <p>Pré condition :</p> <ul style="list-style-type: none">- L'employé s'authentifie.- L'agent de saisie s'authentifie.- IL existe au moins un dossier à envoyer. <p>Post conditions : L'envoi est effectué.</p> <p>Scénario nominal :</p> <ol style="list-style-type: none">1- Le système affiche le formulaire « Envoyer dossier ».2- Le système lit les codes-barres des dossiers envoyés et l'identité de récepteur.3- Le système enregistre les codes-barres, l'identité de l'émetteur, la date d'envoi, l'heure d'envoi, l'identité de récepteur.4- Le système génère la liste des dossiers envoyés dans un fichier.5- Le système affiche une notification «Le fichier est généré».6- Le système attache le fichier envoyé au récepteur.7- Le système envoie le fichier attaché.8- Le système affiche une notification «Le fichier est transmis».9- Le système imprime le bordereau des dossiers envoyés.

Tableau 2.2.11: Fiche descriptive textuelle du cas «Envoyer dossiers».

❖ Diagramme d'activité «Envoyer dossiers »

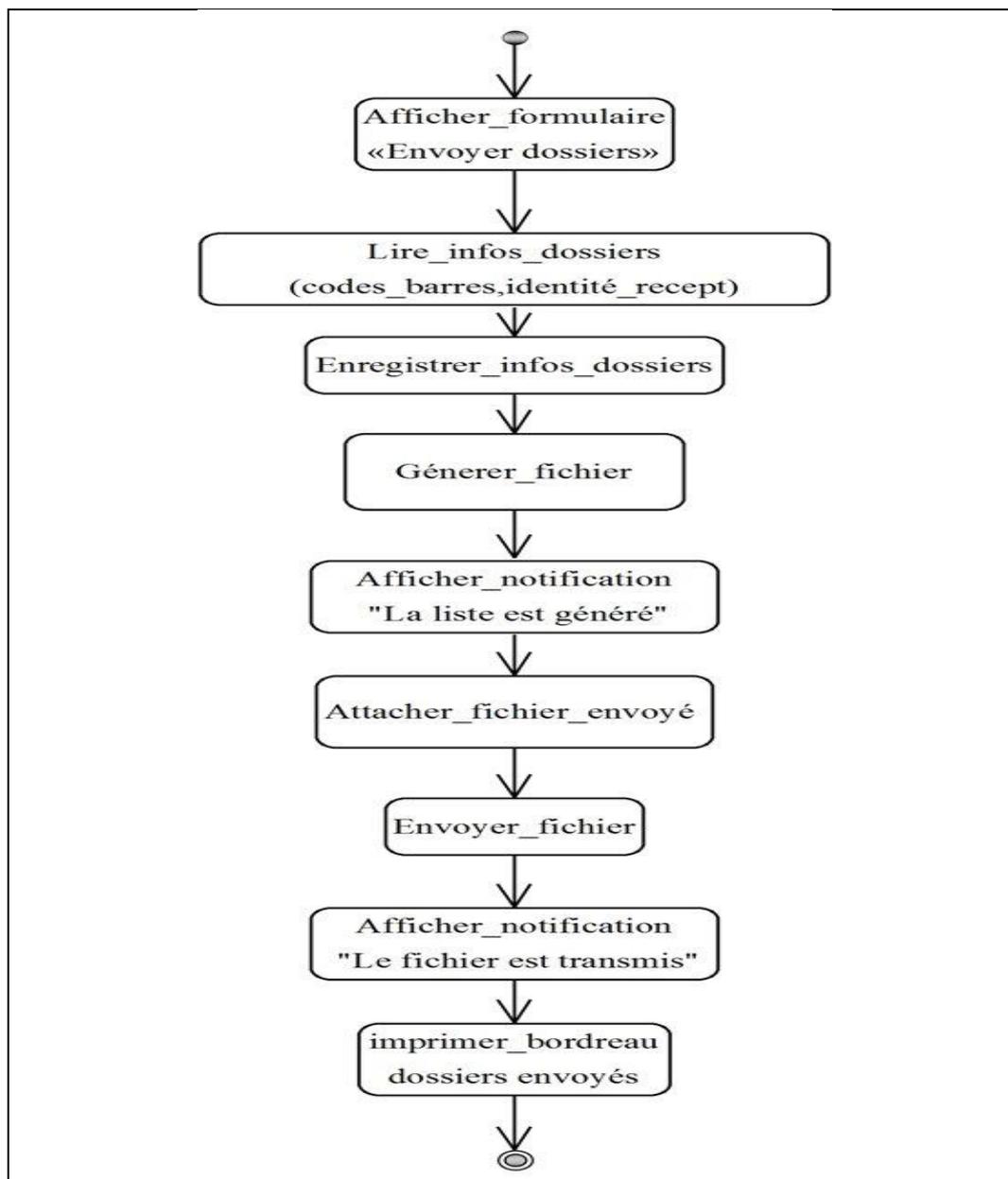


Figure 2.2.17: Diagramme d'activité de cas « Envoyer dossier ».

❖ Diagramme de séquence «Envoyer dossier »

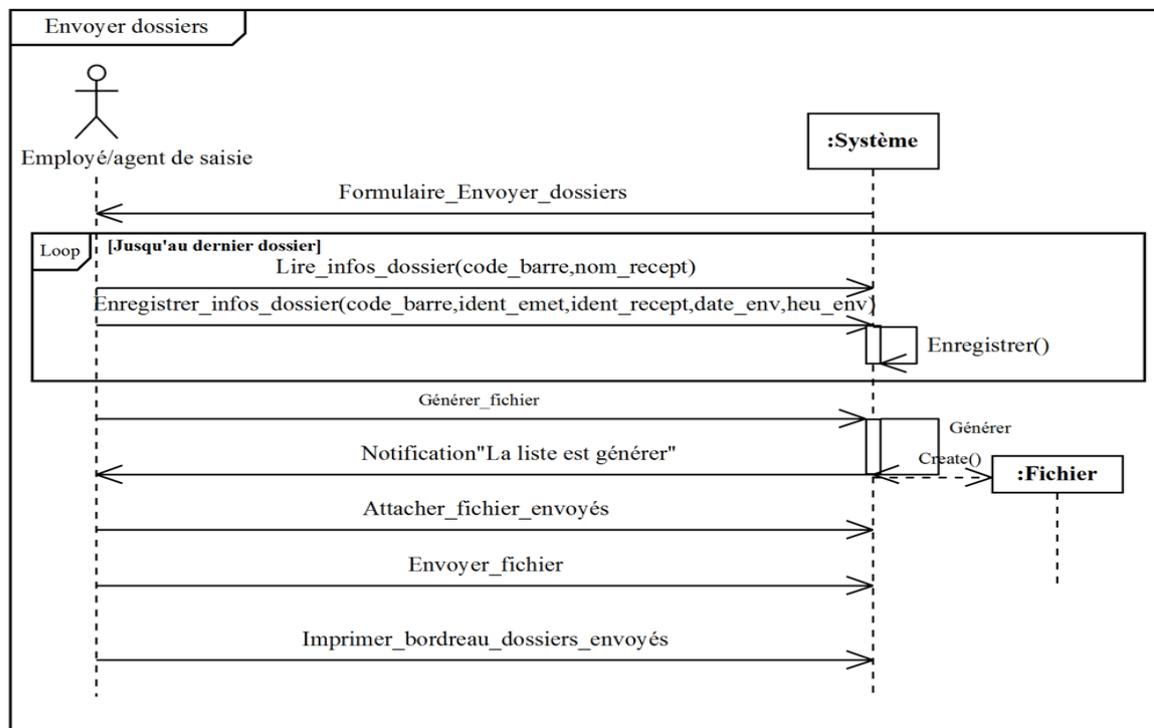


Figure 2.2.18: Diagramme de séquence de cas « Envoyer dossier ».

2.3.Réceptionner dossiers

<p>Titre : Réceptionner dossiers</p> <p>Acteur : Employé, Agent de saisie.</p> <p>But : Réceptionner dossiers.</p> <p>Pré condition :</p> <ul style="list-style-type: none"> - L'employé s'authentifie. - L'agent de saisie s'authentifie. - IL existe au moins un dossier à réceptionner. <p>Post conditions : La réception est effectuée.</p> <p>Scénario nominal :</p> <ol style="list-style-type: none"> 1- Le système affiche le formulaire « Réceptionner dossiers ». 2- Le système affiche le contenu de la boîte de réception. 3- Le système ouvre le fichier contenant la liste des dossiers envoyés. 4- Le système télécharge le fichier. 5- Le système lit les informations des dossiers reçus (code barre). 6- Le système enregistre la réception des dossiers.
--

Tableau 2.2.12: Fiche descriptive textuelle du cas « Réceptionner dossiers».

❖ Diagramme d'activité «Réceptionner dossiers »

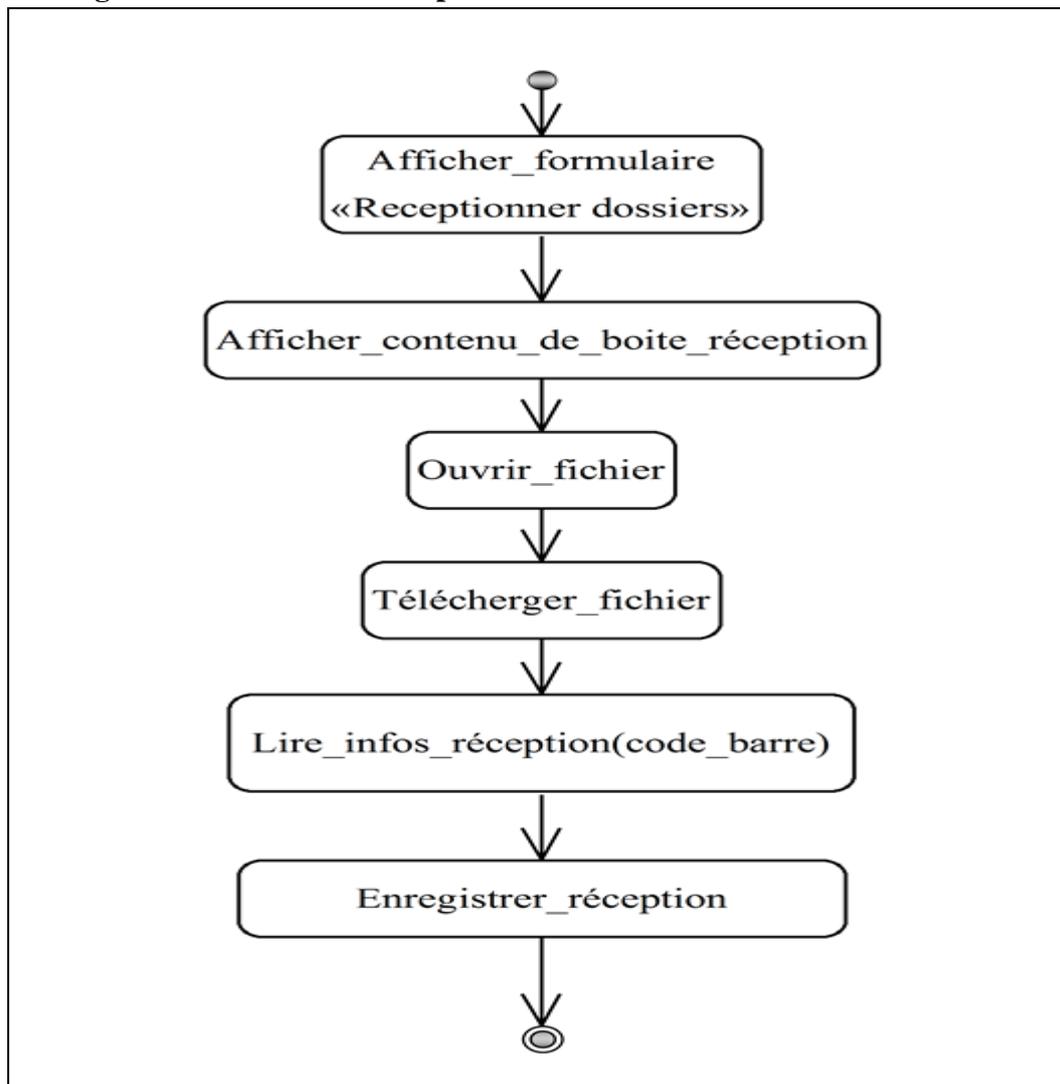


Figure 2.2.19: Diagramme d'activité de cas « Réceptionner dossier ».

❖ Diagramme de séquence «Réceptionner dossier »

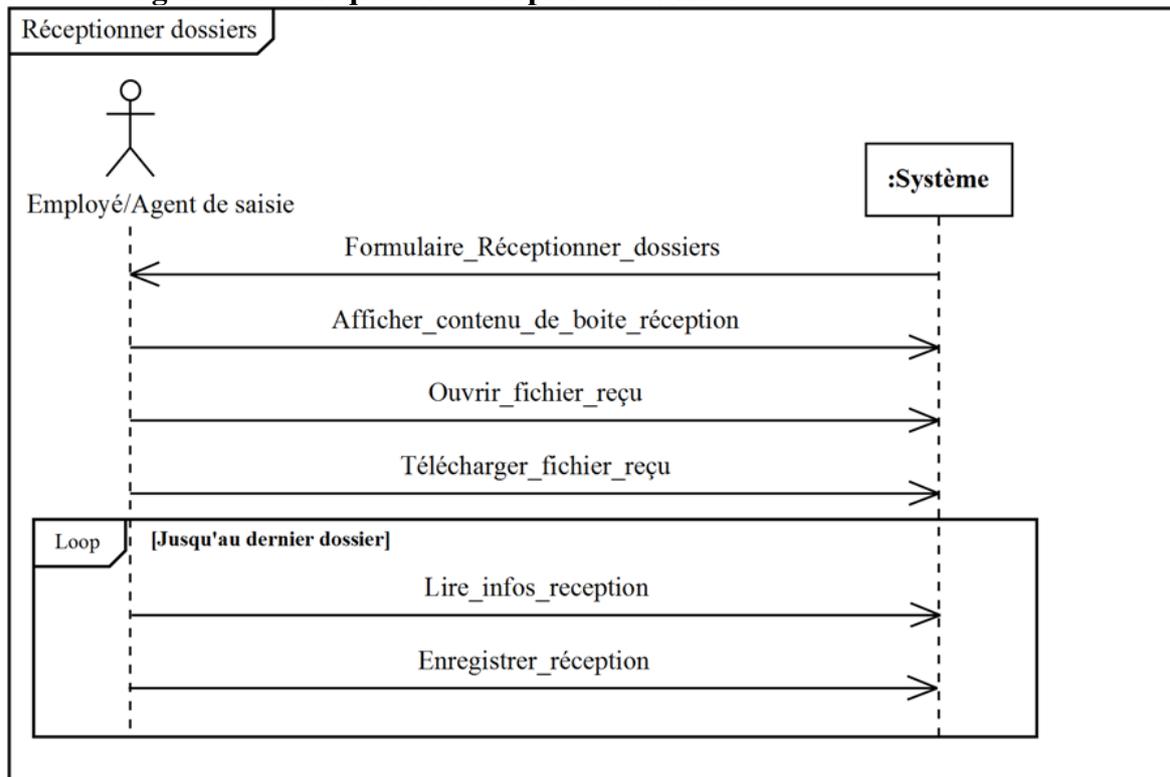


Figure 2.2.20: Diagramme de séquence « Réceptionner dossier ».

2.4. Saisir alpha

Titre : Saisir alpha.

Acteur : Agent de saisie.

But : attribuer le numéro liquidé au dossier.

Pré condition :

- L'agent de saisie s'authentifie.

Post condition : le numéro liquidé est ajouté.

Scénario nominal :

- 1- Le système affiche le formulaire «Saisie alpha».
- 2- Le système lit le code barre ou le numéro pré liquidé.
- 3- Le système affiche les informations du dossier.
- 4- Le système lit le numéro liquidé et l'enregistre.
- 5- Le système affiche une notification «Numéro liquidé ajouté avec succès».

Tableau 2.2.13: Fiche descriptive textuelle du cas « Saisir alpha ».

❖ Diagramme d'activité «Saisir alpha »

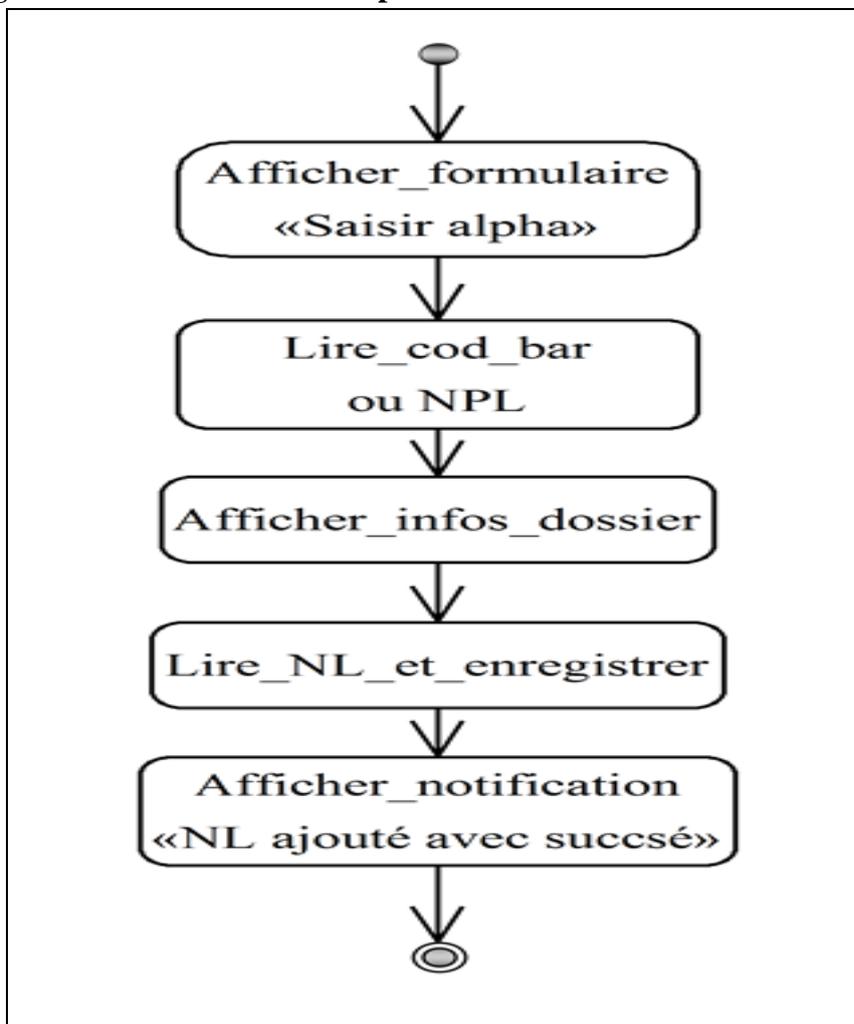


Figure 2.2.21: Diagramme d'activité «Saisir alpha »

❖ Diagramme de séquence « Saisir alpha »

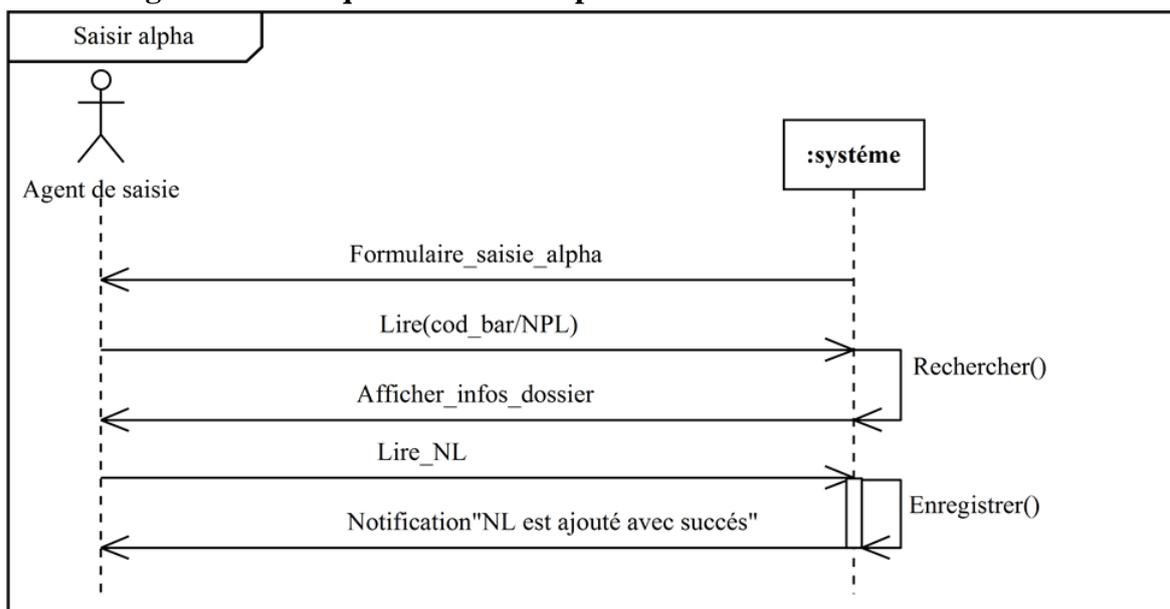


Figure 2.2.22: Diagramme de séquence « Saisir alpha ».

