



N° Réf :.....

Centre Universitaire
Abd elhafid boussouf 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 gestion des
programmes propres (SONELGAZ)**

Préparé par : Ilyes AZIZI.

Soutenue devant le jury

Président : M. Mohammed BOUBAKIR
Examineur : M. Abederrahim DIB
Encadrer : M. Ali LALOUCI

Grade : MAA, C.U.M
Grade : MAA, C.U.M
Grade : MAB, C.U.M

Année universitaire : 2014/2015

REMERCIEMENTS

*Je Remercie Le Bon Dieu Tout Puissant, de nous avoir donné le
Bon sens et grande volonté pour réaliser ce travail.*

*Je Remercie du fond du cœur **Mr**
LALOUCI ALI pour avoir accepté d'encadrer ce mémoire de fin d'étude, et
pour nous avoir guidés en tout s'sincérité.*

*Je Remercie tous Nos Enseignants de Centre Universitaire de Mila pour leur
contribution à notre formation scientifique.*

*Je remercie mon père et ma mère **Chaabane** et **Malika** et mes frères **Nasser** et
Hana et toute la famille qui me donnent tous les possibilités pour une bonne fin
d'étude.*

Je Remercie tous ce qui nous a aidés à réaliser ce travail.

DÉDICACES

Je dédie ce modeste travail :

Aux êtres les plus chers, mon père et ma mère, pour leurs exprimer tout le respect que j'ai pour eux et ainsi pour leurs témoigner ma reconnaissance pour tout les efforts et sacrifices qu'ils ont entrepris.

A mes frères et mes sœurs pour leur aide aussi.

A tous les membres de ma famille, oncle, tantes, cousins, cousines, voisins.

A Toute personne qui a contribué de près ou de loin a la réalisation de ce travail.

Ilyes

LISTE DES FIGURES

Figure1.1: le processus de développement en Y.....	9
Figure1.2 : Architecture à deux niveaux.....	13
Figure1.3 : Architecture à trois niveaux.....	13
Figure1.4: Catégories de réseaux informatiques.....	15
Figure1.5 : Société de Distribution d'Electricité et du Gaz de l'Est (SDE).....	19
Figure1.6: l'organigramme de la direction de la distribution.....	20
Figure 2.1: Le diagramme de contexte dynamique.....	28
Figure 3.1 : Diagramme de cas d'utilisation.....	35
Figure 3.2 : Diagramme de séquence « Proposer Affaire ».....	55
Figure 3.3 : Diagramme de séquence « Modifier Affaire ».....	56
Figure 3.4 : Diagramme de séquence « Supprimer Affaire ».....	56
Figure 3.5 : Diagramme de séquence « Valider Affaire ».....	57
Figure 3.6 : Diagramme de séquence « Etablir un avant projet».....	57
Figure 3.7 : Diagramme de séquence « Affecter élément».....	58
Figure 3.8 : Diagramme de séquence «Modifier Elément affecté».....	59
Figure 3.9 : Diagramme de séquence «Supprimer Elément affecté».....	60
Figure 3.10 : Diagramme de séquence «Ajouter Elément».....	61
Figure 3.11 : Diagramme de séquence «Modifier Elément».....	61
Figure 3.12 : Diagramme de séquence «Supprimer Elément».....	62
Figure 3.13: Diagramme de séquence «Consulter quantitatif».....	62
Figure 3.14: Diagramme de séquence «Ajouter Entreprise».....	63

Figure 3.15: Diagramme de séquence «Modifier Entreprise».....	64
Figure 3.16: Diagramme de séquence «Supprimer Entreprise».....	64
Figure 3.17 : Diagramme de séquence «Etablir Contrat».....	65
Figure 3.18 : Diagramme de séquence «Ajouter Contrat».....	66
Figure 3.19 : Diagramme de séquence «Modifier Contrat».....	66
Figure 3.20 : Diagramme de séquence «Supprimer Contrat».....	67
Figure 3.21 : Diagramme de séquence «Valider contrat».....	67
Figure 3.22 : Configuration matérielle du système de gestion des programmes propres.....	72
Figure 3.23: Modèle préliminaire des composants du SGPP.....	73
Figure3.24: Modèle de spécification logicielle.....	74
Figure 4.1 : Premier découpage en Catégories du SGPP.....	77
Figure 4.2 : Modèle structurel d'analyse du système SGPP.....	79
Figure 4.3 : Diagramme d'interaction « Proposer Affaire ».....	82
Figure 4.4 : Diagramme d'interaction « Modifier Affaire ».....	83
Figure 4.5 : Diagramme d'interaction « Supprimer Affaire ».....	84
Figure 4.6 : Diagramme d'interaction « Valider Affaire ».....	85
Figure 4.7 : Diagramme d'interaction «Etablir Avant projet ».....	86
Figure 4.8 : Diagramme d'interaction «Affecter Elément».....	87
Figure 4.9 : Diagramme d'interaction «Modifier Elément affecter».....	88
Figure 4.10 : Diagramme d'interaction «Supprimer Elément affecter».....	89
Figure 4.11 : Diagramme d'interaction «Ajouter Elément».....	90
Figure 4.12 : Diagramme d'interaction «Modifier Elément».....	91
Figure 4.13 : Diagramme d'interaction «Supprimer Elément».....	92
Figure 4.14 : Diagramme d'interaction «Consulter Quantitatif».....	93

Figure 4.15 : Diagramme d'interaction «Ajouter Entreprise».....	94
Figure 4.16 : Diagramme d'interaction «Modifier Entreprise».....	95
Figure 4.17 : Diagramme d'interaction «Supprimer Entreprise».....	96
Figure 4.18 : Diagramme d'interaction «Etablir Contrat».....	97
Figure 4.19 : Diagramme d'interaction «Ajouter Contrat».....	98
Figure 4.20 : Diagramme d'interaction «Modifier Contrat».....	99
Figure 4.21 : Diagramme d'interaction «Supprimer Contrat».....	100
Figure 4.22 : Diagramme d'interaction «Valider Contrat».....	101
 Figure 5.1 : Le modèle de déploiement du système.....	 103
 Figure 6.1 : Fenêtre « Authentification »	 113
Figure 6.2 : Fenêtre « Administrateur »	114
Figure 6.3 : Fenêtre « Chef district »	114
Figure 6.4 : Fenêtre « Ajouter Affaire »	115
Figure 6.5 : Fenêtre « Modifier Affaire».....	116
Figure 6.6 : Fenêtre « Supprimer Affaire »	117
Figure 6.7 : Fenêtre « Valider Affaire ».....	117
Figure 6.8 : Fenêtre « Etablir Quantitatif ».....	118

LISTE DES TABLEAUX

Tableau 2.1 : Légende des messages du diagramme de contexte dynamique.....	29
Tableau3.1 : la liste des acteurs et les messages par cas d'utilisation.....	34
Tableau 3.2 : Relation entre les cas d'utilisation.....	37
Tableau 3.3 : Description textuelle des cas d'utilisation fonctionnelle.....	55
Tableau 3.4: liste des classes candidates du SGPP.....	68
Tableau 3.5: Responsabilité des classes candidates.....	70
Tableau 3.6 : Diagrammes des classes participantes des cas d'utilisation.....	71
Tableau 3.7: Description textuelle des cas d'utilisation techniques.....	75
Tableau 4.1 : Diagramme de classes préliminaire par catégorie.....	78
Tableau 4.2 : Diagrammes de classes détaillées par catégorie.....	80
Tableau 4.3 : Diagrammes de classe globale.....	81
Tableau 5.1: Table des classes d'objet.....	103
Tableau 5.2: Typage des attributs des classes du système SGPP.....	105
Tableau 5.3:Traduction des associations du système SGPP.....	107
Tableau 5.4: les classes modifiées du système SGPP.....	108
Tableau 5.5: Table de la base de données.....	109

SOMMAIRE

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

Chapitre 1 : Notions introductives

1. Introduction	4
2. Méthodes de conceptions	4
2.1. Définition d'une méthode.....	4
2.2. Pourquoi une méthode	4
2.3. Classification des méthodes	5
2.4. Les méthodes orientées objets.....	5
2.5. Langage de modélisation UML.....	6
2.6. Le processus de développement en Y (2TUP).....	9
3. Réseaux	11
3.1. Pourquoi un réseau.....	11
3.2. Généralités sur les réseaux.....	11
3.3. Les modes de fonctionnements d'un réseau.....	12
3.3.1. Le mode client/serveur.....	12
3.3.2. Le mode égale à égale.....	14
3.4. Les modèles architecturaux des réseaux.....	14
3.5. Les réseaux locaux	15
4. Présentation de SONALGAZ.....	19
4.1. Société de Distribution d'Electricité et du Gaz de l'Est (SDE).....	19
4.2. La direction de la distribution de Mila	19
4.3. L'organigramme de la direction régionale de la distribution	20
5. Conclusion	20

Chapitre 2 : Etude préliminaire

1. Introduction	22
2. Elaboration du cahier des charges.....	22
2.1. Présentation de l'organisme d'accueil	22
2.2. Problématique	23
2.3. Objectifs du nouveau système	23

2.4. Grand choix techniques.....	24
2.5. Recueil des besoins fonctionnels.....	24
2.6. Recueil des besoins opérationnels.....	25
3. Description du contexte du système	26
3.1. Identification des acteurs.....	27
3.2. Identification des messages entre le système et ses acteurs.....	27
3.3. Diagramme de contexte dynamique.....	28
4. Conclusion	31

Chapitre 3 : Capture des besoins

1. Introduction	33
2. Capture des besoins fonctionnels	33
2.1. Liste des cas d'utilisations	33
2.2. Identification des acteurs principaux et secondaires.....	34
2.3. Elaboration de diagramme des cas d'utilisations.....	35
2.4. Organisation des cas d'utilisations.....	35
2.5. Description des cas d'utilisations.....	38
2.5.1. Description textuelle des cas d'utilisations.....	38
2.5.2. Description graphique des cas d'utilisations.....	55
2.6. Identification des classes candidates	68
2.6.1. La liste des classes candidates.....	68
2.6.2. Responsabilités des classes.....	68
2.6.3. Diagrammes de classes participantes.....	71
3. Capture des besoins techniques	72
3.1. Spécification technique du point de vue matériel	72
3.1.1. Style d'architecture utilisée	72
3.1.2. Configuration matériel du système SGPP.....	72
3.2. Spécification d'architecture	73
3.2.1. Style d'architecture en tiers	73
3.2.2. Modèle préliminaires des composants du système.....	73
3.3. Elaboration du modèle de spécification logicielle	73
3.3.1. Définitions.....	73
3.3.2. Modèle de spécification logicielle.....	74
3.3.3. Description textuelle des cas d'utilisation techniques	74
4. Conclusion	75

Chapitre 4 : Analyse

1. Introduction	77
2. Découpage en catégories	77
2.1. Les catégories.....	77
2.2. Élaboration des diagrammes de classes préliminaire par catégorie.....	78
2.3. Modèle structurel d'analyse	79
3. Développement du modèle statique	79
3.1. Diagrammes de classes détaillées par catégorie	80
3.2. Diagrammes de classe globale.....	81
4. Développement du modèle dynamique.....	81
4.1. Diagrammes d'interaction	81
4.1.1. Cas d'utilisation « Proposer Affaire ».....	82
4.1.2. Cas d'utilisation « Modifier Affaire ».....	80
4.1.3. Cas d'utilisation « Supprimer Affaire ».....	84
4.1.4. Cas d'utilisation « Valider Affaire ».....	85
4.1.5. Cas d'utilisation « Etablir Avant projet »	86
4.1.6. Cas d'utilisation « Affecter Elément ».....	87
4.1.7. Cas d'utilisation « Modifier Elément affecté ».....	88
4.1.8. Cas d'utilisation « Supprimer Elément affecté ».....	89
4.1.9. Cas d'utilisation «Ajouter Elément».....	90
4.1.10. Cas d'utilisation «Modifier Elément».....	91
4.1.11. Cas d'utilisation «Supprimer Elément».....	92
4.1.12. Cas d'utilisation « Consulter quantitatif».....	93
4.1.13. Cas d'utilisation « Ajouter Entreprise».....	94
4.1.14. Cas d'utilisation « Modifier Entreprise».....	95
4.1.15. Cas d'utilisation «Supprimer Entreprise».....	96
4.1.16. Cas d'utilisation «Etablir Contrat».....	97
4.1.17. Cas d'utilisation «Ajouter Contrat».....	98
4.1.18. Cas d'utilisation «Modifier Contrat».....	99
4.1.19. Cas d'utilisation «Supprimer Contrat».....	100
4.1.20. Cas d'utilisation «Valider Contrat».....	101

5. Conclusion	101
----------------------------	------------

Chapitre 5: Conception

1. Introduction	103
2. Conception préliminaire	103
2.1. Développement du modèle de déploiement	103
3. Conception détaillée	104
3.1. Typage des attributs	104
3.2. Règle de transformation	106
3.3. Traduction des associations	106
3.4. Les classes modifiées	107
3.5. Passage du modèle objet au modèle relationnel	108
3.6. Modèle relationnel du système SGPP.....	109
4. Conclusion	109

Chapitre 6: Réalisation

1. Introduction	111
2. Présentation de l'environnement de programmation.....	111
2.1. Langage de programmation java	111
2.2. NetBeans.....	111
2.3. MySQL.....	112
3. La présentation de l'IHM.....	112
3.1. Le guidage.....	112
3.2. charge de travail	112
3.3. Adaptabilité.....	113
3.4. Gestion des erreurs.....	113
3.5. homogénéité et cohérence	113
4. Quelques interfaces de l'application.....	113
4.1. Fenêtre d'authentification.....	113
4.2. Fenêtre d'administrateur.....	114
4.3. Fenêtre de chef district.....	114
4.4. Fenêtre d'ajouter affaire.....	115

4.5. Fenêtre de modifier affaire.....	111
4.6. Fenêtre de supprimer affaire.....	112
4.7. Fenêtre de valider affaire.....	112
4.8. Fenêtre d'établir quantitatif.....	113
5. Conclusion	113
Conclusion général	114

INTRODUCTION GÉNÉRALE

INTRODUCTION GÉNÉRALE

L'informatisation de l'entreprise est devenue indispensable et d'un intérêt absolu, elle n'a cessé de gagner places depuis les années 60; au premier temps elle s'est limitée aux tâches administratives et gestionnaires, puis elle s'est étendue graduellement à toutes les facettes de l'activité organisationnelle.

A ce titre, la direction de distribution (**SONELGAZ**) de MILA est une société à caractère commerciale et industriel qui possède de grands réseaux électrique et gazier, et dans le but de les maintenir dans un bon état, elle prépare chaque année un programme travaux pour être réalisé l'année suivante. L'élaboration de ce dernier est une tâche très difficile. Malgré l'existence d'un réseau local au niveau de la direction, le travail se fait manuellement, absence d'un logiciel unifié, et absence de parallélisme sachant que la préparation de ce programme s'effectue au niveau des unités opérationnelles (districts) distribuées géographiquement sur le territoire de la wilaya.

Dans ce contexte, on se propose de construire un système d'information sous réseau local pour avoir une meilleure gestion des informations, minimiser au maximum le temps et l'effort de remplissage des documents, automatiser tout les calculs de formule utilisée pour le coût de réalisation, faciliter la circulation des informations entre les différents services, améliorer les conditions de travail et assurer la sécurité des informations.

La première tâche à réaliser dans une démarche de construction logicielle est l'identification de système à construire. Tout au long de l'élaboration du produit (logiciel), l'essentiel est de garder à l'esprit ses objectifs, et de bâtir un produit cohérent et évolutif à ses besoins

Le développement de ce produit nécessite une méthode de conception. Dans le monde technique et industriel, les méthodes orientées objets trouvent un premier niveau d'application dans le domaine de l'informatique de gestion, l'événement marquant dans ce domaine a été la consécration d'UML en tant que norme du langage de modélisation objet. Afin d'atteindre notre objectif, les orientations techniques qui nous ont été faites sont :

- Sur le plan méthodologique :
 - ✓ Le langage de modélisation orienté objet UML (avec la démarche 2TUP) pour la conception.
- Sur le plan de données :
 - ✓ Le SGBDR : MYSQL pour l'implémentation de la base de données.
- Sur le plan développement :
 - ✓ Langage Java pour la réalisation du logiciel.

Notre mémoire est organisé comme suit :

Chapitre 1 : une présentation des notions générales que nous allons utiliser pour le développement de notre système. Puis une présentation de l'organisme d'accueil qui englobe une description générale de l'entreprise ainsi que le champ d'étude.

Chapitre 2 : présente le sujet de notre étude, et introduit la modélisation de son contexte.

Chapitre 3 : il est consacré à la capture des besoins qui comportent deux étapes : la capture des besoins fonctionnels et celles des besoins techniques.

Chapitre 4 : définit l'analyse de notre système en le découpant en catégories jusqu'à l'arrivée au modèle dynamique.

Chapitre 5 : il définit la conception qui décrit le passage du modèle à objet au modèle relationnel, ce passage est justifié par le SGBD à utiliser.

Chapitre 6 : c'est l'étape de réalisation du système qui présente l'environnement de développement, Avec quelques interfaces graphiques de l'application.

Enfin, nous terminons par une conclusion générale. Ainsi a la fin du document, nous présentons :

- ✓ La bibliographie : qui mentionne les noms des ouvrages et mémoires.
- ✓ Le web graphie : qui mentionne les sites Web utilisés à la réalisation de notre projet.

CHAPITRE 1

NOTION INTRODUCTIFS

1. Introduction

Le développement des systèmes d'information est une activité qui doit être menée selon et suivant une méthode. Dans le monde technique et industriel, les méthodes orientées objets trouvent un premier niveau d'application dans le domaine de l'informatique de gestion, l'événement marquant dans ce domaine a été la consécration d'UML en tant que norme du langage de modélisation objet que nous utiliserons comme un langage de modélisation pour développer notre système.

Comme la notion de base de données est généralement couplée à celle de réseau afin de pouvoir mettre en communication les informations qu'elle regroupe, et qu'une entreprise doit assurer la communication entre ses services, elle a besoin d'un réseau local d'entreprise.

Dans ce chapitre, on va présenter quelques notions des méthodes de conceptions, UML et le processus 2TUP, quelques notions et généralités sur les réseaux.