2.5. Changer état dossier

Titre : Changer état dossier.

Acteur : Employé, Agent de saisie.

But : Changer l'état d'un dossier.

Pré condition :

- L'employé /agent de saisie s'authentifie.
- L'Agent de saisie s'authentifie.

Post condition : l'état du dossier est changé.

Scenario nominal :

- 1- Le système affiche le formulaire «Changer l'état».
- 2- Le système lit le code barre ou le numéro pré liquidé ou (numéro liquidé si le dossier est liquidé).
- 3- Le système affiche les informations du dossier.
- 4- Le système lit l'état de dossier et le date de changement d'état et les enregistre.
- 5- Le système affiche une notification «l'état de dossier est enregistré avec succès».

Tableau 2.2.14: Fiche descriptive textuelle du cas « Changer état dossier ».

❖ Diagramme d'activité «Changer état dossier»

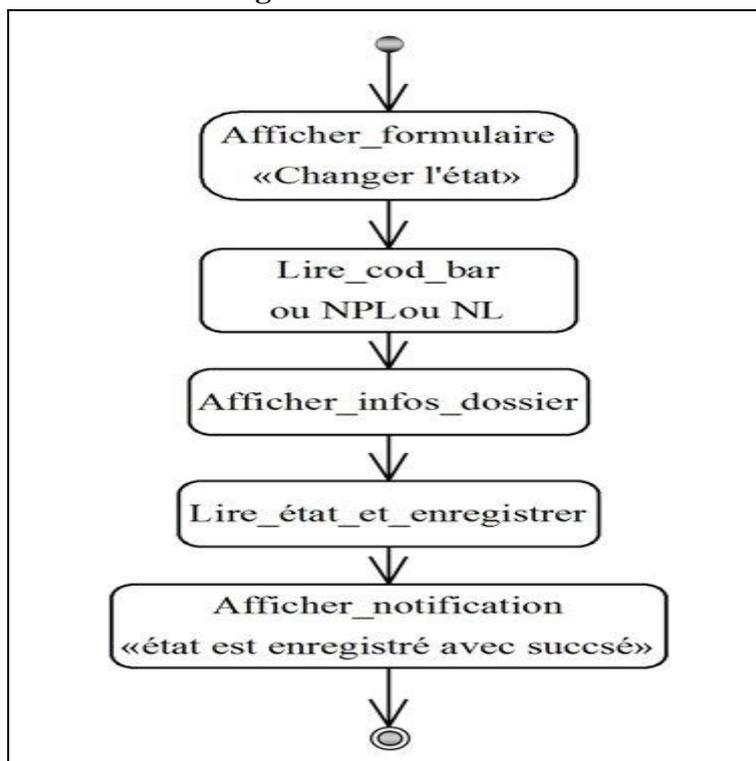


Figure 2.2.23 : Diagramme d'activité «Changer état dossier».

❖ Diagramme de séquence « Changer l'état dossier »

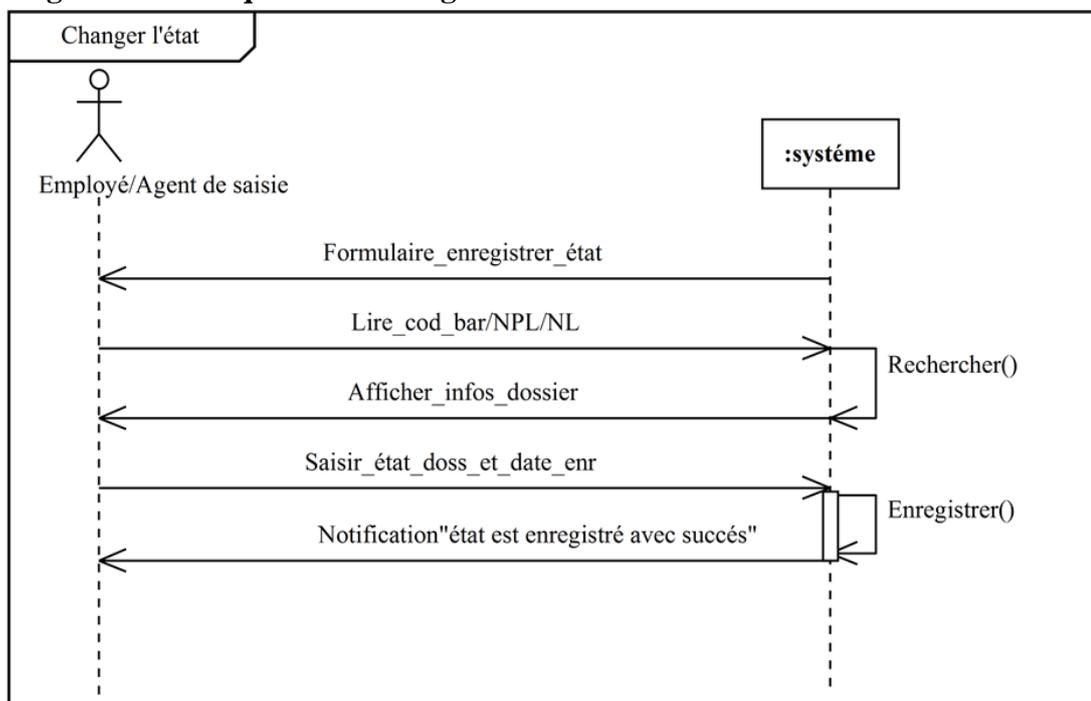


Figure 2.2.24 : Diagramme de séquence «Changer état dossier».

2.6. Tracer dossier

Titre : Tracer dossier.

Acteur : Administrateur.

But : Connaitre la cheminement du dossier à travers les services.

Pré condition :

- L'administrateur s'authentifie.
- Il existe au moins un dossier dans la base de données.

Post condition : la trace du dossier est affichée.

Scenario nominal :

- 1- Le système affiche le formulaire «Trace dossier».
- 2- Le système lit le critère sélectionné (par numéro pré liquidé, par numéro liquidé, par nom et prénom de retraité et date de naissance).
- 3- Le système recherche le dossier.
- 4- Le système affiche la trace du dossier (date envoi, date dépôt, nom émetteur, nom service, état de dossier,..).

Tableau 2.2.15: Fiche descriptive textuelle du cas « Tracer dossier ».

❖ Diagramme d'activité « Tracer dossier »

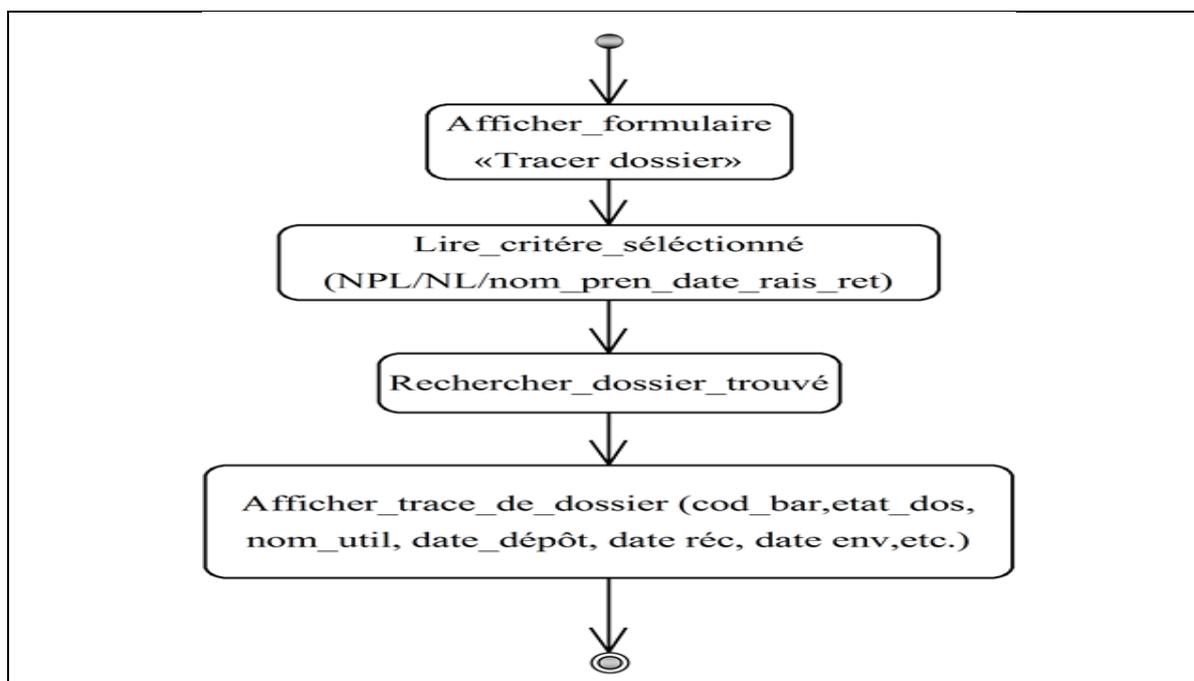


Figure 2.2.25: Diagramme d'activité « Tracer dossier ».

❖ Diagramme de séquence « Tracer dossier »

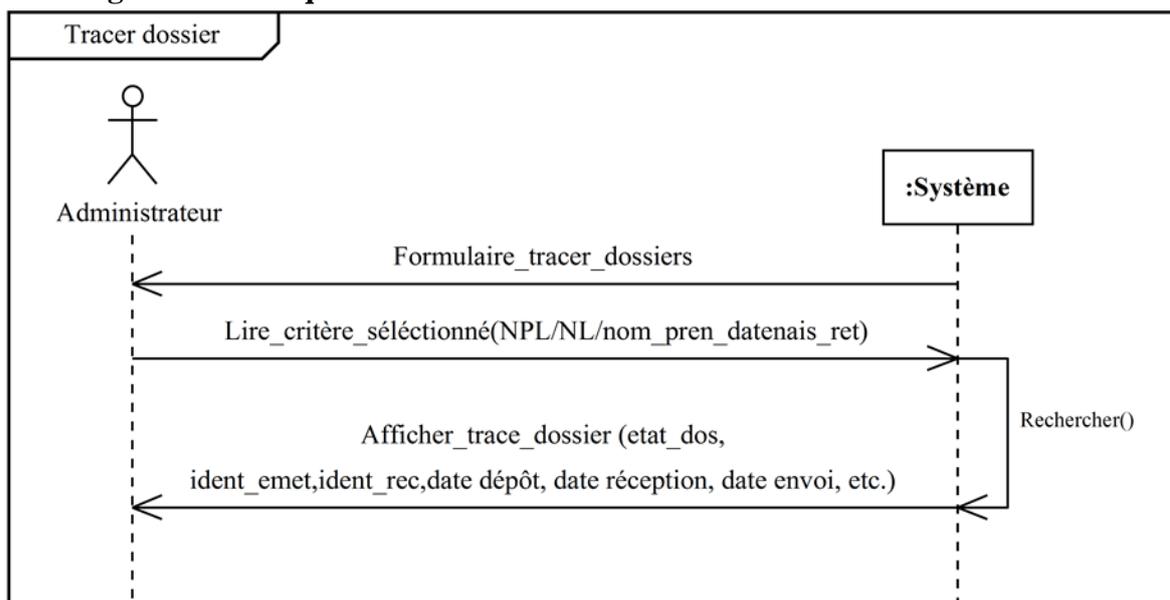


Figure 2.2.26: Diagramme de séquence « Tracer dossier ».

2.7. Localiser dossier

Titre : Localiser dossier.

Acteur : Administrateur/employé.

But : Connaitre la position et l'état de dossier.

Pré condition :

- L'administrateur /l'employé s'authentifie.
- Il existe au moins un dossier dans la base de données.

Post condition : position actuelle du dossier et détails.

Scenarionominal :

- 6- Le système affiche le formulaire «Localiser dossier».
- 7- Le système lit le critère de recherche sélectionné (par numéro pré liquidé/ par numéro liquidé/ par nom et prénom et date de naissance de retraite).
- 8- Le système recherche le dossier.
- 9- Le système affiche les détails du dossier (code barre, état de dossier, numéro pré liquidé, numéro liquidé, nom service actuel, nom utilisateur actuel).

Tableau 2.2.16: Fiche descriptive textuelle du cas « Localiser dossier ».

❖ Diagramme d'activité «Localiser dossier »

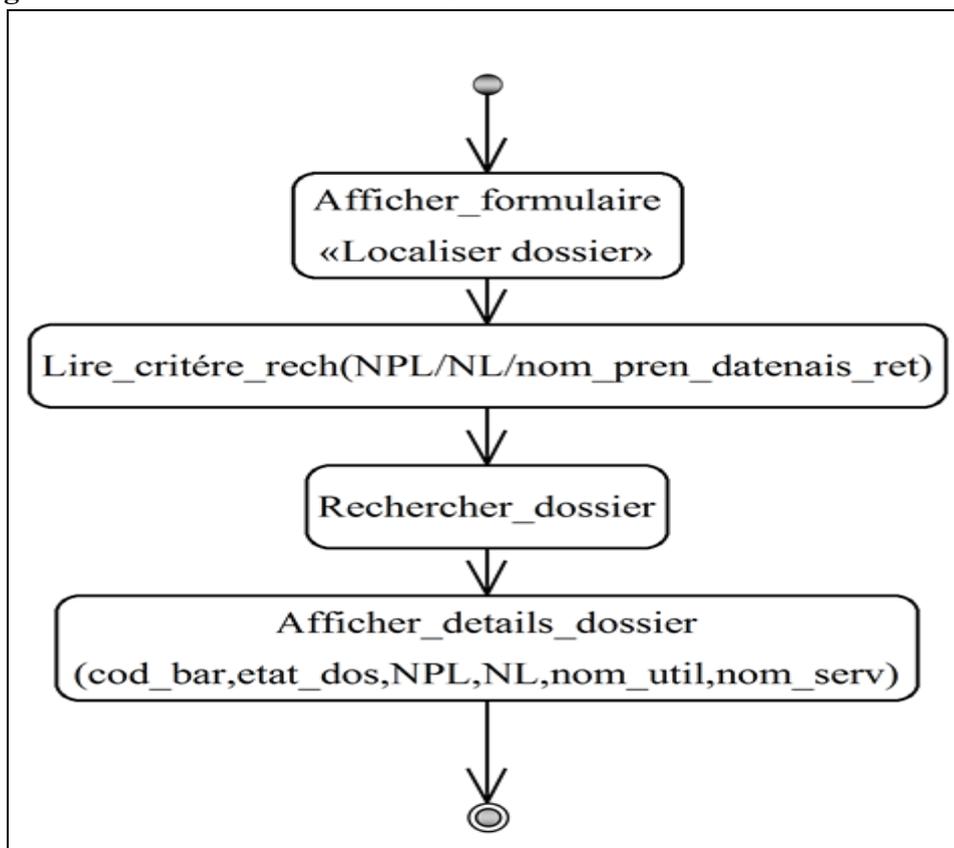


Figure 2.2.27 : Diagramme d'activité de cas «Localiser dossier ».

❖ Diagramme de séquence «Localiser dossier »

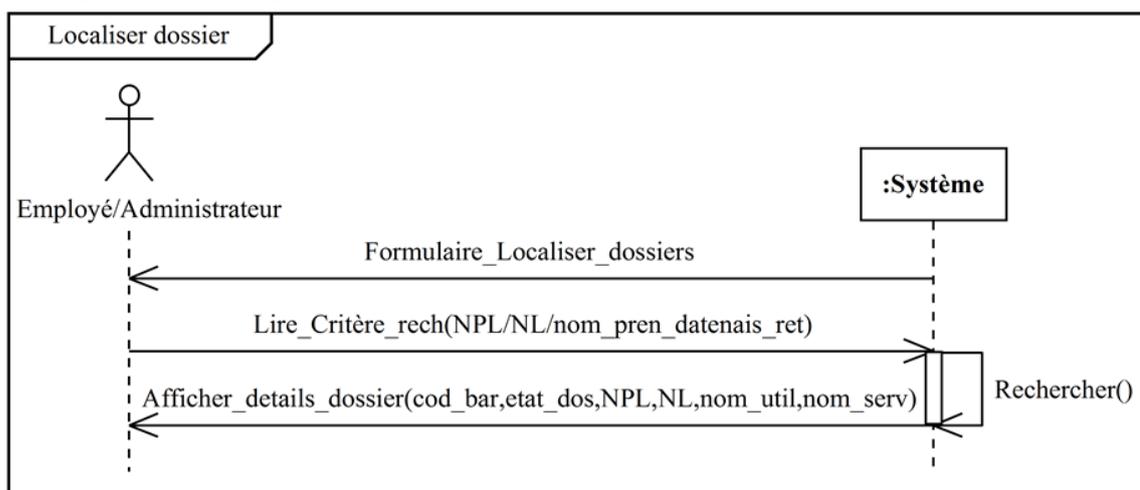


Figure 2.2.28: Diagramme de séquence « Localiser dossier ».

2.8. Etablir statistiques dossiers

2.8.1. Etablir statistiques dossiers par utilisateur

Titre : Etablir statistiques dossiers par utilisateur.

Acteur : Administrateur.

But : Calculer et éditer un rapport de synthèse concernant les dossiers traités par un utilisateur donné, Ce rapport contient des données calculées telles que: le délai moyen de traitement des dossiers, le nombre des dossiers traité pendant une période donnée).

Pré condition : Administrateur s'authentifie.

Post condition : Les statistiques par utilisateur affichées.

Scenarion nominal :

- 1- Le système affiche le formulaire «Etablir statistiques dossiers par utilisateur».
- 2- Le système lit le nom et prénom de l'utilisateur sélectionné.
- 3- Le système lit la date début et la date fin de la période voulue.
- 4- Le système calcule les statistiques concernant l'utilisateur.
- 5- Le système affiche un rapport de synthèse.

Tableau 2.2.17: Fiche descriptive textuelle du cas « Etablir statistiques dossiers par utilisateur ».

❖ Diagramme d'activité « Etablir statistiques dossiers par utilisateur »

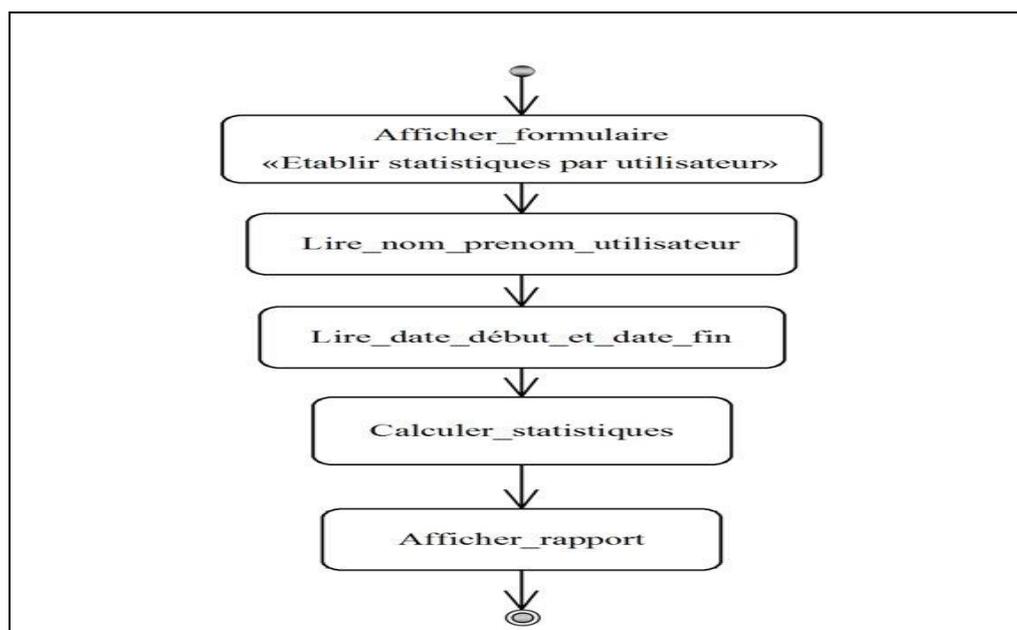


Figure 2.2.29 : Diagramme d'activité « Etablir statistiques dossiers par utilisateur ».

❖ Diagramme de séquence « Etablir statistiques dossiers par utilisateur »

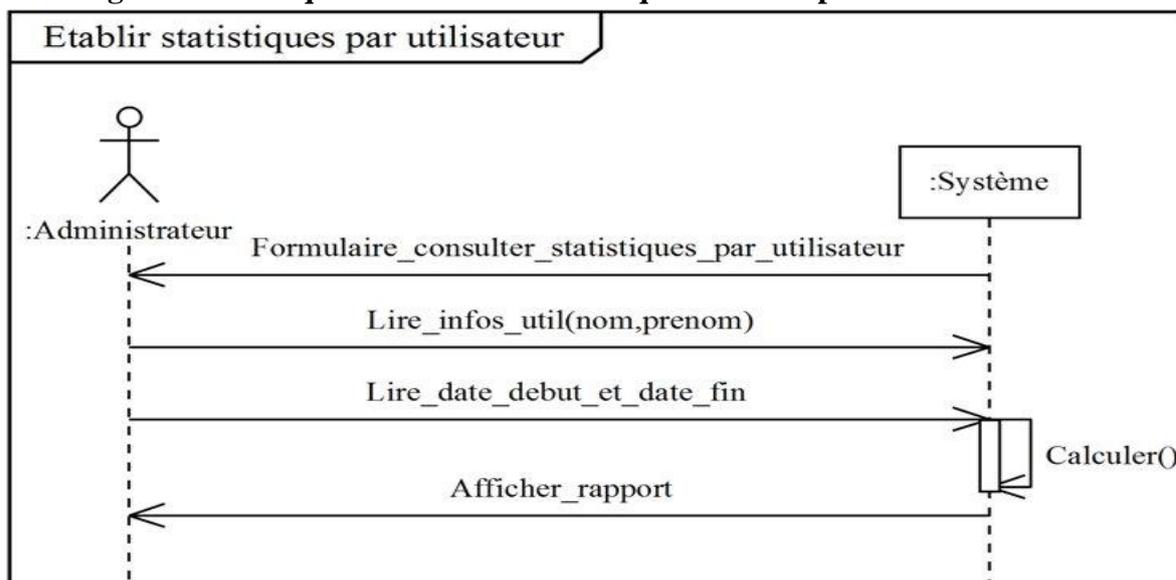


Figure 2.2.30 : Diagramme de séquence « Etablir statistiques dossiers par utilisateur ».

2.8.2. Etablir statistiques dossiers par service

Titre : Etablir statistiques dossiers par service.

Acteur : Administrateur.

But : Calculer et éditer un rapport de synthèse concernant les dossiers traité par un service donné, Ce rapport contient des données calculées telles que : le délai moyen de traitement des dossiers, le nombre dossiers traité pendant une période donnée).

Pré condition : Administrateur s'authentifie.

Post condition : Les statistiques par service affichées.

Scenarionominal :

- 1- Le système affiche le formulaire «Etablir statistiques dossiers par service».
- 2- Le système affiche la liste des services.
- 3- Le système lit le nom de service sélectionné.
- 4- Le système lit la date début et la date fin du période voulue.
- 5- Le système calcule les statistiques concernant le service.
- 6- Le système affiche un rapport de synthèse.

Tableau 2.2.18: Fiche descriptive textuelle du cas « Etablir statistiques dossiers par service ».

❖ Diagramme d'activité « Etablir statistiques dossiers par service ».

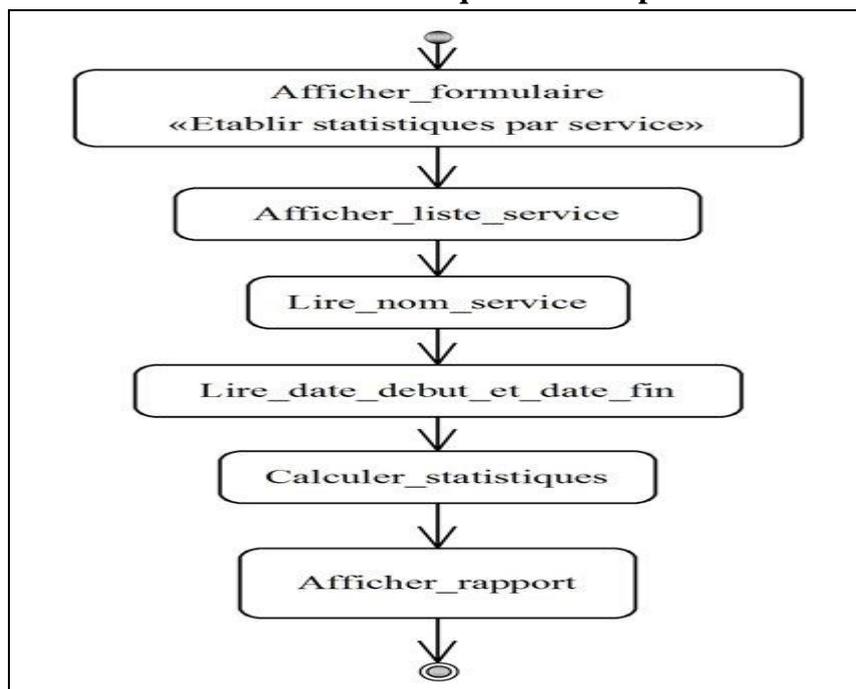


Figure 2.2.31: Diagramme d'activité « Etablir statistiques dossiers par service ».

❖ Diagramme de séquence « Etablir statistiques dossiers par service »

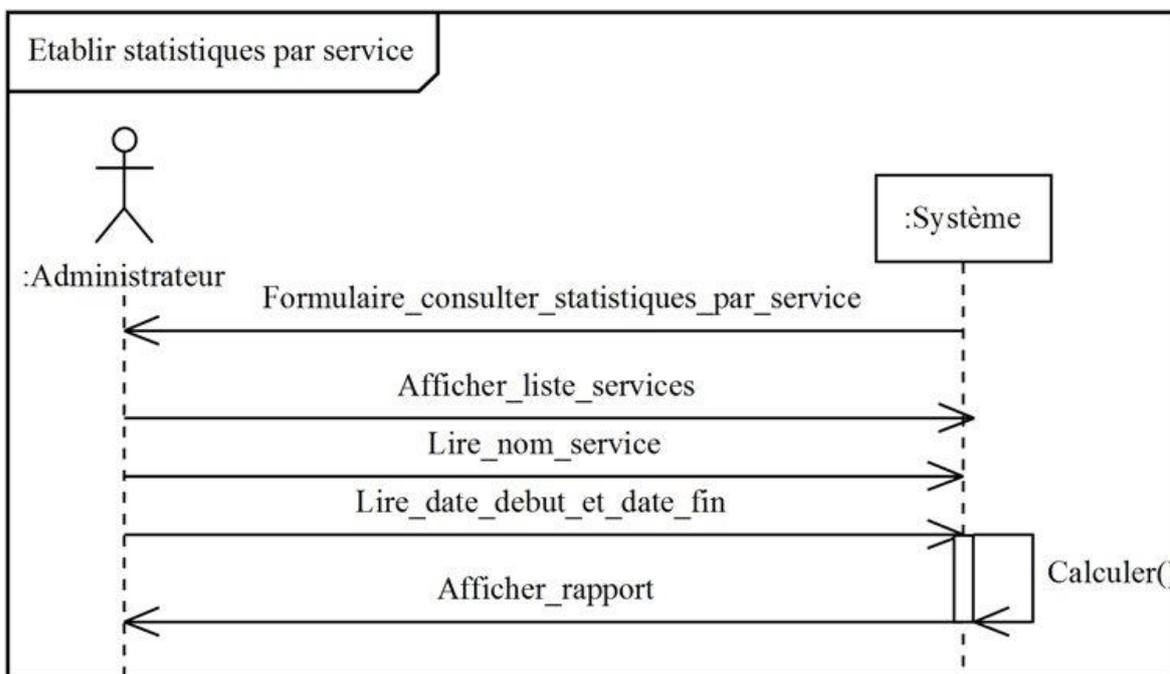


Figure 2.2.32 : Diagramme de séquence « Etablir statistiques dossiers par service ».

Conclusion

Cette phase a permis de compléter le recueil initial des besoins effectués pendant l'étude préliminaire afin de pouvoir fixer les principales fonctionnalités dont doit disposer notre futur système, bien que ce ne soit qu'une première étape pour la mise en place totale de notre système, Dans le prochain chapitre nous allons atteler à la capture des besoins techniques.

Chapitre 03 :

CAPTURE
DES
BESOINS TECHNIQUE

Introduction

On va s'intéresser à la branche droite du cycle en Y qui est « la capture des besoins techniques» en couvrant les contraintes qui ne traitent pas la description applicative, C'est une phase qui sert à compléter par des besoins transmis vers aux la capture des besoins fonctionnels.

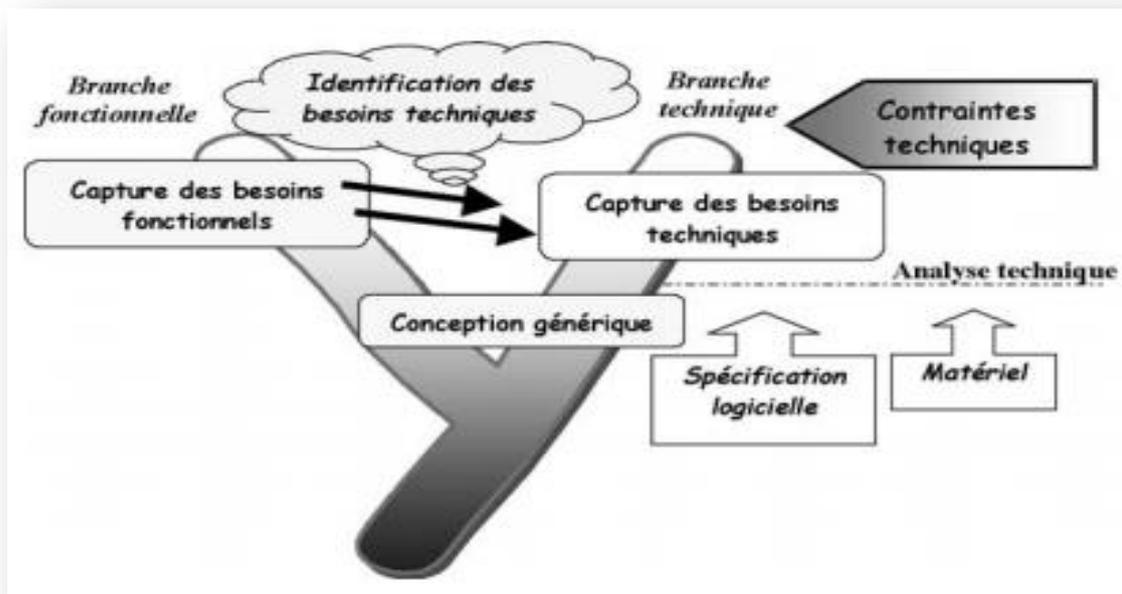


Figure 2.3.33 : Situation de la capture des besoins techniques dans 2TUP

Cette étape peut être appréhendée lorsque les concepteurs de l'application ont obtenu suffisamment d'informations sur l'environnement de travail ainsi que l'architecture globale utilisée pour le système et les prés requis techniques. La capture des besoins techniques se présente comme suit :

- Capture des spécifications matérielles.
- Capture des spécifications logicielles.

1. Capture des spécifications matérielles

1.1. Configuration matérielle

Caractéristiques techniques du :

- **Serveur:**
 - ✓ IBM 2003 serveur 512/40GB.

- ✓ Fujitsu 2008 serveur 860/6*300GB.
- **Client:**
 - ✓ 36 postes.
 - ✓ Windows (7, XP).
- **Réseau:**
 - ✓ 01 routeur CISCO
 - ✓ Réseau local LAN 10/100 mbps.

1.2. Spécification du style d'architecture 2 niveaux

L'architecture à deux niveaux (aussi appelée architecture 2-tiers) caractérise les systèmes clients/serveurs dans lesquels le client demande une ressource et le serveur la lui fournit directement. Cela signifie que le serveur ne fait pas appel à une autre application afin de fournir le service. [7]

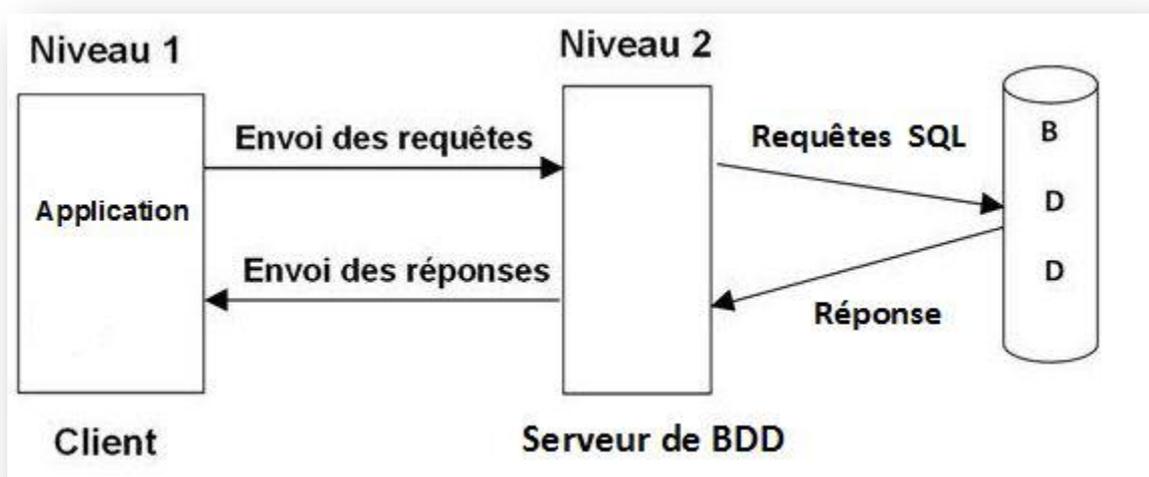


Figure 2.3.34: Architecture 2 niveaux de notre système.

2. Capture des spécifications logicielles

Une fois que les spécifications techniques et d'architecture sont exprimées, on peut s'intéresser aux fonctionnalités propres du système technique en procédant à une spécification logicielle. Pour ceci, on propose d'utiliser les cas d'utilisation de manière différente que pour la spécification fonctionnelle. C'est pourquoi nous avons introduit le concept d'exploitant et de cas d'utilisation technique.

- **Exploitant** : est un acteur au sens d'UML, il bénéficie des fonctionnalités techniques du système.

- **Cas d'utilisation technique (CUT):** un CUT destiné à l'exploitant est une séquence d'actions produisant une valeur ajoutée opérationnelle ou technique mais pas fonctionnelle.

2.1. Identification des exploitants du système

Les exploitants de notre système sont la majorité des acteurs de la branche fonctionnelle. Ce sont : l'administrateur, l'employé, l'agent de saisie.

2.2. Identification des cas d'utilisation technique

Les cas d'utilisation technique de notre système sont :

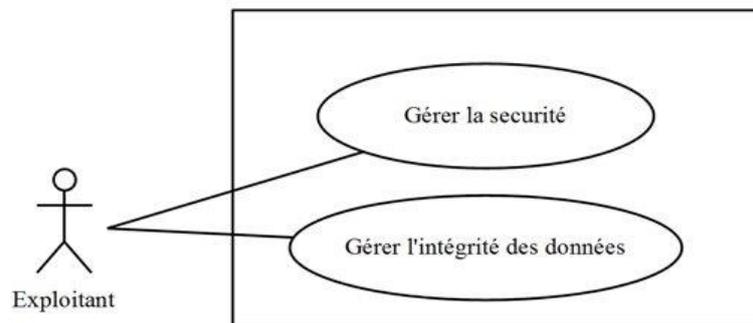


Figure 2.3.35: Modèle de spécification logicielle de système.

2.3. Description des cas d'utilisation technique

Pour des raisons de délais du projet et d'espace dans ce mémoire, nous nous contentons de décrire l'aspect technique « Gérer la sécurité ».

2.3.1. Cas d'utilisation « Gérer la sécurité »

- **S'authentifier**

Titre : S'authentifier.

But : Vérifier l'autorisation d'accéder au système.

Pré condition : L'exploitant possède un compte.

Post conditions : L'exploitant est identifié par le système et la page d'accueil est accessible.

Scénario nominal :

1. L'exploitant lance l'application.
2. Le système lit le nom d'exploitant et le mot de passe .
3. Le système vérifie la validité du nom d'exploitant et du mot de passe et affiche la page d'accueil.

Scénario alternatif :

1. Username de l'exploitant et/ou le mot de passe est erroné.
 - 1.1. Le système affiche une notification « Username ou mot de passe ou rôle est incorrect ».
 - 1.2. Le scénario nominal reprend au point 2.

Exception :

L'exploitant ne saisit pas le bon nom et mot de passe 3 fois. l'application se ferme.

Tableau 2.3.19: Fiche description textuelle du cas « S'authentifier ».

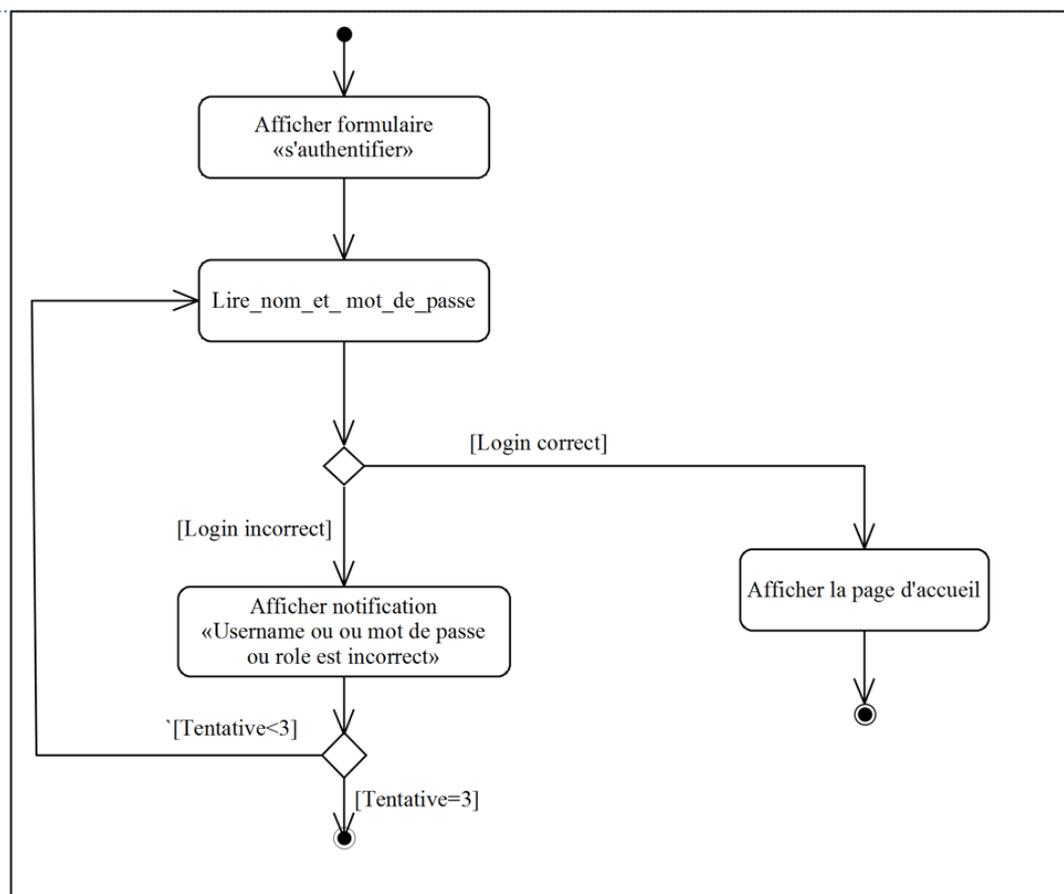
✓ **Diagramme d'activité « S'authentifier »**

Figure 2.3.36 : Diagramme d'activité de cas « S'authentifier ».