2. Méthodes de conceptions

2.1. Définition d'une méthode

- **Littérairement** : Une méthode signifie une manière de faire ou une démarche ordonnée et raisonnée.
- **Dans le domaine des systèmes d'information** : Une méthode est un ensemble de règles et de directives, elle est composée de :
 - ✓ Un ensemble de concepts de modélisation pour capturer la sémantique du problème.
 - ✓ Un ensemble de vues et de notations pour présenter la modélisation.
 - ✓ Un processus interactif, pas à pas, pour la construction des modèles.
 - ✓ Une collection de règles et des suggestions qui conduisent à l'exécution du développement.

2.2. Pourquoi une méthode ?

Une méthode de développement de système d'information guide et encadre le développement en permettant de :

- ✓ Maîtriser le coût (budget).
- ✓ Maîtriser le temps.
- ✓ Satisfaire les besoins de l'utilisateur.
- ✓ Partager les responsabilités des personnes impliquées dans le développement.

- ✓ Maîtriser les risques liés au projet de développement.

2.3. Classification des méthodes

➤ Chronologiquement :

- ✓ **Les méthodes d'analyse** (années 60) :
Ont marqué le début de l'industrialisation de l'informatique.
- ✓ **Les méthodes cartésiennes** (années 60) :
Elles sont associées à une approche basée sur les fonctions.
- ✓ **Les méthodes systématiques** (années 80) :
Ont marqué la rupture avec les méthodes cartésiennes basées sur une approche conceptuelle globale du système d'information sur la recherche d'éléments pertinents.
- ✓ **Les méthodes objets** (années 80-90) :
Sont la synthèse des approches systématiques et cartésiennes.

2.4. Les méthodes orientées objets

Les concepts de l'orienté-objet ont débuté avec la programmation en fin des années 70 (smalltalk). En suite, ils ont été utilisés dans le domaine du développement des logiciels et de système en général.

➤ les motivations de cette utilisation :

- ✓ **Réutilisation** : On conçoit un ensemble des composants qui seront réutilisés dans plusieurs situations (gain d'effort et de temps).
- ✓ **Confrontation données/traitements** : la notion d'objet permet de voir les aspects des données et les aspects des traitements en même temps.

➤ Exemples de méthodes orientée objets :

Durant les années 90, on dénombrait déjà plus de 50 méthodes orienté-objet, Parmi les plus distinguées :

- ✓ OMT (91) —————→ rambaugh.
- ✓ OOA/OOD (291/94) —————→ cood/ yourdon.
- ✓ OOD (91/95) —————→ booch.

En 1996 l'UML est sorti, suite à la fusion de 3 méthodes (OMT, OOD, OOSE). Depuis 1997 UML est prôné par l'objet management group (OMG).

➤ **Le développement orienté objet suit 3 grandes phases:**

- ✓ **L'analyse orienté-Objet:** il s'agit d'abstraire les objets de la vilités en utilisant des modèles conceptuelles.
- ✓ **La conception orienté-Objet:** il s'agit d'organiser la solution informatique avant de passer à l'implantation.
- ✓ **L'implantation orienté-objet:** il s'agit de mettre en ouvre le logiciel en terme de programmes et de base de données avec la solution matérielle qui le supporte.

2.5. Langage de modélisation UML : [1]

UML (Unified Modeling Langage. Traduction = "langage de modélisation objet unifié") est né de la fusion des trois méthodes qui ont le plus influencé la modélisation objet au milieu des années 90 :

- ✓ **OMT:** Object Modeling Technique.
- ✓ **BOOCH:** nom tiré de son inventeur (grady Booch).
- ✓ **OOSE:** Object Oriented Software Engineering.

Issu "du terrain" et fruit d'un travail d'experts reconnus, UML est le résultat d'un large consensus. De très nombreux acteurs industriels de Renom qui ont adopté UML et qui ont participé à son développement.

UML comble une lacune importante des technologies objet, indépendamment de tout langage de programmation. Il a été conçu pour servir de support à une analyse basée sur les concepts objet.

➤ **Les caractéristiques d'UML:**

- ✓ **UML** n'est pas une méthode. C'est un langage graphique qui permet de représenter les divers aspects d'un système.
- ✓ **UML** un cadre d'analyse orienté objet : offre un ensemble de diagrammes qui reflètent un aspect particulier du système.
- ✓ **UML** possède plusieurs facettes, c'est une norme, un langage de modélisation objet, un support de communication, un cadre méthodologique. UML est tout cela à la fois, ce qui semble d'ailleurs engendrer quelques confusions. [W1]
- ✓ **UML** comble une lacune importante des technologies objet. Elle permet d'exprimer et d'élaborer des modèles objet, indépendamment de tout langage de programmation, Elle a été pensée pour servir de support à une analyse basée sur les concepts objet. [W1]

- ✓ Une autre caractéristique importante d'**UML**, est qu'il est cadre d'analyse. **UML** permet de représenter un système selon différentes vues complémentaires : les diagrammes. [W1]
- ✓ Une caractéristique importante des diagrammes d'**UML**, est qu'ils supportent l'abstraction. Cela permet mieux contrôler la complexité dans l'expression et l'élaboration des solutions objet.
- ✓ **UML** propose des diagrammes pour la communication entre les différents acteurs d'un objet, qui vont permettre d'éclaircir les spécifications. Il définit 9 diagrammes pour représenter les différents points de vue de modélisation.

➤ **Les diagrammes d'UML:**

- ✓ **Le diagramme fonctionnel:**
 - **Diagramme de cas d'utilisation:** c'est une représentation des fonctions de système du point de vue de l'utilisateur.
- ✓ **Les diagrammes statiques:**
 - **Diagramme de classe:** c'est une représentation de la structure statique en termes de classes et de relations.
 - **Diagramme d'objet:** c'est une représentation des objets et leurs relations, correspondant à un diagramme de collaboration simplifié, sans représenter des envois de message.
 - **Diagramme de composants:** ce diagramme montre l'organisation et les dépendances d'une série des composants.
 - **Diagramme de déploiement:** c'est une représentation du déploiement des composants sur les dispositifs matériels.
 - **Diagramme de paquetage:** c'est une représentation de l'organisation logique du modèle et des relations entre les packages.
 - **Diagramme de structure composite :** c'est une représentation de l'organisation interne d'un élément statique complexe.
- ✓ **Les diagrammes dynamiques:**
 - **Diagramme de séquence (d'interaction):** c'est une représentation temporelle des objets et leurs interactions.
 - **Diagramme de collaboration (de communication):** représente les objets, leur lien et leur interaction.

- **Diagramme d'état/transition:** c'est une représentation du comportement d'une classe en termes d'état.
- **Diagramme d'activités:** c'est une représentation du comportement d'une opération en termes d'action.
- **Diagramme globale d'interaction:** représente la fusion des diagrammes d'activités et de séquences pour combiner des fragments d'interaction avec des décisions et des flots.
- **Diagramme de temps:** représente la fusion des diagrammes d'état et de séquence pour montrer l'évolution de l'état d'un objet au cours du temps.

➤ **Les points forts d'UML:**

- ✓ UML est avant tout un support de communication performant, qui facilite la représentation et la compréhension des solutions objet.
- ✓ Sa notation graphique permet d'exprimer visuellement une solution objet, ce qui facilite la comparaison et l'évaluation des solutions.
- ✓ L'aspect formel de sa notation, limite les ambiguïtés et les incompréhensions.
- ✓ Son indépendance par rapport aux langages de programmation, aux domaines d'application et aux processus, ont fait un langage universel.
- ✓ La notation graphique d'UML n'est que le support du langage. La véritable force d'UML, c'est qu'il repose sur un méta modèle. En d'autres termes: la puissance et l'intérêt d'UML, c'est qu'il normalise la sémantique des concepts qu'il véhicule.
- ✓ UML est, donc, bien plus qu'un simple outil qui permet de "dessiner" des représentations mentales...Il permet de parler un langage commun, normalisé mais accessible, car visuel. Il représente un juste milieu entre le langage mathématique et naturel.
- ✓ Pas trop complexe mais suffisamment rigoureux, car basé sur un méta modèle.

➤ **Les points faibles d'UML:**

- ✓ Sans contestation la lourdeur de sa mise en place au sein de n'importe quel processus. son apprentissage assez long et rigoureux peut également freiner son utilisation.
- ✓ Le processus de conception n'est pas couvert par UML. UML ne propose pas de méthodologie.

2.6. Le processus de développement en Y (2TUP) :

Afin de maîtriser les risques, nous allons utiliser une approche itérative et incrémentale fondée sur le processus 2TUP.

➤ Description du processus 2TUP:

2TUP signifie "2 Traks Unifed Process" C'est un processus unifié qui apporte une réponse aux contraintes, de changement continu, imposées aux systèmes d'information des entreprises.

L'axiome fondateur du 2TUP a fait constater que toute évolution imposée au système d'information peut décomposer et traiter pareillement suivant un axe fonctionnel et un axe technique. La réalisation du système consiste à fusionner les résultats de ses deux branches du processus, cette fusion conduit à l'obtention d'un processus de développement en forme de Y, comme l'illustre la figure suivante :

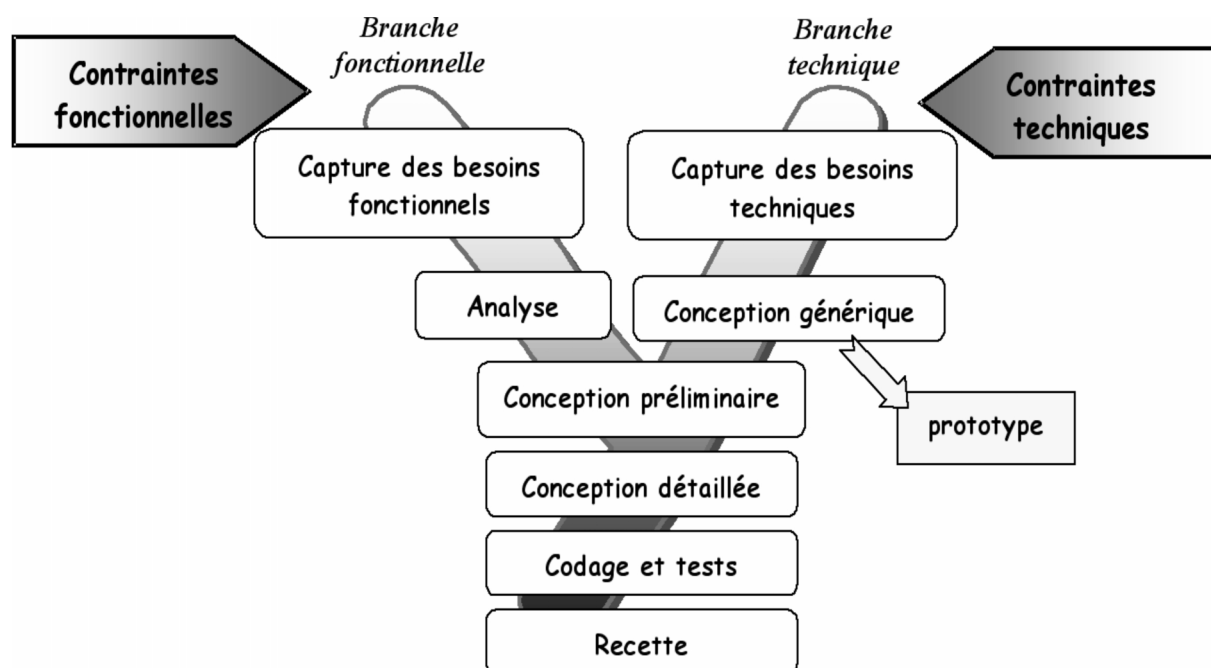


Figure1.1: le processus de développement en Y. [2]

- ✓ **La branche gauche (fonctionnelle)** comporte :
 - **La capture des besoins fonctionnels**, qui produit un 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 la manière à obtenir ce que va réaliser le système en termes de métier. Les résultats de l'analyse ne dépendent d'aucune technologie particulière.
- ✓ **La branche droite (architecture technique)** comporte :
 - **La capture des besoins techniques**, qui recense toutes les contraintes et les choix dimensionnant la conception du système. Les outils et les matériels sélectionnés ainsi que la prise en compte de contrainte d'intégration avec l'existant, conditionnent généralement les pré-requis d'architectures techniques.
 - **La conception générique**, qui définit ensuite les composants nécessaires à la construction de l'architecture technique de manière que cette conception soit complètement indépendante des aspects fonctionnels, elle a pour objectif d'uniformiser et de réutiliser les mêmes mécanismes pour tout système.
- ✓ **La branche du milieu** comporte :
 - **La conception préliminaire**, qui représente une étape délicate puisqu'elle intègre le modèle d'analyse dans l'architecture technique de manière à tracer la cartographie des composants du système à développer.
 - **La conception détaillée**, qui étudie, par la suite, comment réaliser chaque composant. Le 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 fonctions du système développé.
- **Les points forts du 2TUP:**
 - ✓ Itératif.
 - ✓ Fait une large place à la technologie et la gestion de risque.
 - ✓ Définit les profils des intervenants, les livrables, les plannings, les prototypes.
 - ✓ Améliore la compréhension du système.
 - ✓ Meilleur niveau de portabilité grâce à l'utilisation d'UML.
- **Les points faibles du 2TUP:**
 - ✓ Plutôt superficiel sur les phases situées en amont et en aval du développement: Capture des besoins, maintenance, gestion du changement.
 - ✓ Ne propose pas de document type.

3. Réseaux

3.1. Pourquoi un réseau ?

Les besoins de communication des données informatiques entre système plus au moins éloignés sont multiples [3] :

- Transmission des messages.
- Partage des ressources.
- Transfert des fichiers.
- Consultation des bases de données.

3.2. Généralités sur les réseaux

➤ **Terminologies :**

✓ **Les réseaux informatiques :**

Un réseau de transmission de données peut être défini comme l'ensemble des ressources liées à la transmission permettant l'échange des données entre les différents systèmes éloignés. Suivant leur organisation où architecture. Les distances, les vitesses de transmission et la nature des informations transmises. Les spécifications et normes utilisées sont différentes. [3]

✓ **Client d'un réseau :**

C'est un programme qui prend, le plus souvent, la forme d'un navigateur grâce auquel un utilisateur du réseau peut demander et consulter, très simplement, des documents. [4]

✓ **Serveur d'un réseau :**

C'est une station principale chargée d'un logiciel qui permet d'effectuer le traitement localement. Une fois ce traitement terminé, les résultats sont sauvegardés sur le serveur et/ou imprimés. A partir de cette station les données transitent vers les autres stations.

✓ **Protocole d'un réseau :**

Le dialogue entre un programme serveur et un programme client s'effectue selon des règles précises qui constituent un protocole. [4]

✓ **Ressources d'un réseau :**

Les ressources d'un réseau sont communes (disque, programmes, fichier) qui sont accessibles à chaque début et fin d'une session de travail, il y a, donc, un recours à ces ressources pour des volumes limitées, le travail sur des bases de données partagées nécessite un accès permanent à ceci.

3.3. Les modes de fonctionnement d'un réseau

Un réseau peut fonctionner selon deux modes : le mode client/serveur et le mode égale à égale.

3.3.1. Le mode client/serveur :

➤ Présentation de l'architecture d'un système client/serveur :

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des machines clientes (des machines faisant partie du réseau) contactent un serveur; une machine généralement très puissante en terme de capacité d'entrées-sorties, qui leurs fournissent des services.

➤ Fonctionnements d'un système client/serveur :

Un système client/serveur fonctionne comme suit :

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

➤ Types d'architecture client/serveur :

Selon le nombre de niveaux on peut distinguer :

➤ L'architecture à deux niveaux :

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

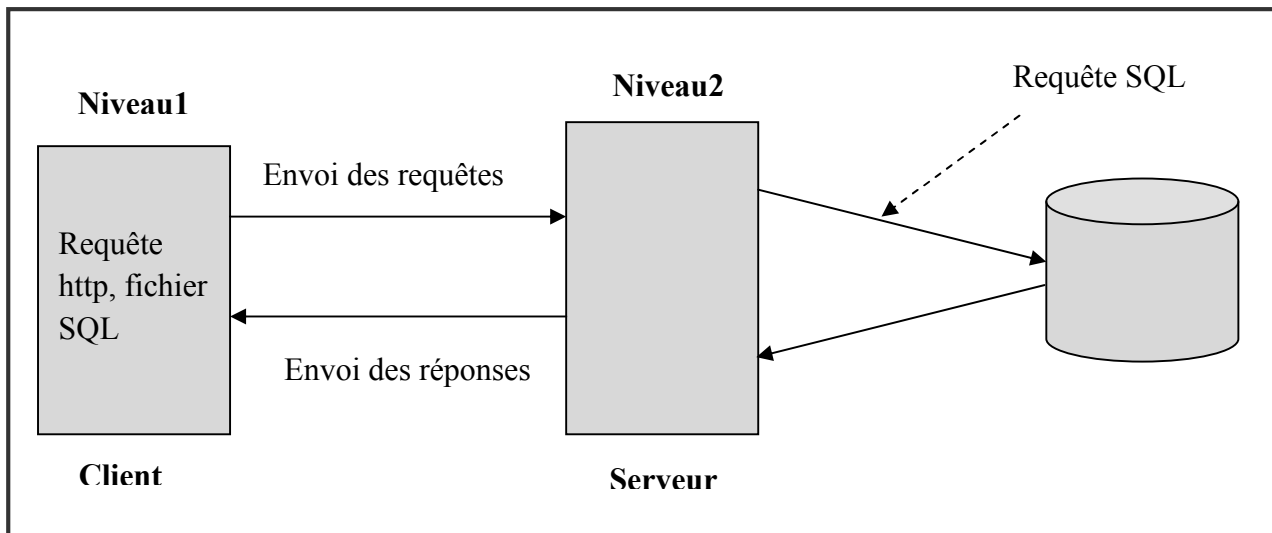


Figure 1.2 : Architecture à deux niveaux.

➤ **L'architecture à 3 niveaux :**

L'architecture à 3 niveaux (appelée architecture 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 service.
- ✓ 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).

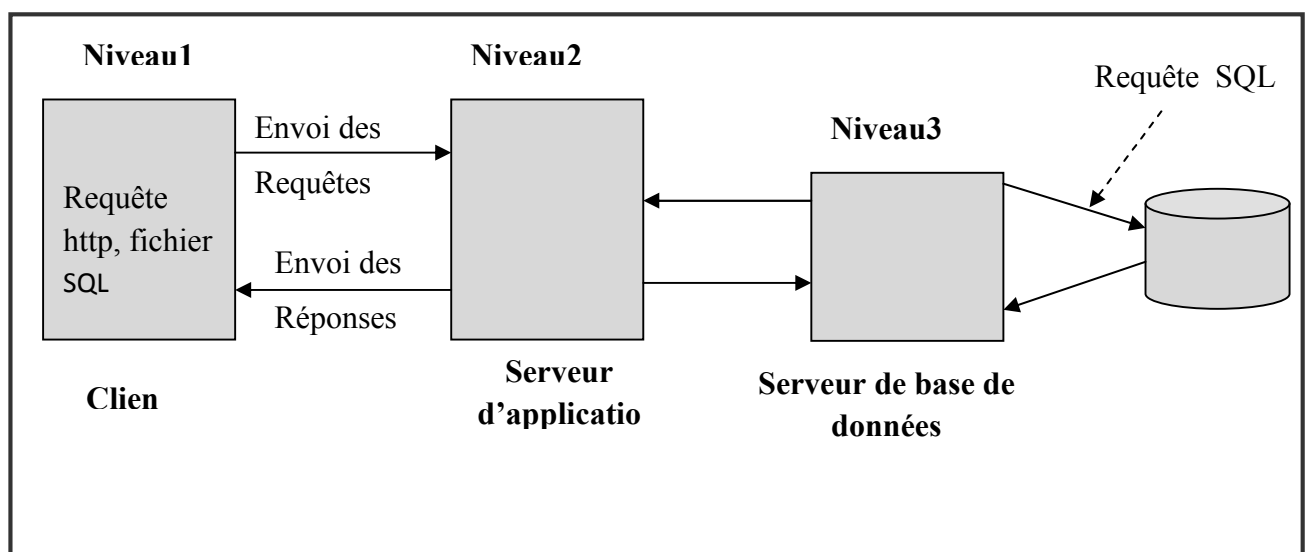


Figure1.3 : Architecture à trois niveaux.

3.3.2. Le mode égale à égale :

Dans le mode égale à égale il n'y a pas d'ordinateur central, mais chaque ordinateur a un rôle similaire.

3.4. Les modèles architecturaux des réseaux

L'apparition des techniques de commutation évoluées, de plus en plus performantes (paquets, trames, cellules...), ainsi que la modélisation des transferts d'information ont amené les constructeurs et les organismes internationaux à définir des architectures de réseaux plus au moins 'standard' et tout particulièrement le modèle OSI ainsi que le modèle TCP/IP. [5]

➤ Modèle OSI :

Le modèle OSI (Open System Interconnexion) dit aussi modèle ISO (interconnexion des systèmes ouverts) a été mis au point par l'organisme de normalisation (ISO International Standard Organisation) en 1978. [5]

Le rôle du modèle OSI consiste à standardiser la communication entre les machines afin que différents constructeurs puissent mettre au point des produits (logiciels, matériels) compatibles(pour peu qu'ils respectent scrupuleusement le modèle OSI) [6].

Le modèle OSI repose sur trois termes importants : les **couches**, les **protocoles**, et les **interfaces** normalisées.

Le modèle OSI est composé de sept couches :

- ✓ Les couches 1 à 3 sont les couches dites basses orientées **transmission**.
- ✓ La couche 4 représente une couche charnière entre couches basse et couches hautes. On peut parler aussi de **middleware** pour désigner cette couche et les couches avoisinantes.
- ✓ Les couches 5 à 7 sont les couches dites hautes orientées **traitement**.

L'organisation en couches permet d'isoler des fonctions réseaux dans les couches et de les implanter indépendamment de l'ensemble du système. Elle facilite ainsi l'évolution des logiciels réseau, en cachant les caractéristiques internes de la couche, au profit de la description des interfaces et des protocoles. [5]

3.5. Les réseaux locaux

Ils servent au transport de toutes les informations numériques de l'entreprise. En règle générale, les bâtiments à câbler s'étendent sur plusieurs centaines de mètres. [7]

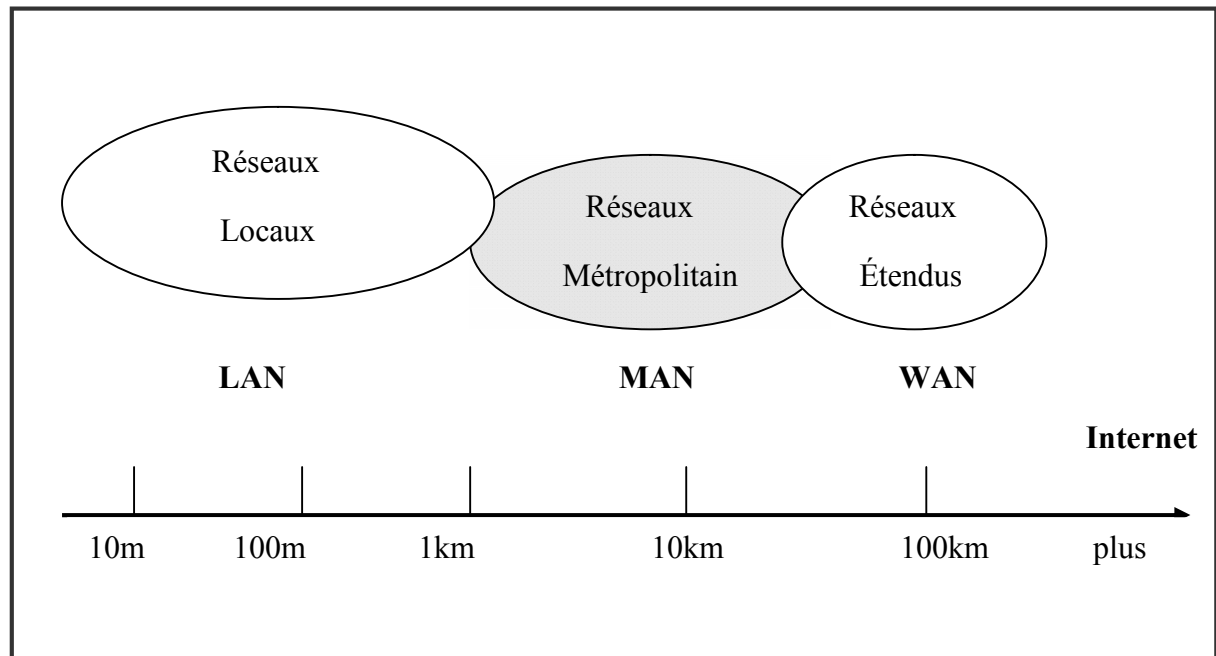


Figure1.4: Catégories de réseaux informatiques.

➤ **Les réseaux locaux d'une entreprise :**

Ce sont les réseaux mis en place par les petites ou grandes entreprises pour permettre la circulation de leurs données et éventuellement de la voix et de la vidéo dans un même réseau. Ces réseaux permettent également d'accéder aux ressources d'information de l'entreprise, qui se situent dans des serveurs, la plupart du temps sous forme de serveurs.

Lorsque le réseau d'entreprise utilise la technologie définie par les protocoles du monde IP, ce réseau prend pour nom Intranet. [7]

➤ **Intranet :**

Le mot Intranet est devenu familier depuis une décennie bien qu'il renferme différentes interprétations. Un Intranet est avant tout aux protocoles d'Internet, IP, TCP, UDP et leurs différentes incarnations. Le paradigme intranet correspond au système d'information de l'entreprise utilisant les applicatifs d'Internet. L'Intranet désigne aussi parfois l'infrastructure de l'entreprise pour réaliser ses communications internes ayant accès à l'Internet depuis l'extérieur.

Le mot Intranet désigne donc tous ces concepts à la fois, avec pour dénominateur commun l'utilisation du protocole IP et protocoles bâtis au-dessus d'IP ou sur lesquels IP peut s'appuyer. [7]

➤ **L'intérêt de l'Intranet :**

Face aux nouvelles contraintes économiques actuelles, une entreprise doit savoir communiquer et réagir en temps réel. Pour cela son information doit être gérée et la seule façon de le faire c'est de la décentraliser et la rendre accessible à l'ensemble de ses acteurs.

La structure de l'entreprise doit évoluer et les outils de l'Intranet seraient pour le moment les mieux adaptés. Intranet repose aujourd'hui sur de nouveaux standards et des innovations technologiques importants. Les nouvelles applications viennent souvent bousculer les architectures réseaux classiques. Du LAN au WAN, nouveaux débits, nouveaux protocoles, gestion de flux et de sécurité nécessitant de nouveaux investissements. De plus, Intranet offre un large panel d'applications :

- ✓ Des forums de discussion entre des équipes projets au sein d'une entreprise.
- ✓ L'interconnexion sécurité des réseaux locaux de l'entreprise et ses filiales en utilisant les infrastructures publiques de l'Internet.
- ✓ La création de serveurs WEB internes.
- ✓ La création d'un serveur WEB, accessible depuis l'entreprise et depuis l'Internet par une communauté fermée d'utilisateur, les clients ou les partenaires de l'entreprise par exemple.
- ✓ La mise en place d'une messagerie.
- ✓ L'interconnexion des différents systèmes d'information de l'entreprise et de ses fournisseurs en utilisant le protocole TCP/IP. De ce fait, de nombreux réseaux d'entreprise migrent vers les réseaux intranet.

➤ **Services et fonctionnalités de l'Intranet :**

Un Intranet est constitué de services destinés aux utilisateurs finaux, aux équipes informatiques et réseau. Il couvre une large palette des besoins de ces populations et propose généralement 8 services qui sont les suivants :

Les services de transport : les services de transport permettent de véhiculer l'information d'un point à un autre d'un Intranet. Un Intranet s'appuie sur les mêmes protocoles de transport que ceux de l'Intranet.

Les services d'administration : Un Intranet peut également comprendre des services d'administration et de gestion du réseau, les mêmes sont utilisés sur l'Intranet. En général trois services sont mis en place :

- ✓ Une plate-forme de supervision : permet de surveiller les entités du réseau (routeurs, imprimantes, logiciels...). elle s'appuie sur le standard SNMP (Simple Network Management Protocol) de l'Intranet.
- ✓ Une plate-forme d'administration et de télémaintenance : pour une prise de contrôle à distance. Elle utilise les protocoles tels que TCP/IP, PPP (Point to Point Protocol ...).
- ✓ Des serveurs cachés : permettent d'alléger les infrastructures télécoms. Ce sont des serveurs cachés 'proxy' comprenant un mécanisme de réplication sélective.

Les services de sécurité :

- ✓ Les services de sécurité sur un Intranet s'appuient sur des mécanismes simples comme les mots de passe, ou des mécanismes plus sophistiqués de calculatrice de chiffrement.
- ✓ Le chiffrement : évite que les messages et trames **IP** soient lues ou modifiées par des personnes non habilitées.
- ✓ Le filtrage des services, des adresses et du contenu : permet d'interdire ou d'autoriser un service quelconque pour une adresse **IP**, de filtrer le passage de ces adresses et le transport de fichiers par rapport à leur contenu.

Les services de partage de l'information : ces services sont destinés à archiver et restituer les informations. Il existe deux types de stockage dans un Intranet.

- ✓ Les services de stockage et d'accès : les serveurs de fichiers classiques des systèmes d'exploitation utilisés. Sur un Intranet, le transfert de fichiers s'appuie plus souvent sur une couche supérieure à celle du système de fichier.
- ✓ Le protocole **FTP** (File Transfer Protocol) permet à deux postes d'un Intranet d'échanger des fichiers, indépendamment de leur système d'exploitation.
- ✓ Les "serveurs de données" : les serveurs de données s'interfaçent avec les serveurs Web pour être consultées à l'aide d'un navigateur. Cette interface nécessite souvent l'utilisation d'un logiciel spécifique (middleware).
- ✓ Les serveurs de documents : sont des serveurs Web parfois couplés à des moteurs de recherche documentaire, et à travers lesquels les utilisateurs accèdent aux documents de l'entreprise.

- ✓ Les services de production et de publication d'information : Permettent à l'utilisateur d'adapter ses documents au standard de l'Intranet avec un minimum d'effort.

Les services de communication et de travail coopératif : l'entreprise cherche à être plus réactive, dans ce but intranet offre généralement cinq services favorisant la communication et le travail coopératif au sein de la société. Ces services sont :

- ✓ La messagerie et les listes de diffusion : la messagerie d'un intranet s'appuie sur les mécanismes de l'intranet **SMTP**, **POP3** (Post Office Protocol version3) et **IMAP** (Internet Message Access Protocol).
- ✓ La circulation de document (Workflow) : est une extension naturelle de la messagerie, elle permet de faire circuler un document selon un schéma préétabli. Il existe plusieurs catégories de circulation de documents, selon le parcours ou le type d'interaction entre les différents acteurs du processus.
- ✓ La visioconférence et l'audioconférence : elles ont été conçues pour un Internet ou la bande passante est précieuse mais elles trouvent toutes leur puissance sur un intranet ou le réseau est mieux maîtrisé.
- ✓ Le travail coopératif en temps partagé : permet à plusieurs acteurs de travailler ensemble et à distance sur un même document.
- ✓ Les forums : suivent de très près la messagerie ou le serveur Web dans la construction d'un Intranet. Il existe deux types de forums : les forums interactifs en temps réel et les forums interactifs en temps différés.

Les services d'annuaires : les annuaires des utilisateurs permettant de retrouver les adresses et les droits des utilisateurs de l'intranet.

Les services de développement applicateur :

- ✓ Langage de développement : permettent de développer des applications de production qui pourront être télédiffusées et utilisées à l'aide d'un navigateur.
- ✓ Atelier de génie logiciel (AGL) : permet de construire ses propres applications intranet.

Les services d'accès aux applications : ces services se concentrent dans le navigateur. Ce "client universel" permet d'accéder aux serveurs Web, aux serveurs de fichiers, ainsi qu'aux bases de données et application développées en java. Il est d'utilisation très simple et va jusqu'à

intégrer la messagerie et les forums. Il permet l'accès aux systèmes d'information de gestion et de production.

4. Présentation de Sonelgaz :

4.1. Société de Distribution d'Electricité et du Gaz de l'Est (SDE) :

La Société de Distribution d'Electricité et du Gaz de l'Est (SDE) a pour mission :

- L'exploitation et la maintenance du réseau de distribution de l'électricité et du gaz.
- Le développement des réseaux électricité et gaz permettant le raccordement des nouveaux clients.
- La commercialisation de l'électricité et du gaz, dans les meilleures conditions de sécurité, de qualité de service au moindre coût.



Figure 1.5 : Société de Distribution d'Electricité et du Gaz de l'Est (SDE)

4.2. La direction de distribution de Mila :

La direction régionale de Mila a les mêmes missions que la SDE donc l'exploitation et la maintenance du réseau de distribution de l'électricité et du gaz, le développement des réseaux électricité et gaz permettant le raccordement des nouveaux clients, la commercialisation de l'électricité et du gaz, dans les meilleures conditions de sécurité, de qualité de service au moindre coût.

La direction régionale de Mila couvre 34 communes de la wilaya, elle est composée de :

- La direction de la SDE
- Cameg Mila.
- Les agences commerciales (Ferdjioua, **Mila**, Tedjannanet).

4.3. L'organigramme de la direction régionale de la distribution :

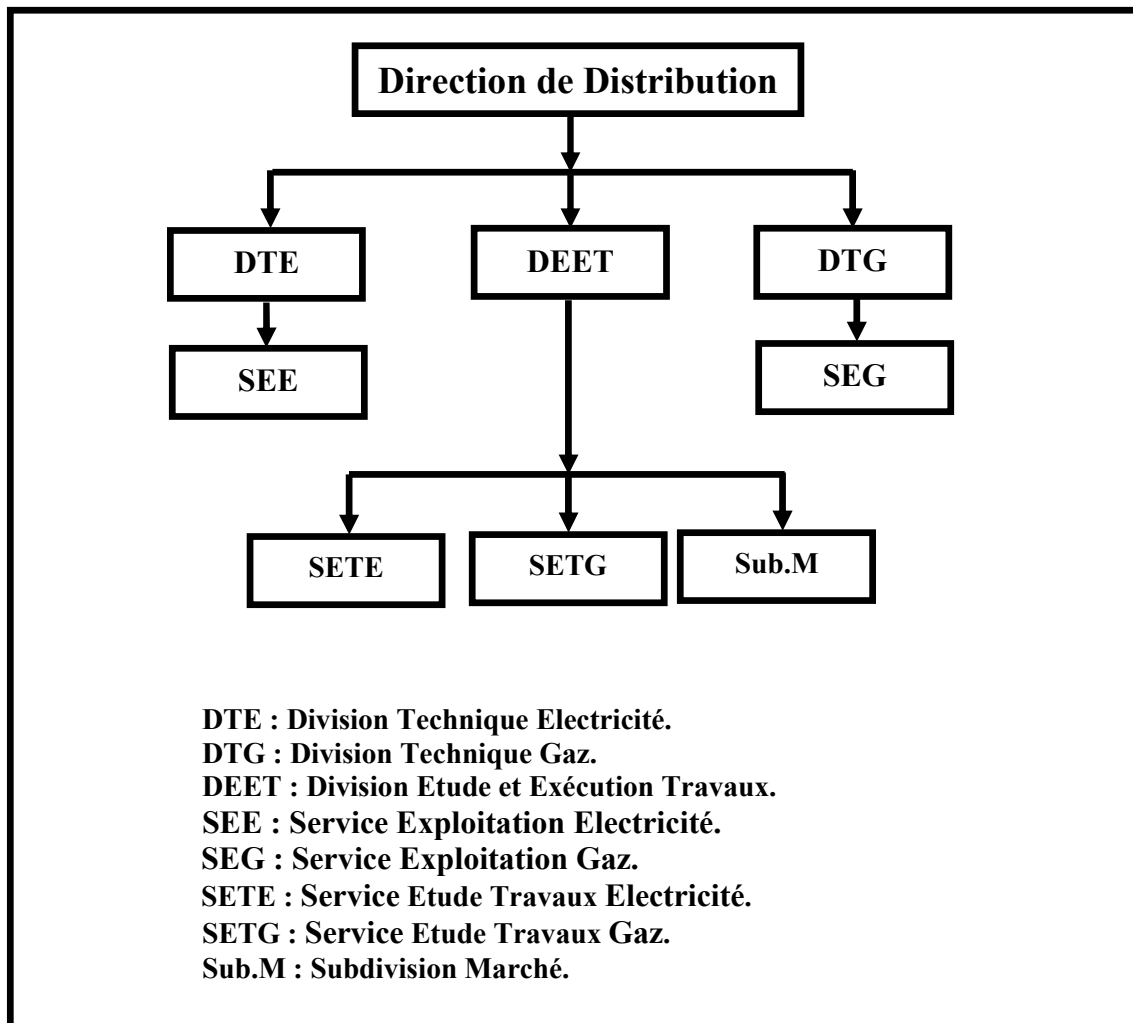


Figure 1.6 : l'organigramme de la direction de la distribution

5. Conclusion

La notion UML est suffisamment complète pour pouvoir remplacer ou compléter les notions à objets l'ayant précédées, et sa souplesse permet de l'utiliser pour modéliser toutes les facettes d'un système d'information de l'entreprise, depuis la phase d'identification jusqu'au test et au déploiement final de système opérationnel. L'entreprise doit coopérer entre leurs différents services et savoir communiquer et réagir en temps réel, cela est réalisable par la mise en place d'un réseau.