❖ Diagramme de séquence « S'authentifier »

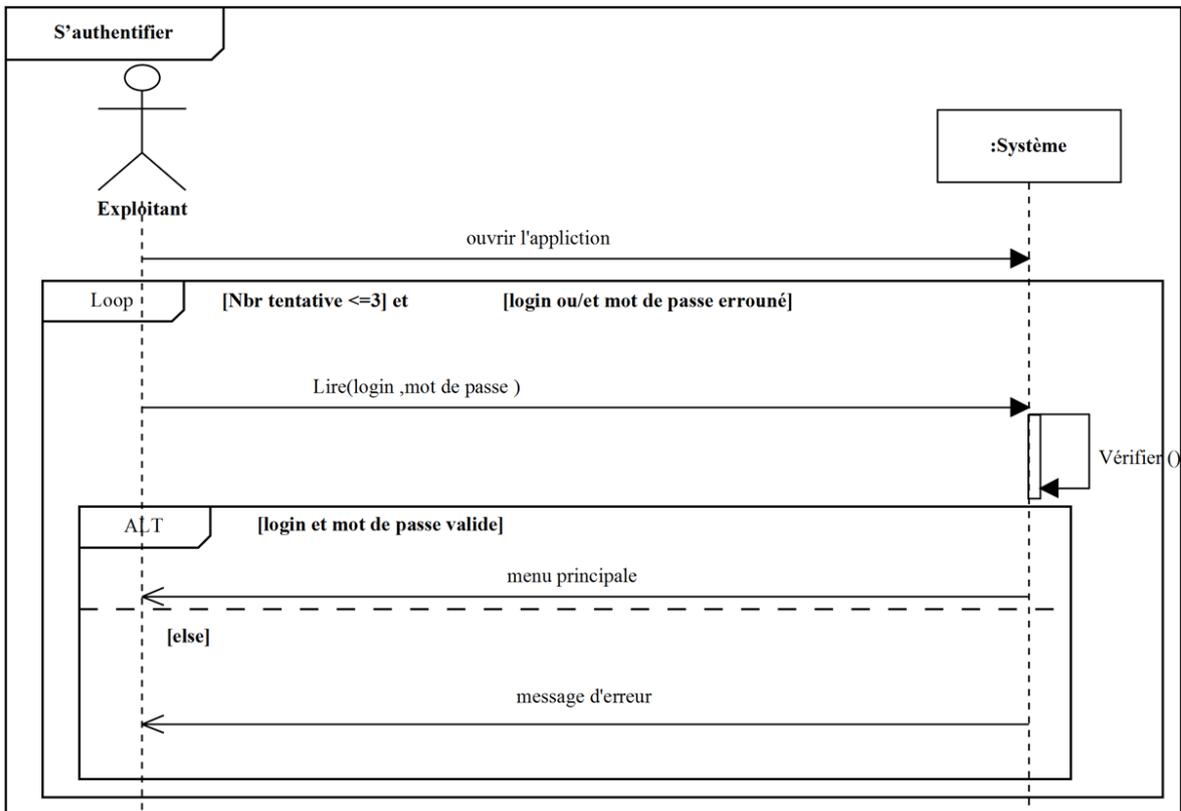


Figure 2.3.37 : Diagramme de séquence « S'authentifier ».

➤ Gérer comptes

Titre : Gérer comptes**But :** Consiste à ajouter, modifier ou supprimer un compte.**Pré condition :**

- L'administrateur s'authentifie.
- En cas de modification ou suppression, il existe au moins un compte dans BDD.

Post conditions :

- La mise à jour est effectuée.
- En cas d'ajout le nouveau compte est enregistré.

🚦 **Cas d'ajouter d'un compte**

Scénario nominal :

L'exploitant accède au système et choisit « Ajouter compte ».

1. Le système affiche le formulaire « Ajoute compte ».

2. Le système lit les informations de ce compte (numéro utilisateur, nom et prénom d'employé, mot de passe, le poste, grade, username, nom de service, droit possible, compte email).
3. Le système vérifie l'existence de compte, enregistre le nouveau et affiche une notification « Ajout avec succès».

Scénario alternatif :

1. Cas où le compte existe déjà.
 - 1.1. le système affiche une notification « Le compte existe déjà ».
 - 1.2.Reprise du scénario nominal au point 2.

 **Cas de modification d'un compte****Scénario nominal :**

L'exploitant accède au système et choisit « Modifier compte ».

1. Le système affiche le formulaire « modifier compte ».
1. Le système affiche la liste des comptes existants.
2. Le système lit le compte sélectionné.
2. Le système affiche toutes les informations de cet employé.
3. Le système lit les nouvelles informations et demande la validation de la modification.
3. Le système lit la validation de la modification.
4. Le système enregistre les nouvelles informations.
4. Le système affiche une notification « modification avec succès ».

Exception :

L'exploitant ne valide pas l'opération, la modification est annulée.

 **Cas de suppression d'un compte****Scénario nominal :**

L'exploitant accède au système et choisit « Supprimer compte ».

5. Le système affiche le formulaire « supprimer compte ».
6. Le système affiche la liste des comptes existants.
7. Le système lit le compte sélectionné, et demande la validation de la suppression.
8. Le système lit la validation de la suppression et supprime le compte.

9. Le système affiche une notification « le compte est supprimé avec succès ».

Exception :

L'exploitant ne valide pas l'opération, la suppression est annulée.

Tableau 2.3.20: Fiche descriptive textuelle du cas « Gérer comptes ».

✓ **Diagramme d'activité «Gérer comptes»**

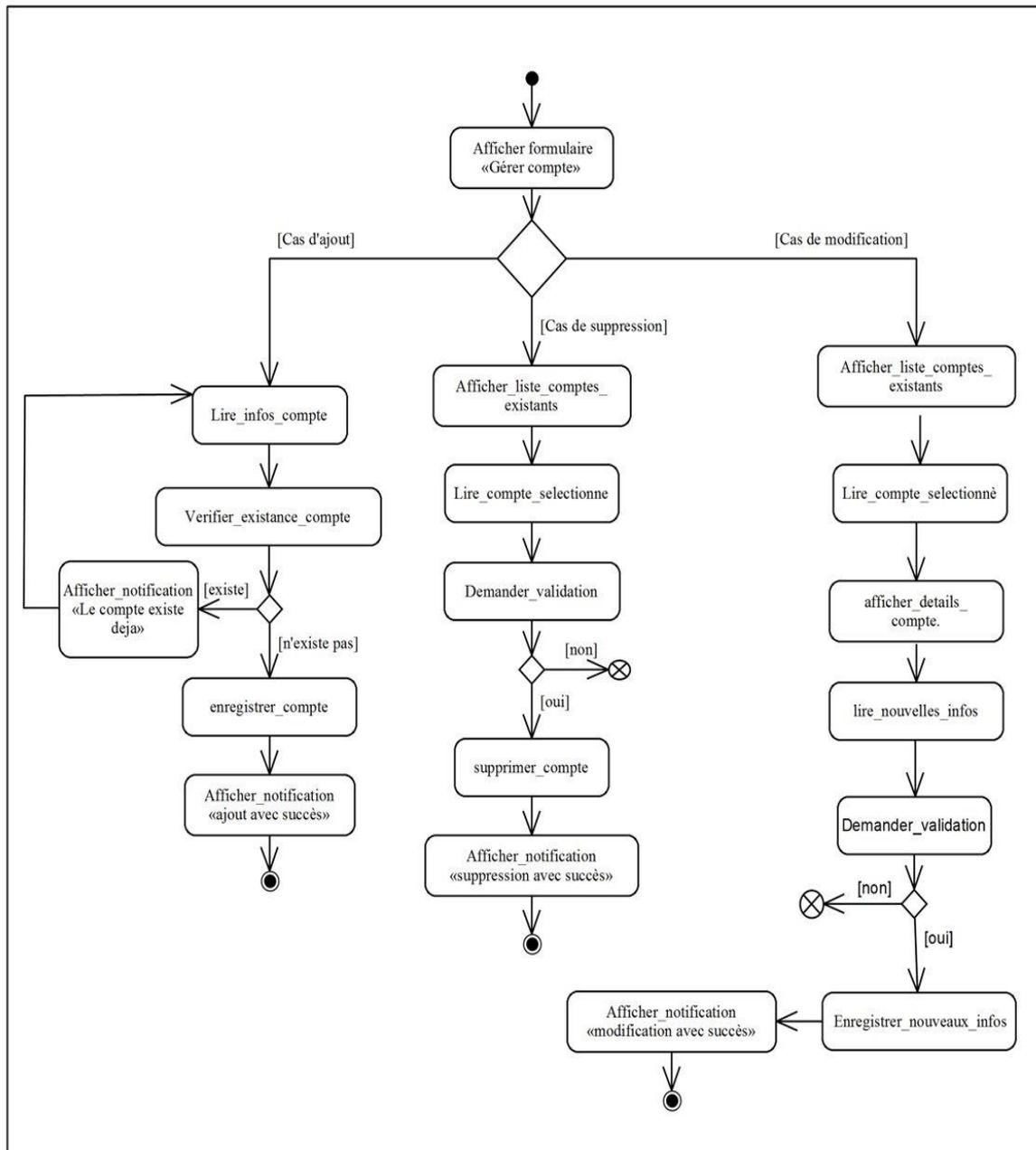


Figure 2.3.38 : Diagramme d'activité de cas « Gérer comptes ».

✓ Diagramme de séquence « Gérer comptes »

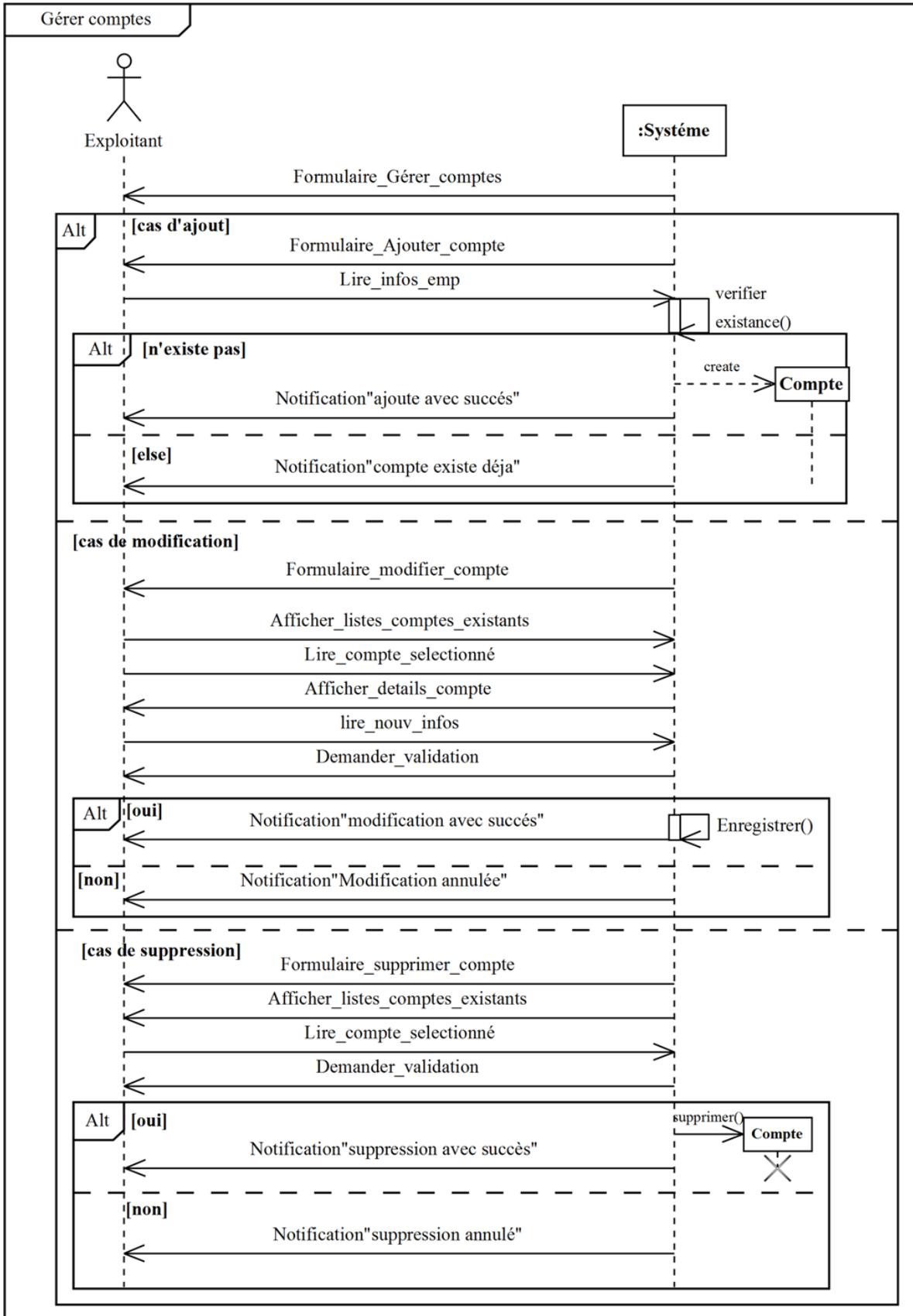


Figure 2.3.39 : Diagramme de séquence « Gérer compte ».

2.3.2. Cas d'utilisation « Gérer l'intégrité des données »

Titre : Gérer l'intégrité des données.

Pré condition : L'exploitant est authentifié et accède à un formulaire.

Scénario nominal:

- 1- L'exploitant saisit les données.
- 2- Le système contrôle la validité de chaque champ.

Scénario alternatif :

- 1- Le système affiche une notification « les champs non valide ».
- 2- Le scénario reprend au point 1.

Tableau 2.3.21: Fiche descriptive textuelle « Gérer l'intégrité des données »

✓ **Diagramme d'activité « Gérer l'intégrité des données »**

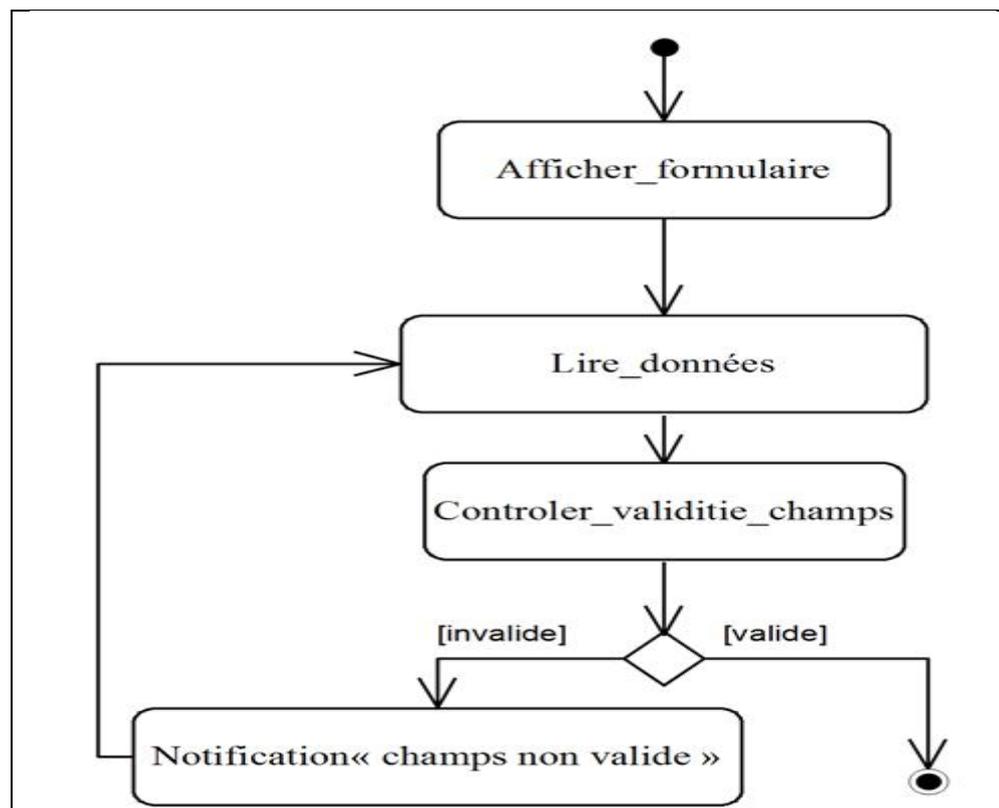


Figure 2.3.40 : Diagramme d'activité de cas « Gérer l'intégrité des données ».

✓ Diagramme de séquence « Gérer l'intégrité des données »

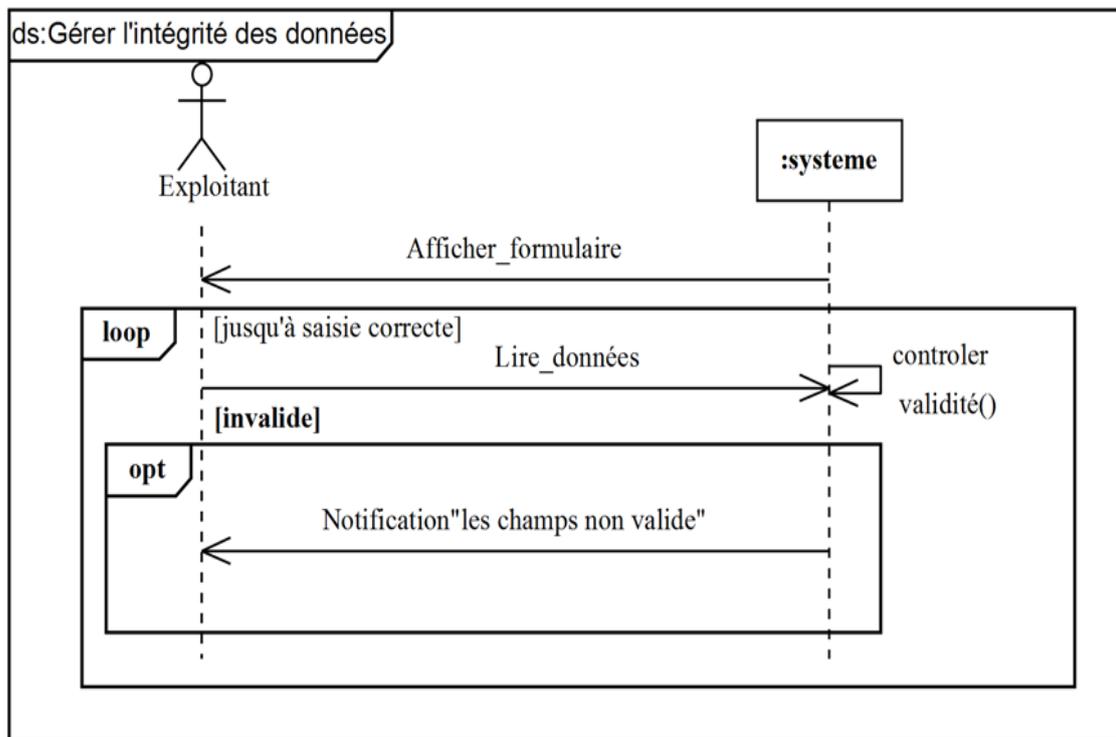


Figure 2.3.41 : Diagramme de séquence « Gérer l'intégrité des données ».

2.4. Organisation du modèle de spécification logicielle

La couche logicielle représente un ensemble de spécifications ou de réalisations qui respectivement expriment ou mettent en œuvre un ensemble de responsabilités techniques et homogènes pour un système logiciel. [1]

Les couches s'empilent en niveaux pour couvrir des transformations logicielles successives, de sorte que la couche d'un niveau ne puisse utiliser que les services des couches des niveaux inférieurs. [1]

Le modèle de spécification logiciel est conditionné par l'architecture déployé. Dans notre cas avec l'architecture 2-tiers, notre système est séparé trois couches logicielles :

- **La couche présentation** ou IHM (Interface Homme/Machine) : gère les interactions utilisateur/machine (la présentation). [9]
 - **La couche traitements :**
 - *Locaux* : contrôles effectués au niveau du dialogue avec l'IHM.
 - *Globaux* : L'application elle-même.[9]
 - **La couche donnée** : Gère le stockage des données et l'accès à ces dernières. [9]
- Les couches présentation et traitements sont implantées sur le client.

- La couche donnée est implantée sur le serveur.
- Ainsi, le contexte de l'application est un contexte multi-utilisateurs avec accès aux données centralisées via un middleware.

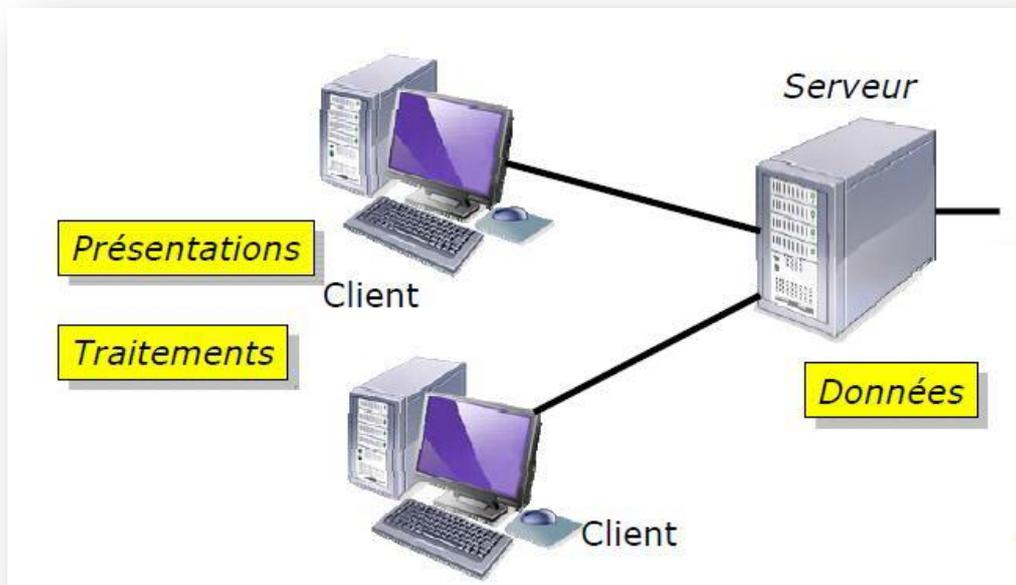


Figure 2.3.42: Organisation du modèle de spécification logicielle.

Conclusion

Au cours de ce chapitre, l'architecture physique a été choisie selon l'environnement adopté, on a pris en compte toutes les contraintes techniques et logicielles pour le choix le plus adapté de notre architecture. Dans le prochain chapitre nous allons présenter la phase d'analyse.

Chapitre 04 :

ANALYSE

Introduction

L'étape d'analyse représente la deuxième étape de la branche gauche du cycle en Y. Elle consiste à étudier précisément les spécifications fonctionnelles de manière à obtenir une idée de ce que va réaliser le système en terme de métier.

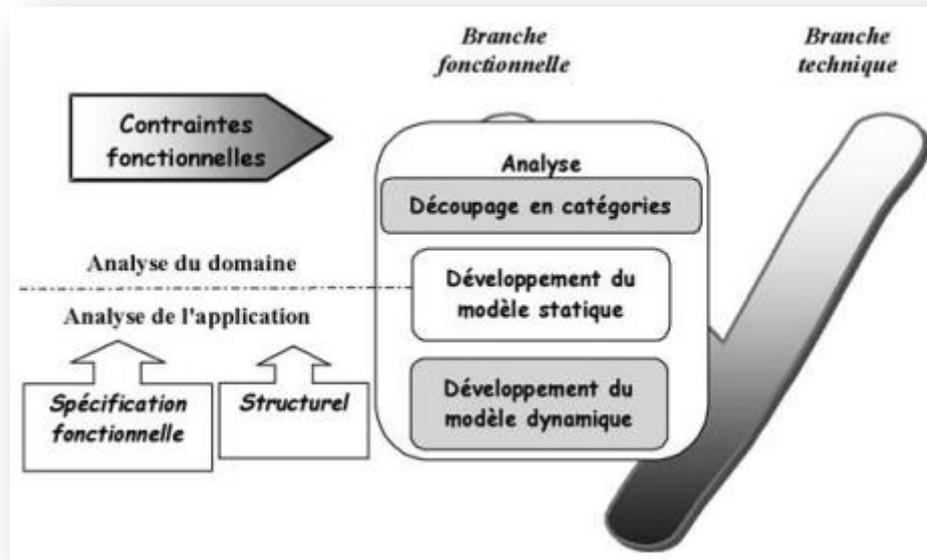


Figure 2.4.43: Situation de l'étape d'analyse dans le 2TUP.

Elle est constituée de 3 activités :

- Le découpage en catégories.
- Le développement du modèle statique.
- Le développement du modèle dynamique.

1. Le découpage en catégories

Le découpage en catégories constitue la première activité de l'étape d'analyse. Il utilise la notion de package pour définir des catégories de classes d'analyse et découper le modèle UML en blocs logiques les plus indépendants possibles.

1.1. La répartition des classes candidates en catégories

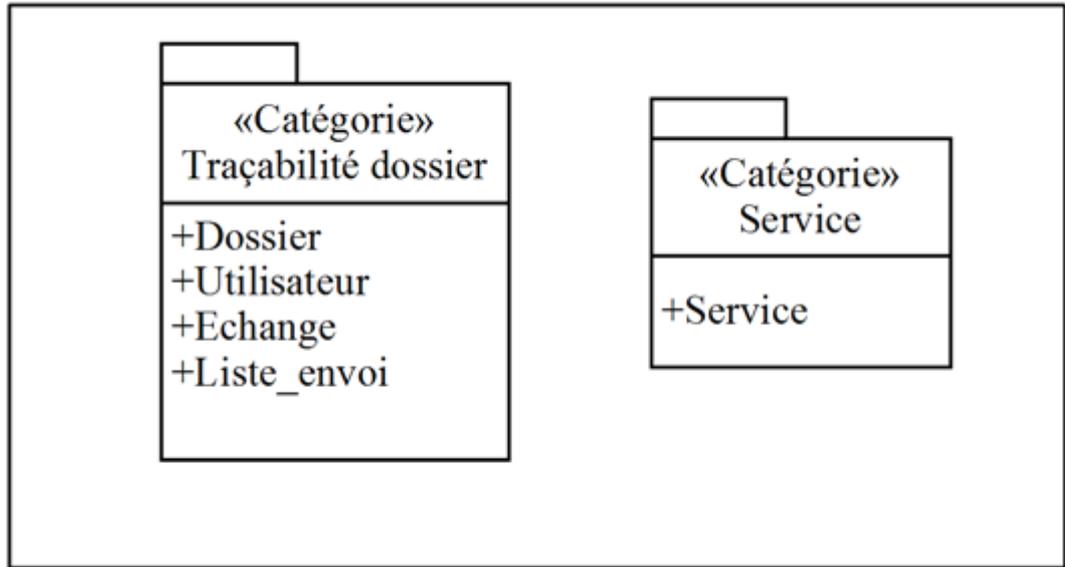
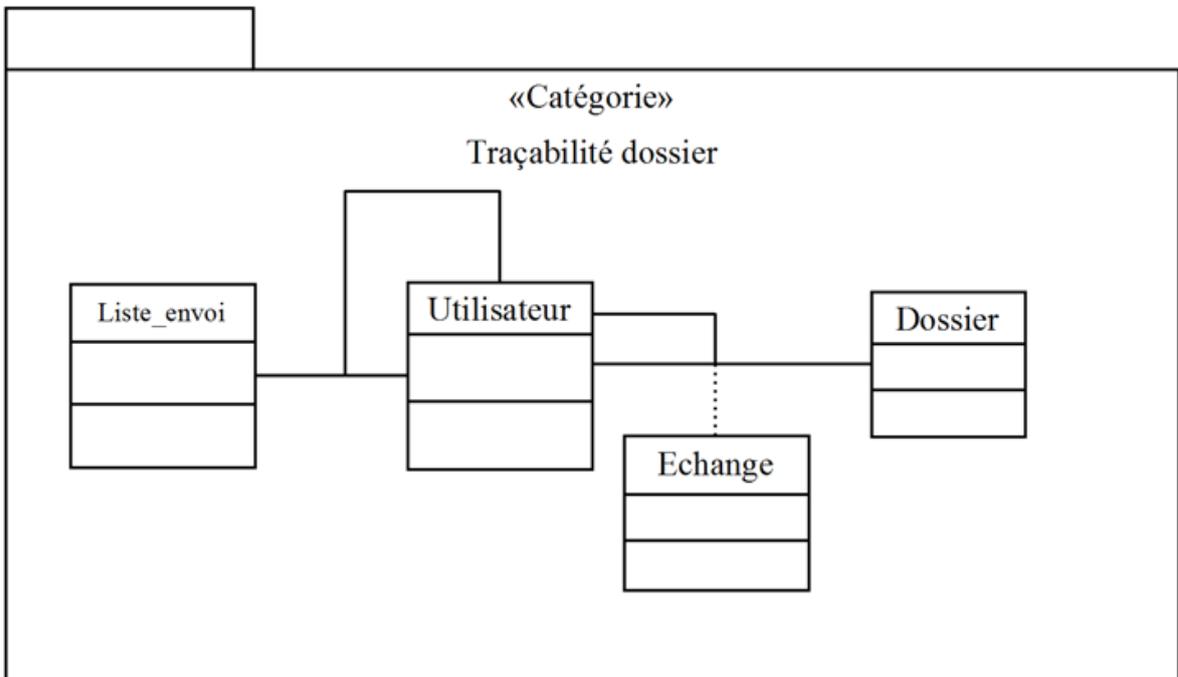


Figure 2.4.44: Découpage en catégories.

1.2. Elaboration des diagrammes de classes préliminaires par catégorie



➤ Diagramme de classe de catégorie traçabilité dossier

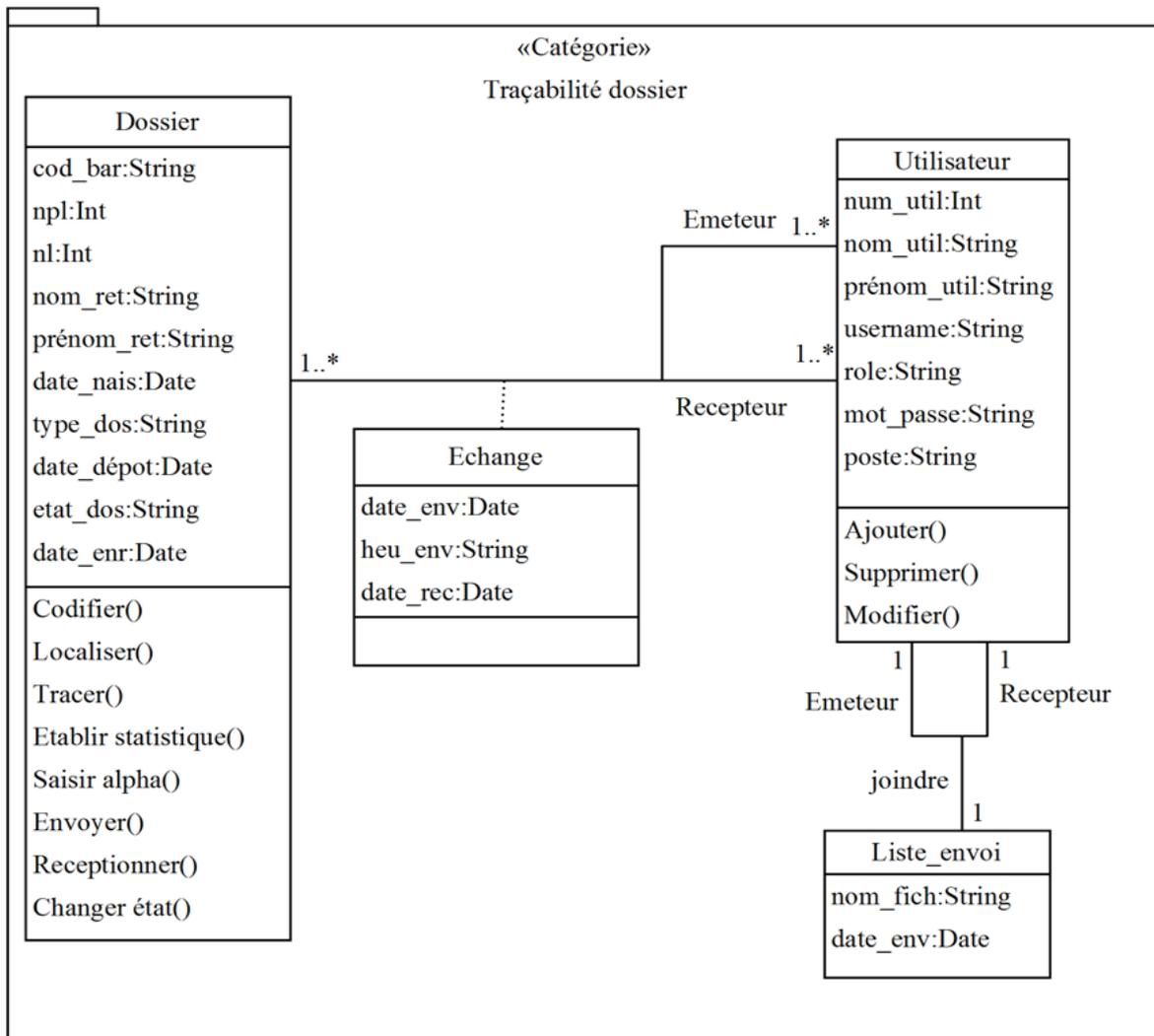


Figure 2.4.47 : Diagramme de classe de catégorie traçabilité dossier

➤ Diagramme de classe de catégorie Service

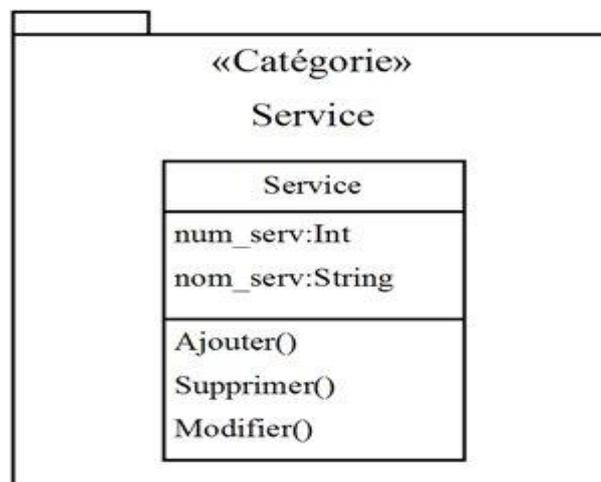


Figure 2.4.48 : Diagramme de classe de catégorie service.

3. Développement du modèle dynamique

Le développement du modèle dynamique est la troisième activité de l'étape d'analyse. Lors de cette étape, nous décrivons les différentes interactions entre les objets de notre application. En effet, nous avons utilisé le modèle dynamique : le diagramme d'interaction.

1.1. Diagrammes d'interaction

1.1.1. Diagramme d'interaction de cas « Codifier dossier »

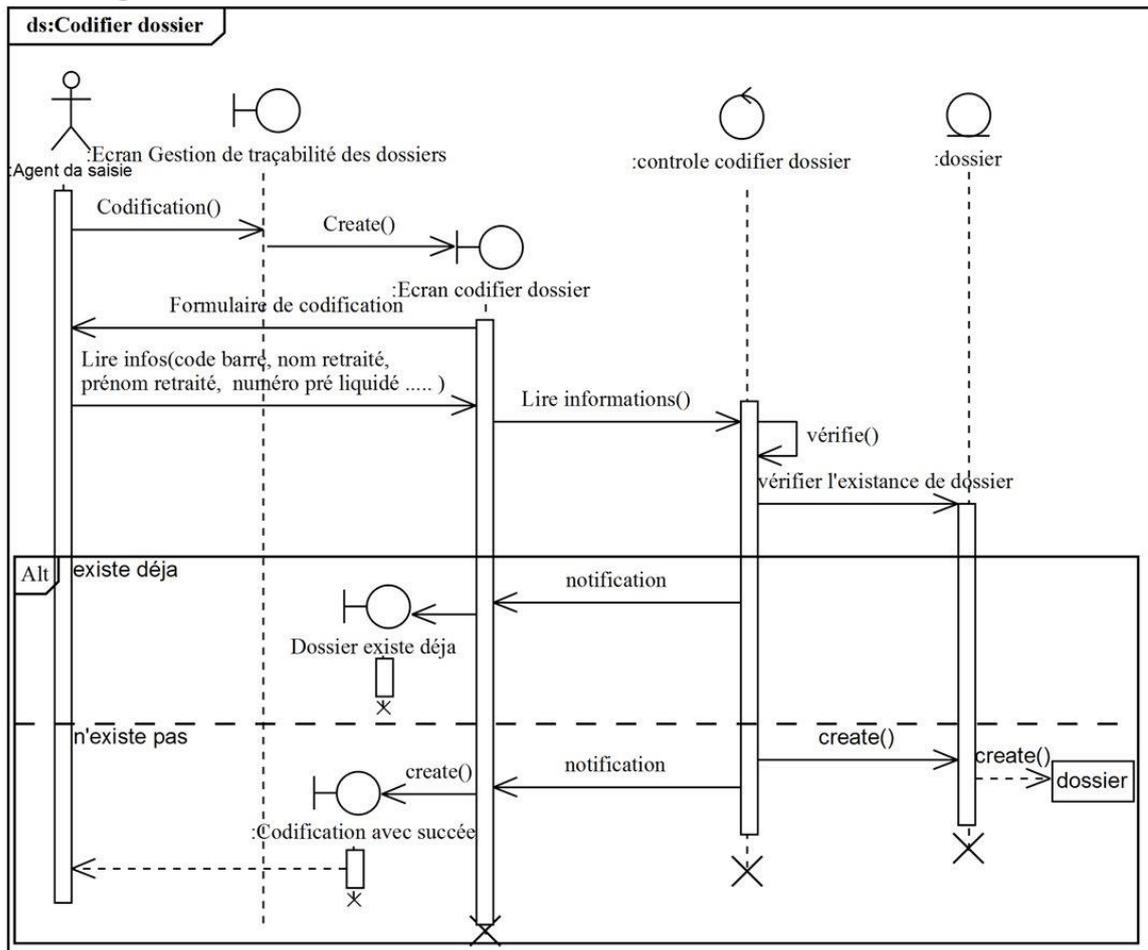


Figure 2.4.49: Diagramme d'interaction de cas d'utilisation « Codifier dossier ».