CHAPITRE 2

ETUDE PRÉLIMINAIRE

1. Introduction :

L'étude préliminaire est la première phase de notre processus de développement. Elle consiste à effectuer un repérage des besoins fonctionnels et opérationnels, en utilisant principalement le texte, ou des diagrammes très simples.

Dans ce chapitre on va présenter d'une façon générale le cahier des charges de notre projet, et par la suite décrire le contexte du notre système.

2. Élaboration du cahier des charges :

Le cahier des charges du projet rassemble les spécifications du projet. A cette étape s'effectue l'identification de l'entreprise, ses activités, et les activités concernées par le projet. Il contient aussi les choix techniques, les besoins fonctionnels et opérationnels.

2.1. Présentation de l'organisme d'accueil :

La direction de distribution SONALGAZ est une société algérienne à caractère industriel et commercial qui a pour fonction principale la distribution de l'Electricité et du Gaz. La DD de MILA possède un grand réseau d'électricité et de gaz aux niveaux de la wilaya de MILA, ses fonction principales sont :

- Assurer la gestion (conduite, exploitation et maintenance) et le développement des réseaux MT/BT et MP/BP et des installations annexes.
- Etablir et mettre en œuvre le développement de la construction et de la maintenance et l'exploitation des ouvrages.
- Assurer la gestion et le développement de la ressource humaine et des moyens matériels nécessaire au fonctionnement du centre.

Ces réseaux sont confrontés à des incidents et des pannes, maintenance tel que : les réclamations des abonnés et suggestions des exploitants, mesures surcharges, chute tension...

La Direction de Distribution de MILA prépare durant l'année un ensemble de dossiers travaux pour être réalisés l'année suivante dans le but de résoudre ces problèmes des réseaux, d'améliorer la qualité et la continuité de service fourni pour les abonnés et minimiser le plus possible l'insuffisance des deux énergies (Electricité & Gaz) pour mieux distribution.

La DD de MILA souhaite se doter d'un système informatique performant afin de maîtriser la gestion des programmes propres électricité et gaz au niveau local, ce nouveau système est appelé **SGPP (Système de Gestion de Programme Propre)**.

2.2.Problématique :

Durant le stage que nous avons fait dans la DD de MILA, au niveau des différents services concernés par la gestion de programme propres, nous avons synthétisés les problèmes suivants :

- Le travail se fait manuellement.
- L'insuffisance des ressources matérielles et informatique pour la gestion (absence d'un logiciel pour suivre des programmes propres).
- Pas de collaboration entre les services pour travailler simultanément (chacun fait individuellement).
- Absence du suivi des dépenses et également au niveau à la subdivision des investissements.
- Perte un beaucoup temps durant le suivi d'une affaire.
- Mauvaise exploitation du réseau local au niveau de la direction.

2.3.Objectifs du nouveau système :

La gestion des programmes propres au niveau de la DD est une tâche très difficile plus qu'elle est trop compliquée, elle demande beaucoup de temps dans le sens qu'on présente notre sujet qui consiste à réaliser un système d'information pour la gestion de ces programmes au niveau de la Direction de Distribution de MILA.

Notre objectif est la réalisation d'un système d'information qui assure le suivi et l'élaboration des programmes propres ELEC et GAZ à développer autour d'une base de données globale sous forme d'une application client/serveur en but de résoudre ces problèmes définitivement.

Notre système doit répliquer aux exigences suivantes :

- ✓ Informatiser les données nécessaires pour la gestion des programmes propres.
- ✓ Réduire le temps de travail entre les différents postes de système.
- ✓ Faire le travail de manière automatique.

- ✓ Un bon suivi pour les réalisations financières.
- ✓ Faciliter la circulation des informations entre les différents services.
- ✓ Améliorer les conditions de travail.
- ✓ Assurer la sécurité des informations (gestion des accès).

2.4.Grands choix techniques :

Pour réaliser ce projet, nous allons utiliser une approche itérative et incrémentale, fondée sur le processus de développement en Y. On a choisi un certain nombre de techniques clés :

- Le processus de développement : 2TUP.
- Le langage de modélisation : UML.
- L'architecture : clients/serveur.
- Le réseau utilisé : réseau local (LAN).
- Langage de programmation : JAVA.
- Le serveur de la base de données : MySQL

2.5.Recueil des besoins fonctionnels :

Un premier tour d'horizon des besoins exprimés par les employés de la direction de la distribution, permet d'établir un cahier des charges préliminaire :

Durant une année, un programme annuel d'entretien est réalisé par une équipe de travaux composé d'un chef d'équipe et des agents et ce programme se fait chaque jour, ses tâches tel que : surveillance réseau, réparation des chutes électricité et des fuites gaz, etc....lors de ce travail on constate des anomalies, les qui ne sont pas réalisées, on les propose des affaires programme propre. Ces affaires sont proposées par :

- ✓ Le chef district, le chef d'équipe.

Chaque année, les affaires proposées doivent être classées par des services d'exploitation (électricité et gaz) au niveau de chaque district, puis les enregistre pour devoir réaliser l'année prochaine, c'est à dire comment faire la création de l'affaire et sa proposition, validation et sa consultation. Ces derniers doit être consultées par :

- ✓ Le directeur, le chef district, le chef de division technique (Electricité & Gaz), le chef de service étude et travaux, le technicien d'étude, le chef de subdivision marchée et le chef de subdivision investissement.

Après l'enregistrement des affaires, des avants projets ont été créés pour chaque affaire pour améliorer la qualité et la continuité de service. Ces avants projets sont créés par :

- ✓ l'ingénieur basse tension (BT).

Après la validation par le chef district, ces avants projets seront envoyés aux divisions techniques (Electricité & Gaz) pour les traiter et les approuver.

- Le chef division technique reçoit des propositions qui sont validées après le traitement et transmis les avants projets à la division d'étude et exécution travaux pour l'étude.
- Le technicien d'étude fait sortie sur terrains pour étudier les affaires validées par le chef division technique (Electricité & Gaz) établit le quantitatif et le plan d'étude de chaque affaire pour la réalisation.
 - ✓ Ces fiches validées par le chef service étude et travaux (Electricité & Gaz).
- Etablissement un report d'étude de chaque affaire au bureau et l'approuver par le technicien et le chef service (Electricité & Gaz).
- Le chef service (Electricité & Gaz) transmet ce rapport à la subdivision marchée pour lancer la consultation.
- Le chef subdivision marchée établit le contrat avec l'entreprise et le transmet la au pour lancer le travail avec l'entreprise de réalisation.

Le suivi des travaux est consulté par le chef de service étude et travaux, le chef subdivision marché et surveillant travaux.

- ✓ Le directeur valide le contrat de réalisation avec l'entreprise.
- L'administrateur du système, définit les profils utilisateurs.

2.6. Recueil des besoins opérationnels :

- **Limiter les risques d'accidents :**
 - ✓ Vérification et maintenance périodique des installations techniques.

- ✓ Le contrôle de saisie des informations (géré également par MySQL, utilisation l'authentification tel que les mots de passe).

➤ **Contrôle d'accès :**

- ✓ L'authentification des utilisateurs par login et mot de passe.
- ✓ L'administrateur est la seule personne autorisée à définir et modifier les mots de passe.
- ✓ La modification périodique des mots de passe est obligatoire.

➤ **L'accès au système**

- ✓ Autorisation d'accès : lors de sa connexion, chaque utilisateur doit être reconnu par un nom et un mot de passe.

3. Description du contexte du système :

Une fois, ce premier recueil de besoin effectué, la description du contexte du système peut commencer. Elle consiste en trois activités successives :

- ✓ L'identification des entités externes au système qui interagissent directement avec lui (acteurs).
- ✓ L'identification des messages et des interactions (émission/réception des messages) entre ces acteurs et le système.
- ✓ La représentation de tous les messages (système <-> acteurs) sur un diagramme de contexte dynamique.

3.1. Identification des acteurs :

Les acteurs de système sont :

➤ **Le chef district :** son rôle est :

- ✓ Proposer Affaire.
- ✓ Valider Affaire.

➤ **Ingénieur BT :** son rôle est :

- ✓ Etablir Avant projet.

➤ **Le chef équipe :** son rôle est :

- ✓ Proposer Affaire.
- ✓ Consulter Affaire.

➤ **Le chef service exploitation :** son rôle est :

- ✓ Valider Affaire.

- **Le chef de division technique :** le rôle de cet acteur :
 - ✓ Valider Affaire.
- **Le technicien d'étude :** son rôle est :
 - ✓ Etablir Quantitatif.
 - ✓ Mettre à jour Elément.
- **Le chef de service étude et travaux :** son rôle est :
 - ✓ Consulter Quantitatif.
- **Le chef de subdivision marché :** son rôle est :
 - ✓ Consulter Quantitatif.
 - ✓ Etablir Contrat.
 - ✓ Mettre à jour Entreprise.
- **Le surveillant travaux :** son rôle est :
 - ✓ Consulter Quantitatif
- **Le directeur :** son rôle est :
 - ✓ Valider Contrat.
 - ✓ Consultation Affaire.
- **L'administrateur :** son rôle est :
 - ✓ Mettre à jour Employé.

3.2. Identification des messages entre le système et ses acteurs :

- Les messages émis par le système sont :
 - ✓ Demande des informations sur l'affaire proposée.
 - ✓ Demande la validation de l'affaire.
 - ✓ Demande la consultation de l'affaire.
 - ✓ Demande des informations pour extraire l'avants projet.
 - ✓ Demande des informations sur les éléments.
 - ✓ Demande des informations pour extraire le quantitatif.
 - ✓ Demande des informations pour valider le contrat.
 - ✓ Demande des informations sur les employés.

- ✓ Demande des informations sur les éléments.
- ✓ Demande des informations sur les entreprises.
- Les messages reçus par le système sont :
 - ✓ Informations détaillées sur l'affaire proposée.
 - ✓ Informations détaillées sur la validation de l'affaire.
 - ✓ Informations détaillées pour extraire les avant projets.
 - ✓ Informations détaillées sur les éléments.
 - ✓ Informations détaillées pour extraire le quantitatif.
 - ✓ Informations détaillées sur le contrat.
 - ✓ Informations détaillées sur les éléments.
 - ✓ Informations détaillées sur l'entreprise.

3.3. Diagramme de contexte dynamique de système :

Tous les messages (systèmes acteurs) identifiés précédemment peuvent être représentés de façon synthétique sur un diagramme, que l'on peut qualifier de diagramme de contexte dynamique.

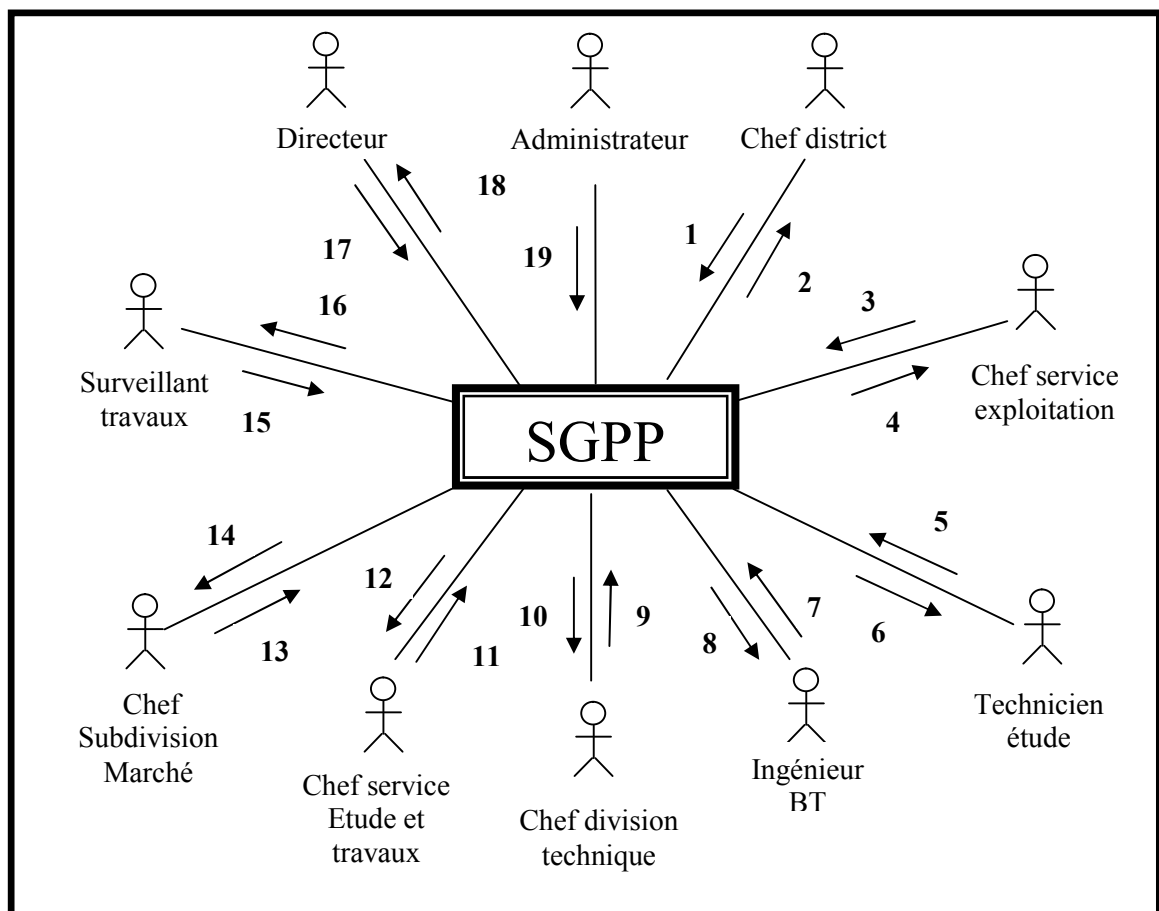


Figure 2.1: Le diagramme de contexte dynamique

Légende des messages du diagramme de contexte dynamique

1. Chef district ➡ SGPP

- Informations détaillées sur l'affaire proposée.

2. SGPP ➡ Chef district

- Demande des informations sur l'affaire a proposée.

3. Chef service exploitation ➡ SGPP

- Informations détaillées sur la validation de l'affaire.

4. SGPP ➡ Chef service exploitation

- Demande des informations sur la validation de l'affaire.

5. Technicien étude ➡ SGPP

- Informations détaillées sur le quantitatif.
- Informations détaillées sur les éléments.

6. SGPP ➡ Technicien d'étude

- Demande des informations sur la quantitatif.
- Demande des informations sur les éléments.

7. Ingénieur BT ➡ SGPP

- Informations détaillées sur l'avant projet.

8. SGPP ➡ Ingénieur BT

- Demande des informations sur l'avant projet.

9. Chef division technique ➡ SGPP

- Informations détaillées sur la validation de l'affaire.

10. SGPP ➡ Chef division technique

- Demande des informations sur la validation de l'affaire.

11. Chef service étude et travaux ➡ SGPP

➤ Informations détaillées sur le quantitatif. 12. SGPP ➡ Chef service étude et travaux ➤ Demande des informations sur le quantitatif.
13. Chef subdivision marché ➡ SGPP ➤ Informations détaillées sur le quantitatif. ➤ Informations détaillées sur le contrat. ➤ Informations détaillées sur l'entreprise. 14. SGPP ➡ Chef subdivision marché ➤ Demande des informations sur le quantitatif. ➤ Demande des informations sur le contrat. ➤ Demande des informations sur les entreprises.
15. Surveillant travaux ➡ SGPP ➤ Informations détaillées sur le quantitatif. 16. SGPP ➡ Surveillant travaux ➤ Demande des informations sur le quantitatif.
17. Directeur ➡ SGPP ➤ Informations détaillées sur la validation du contrat avec l'entreprise. 18. SGPP ➡ Directeur ➤ Demande des informations sur le contrat de travail.
19. Administrateur ➡ SGPP ➤ Informations détaillées sur les employés.

Tableau 2.1 : Légende des messages du diagramme de contexte dynamique.

4. Conclusion :

Dans ce chapitre on a :

- Etablit un recueil initial des besoins fonctionnels et opérationnels.
- Modélisé le contexte du système SGPP, considéré comme une boîte noire permettant de :
 - ✓ Identifié les entités externes du système qui interagissent directement avec lui (acteurs).
 - ✓ Répertorie les interactions entre les acteurs et le système.
 - ✓ Répertorie l'ensemble des interactions sur un modèle de contexte dynamique.

L'étude préliminaire prépare les étapes plus formelles de capture des besoins fonctionnels et techniques que nous allons décrire au chapitre suivant.

CHAPITRE 3

CAPTURE DES BESOINS

1. Introduction :

Cette phase est très importante dans le cycle de développement en Y, elle explique comment identifier les besoins exprimés selon le métier des utilisateurs, les reformuler, les structurer et les documenter en UML. Elle s'appuie pour une large part sur la technique des cas d'utilisation.

La phase des captures des besoins est constituée de deux étapes :

- ✓ La capture des besoins fonctionnels.
- ✓ La capture des besoins techniques.

2. Capture des besoins fonctionnels :

La capture des besoins fonctionnels est la première étape de la branche gauche du cycle en Y. Elle formalise et détaille ce qui a été ébauché au cours de l'étude préliminaire. Elle est complétée au niveau de la branche droite du Y par la capture des besoins technique et prépare l'étape suivante de la branche gauche.