1.1.2. Diagramme d'interaction de cas « Saisir alpha »

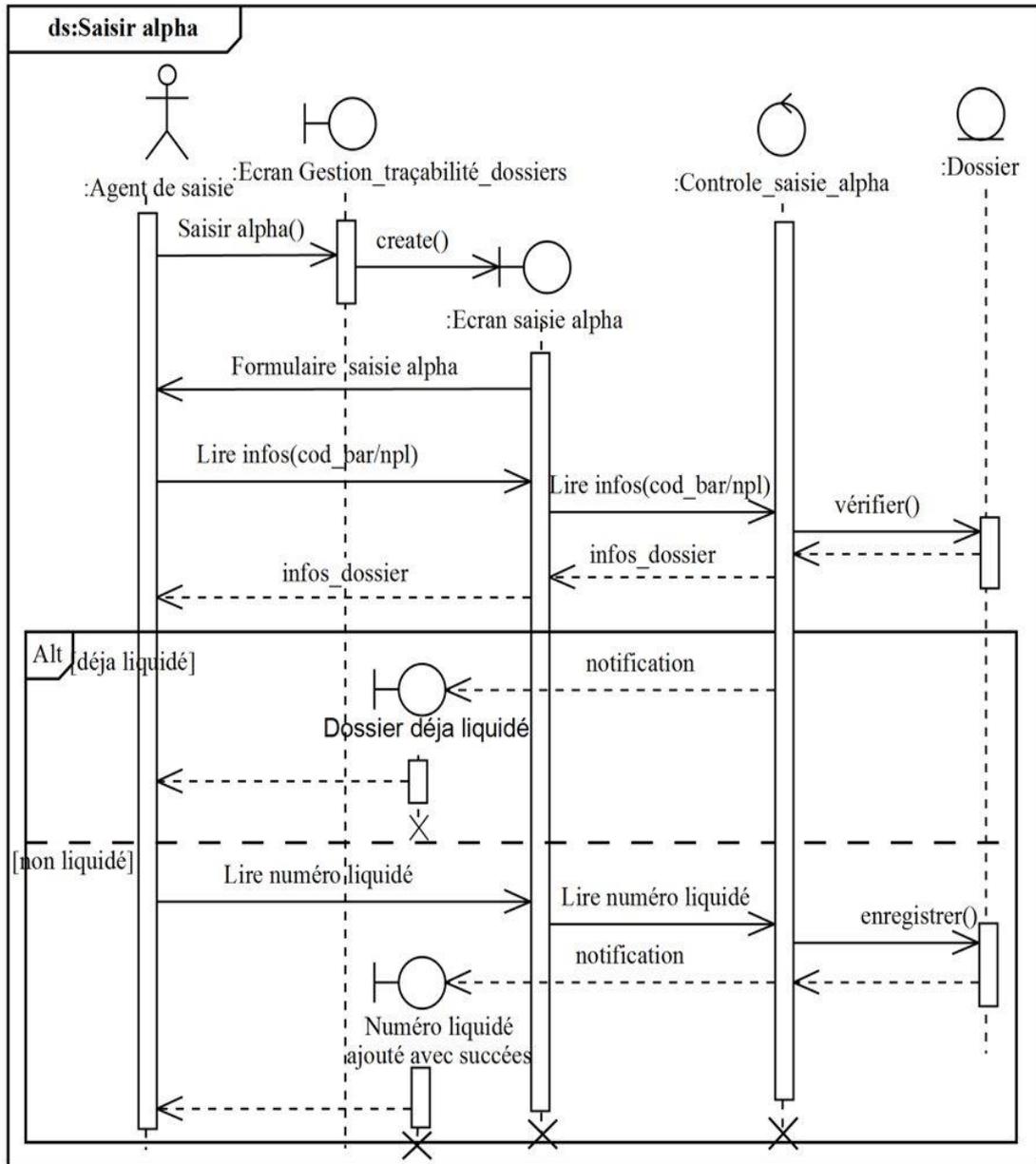


Figure 2.4.50 : Diagramme d'interaction de cas d'utilisation « Saisir alpha »

1.1.3. Diagramme d'interaction de cas « Changer état »

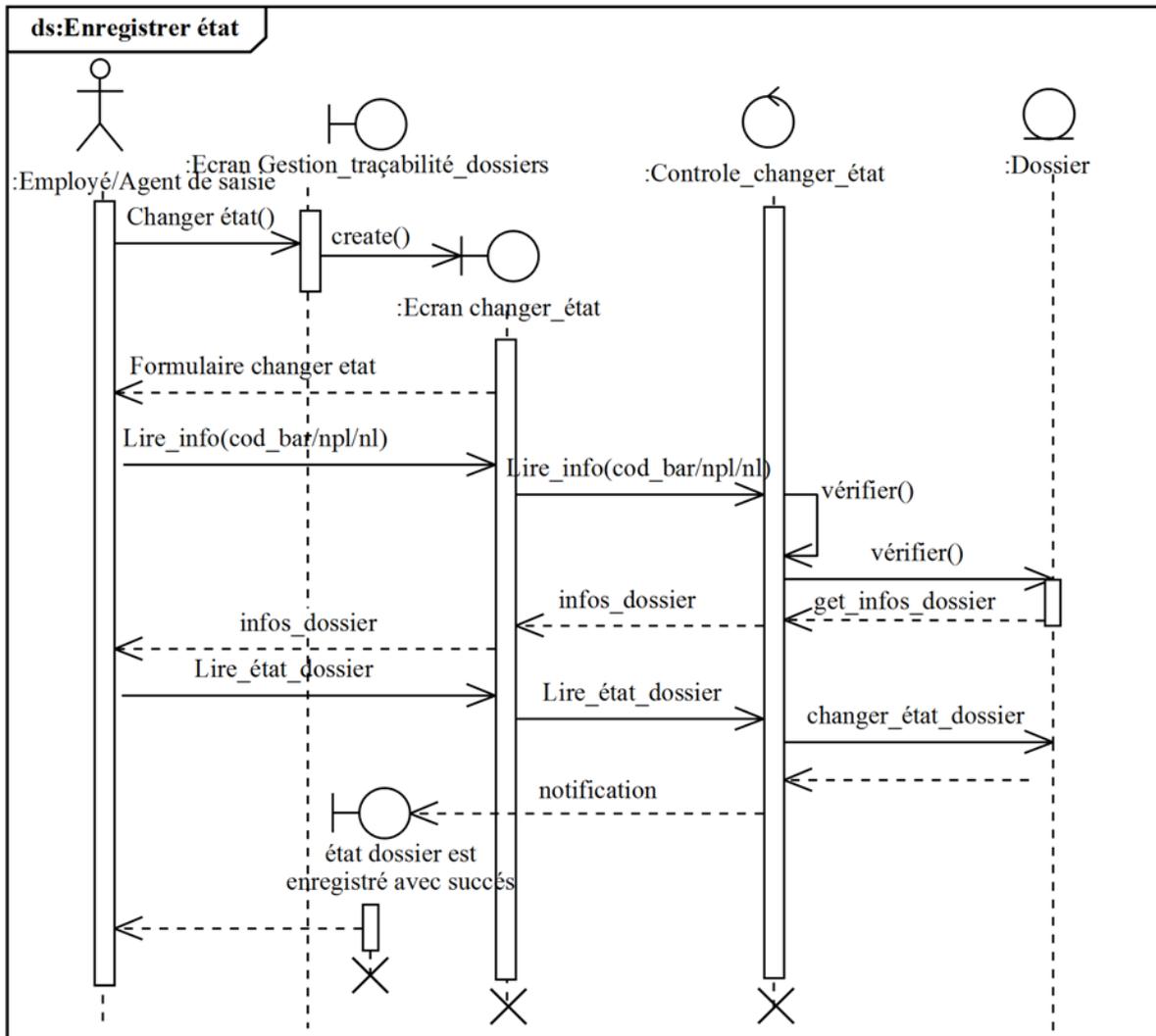


Figure 2.4.51: Diagramme d'interaction de cas d'utilisation « Changer état »

1.1.4. Diagramme d'interaction de cas « Localiser dossier »

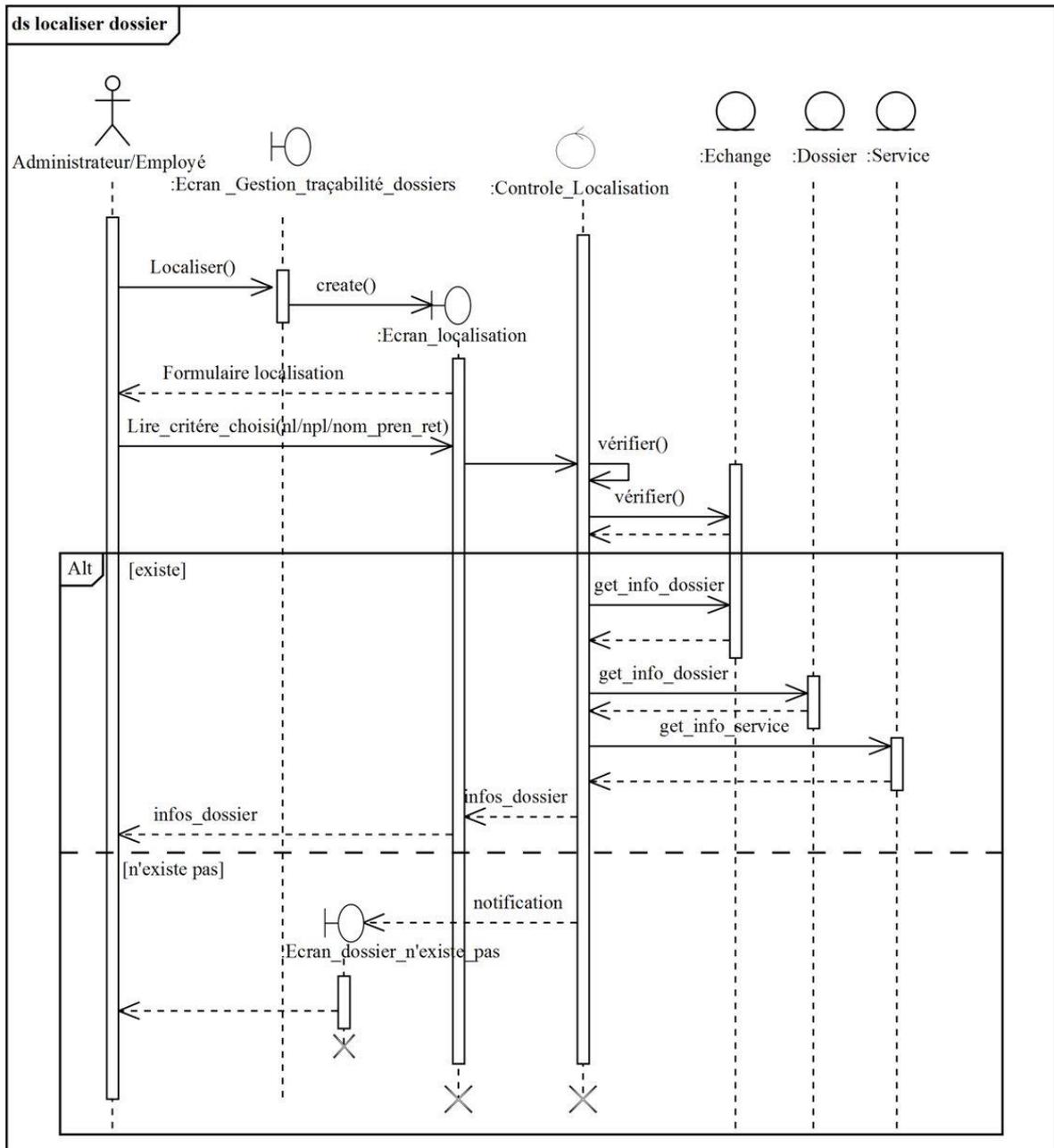


Figure 2.4.52: Diagramme d'interaction de cas d'utilisation « Localiser dossier »

1.1.5. Diagramme d'interaction de cas « Tracer dossier »

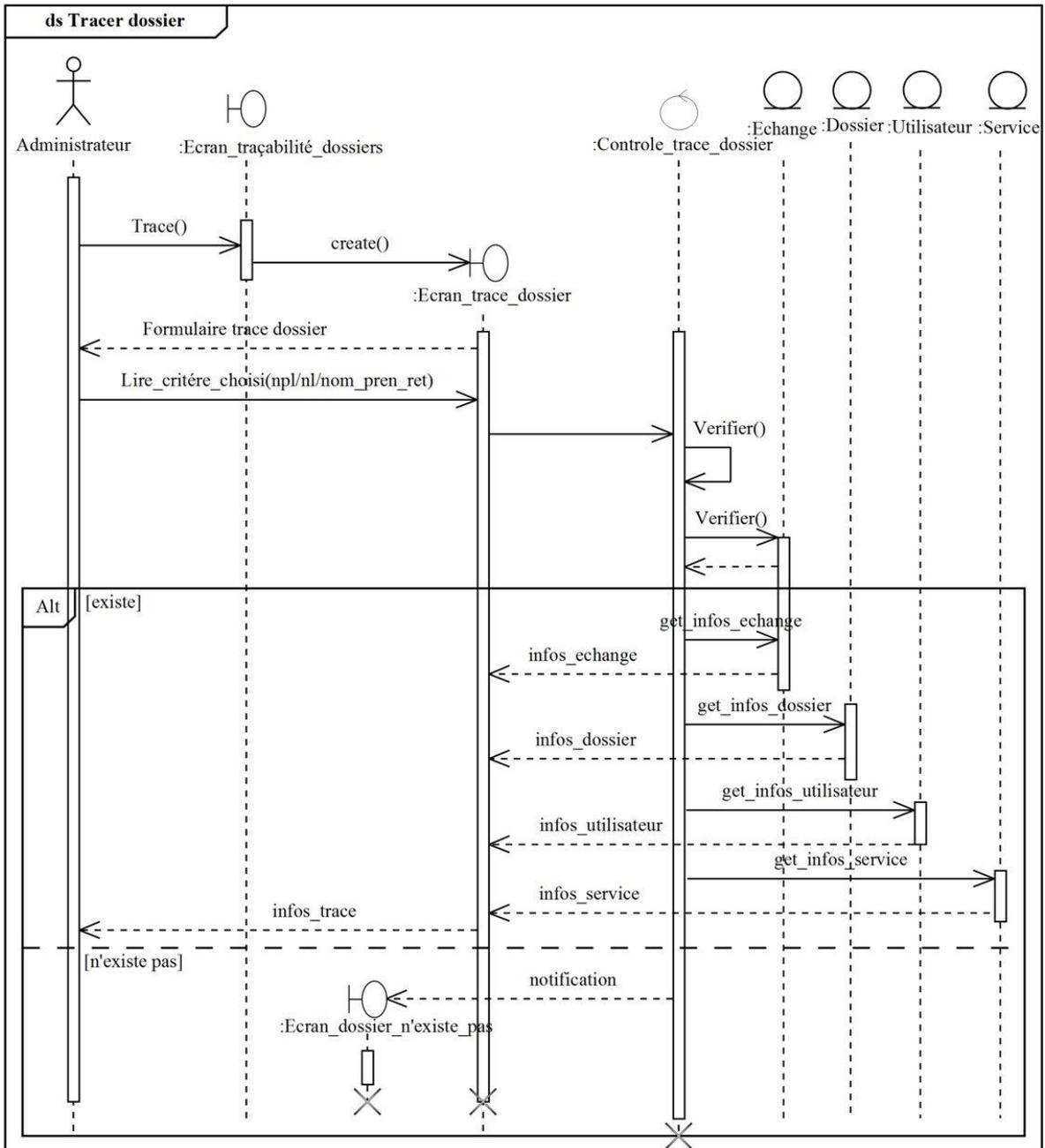


Figure 2.4.53: Diagramme d'interaction de cas d'utilisation « Tracer dossier »

1.1.6. Diagramme d'interaction de cas « Envoyer dossier »

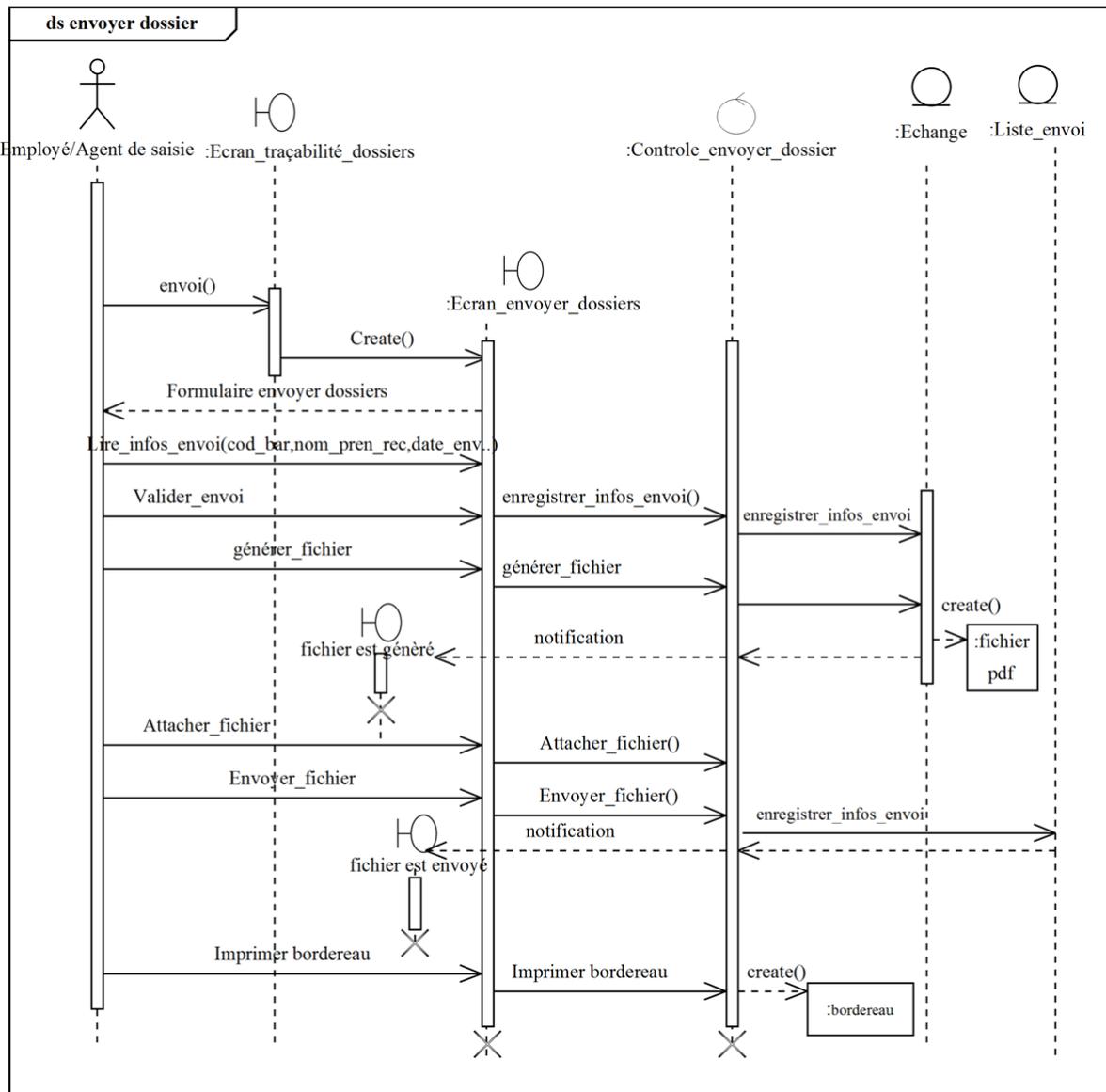


Figure 2.4.54 : Diagramme d'interaction de cas d'utilisation « Envoyer dossiers »

1.2. Diagramme d'état transition

Nous allons maintenant présenter le cycle de vie d'un objet. Nous spécifions les états possibles d'une classe et leurs enchaînement grâce au diagramme d'état transition.

Nous allons spécifier le cycle de vie de classe « dossier » parce qu'il est le plus important dans notre système.

➤ Diagramme d'état transition de classe « dossier »

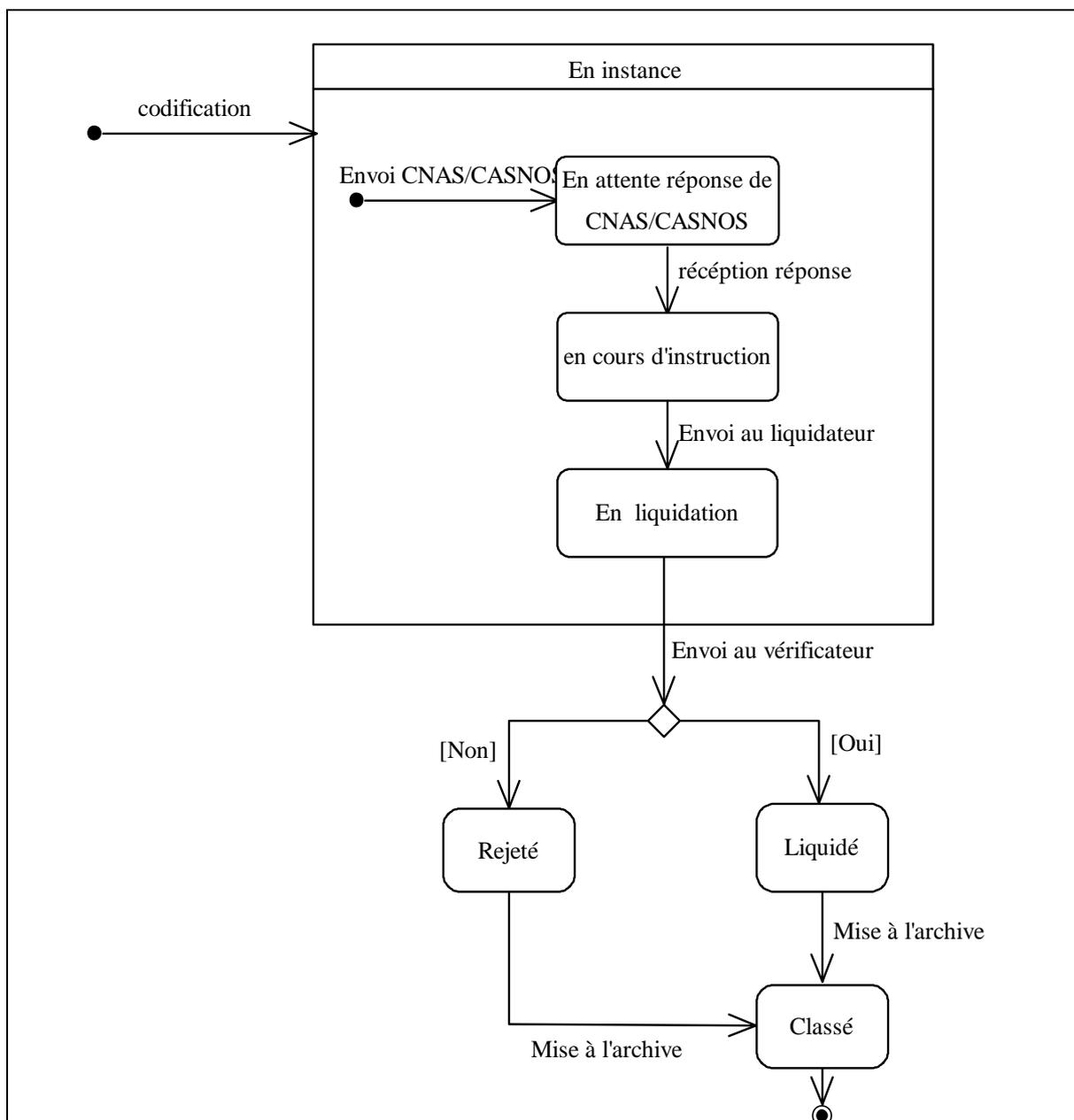


Figure 2.4.55 : Diagramme d'état transition « dossier »

Conclusion

Au cours de ce chapitre, nous avons présenté l'étape d'analyse qui nous a permis de passer d'une structuration fonctionnelle via les cas d'utilisations et les packages à une structuration objet via les classes et les catégories, Dans le prochain chapitre nous allons suivre la conception préliminaire.

Chapitre 05 :

CONCEPTION PRELIMINAIRE

Introduction

La conception préliminaire est l'étape la plus délicate du processus 2TUP car elle en représente le cœur. C'est en effet à cette occasion que s'effectue la fusion des études fonctionnelles et techniques. En conséquence, cette étape permet de passer de l'analyse objet à la conception, c'est-à-dire adapter la conception aux spécifications fournies par l'analyse et intégrer les fonctions métier et applicatives du système dans l'architecture technique.

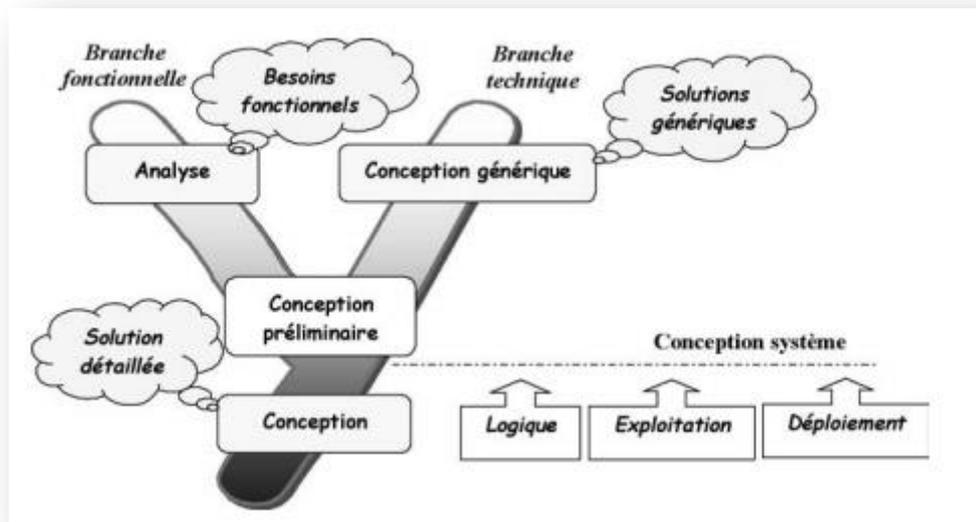


Figure 2.5.56 : Situation de la conception préliminaire dans 2TUP.

1. Développement modèle de déploiement

Le diagramme de déploiement permet de représenter l'architecture physique supportant l'exploitation du système, le déploiement d'une solution client/serveur se construit sur la définition des postes de travail.

1.1. Architecture adoptée

Le choix de notre solution s'est porté sur une architecture 2 tiers (Client/serveur) c'est à-dire plusieurs clients, connectés à un serveur (BDD, Application), et cela en se basant sur un réseau LAN au CNR. Ce choix repose sur les arguments suivants :

- L'ensemble des utilisateurs se trouve dans la même zone géographique.
- Le nombre restreint d'utilisateurs.
- Le matériel disponible n'est pas assez puissant pour supporter des architectures plus compliquées. Ainsi, l'acquisition d'un serveur puissant exclut le

risque de surcharge de réseau qui peut être engendré par le choix d'une architecture différente.

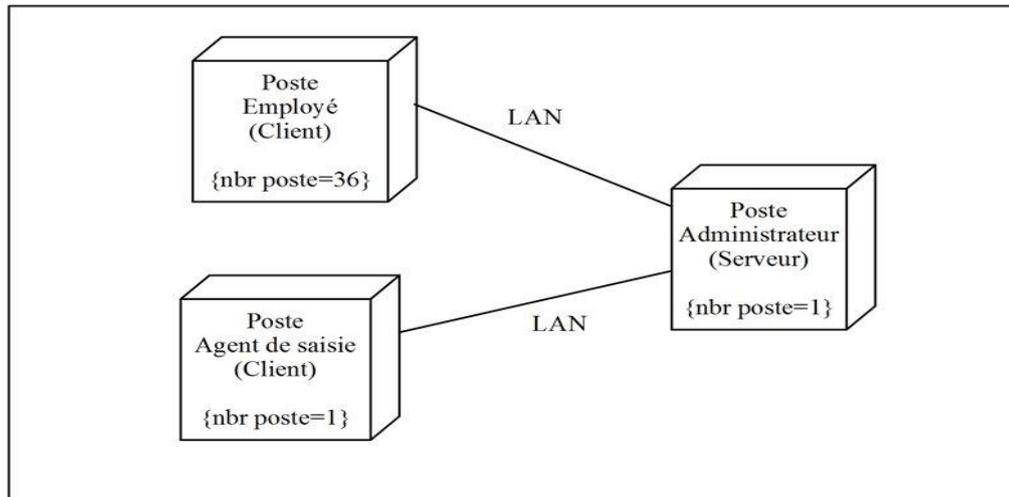


Figure 2.5.57 : Schéma du modèle de déploiement de notre système.

1.2. Déploiement du modèle d'exploitation

Par le biais du modèle d'exploitation de notre système, nous définissons les applications installées sur les postes de travail, les composants métier déployés sur les serveurs mais également les instances de base de données implantées sur les serveurs.

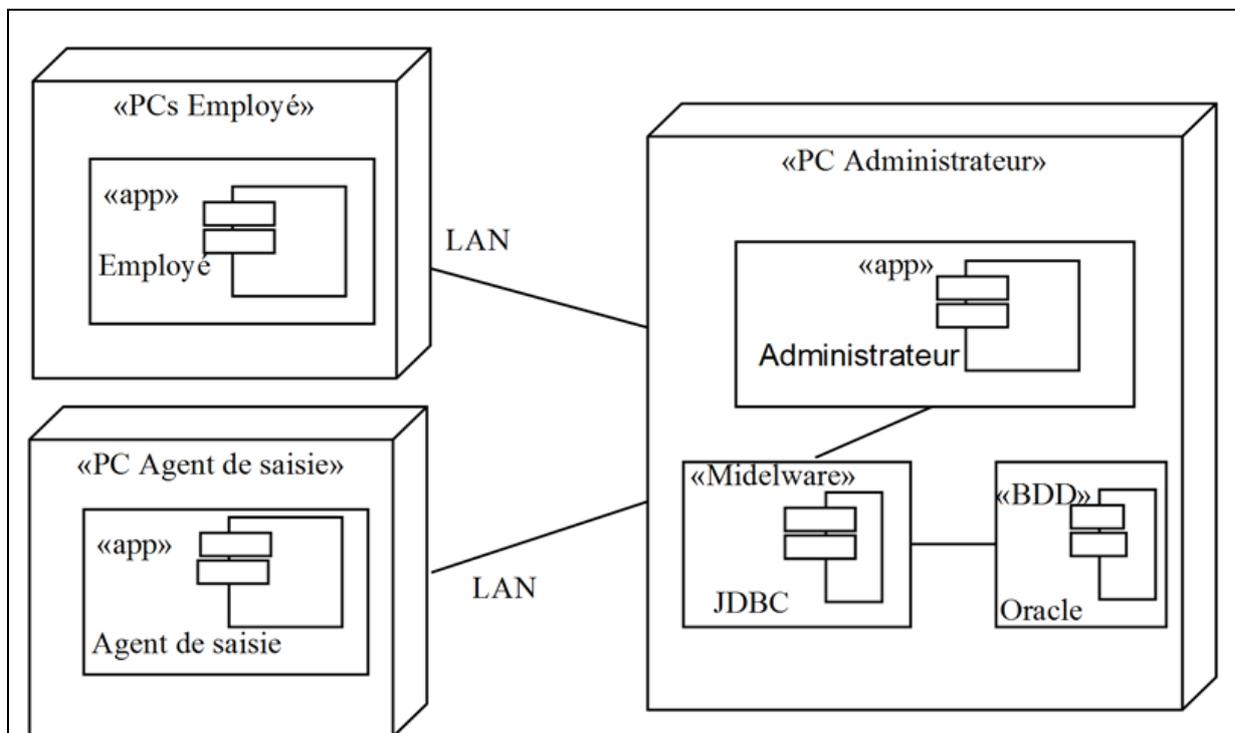


Figure 2.5.58 : Définition des applications dans le modèle d'exploitation.

2. Définition des interfaces

Espace	Interface	description
Administrateur	Gestion compte	Ajouter, modifier et supprimer un compte.
	Statistiques	Etablir statistique par employé ou par service.
	Trace dossier Localisation	Tracer dossier. Localiser dossier.
Agent de saisie	Dossier	Codifier dossier. Saisie alpha. Changer état.
	Echange	Envoyer liste des dossiers. Réceptionner liste des dossiers.
Employé	Echange	Envoyer liste des dossiers. Réceptionner liste des dossiers.
	Localisation dossier	Localiser dossier.
	Dossier	Changer état.

Tableau 2.3.22 : les interfaces de notre système

Conclusion

Au cœur de ce chapitre, nous avons présenté l'étape de la conception préliminaire qui nous a permis de décrire le diagramme de déploiement, d'exploitation, et définir les interfaces de notre système. Dans le prochain chapitre nous présentons la conception détaillée.

Chapitre 06 :

CONCEPTION DETAILLÉE

Introduction

La conception détaillée est la phase ultime de la modélisation avec UML. Après la modélisation des besoins puis l'organisation de la structure de la solution, la Conception détaillée vient construire et documenter précisément les classes, les interfaces, les tables et les méthodes qui constituent le codage de la solution.

1. Dictionnaire des données

1.1. Les classes et les attributs

Classe	Description	Code	Type	Taille
Dossier	Code barre	cod_bar	Chaîne de caractère	8
	Numéro pré liquidé	npl	Entier	6
	Numéro liquidé	nl	Entier	9
	Nom retraité	nom_ret	Chaîne de caractère	50
	Prénom retraité	prénom_ret	Chaîne de caractère	50
	Date de naissance	date_nais	Date	
	Type dossier	type_dos	Chaîne de caractère	50
	Date dépôt	date_depot	Date	
	Etat dossier	etat_doss	Chaîne de caractère	100
	Date enregistrement	date_enr	Date	
Utilisateur	Numéro d'utilisateur	num_util	Entier	4
	Nom d'utilisateur	nom_util	Chaîne de caractère	50
	Prénom d'utilisateur	pénom_util	Chaîne de caractère	50
	Username	username	Chaîne de caractère	50
	Rôle	role	Chaîne de caractère	50
	Mot de passe	mot_passe	Chaîne de caractère	50
	Poste	poste	Chaîne de caractère	50
Service	Numéro de service	num_serv	Entier	4
	Nom de service	nom_serv	Chaîne de caractère	50
Liste_envoi	Nom de fichier	nom_fich	Chaîne de caractère	50
	Date envoi	date_env	Date	

Tableau 2.3.23: Tableau des classes et des attribues.

1.2.les classes association

classes association	Description	Code	Type	Taille
Echange	Date d'envoi Heure d'envoi Date de réception	date_env heu_env date_réc	Date Chaine de caractère Date	5

Tableau 2.3.24: Tableau des classes association et des attribues.

1.3.Les opérations

Classe/Classe association	Opération	Description
Dossier	Codifier Localiser Tracer Etablir statistiques Saisir Alpha Changer l'état	Codifier un dossier Localiser un dossier Tracer un dossier Etablir statistiques des dossiers Saisir le numéro liquidé d'un dossier Changer l'état d'un dossier.
Utilisateur	Ajouter Supprimer Modifier	Ajouter un compte utilisateur Supprimer un compte utilisateur Modifier un compte utilisateur
Service		
Echange	Envoyer Réceptionner	Envoyer la liste des dossiers Réceptionner la liste des dossiers
Liste_envoi		

Tableau 2.3.25 : Tableau des opérations.

2. Diagramme de classe

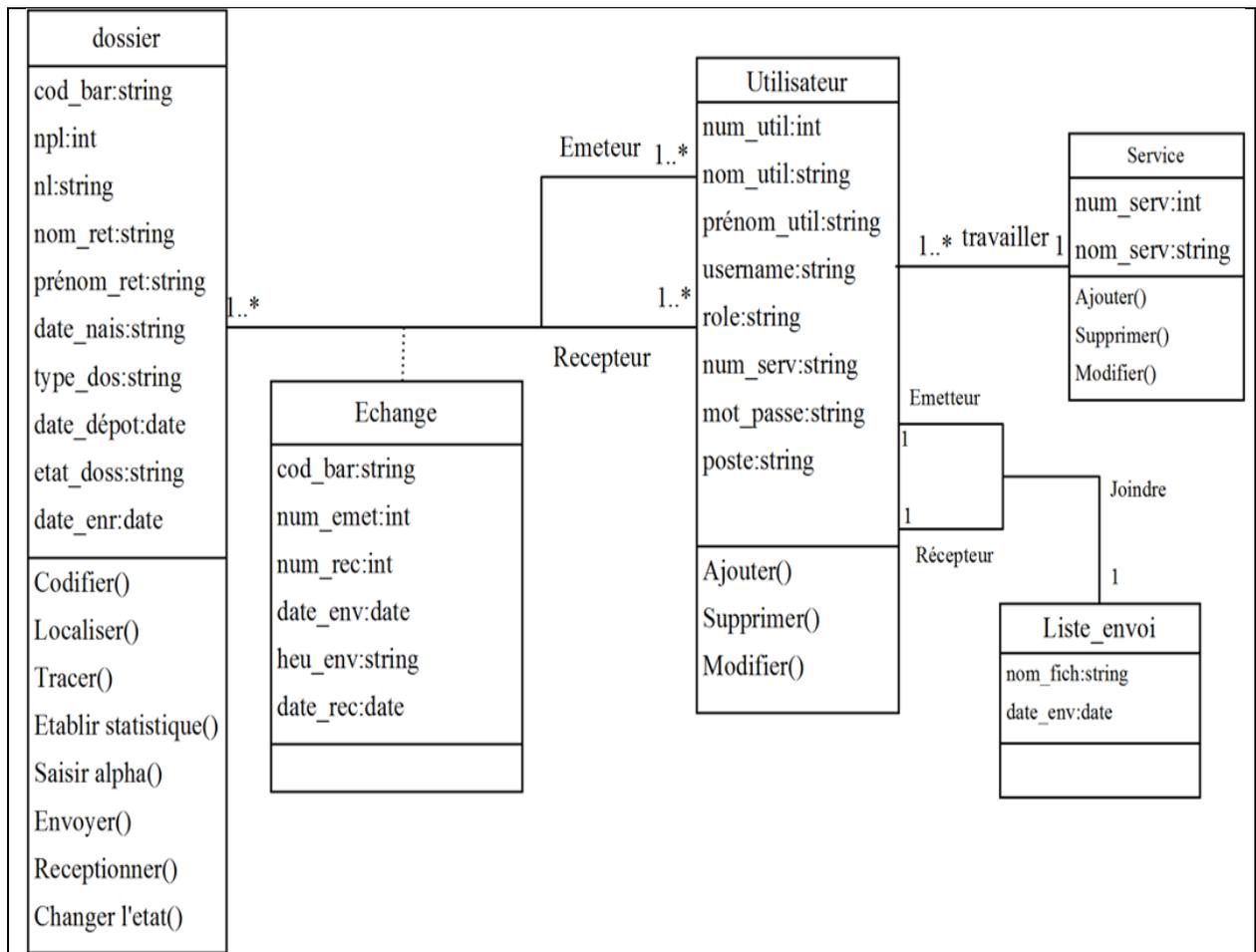


Figure 2.6.59 : Diagramme de classe

3. Passage vers le model relationnel

3.1. Les règles de passage

La réalisation du modèle relationnel peut être faite à partir de la description conceptuelle. Ce passage de modèle doit respecter un certain nombre de règle.

Règle1 : Toute entité devient une relation ayant pour clé primaire son identifiant. Chaque propriété se transforme en attribut.

Règle2 : Toute association hiérarchique (de type [1, n]) se traduit par une clé étrangère

Règle3 : Toute association non hiérarchique (de type [n, n] ou de dimension > 2) devient une relation.

Règle4 : Les entités n'ayant que leur identifiant comme attribut ne deviennent pas des relations, mais des attributs dans les autres relations liées.

Règle5 : cas de héritage, transformer chaque sous classe en une relation, la clé primaire de la super classe devient clé primaire de chaque sous classe.

Règle6 : cas de composition, la clé primaire de la classe composée devient clé étrangère de la classe composant.

Règle7 : cas d'agrégation, le même principe que la Règle 2.

3.2. Les tables de la base de données

En se basant sur les règles ci-dessus, nous avons converti les classes entités et leurs associations, à des tables dans la base données. Les tables générées sont :

<p>Dossier(+<u>cod_bar</u>, npl, nl, nom_ret, prénom_ret, date_nais, type_dos, date_dépot, état_doss, date_enr).</p> <p>Utilisateur(+<u>num_util</u>, nom_util, prénom_util, username, mot_passe, rôle, poste, #num_serv).</p> <p>Service(+<u>num_serv</u>, nom_serv).</p> <p>Echange(+<u>date_env</u>, +<u>#cod_bar</u>, +<u>#num_emet</u>, +<u>#num_rec</u>, heu_env, date_rec, heure_rec).</p> <p>Liste_envoi(+<u>nom_fich</u>, #num_emet, #num_rec, date_env)</p>
--

Tableau 2.3.26: les tables de la base de données

Conclusion

La conception détaillée consiste à concevoir 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. Le produit d'une conception détaillé consiste en l'obtention d'un model près à codé lorsque on utilise des langages orientés objet. Cette propriété facilite la compréhension des modèles de conception et donne encore plus d'intérêt à la réalisation d'une conception détaillé avec «UML».

Le chapitre suivant, qui est le dernier, explique les détails des choix d'implémentation de notre application.

Chapitre 07 :

DOSSIER TECHNIQUE

Introduction

Dans ce chapitre, nous présentons la partie réalisation et mise en œuvre de notre travail. Pour cela, nous présentons, en premier lieu, l'environnement de travail et les outils de développement utilisés. En second lieu, nous élaborons une présentation des différentes interfaces créées.

1. Environnement de travail

Pour implémenter notre système nous avons utilisé le langage de programmation Java avec l'IDE NetBeans. pour implémenter notre base de données nous avons utilisé le SGBD Oracle.

1.1. Le langage de programmation Java

Java est un langage de programmation évolué et orienté objet développé par Sun Micro Systems. il est très largement utilisé dans le domaine de développement notamment le développement d'application d'entreprises et mobiles.

Java possède un certain nombre des caractéristiques qu'ont contribué à son énorme succès:

- Java est portable: il est indépendant de toute plate-forme.
- Java est simple (pas de pointeur).
- Java est fortement typé (tout les variables sont typées).
- Java contient une très riche bibliothèque de classes.
- Java assure la gestion de la mémoire.

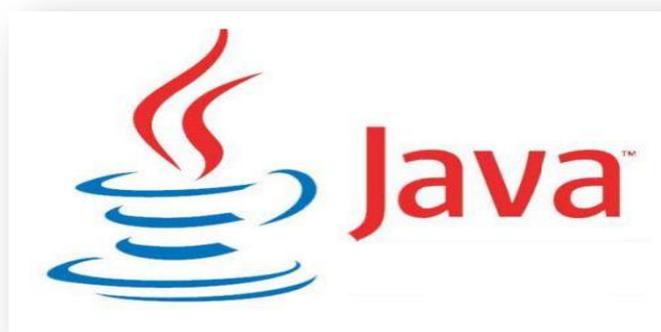


Figure 2.7.60: Logo du langage de programmation Java.