Par besoins fonctionnels on désigne l'ensemble des besoins liés au métier et domaine traité par le système. La technique des cas d'utilisation est la pierre angulaire de l'étape de capture des besoins fonctionnels. Elle va nous permettre de préciser l'étude du contexte fonctionnel du système SGPP, on décrivant les différentes façons qu'auront les acteurs pour utiliser le futur système SGPP.

2.1.Liste des cas d'utilisations :

Un cas d'utilisation représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier.

La liste des cas d'utilisations est la suivante:

- ✓ Proposer Affaire.
- ✓ Valider Affaire.
- ✓ Affecter Élément.
- ✓ Etablir Avants projet.
- ✓ Etablir Quantitatif.
- ✓ Etablir Contrat.
- ✓ Valider Contrat.
- ✓ Consulter Affaire.
- ✓ Mettre à jour Employé.

- ✓ Mettre à jour Elément.
- ✓ Mettre à jour Entreprise.

2.2. Identification des acteurs principaux et secondaires:

Cas d'utilisation	Acteur principal et Acteur secondaire	Messages émis et reçus
Proposer Affaire	Chef district Chef équipe	Emet : Information sur l'affaire proposée.
Valider Affaire	Chef district Chef service exploitation Chef division technique	Emet : Informations de validation de l'affaire proposées.
Etablir Avants projet	Ingénieur BT	Emet : Informations pour extraire l'avants projet.
Etablir Quantitatif	Technicien étude	Emet : Informations sur le quantitatif.
Consulter Quantitatif	Surveillant travaux Chef service étude et travaux Chef subdivision marché	Reçoit : Le quantitatif.
Etablir le contrat.	Chef subdivision marché	Emet: Informations détaillées sur le contrat.
Valider le contrat.	Directeur	Emet: Informations pour valider le contrat.
Mettre à jour Employés	Administrateur	Emet: Informations sur les employés.
Mettre à jour Elément	Technicien étude	Emet: Informations sur les éléments
Mettre à jour Entreprise	Chef subdivision marché	Emet: Informations sur les entreprises

Tableau 3.1 : la liste des acteurs et les messages par cas d'utilisation.

2.3.Elaboration de diagramme des cas d'utilisation :

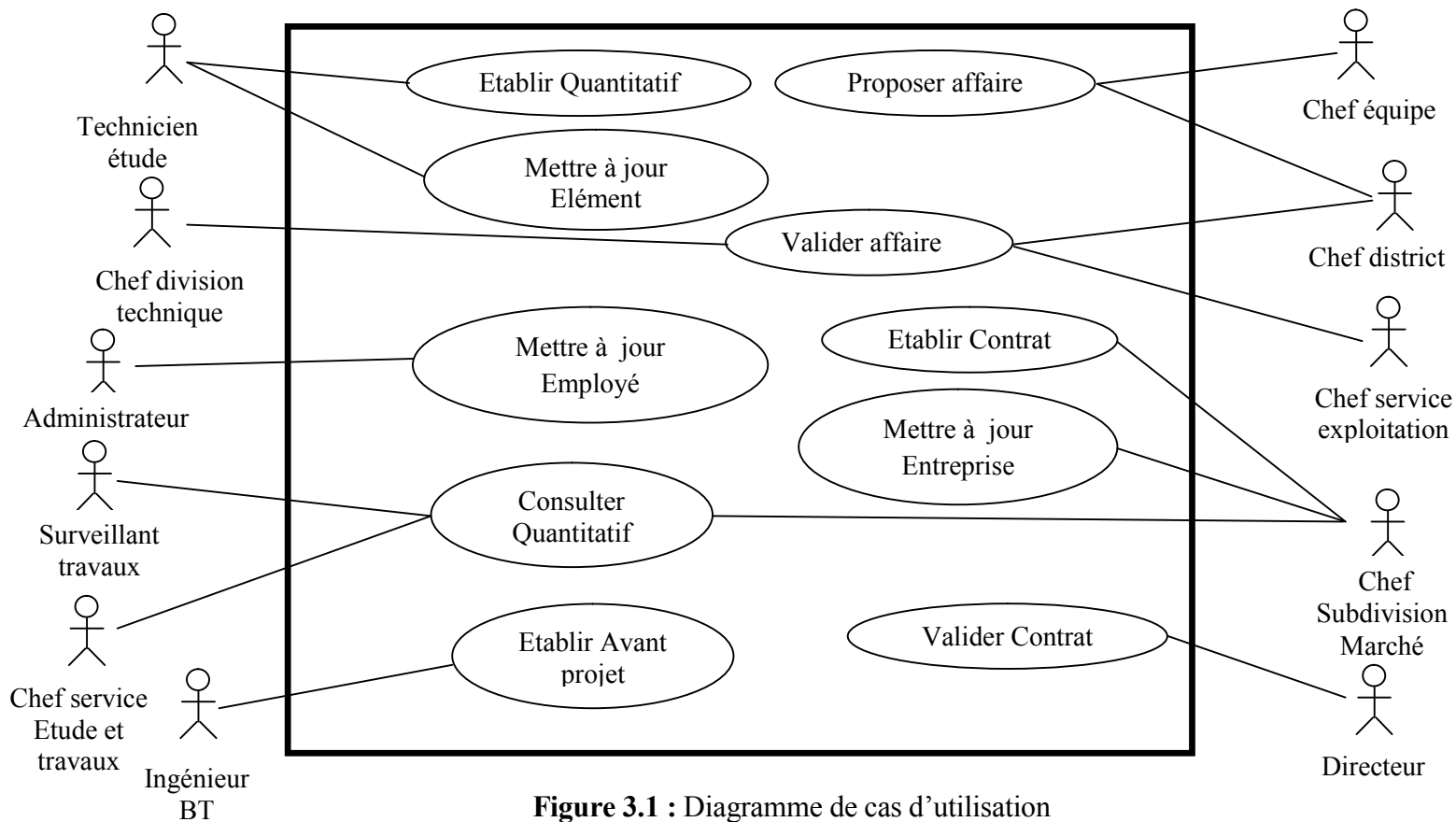
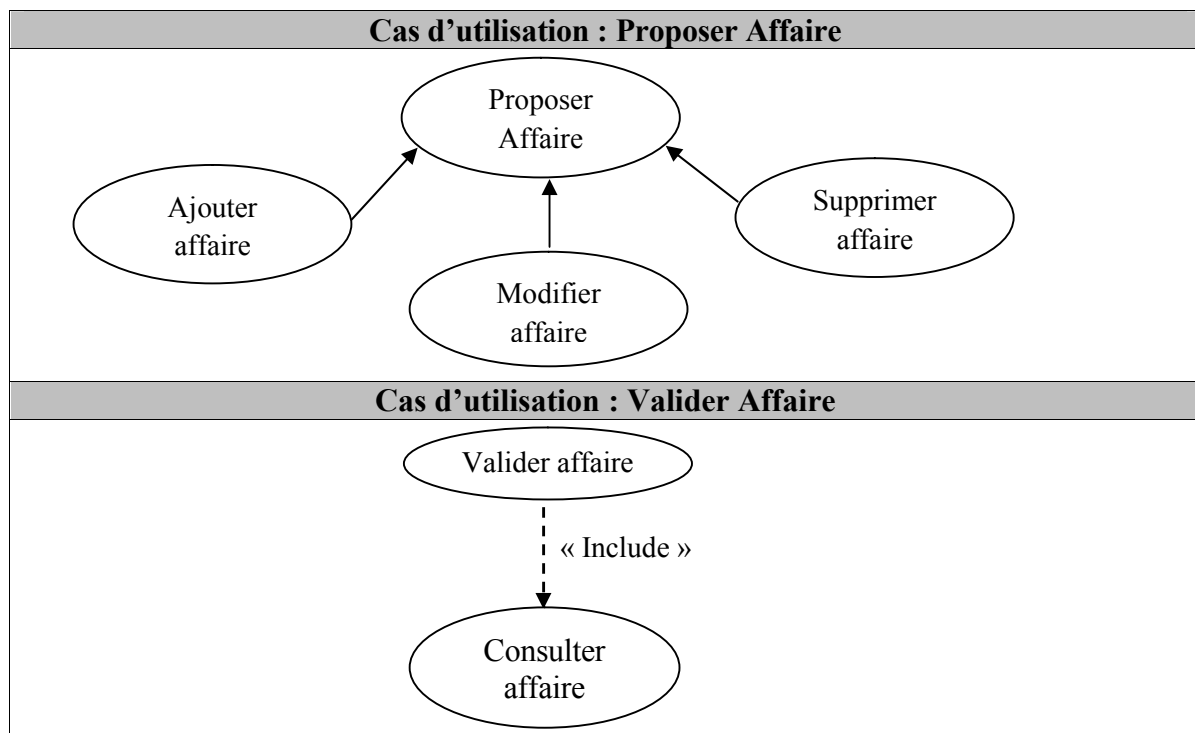
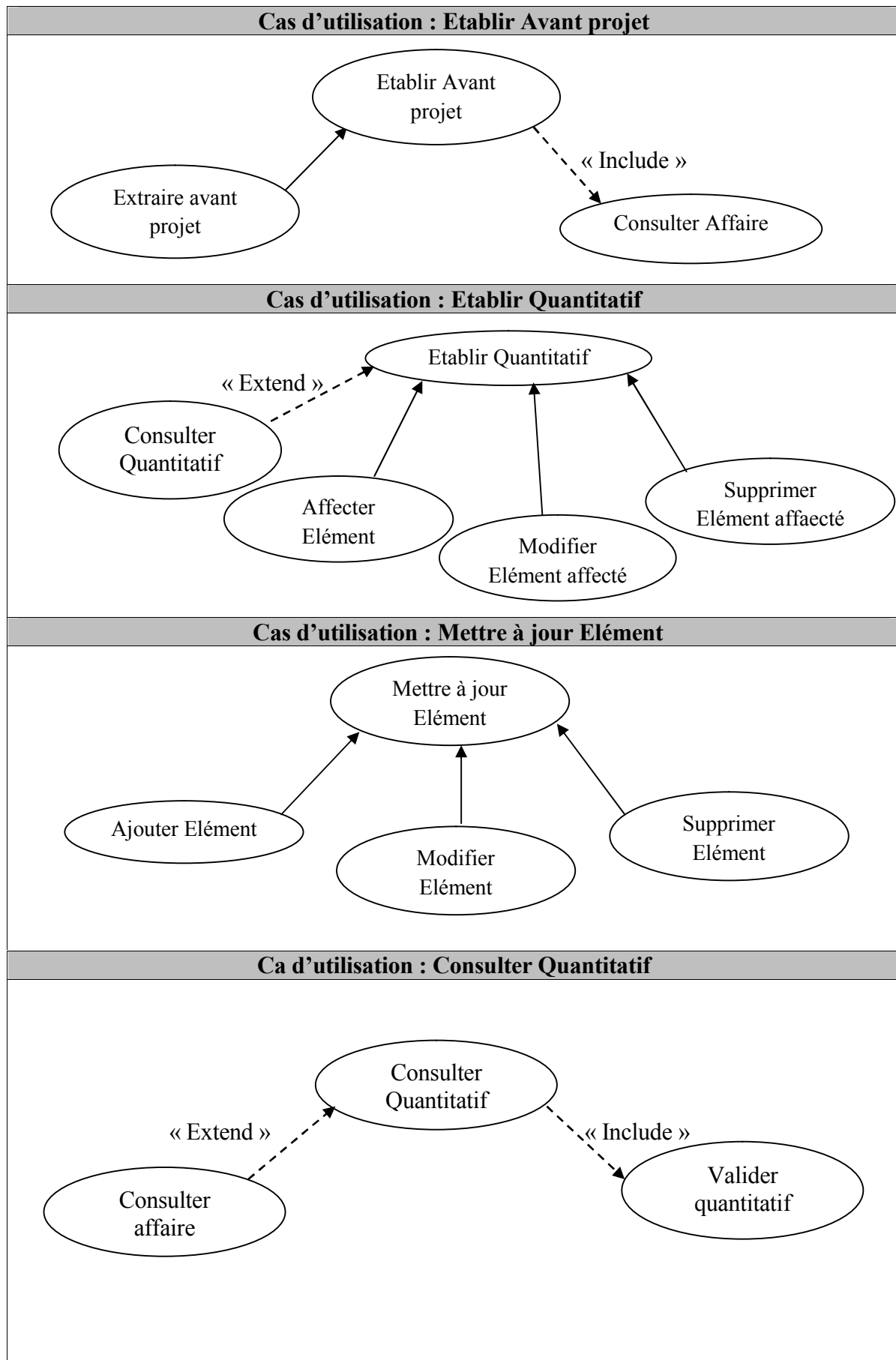


Figure 3.1 : Diagramme de cas d'utilisation

2.4.Relations entre cas d'utilisation :





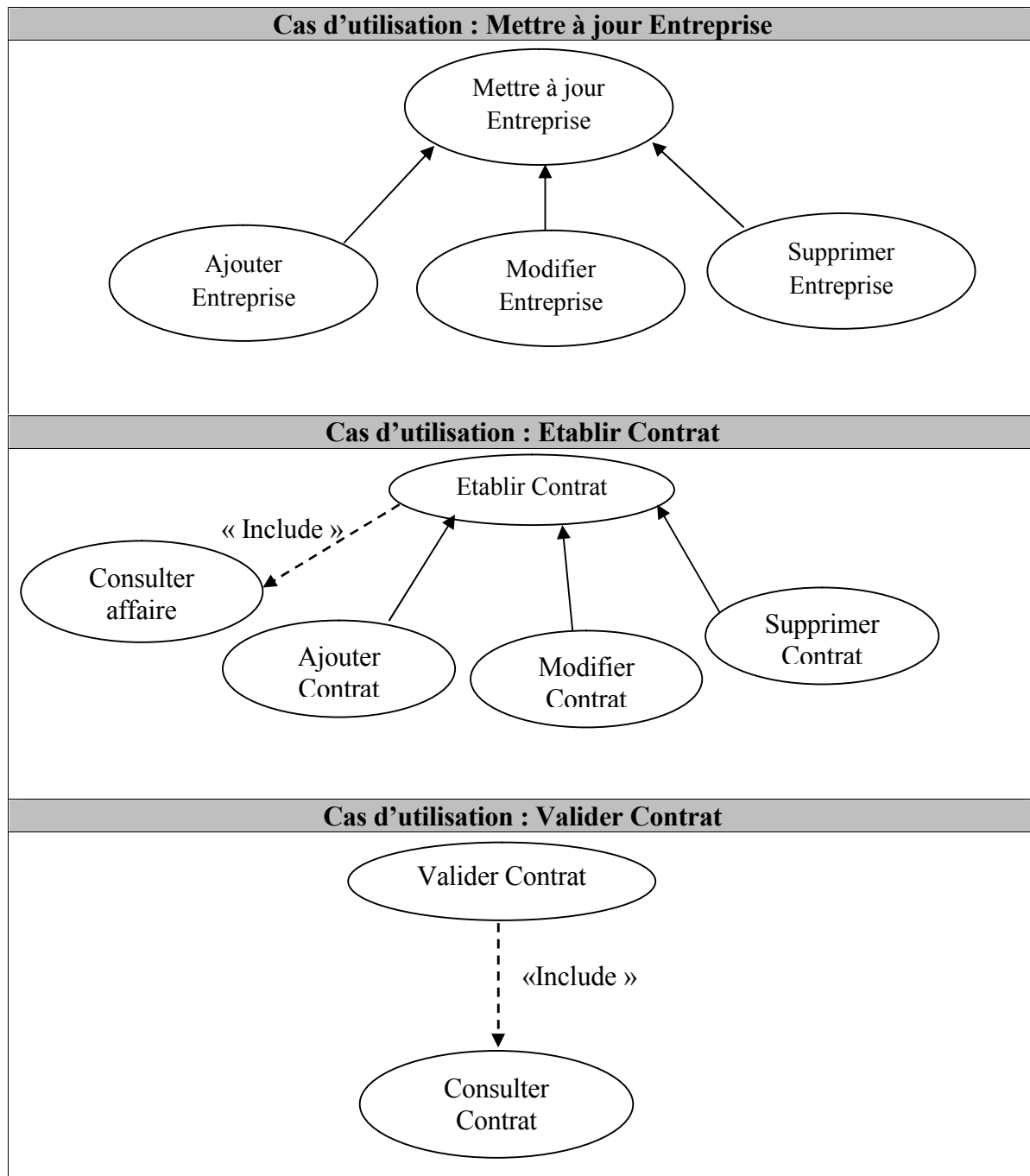


Tableau 3.2 : Relation entre les cas d'utilisation.

2.5.Description des cas d'utilisation :

Pour décrire la dynamique d'un cas d'utilisation. Le plus naturel consiste à recenser toutes les interactions de façon textuelle. Le cas d'utilisation doit par ailleurs avoir un début et une fin clairement identifiés. Il doit préciser les interactions entre les acteurs et le système, et quels sont les messages échangés. Il faut également préciser les variantes possibles, tout en essayant d'ordonner séquentiellement les descriptions, afin d'améliorer la lisibilité. Chaque unité de description de séquences d'actions est appelée enchaînement. Un scénario représente une succession particulière d'enchaînement, qui s'exécute du début à la fin de cas d'utilisation.

2.5.1. Description textuelle des cas d'utilisation :

Cas d'utilisation : Proposer affaire	
Sommaire d'authentification :	
➤ Titre : Proposer Affaire. ➤ But : Proposer une nouvelle affaire. ➤ Acteur : Chef district Chef équipe	
Enchainements :	
➤ Préconditions : l'utilisateur est authentifié. ➤ Démarrage : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour ajouter une nouvelle affaire. ➤ Enchaînements : - a : l'utilisateur demande la proposition d'une nouvelle affaire. - b : le système affiche le formulaire. - c : l'utilisateur saisit les informations d'une affaire. - d : l'utilisateur demande au système d'enregistrer. - e : le système effectue une vérification. [Si Erreur] le système affiche un message d'erreur. [Si non] le système enregistre l'affaire et indique la fin de l'opération. ➤ Arrêt : ce cas d'utilisation termine lorsque l'ajout est terminé ou annulé.	

- **Postconditions** : une nouvelle affaire est ajoutée à l'ensemble des affaires.