1.2.IDE NetBeans

Pour la sélection de l'environnement de développement, Notre choix s'est porté vers l'EDI NetBeans. C'est un environnement de développement intégré, open source, très utile qui permet le développement en java. Son grand atout est le confort et la Simplicité qui offre pour un développement propre et rapide.

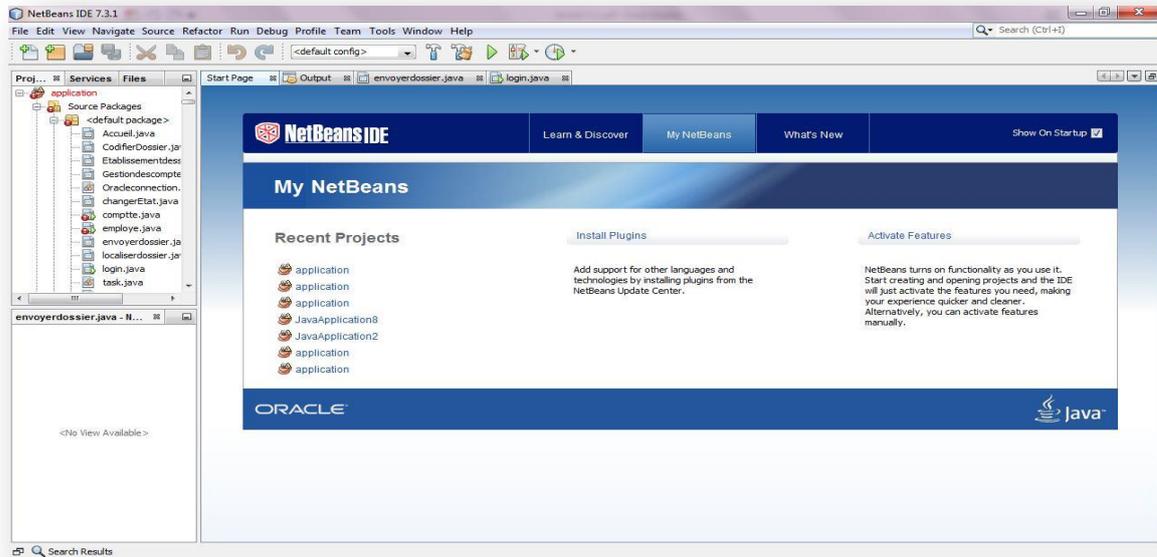


Figure 2.7.61: Capture écran de l'EDI NetBeans.

1.3.Implémentation de la base de données

Pour l'implémentation de la base de données que notre système utilise, nous avons sélectionné le SGBD Oracle, qu'est un système de gestion de bases de données édité par la société du même nom (Oracle Corporation), leader mondial des bases de données. Oracle se décline en plusieurs versions.

Oracle Server **Standard**, une version comprenant les outils les plus courants de la solution Oracle.

Oracle Server **Enterprise Edition**, Oracle est un SGBD permettant d'assurer :

- La définition et la manipulation des données.
- La cohérence des données.
- La confidentialité des données.
- L'intégrité des données.
- La sauvegarde et la restauration des données.
- La gestion des accès concurrents.

2. Quelques interfaces de notre système

2.1. Interface authentification



Figure 2.7.62: Capture fenêtre authentification.

2.2. Interface Accueil



Figure 2.7.63: Capture interface Accueil

2.3. Interface Codification

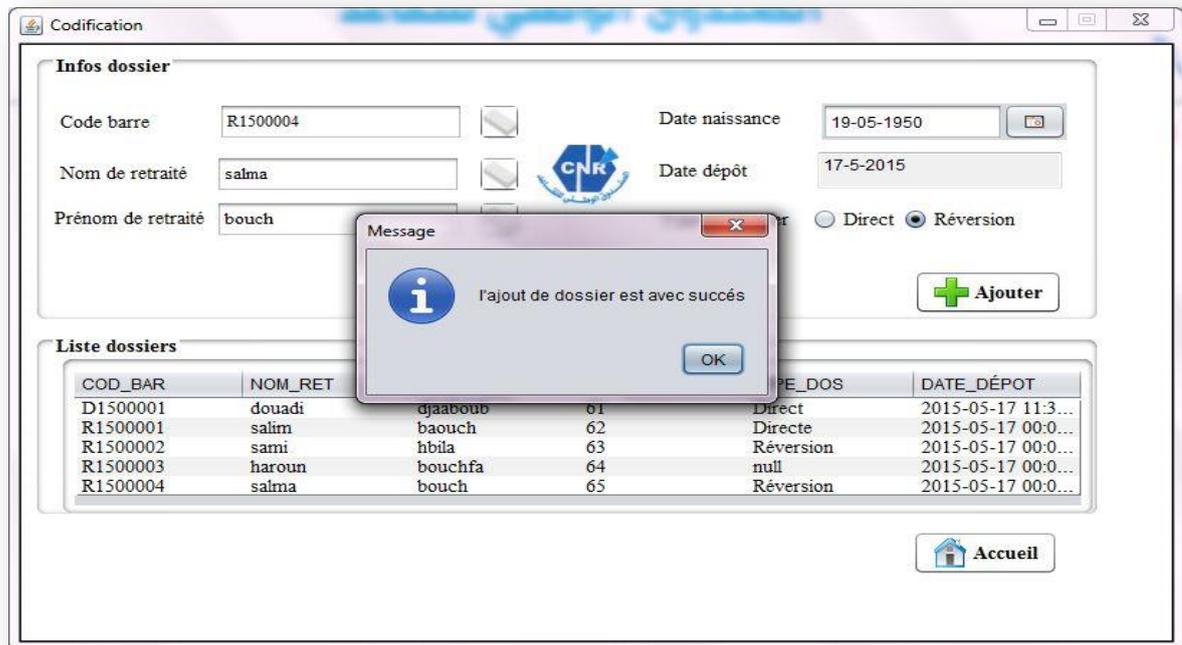


Figure 2.7.64 : Capture fenêtre codification.

2.4. Interface Envoi dossier

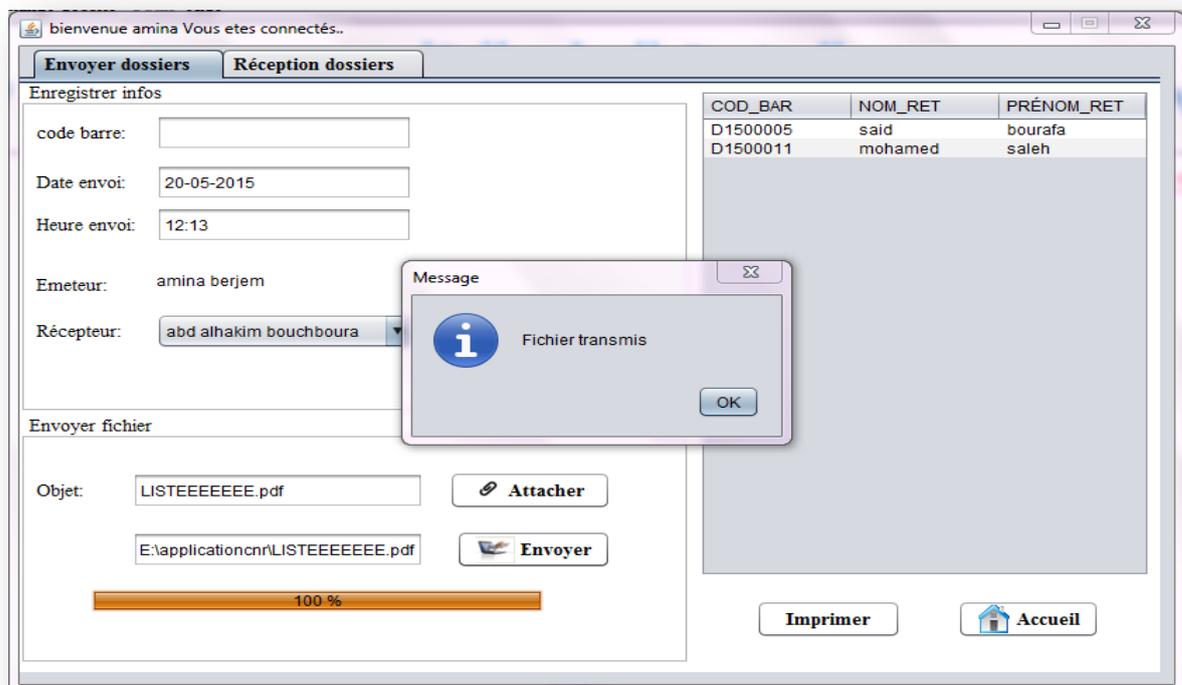


Figure 2.7.65: Capture fenêtre Envoi dossier

2.5. Interface Réception dossiers

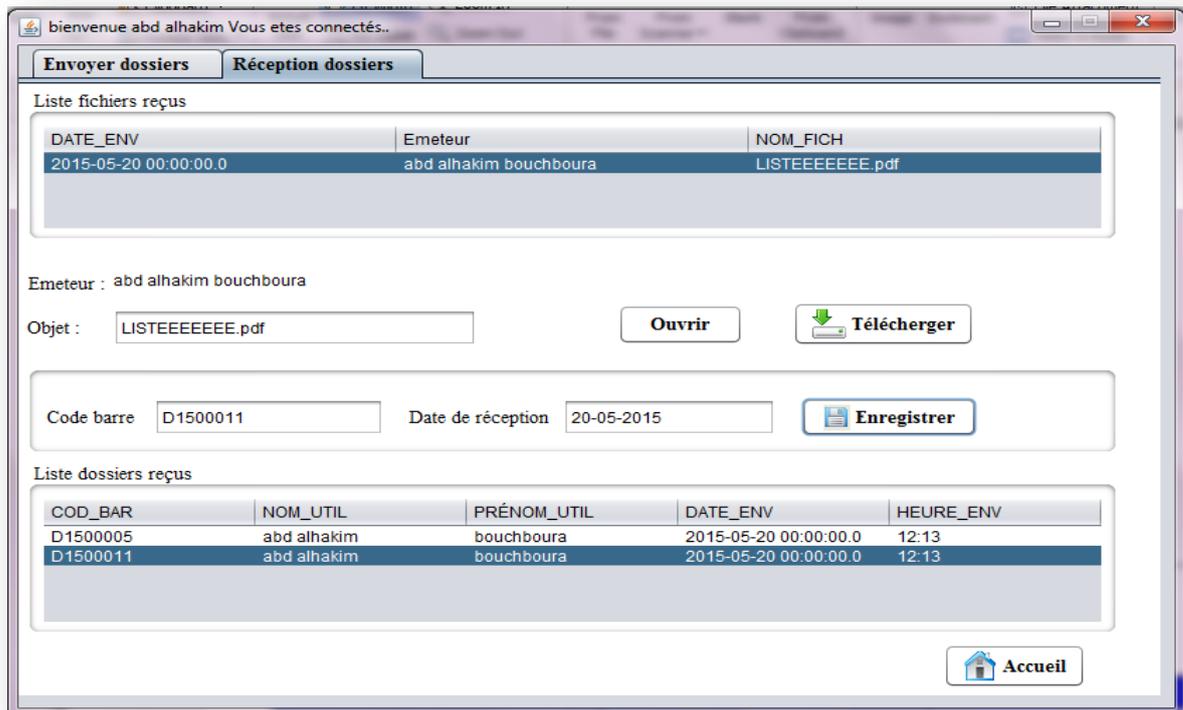


Figure 2.7.66: Capture fenêtre Réception dossiers

2.6. Interface Localisation dossier

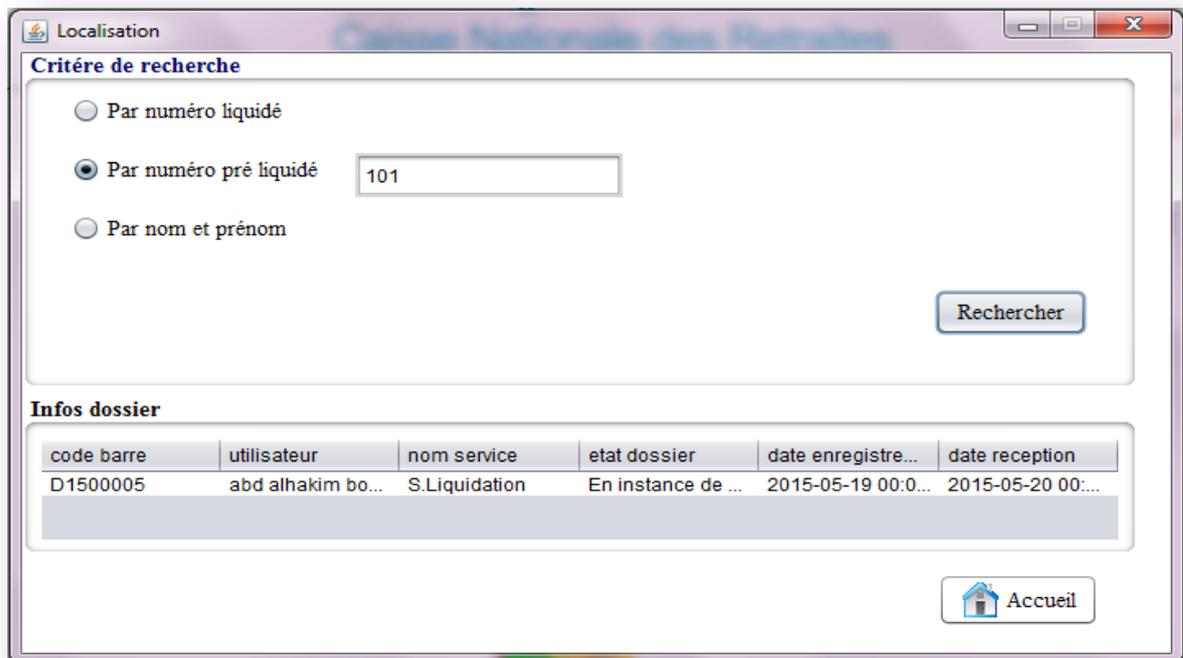
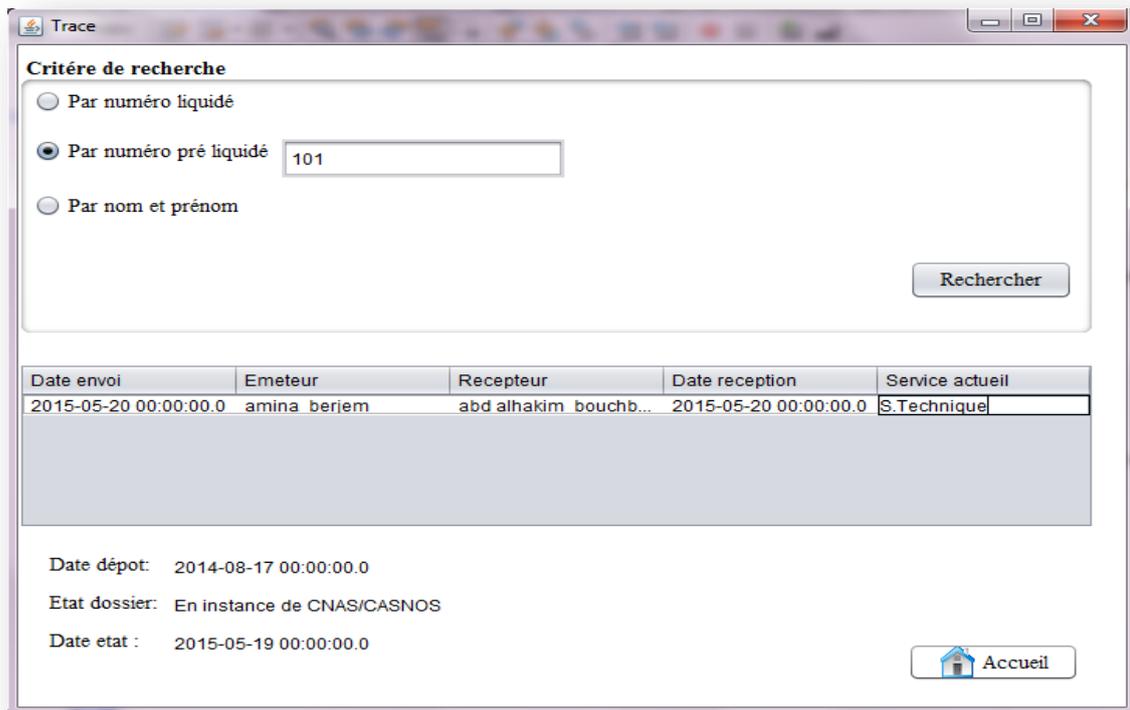


Figure 2.7.67: Capture fenêtre Localisation dossier

2.7.Interface Trace dossier



The screenshot shows a window titled 'Trace' with a search interface. Under 'Critère de recherche', the 'Par numéro pré liquidé' option is selected, and the value '101' is entered in the adjacent text box. A 'Rechercher' button is located to the right. Below the search area is a table with the following data:

Date envoi	Emeteur	Recepteur	Date reception	Service actuel
2015-05-20 00:00:00.0	amina beriem	abd alhakim bouchb...	2015-05-20 00:00:00.0	S.Technique

At the bottom of the window, there are three status fields: 'Date dépôt: 2014-08-17 00:00:00.0', 'Etat dossier: En instance de CNAS/CASNOS', and 'Date etat : 2015-05-19 00:00:00.0'. An 'Accueil' button with a home icon is positioned in the bottom right corner.

Figure 2.7.68: Capture fenêtre Trace dossier

Conclusion

Dans cette partie nous avons présenté l'environnement de développement et quelques interfaces du système. Cependant cette phase ne présente pas la fin du processus de développement. C'est au tour des deux phases test et validation.

Il faut remettre le système aux utilisateurs testeurs dans le but de détecter les éventuels bugs et anomalies pour les rectifier et assurer la stabilité et la fiabilité du système.

CONCLUSION

GENERALE

Conclusion générale

Nous avons tenté à travers ce projet de mettre en place une application client/serveur pour la mise en œuvre du suivi de la traçabilité des dossiers à travers les services de la Caisse Nationale de la Retraite(CNR) depuis le dépôt du dossier jusqu'au mandatement, notamment, la codification, la localisation, la trace, la gestion des flux des dossiers, etc. Ceci a permis de diagnostiquer rapidement les facteurs internes ou externes liés aux retards accusés lors du transit des dossiers à travers les services et d'y remédier.

Le projet s'est déroulé selon trois axes principaux afin de passer par les étapes essentielles de tout projet: l'analyse, la conception et la réalisation.

Pour la conception de notre application, nous avons eu recours à la méthode 2TUP. Cette approche nous a permis de bien comprendre la problématique et de bien modéliser les objectifs à atteindre. Ce qui nous a donné la possibilité de réaliser un système stable et évolutif.

Pour la réalisation, nous avons utilisé l'environnement NetBeans comme environnement de développement et ORACLE comme système de gestion de base de données.

En outre, ce projet nous a permis d'enrichir nos connaissances dans des domaines variés comme : L'orienté objet, UML, 2TUP, le langage JAVA,

D'autre part, ce projet a été une expérience réussie qui nous a permis de bénéficier d'un stage pratique. Ce stage a contribué à l'augmentation de nos connaissances dans le domaine de la recherche d'information. Il a également eu le rôle de nous rapprocher au monde du travail.

Références Bibliographiques

Ouvrage

[1].Pascal Roques-Franck Vallée ; UML 2 en action: De l'analyse des besoins à la conception.

[6]. JM Alliot ; Qu'est-ce que le "Middleware" ;13 mars 2003.

Sites webs

[2].2TUP de site <https://fr.scribd.com/doc/37222279/2TUP>

[4].http://bibfac.univ-tlemcen.dz/bibfs/opac_css/doc_num.php?explnum_id=802

[5].Composants-de-larchitecture-clientserveur de site <http://telechargercours.com/reseaux-telecoms/composants-de-larchitecture-clientserveur/>

[8].Reseaux-architecture-client-serveur-a-3-niveaux de site <http://www.commentcamarche.net/contents/221-reseaux-architecture-client-serveur-a-3-niveaux>

[9].WebServices_Intro de site http://cours-info.iut-bm.univ-fcomte.fr/upload/supports/LP/ProgMultiTiers/WebServices/WebServices_Intro.pdf

Mémoire de fin d'étude

[3]. Belkhouche ; Etude et Administration des Systèmes de Supervision dans un Réseau Local ; Université Abou Bakr Belkaid– Tlemcen ; 2010.

[7].F Besma,K Imane ; Conception d'une application client/serveur pour le vote en ligne ; Centre Universitaire de Mila ;2013.