Cas d'utilisation : Modifier affaire

Sommaire d'authentification :

- **Titre** : Modifier Affaire.
- **But** : Modifier des informations d'une affaire existante.
- **Acteur** : Chef district
Chef équipe

Enchainements :

- **Préconditions** : 1. Le chef de district est authentifié.
2. L'existence d'une affaire non validée.
- **Démarrage** : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour modifier des informations d'une affaire proposée et non acceptée.
- **Enchaînements** :
 - a : l'utilisateur demande la modification d'une affaire.
 - b : le système affiche le formulaire.
 - c : l'utilisateur sélectionne une affaire.
 - c : le système affiche les informations de l'affaire sélectionnée.
 - d : l'utilisateur modifie (le numéro, l'intitulé, le type de l'affaire...).
 - e : l'utilisateur demande au système d'enregistrer.
 - f : le système effectue une vérification et affiche un message de confirmation de modifications des informations.
 - [Si Erreur] le système affiche un message d'erreur.
 - [Si non] le système enregistre les modifications et indique la fin de l'opération.
 - [Exception 1 : affaire n'existe pas] un message d'erreur est affiché lorsque l'affaire n'existe pas.
- **Arrêt** : Ce cas d'utilisation termine lorsque la modification d'une affaire est terminée ou bien annulée.
- **Postconditions** : Affaire modifiée.

Cas d'utilisation : Supprimer affaire	
Sommaire d'authentification :	
➤ Titre : Supprimer une affaire. ➤ But : Supprimer des informations d'une affaire existante. ➤ Acteur : Chef district Chef équipe	
Enchainements :	
➤ Préconditions : Le chef de district est authentifié. ➤ Démarrage : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour supprimer des informations d'une affaire proposée ➤ Enchainement : - a : L'utilisateur demande la suppression d'une affaire. - b : Le système affiche le formulaire. - c : L'utilisateur sélectionne une affaire et la supprime. - d : Le système valide la suppression en éliminant cette affaire de la base de donnée. ➤ Arrêt : Ce cas d'utilisation termine lorsque la suppression est terminée ou bien annulée. ➤ Postconditions : L'affaire est supprimée.	

Cas d'utilisation : Valider affaire	
Sommaire d'authentification :	
➤ Titre : Valider une affaire. ➤ But : Valider des affaires proposées.	

➤ Acteur : Chef division technique Chef district Chef service exploitation
Enchainements :
➤ Préconditions : 1. L'utilisateur est authentifié. 2. L'existence d'une autre affaire proposée non validée. ➤ Démarrage : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour valider une affaire proposé. ➤ Enchaînements : a : l'utilisateur demande la validation d'une nouvelle affaire. b : l'utilisateur saisit le numéro d'affaire et demande d'afficher. Exception 1 : affaire n'existe pas. Exception 2 : affaire existe et n'est pas validée. d : le système affiche le formulaire de l'affaire. e : l'utilisateur saisit la décision. f : l'utilisateur demande au système d'enregistrer. g : le système effectue une vérification. [Si Erreur] le système affiche un message d'erreur. [Si non] le système enregistre la validation et indique la fin de l'opération. Exceptions : [Exception 1 : affaire n'existe pas] un message d'erreur est affiché lorsque l'affaire n'existe pas ou affaire n'est pas validée. ➤ Arrêt: Ce cas d'utilisation termine lorsque la validation d'une affaire proposée est terminée ou bien annulée. ➤ Postconditions : Affaire validée.

Cas d'utilisation : Etablir Avant projet
Sommaire d'authentification :

➤ Titre : Etablir Avant projet. ➤ But : Extraire un nouveau avant projet. ➤ Acteur : Ingénieur BT.
Enchainements :
➤ Préconditions : L'ingénieur BT est authentifié. ➤ Démarrage : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour extraire un nouveau avant projet. ➤ Enchaînements : - a : l'utilisateur demande la création d'un nouveau avant projet de l'affaire. - b : le système affiche les affaires proposées. - b : L'ingénieur BT choisi une des affaires et extraire l'avant projet. - c : l'utilisateur sélectionne un type et un emplacement, et saisit les autres informations et l'extraire. - d : l'utilisateur demande au système d'enregistrer. - e : le système effectue une vérification. [Si Erreur] le système affiche un message d'erreur. [Si non] le système enregistre l'avant et indique la fin de l'opération. ➤ Arrêt : ce cas d'utilisation termine lorsque l'ajout est terminé ou annulé. ➤ Postconditions : une nouvelle affaire est ajoutée à l'ensemble des affaires.

Cas d'utilisation : Affecter Elément
Sommaire d'authentification :
➤ Titre : Affecter élément. ➤ But : Affecter les éléments pour des affaires. ➤ Acteur : Technicien étude.
Enchainements :

- **Préconditions** : 1. L'utilisateur est authentifié.
2. L'existence d'une affaire acceptée.
- **Démarrage** : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour ajouter les éléments nécessaires pour la réalisation de l'affaire.
- **Enchaînements** :
 - a : l'utilisateur demande d'affecter un élément pour l'affaire.
 - b : le système affiche la liste des affaires.
 - c : l'utilisateur sélectionne une affaire.
 - d : le système affiche la liste des éléments.
 - e : l'utilisateur sélectionne un ou plusieurs et saisit la quantité.
 - f : l'utilisateur demande au système d'enregistrer.
 - g : le système effectue une vérification.

[Si Erreur] le système affiche un message d'erreur.

[Si non] le système enregistre la quantité et indique la fin de l'opération.
- **Exceptions** : [Exception 1 : affaire n'existe pas] un message d'erreur est affiché lorsque l'affaire n'existe pas.
- **Arrêt**: Ce cas d'utilisation termine lorsque l'ajout est terminé ou bien annulé.
- **Postconditions** : éléments ajoutés pour l'affaire.

Cas d'utilisation : Modifier Elément affecté
Sommaire d'authentification :
➤ Titre : Modifier Elément affecté ➤ But : Modifie les informations d'un quantitatif existe. ➤ Acteur : Technicien étude.
Enchainements :
➤ Préconditions : 1. L'utilisateur est authentifié. 2. L'existence d'un quantitatif non validé. ➤ Démarrage : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour modifier les informations des éléments affectés pour l'affaire.

➤ **Enchaînements :**

a : l'utilisateur demande de modifier un élément pour l'affaire.

b : le système affiche le formulaire.

c : l'utilisateur sélectionne une affaire.

d : le système affiche la liste des éléments.

e : l'utilisateur modifier la quantité d'un ou plusieurs élément.

f : l'utilisateur demande au système d'enregistrer.

g : le système effectue une vérification.

[Si **Erreur**] le système affiche un message d'erreur.

[Si **non**] le système enregistre la quantité et indique la fin de l'opération.

➤ **Exceptions :** [Exception 1 : élément n'existe pas] un message d'erreur est affiché lorsque l'élément n'existe pas.

➤ **Arrêt:** Ce cas d'utilisation termine lorsque l'ajout est terminé ou bien annulé.

➤ **Postconditions :** éléments modifiés pour l'affaire.

Cas d'utilisation : Supprimer Élément affecté

Sommaire d'authentification :

- **Titre :** Supprimer Élément affecté
- **But :** Supprimer un élément d'un quantitatif existe.
- **Acteur :** Technicien étude.

Enchaînements :

- **Préconditions :** 1. L'utilisateur est authentifié.
2. L'existence d'un quantitatif non validé.
- **Démarrage :** ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour supprimer les informations des éléments affectés pour l'affaire.
- **Enchaînements :**
 - a : l'utilisateur demande de supprimer un élément pour l'affaire.
 - b : le système affiche le formulaire.
 - c : l'utilisateur sélectionne une affaire.
 - d : le système affiche la liste des éléments.

e : l'utilisateur supprime la quantité d'un ou plusieurs élément. f : l'utilisateur demande au système d'enregistrer. g : le système effectue une vérification. [Si Erreur] le système affiche un message d'erreur. [Si non] le système enregistre la quantité et indique la fin de l'opération. ➤ Exceptions : [Exception 1 : affaire n'existe pas] un message d'erreur est affiché lorsque l'affaire n'existe pas. ➤ Arrêt: Ce cas d'utilisation termine lorsque l'ajout est terminé ou bien annulé. ➤ Postconditions : éléments supprimés pour l'affaire.

Cas d'utilisation : Ajouter Élément
Sommaire d'authentification : ➤ Titre : Ajouter Élément. ➤ But : Ajouter un nouveau élément. ➤ Acteur : Technicien étude.
Enchaînements : ➤ Préconditions : l'utilisateur est authentifié. ➤ Démarrage : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour ajouter un nouveau élément. ➤ Enchaînements : a : l'utilisateur demande d'ajouter d'un nouveau élément. b : le système affiche le formulaire. c : l'utilisateur saisit les informations d'un élément. d : l'utilisateur demande au système d'enregistrer. e : le système effectue une vérification. [Si Erreur] le système affiche un message d'erreur. [Si non] le système enregistre l'affaire et indique la fin de l'opération. ➤ Arrêt : ce cas d'utilisation termine lorsque l'ajout est terminé ou annulé. ➤ Postconditions : un nouveau élément est ajouté à l'ensemble des éléments.

Cas d'utilisation : Modifier Elément
Sommaire d'authentification :
➤ Titre : Modifier Affaire. ➤ But : Modifier des informations d'un élément existant. ➤ Acteur : Technicien étude.
Enchainements :
➤ Préconditions : 1. L'utilisateur est authentifié. 2. L'existence d'un élément non validé. ➤ Démarrage : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour modifier des informations d'un élément non accepté. ➤ Enchaînements : - a : l'utilisateur demande la modification d'un élément. - b : le système affiche le formulaire. - c : l'utilisateur sélectionne un élément. - c : le système affiche les informations de l'élément. - d : l'utilisateur modifie l'élément. - e : l'utilisateur demande au système d'enregistrer. - f : le système effectue une vérification et affiche un message de confirmation de modifications des informations. [Si Erreur] le système affiche un message d'erreur. [Si non] le système enregistre les modifications et indique la fin de l'opération. [Exception 1 : élément n'existe pas] un message d'erreur est affiché lorsque l'élément n'existe pas. ➤ Arrêt : Ce cas d'utilisation termine lorsque la modification d'un élément est terminée ou bien annulé. ➤ Postconditions : Elément modifié.

Cas d'utilisation : Supprimer Elément	
Sommaire d'authentification :	
➤ Titre : Supprimer une affaire. ➤ But : Supprimer des informations d'un élément existant. ➤ Acteur : Technicien étude.	
Enchainements :	
➤ Préconditions : L'utilisateur est authentifié. ➤ Démarrage : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour supprimer des informations d'un élément. ➤ Enchaînement : - a : L'utilisateur demande la suppression d'un élément. - b : Le système affiche le formulaire. - c : L'utilisateur sélectionne un élément et le supprime. - d : Le système valide la suppression en éliminant cet élément de la base de donnée. ➤ Arrêt : Ce cas d'utilisation termine lorsque la suppression est terminée ou bien annulée. ➤ Postconditions : L'élément est supprimé.	

Cas d'utilisation : Consulter Quantitatif	
Sommaire d'authentification :	
➤ Titre : Consulter un quantitatif. ➤ But : Accepter ou bien refuser un quantitatif. ➤ Acteur : Surveillant travaux	

Chef subdivision marché Chef service étude et travaux
Enchainements :
➤ Préconditions : 1. Le surveillant travaux est authentifié. ➤ Démarrage : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour consulter les éléments concernant une affaire. ➤ Enchaînements : - a : l'utilisateur demande la consultation d'un quantitatif. - b : le système affiche l'interface de consultation. - c : l'utilisateur choisit une affaire et demande l'affichage du quantitatif de cette affaire. - d : le système affiche les informations du quantitatif. ➤ Exceptions : [Exception 1 : affaire n'existe pas] un message d'erreur est affiché lorsque l'affaire n'existe pas. ➤ Arrêt: Ce cas d'utilisation termine lorsque la consultation est terminée ou bien annulée. ➤ Postconditions : Néant.

Cas d'utilisation : Ajouter Entreprise
Sommaire d'authentification :
➤ Titre : Ajouter Entreprise. ➤ But : Ajouter une nouvelle entreprise. ➤ Acteur : Chef subdivision marché.
Enchainements :

➤ Préconditions : l'utilisateur est authentifié. ➤ Démarrage : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour ajouter une nouvelle entreprise. ➤ Enchaînements : - a : l'utilisateur demande d'ajouter une nouvelle entreprise. - b : le système affiche le formulaire. - c : l'utilisateur saisit les informations d'une entreprise. - d : l'utilisateur demande au système d'enregistrer. - e : le système effectue une vérification. [Si Erreur] le système affiche un message d'erreur. [Si non] le système enregistre l'affaire et indique la fin de l'opération. ➤ Arrêt : ce cas d'utilisation termine lorsque l'ajout est terminé ou annulé. ➤ Postconditions : une nouvelle entreprise est ajoutée à l'ensemble des entreprises.
--

Cas d'utilisation : Modifier Entreprise
Sommaire d'authentification :
➤ Titre : Modifier Entreprise. ➤ But : Modifier des informations d'une entreprise existante. ➤ Acteur : Chef subdivision marché.
Enchaînements :
➤ Préconditions : 1. L'utilisateur est authentifié. 2. L'existence d'une entreprise non validée. ➤ Démarrage : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour modifier des informations d'une entreprise non acceptée. ➤ Enchaînements : - a : l'utilisateur demande la modification d'une entreprise. - b : le système affiche le formulaire. - c : l'utilisateur sélectionne une entreprise. - c : le système affiche les informations de l'entreprise.

d : l'utilisateur modifie l'entreprise.

e : l'utilisateur demande au système d'enregistrer.

f : le système effectue une vérification et affiche un message de confirmation de modifications des informations.

[Si Erreur] le système affiche un message d'erreur.

[Si non] le système enregistre les modifications et indique la fin de l'opération.

[Exception 1 : entreprise n'existe pas] un message d'erreur est affiché lorsque l'entreprise n'existe pas.

- **Arrêt** : Ce cas d'utilisation termine lorsque la modification d'une entreprise est terminée ou bien annulée.
- **Postconditions** : Entreprise modifiée.

Cas d'utilisation : Supprimer Entreprise

Sommaire d'authentification :

- **Titre** : Supprimer Entreprise.
- **But** : Supprimer des informations d'une entreprise existante.
- **Acteur** : Chef subdivision marché.

Enchainements :

- **Préconditions** : L'utilisateur est authentifié.
- **Démarrage** : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour supprimer des informations d'une entreprise.
- **Enchainement** :
 - a : L'utilisateur demande la suppression d'une entreprise.
 - b : Le système affiche le formulaire.
 - c : L'utilisateur sélectionne une entreprise et la supprime.
 - d : Le système valide la suppression en éliminant cette entreprise de la base de donnée.
- **Arrêt** : Ce cas d'utilisation termine lorsque la suppression est terminée ou bien annulée.

- **Postconditions** : L'entreprise est supprimée.

Cas d'utilisation : Etablir Contrat

Sommaire d'authentification :

- **Titre** : Etablir Contrat.
- **But** : Créer un nouveau contrat avec une entreprise.
- **Acteur** : Chef subdivision marché.

Enchaînements :

- **Préconditions** : L'utilisateur est authentifié.
- **Démarrage** : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour créer un nouveau contrat avec une entreprise de réalisation.
- **Enchaînements** :
 - a : l'utilisateur demande la création d'un nouveau contrat.
 - b : l'utilisateur saisit le numéro d'affaire acceptée et demande d'afficher.
 - c : le système affiche le formulaire de l'affaire.
 - d : le système affiche la liste des entreprises.
 - e : l'utilisateur demande d'enregistrer.
 - f : l'utilisateur entre les informations du contrat.
 - g : l'utilisateur demande au système l'enregistrement.
 - h : le système effectue une vérification.

[Si Erreur] le système affiche un message d'erreur.

[Si non] le système enregistre le contrat et indique la fin de l'opération
- **Arrêt**: Ce cas d'utilisation termine lorsque l'ajout est terminé ou bien annulée.
- **Postconditions** : Un nouveau contrat est ajouté à l'ensemble des contrats.

Cas d'utilisation : Ajouter Contrat	
Sommaire d'authentification :	
➤ Titre : Ajouter Contrat. ➤ But : Ajouter un nouveau contrat. ➤ Acteur : Chef subdivision marché.	
Enchainements :	
➤ Préconditions : l'utilisateur est authentifié. ➤ Démarrage : ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour ajouter un nouveau contrat. ➤ Enchaînements : - a : l'utilisateur demande d'ajouter un nouveau contrat. - b : le système affiche le formulaire. - c : l'utilisateur saisit les informations d'un contrat. - d : l'utilisateur demande au système d'enregistrer. - e : le système effectue une vérification. [Si Erreur] le système affiche un message d'erreur. [Si non] le système enregistre l'affaire et indique la fin de l'opération. ➤ Arrêt : ce cas d'utilisation termine lorsque l'ajout est terminé ou annulé. ➤ Postconditions : un nouveau contrat est ajouté à l'ensemble des contrats.	

Cas d'utilisation : Modifier Contrat	
Sommaire d'authentification :	
➤ Titre : Modifier Contrat. ➤ But : Modifier des informations d'un contrat existant. ➤ Acteur : Chef subdivision marché.	

Enchainements :

- **Préconditions** : 1. L'utilisateur est authentifié.
2. L'existence d'un contrat non validé.
- **Démarrage** : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour modifier des informations d'un contrat non accepté.
- **Enchaînements** :
 - a : l'utilisateur demande la modification d'un contrat.
 - b : le système affiche le formulaire.
 - c : l'utilisateur sélectionne un contrat.
 - c : le système affiche les informations du contrat.
 - d : l'utilisateur modifie l'élément.
 - e : l'utilisateur demande au système d'enregistrer.
 - f : le système effectue une vérification et affiche un message de confirmation de modifications des informations.
 - [Si Erreur] le système affiche un message d'erreur.
 - [Si non] le système enregistre les modifications et indique la fin de l'opération.
 - [Exception 1 : contrat n'existe pas] un message d'erreur est affiché lorsque Le contrat n'existe pas.
- **Arrêt** : Ce cas d'utilisation termine lorsque la modification d'un contrat est terminée ou bien annulé.
- **Postconditions** : contrat modifié.

Cas d'utilisation : Supprimer Contrat

Sommaire d'authentification :

- **Titre** : Supprimer Contrat.
- **But** : Supprimer des informations d'un contrat existant.
- **Acteur** : Chef subdivision marché.

Enchainements :

- **Préconditions** : L'utilisateur est authentifié.
- **Démarrage** : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour supprimer des informations d'un contrat.
- **Enchaînement** :
 - a : L'utilisateur demande la suppression d'un contrat.
 - b : Le système affiche le formulaire.
 - c : L'utilisateur sélectionne un contrat et le supprime.
 - d : Le système valide la suppression en éliminant ce contrat de la base de donnée.
- **Arrêt** : Ce cas d'utilisation termine lorsque la suppression est terminée ou bien annulée.
- **Postconditions** : Le contrat est supprimé.

Cas d'utilisation : Valider Contrat

Sommaire d'authentification :

- **Titre** : Valider le contrat avec l'entreprise.
- **But** : Accepter ou bien refuser un contrat.
- **Acteur** : Directeur.

Enchaînements :

- **Préconditions** : 1. Le directeur est authentifié.
2. L'existence d'un nouveau contrat non validé.
- **Démarrage** : Ce cas d'utilisation commence lorsque l'utilisateur se connecte au système pour valider un contrat d'une entreprise de réalisation.
- **Enchaînements** :
 - a : l'utilisateur demande la validation d'un contrat.
 - b : l'utilisateur choisit une affaire et demande l'affichage du contrat.
 - d : Le système affiche le formulaire du contrat.

e : L'utilisateur saisit la décision.

f : L'utilisateur demande au système l'enregistrement.

g : Le système enregistre la validation et indique la fin de l'opération.

- **Arrêt** : Ce cas d'utilisation termine lorsque la validation d'un contrat est terminée ou bien annulé.
- **Postconditions** : le contrat validé.

Tableau 3.3 : Description textuelle des cas d'utilisation fonctionnelle.

2.5.2. Description graphique des cas d'utilisation :

La description textuelle des cas d'utilisation est insuffisante pour leur documentation, elles permettent de communiquer aux utilisateurs de manière facile et précise donc il doit compléter la description textuelle par les diagrammes dynamiques correspondants.

Nous avons choisis de compléter la description par les diagrammes de séquence.

Diagrammes de séquence :

- **Diagramme de séquence « Proposer affaire »**

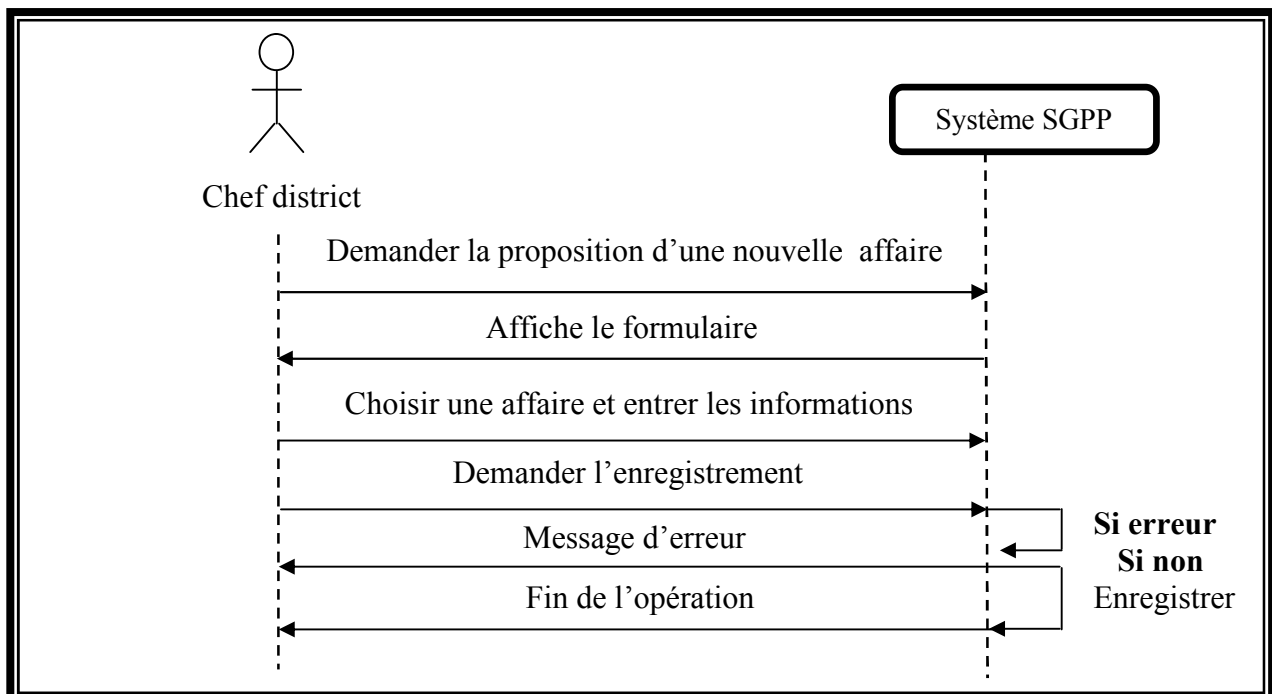


Figure 3.2 : Diagramme de séquence « Proposer affaire »

➤ **Diagramme de séquence « Modifier affaire »**

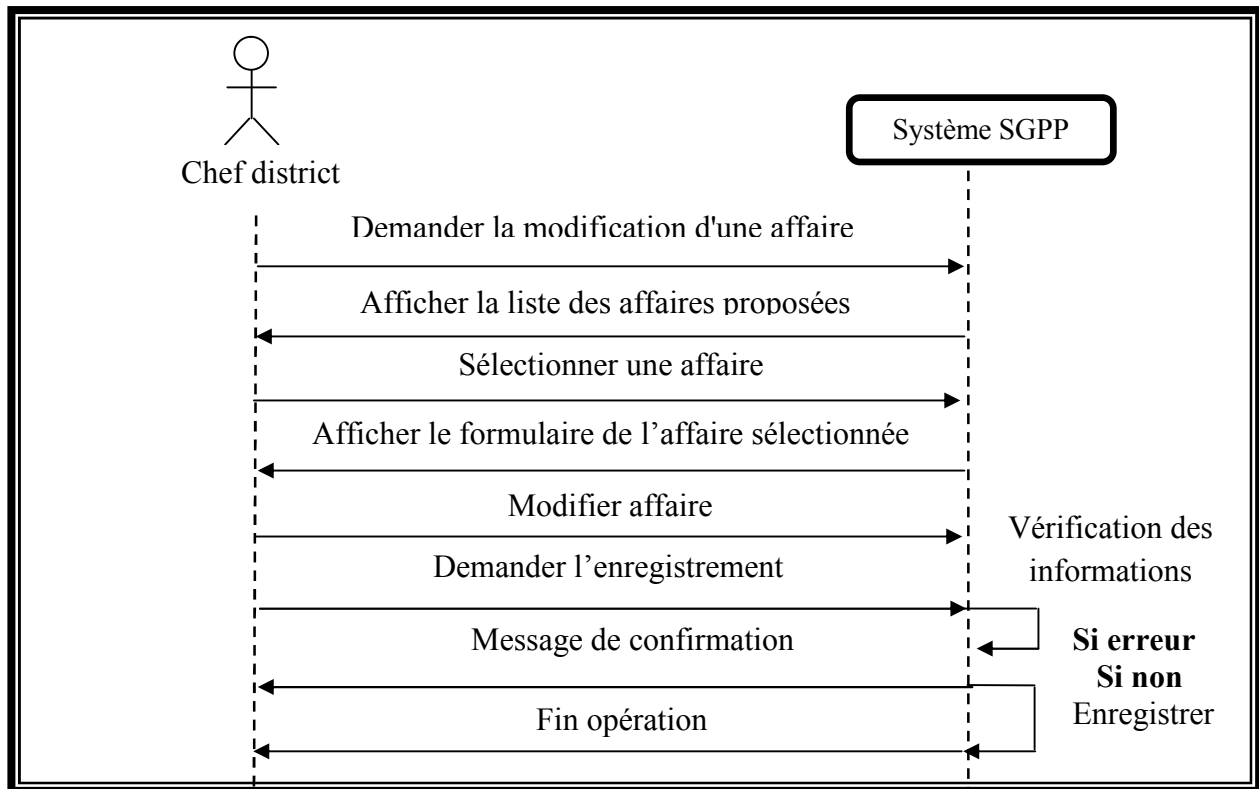


Figure 3.3 : Diagramme de séquence « Modifier affaire »

➤ **Diagramme de séquence « Supprimer affaire »**

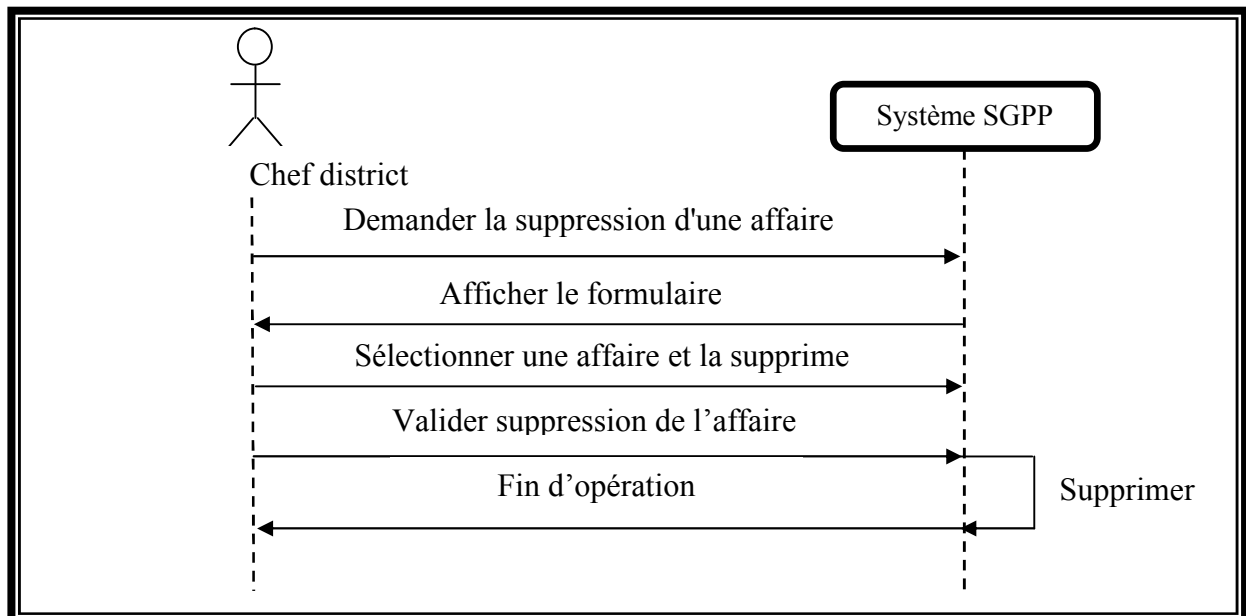


Figure 3.4 : Diagramme de séquence « Supprimer affaire »

➤ Diagramme de séquence « Valider affaire »

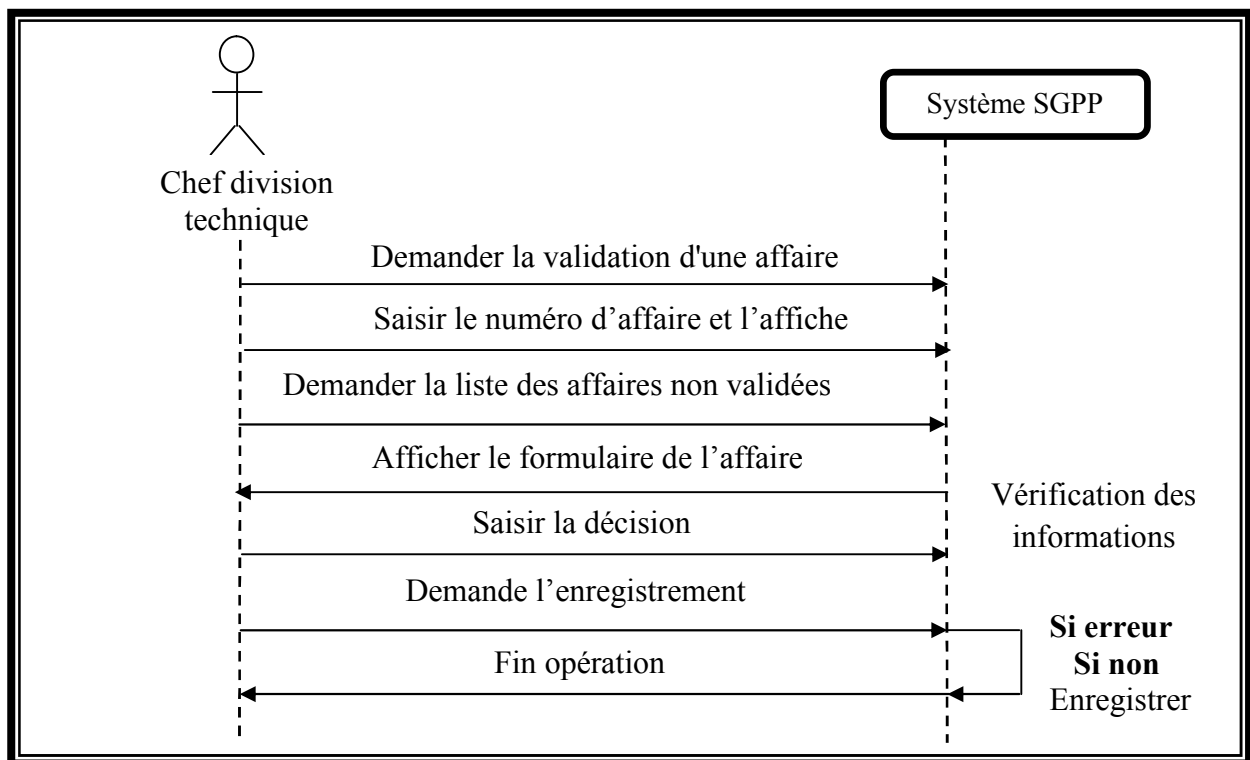


Figure 3.5 : Diagramme de séquence « Valider affaire »

➤ Diagramme de séquence « Etablir Avant projet »

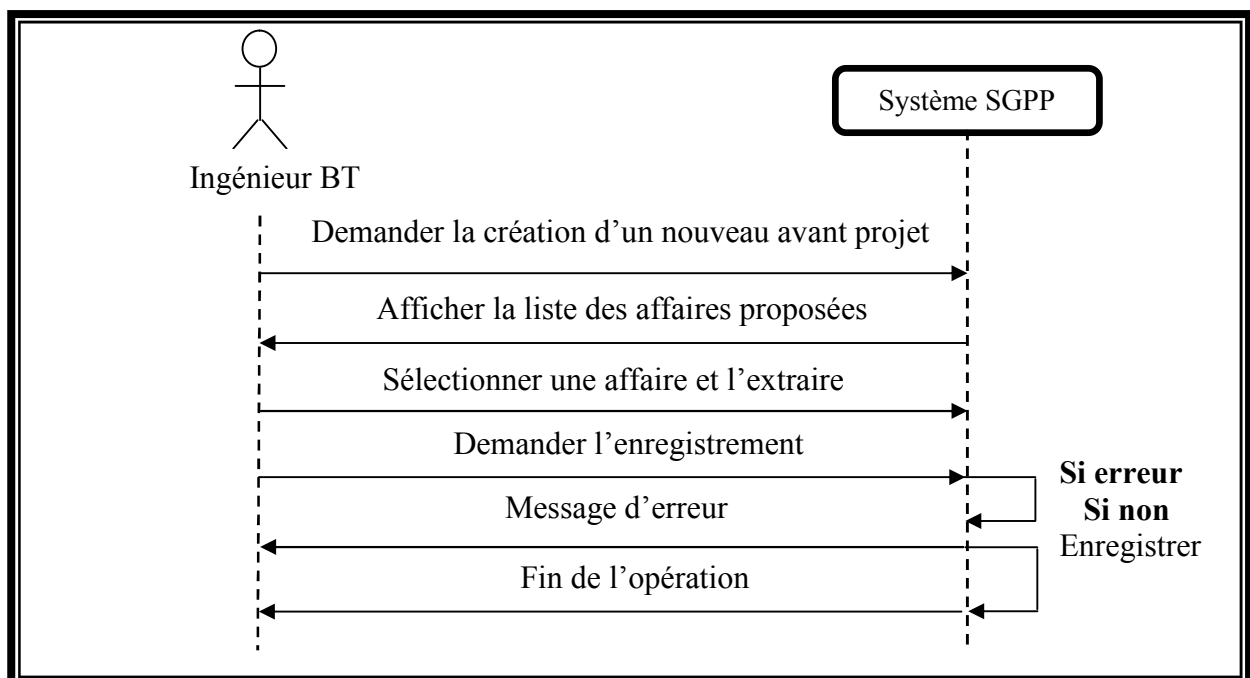


Figure 3.6 : Diagramme de séquence « Etablir Avant projet »

➤ Diagramme de séquence « Affecter élément»

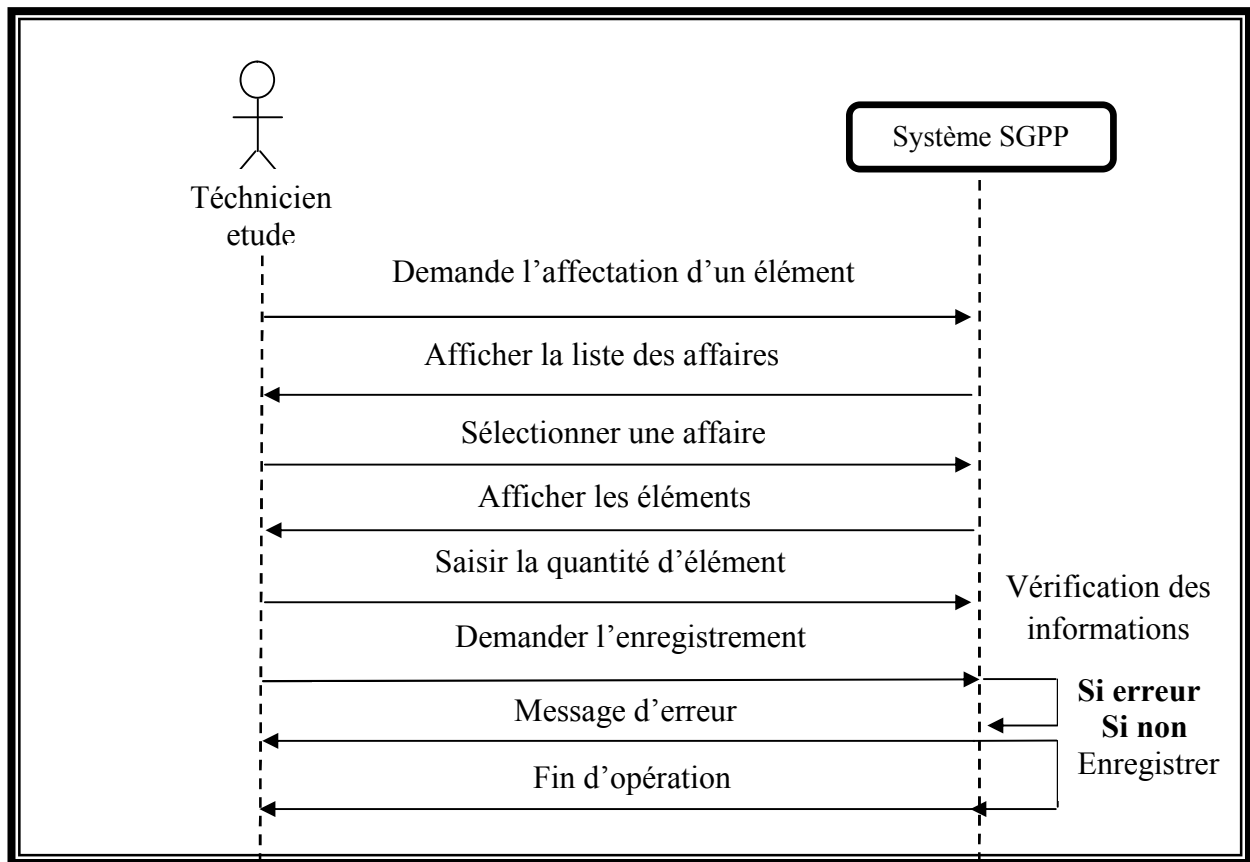


Figure 3.7 : Diagramme de séquence « Affecter élément»

➤ Diagramme de séquence « Modifier Élément affecté »

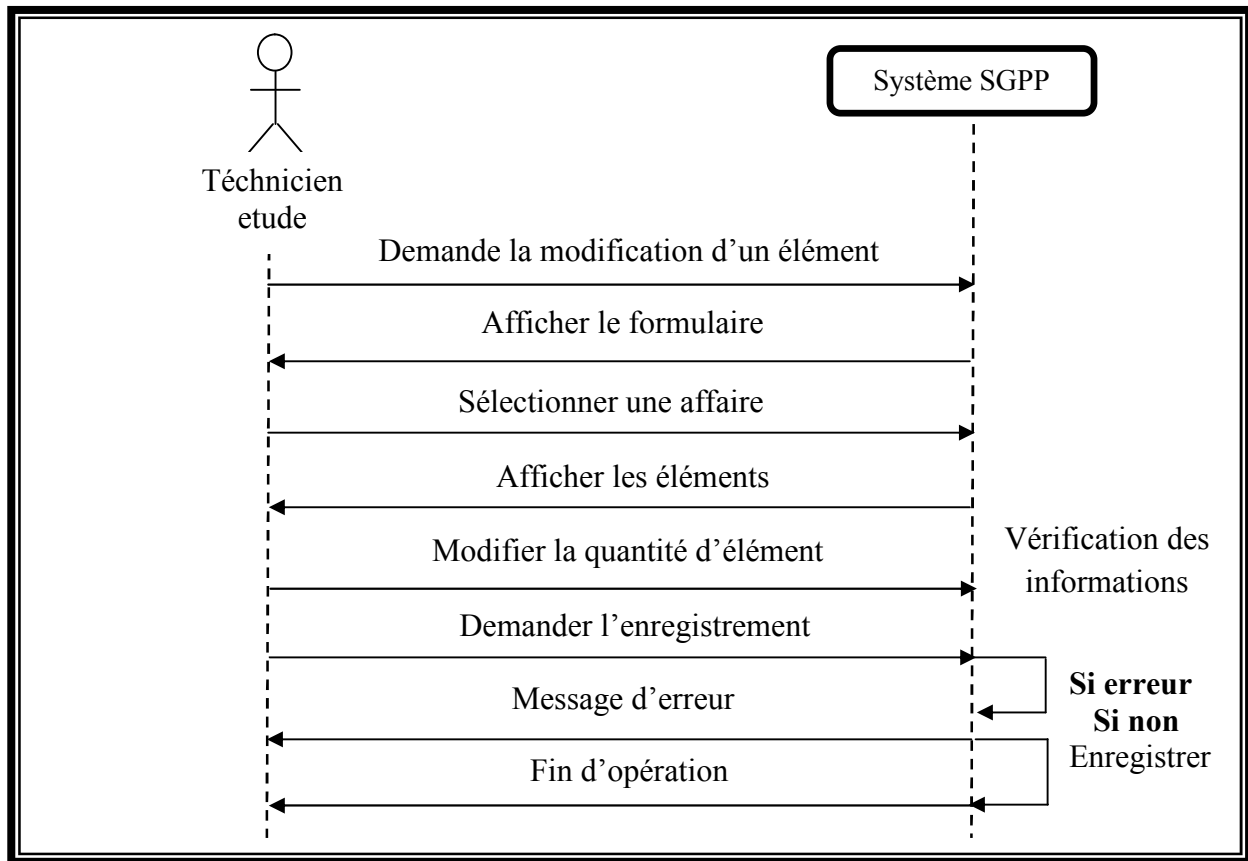


Figure 3.8 : Diagramme de séquence «Modifier Élément affecté»

➤ Diagramme de séquence « Supprimer Élément affecté »

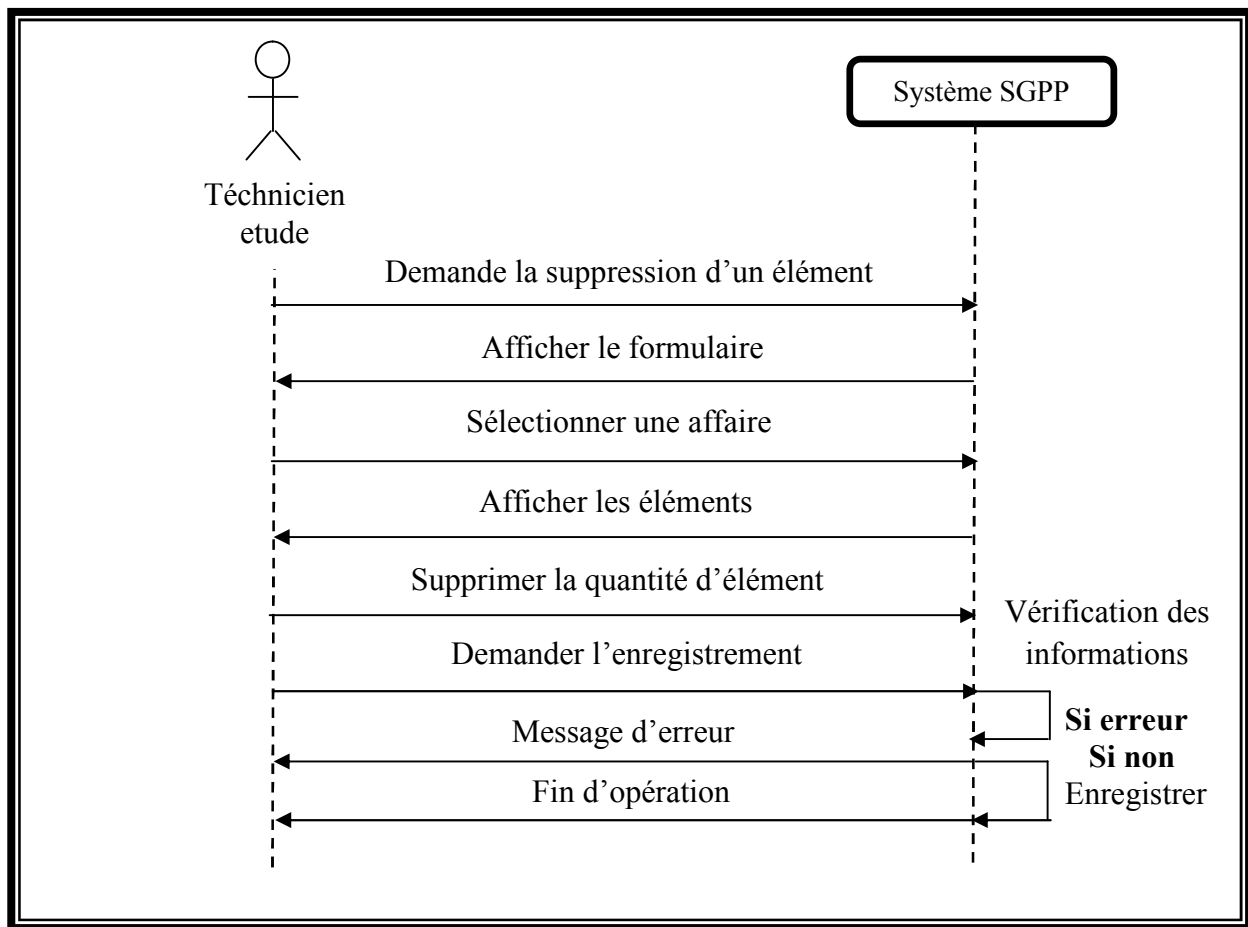


Figure 3.9 : Diagramme de séquence «Supprimer Élément affecté»

➤ Diagramme de séquence « Ajouter Élément »

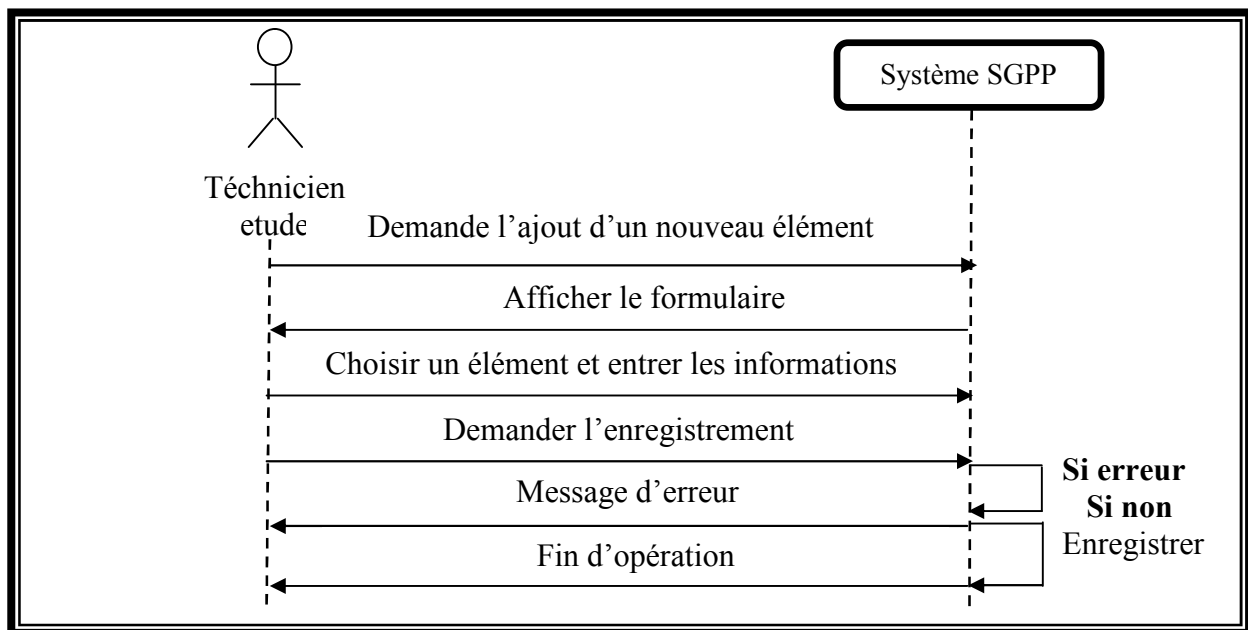


Figure 3.10 : Diagramme de séquence «Ajouter Élément»

➤ Diagramme de séquence « Modifier Élément »

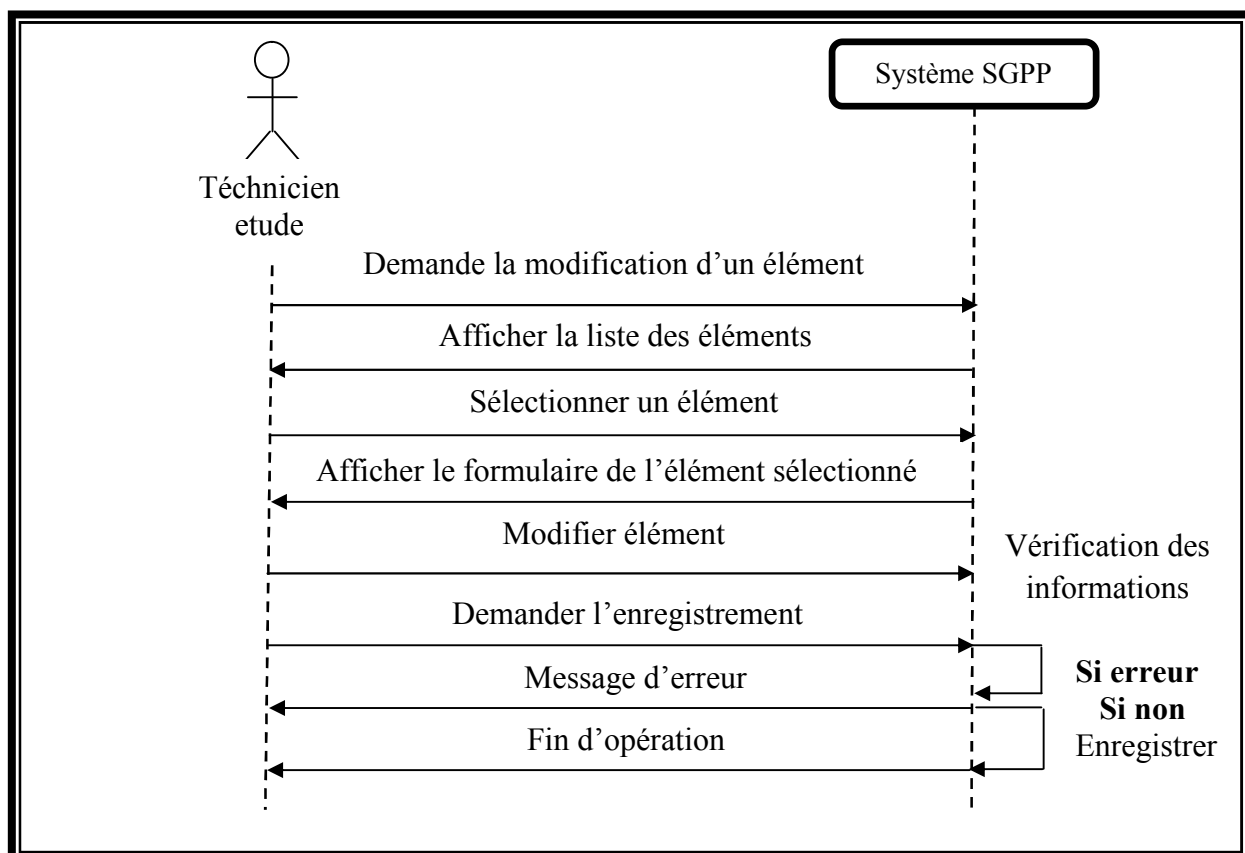


Figure 3.11 : Diagramme de séquence «Modifier Élément»

➤ **Diagramme de séquence «Supprimer Elément»**

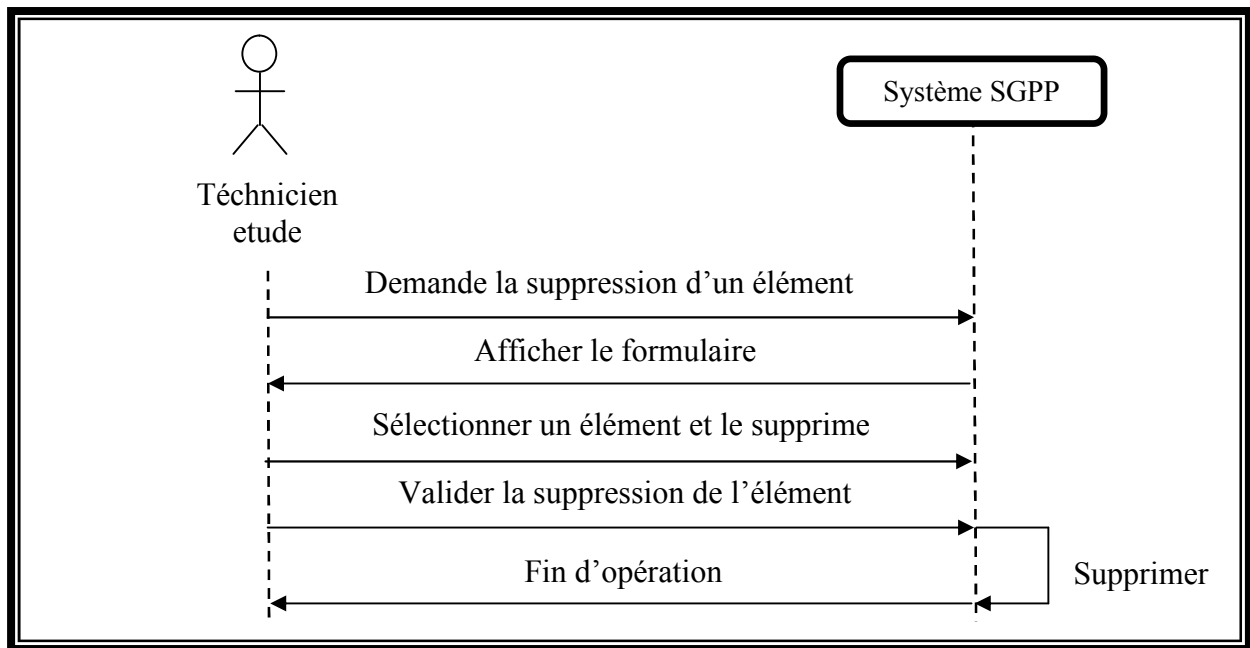


Figure 3.12 : Diagramme de séquence «Supprimer Elément»

➤ **Diagramme de séquence « Consulter quantitatif»**

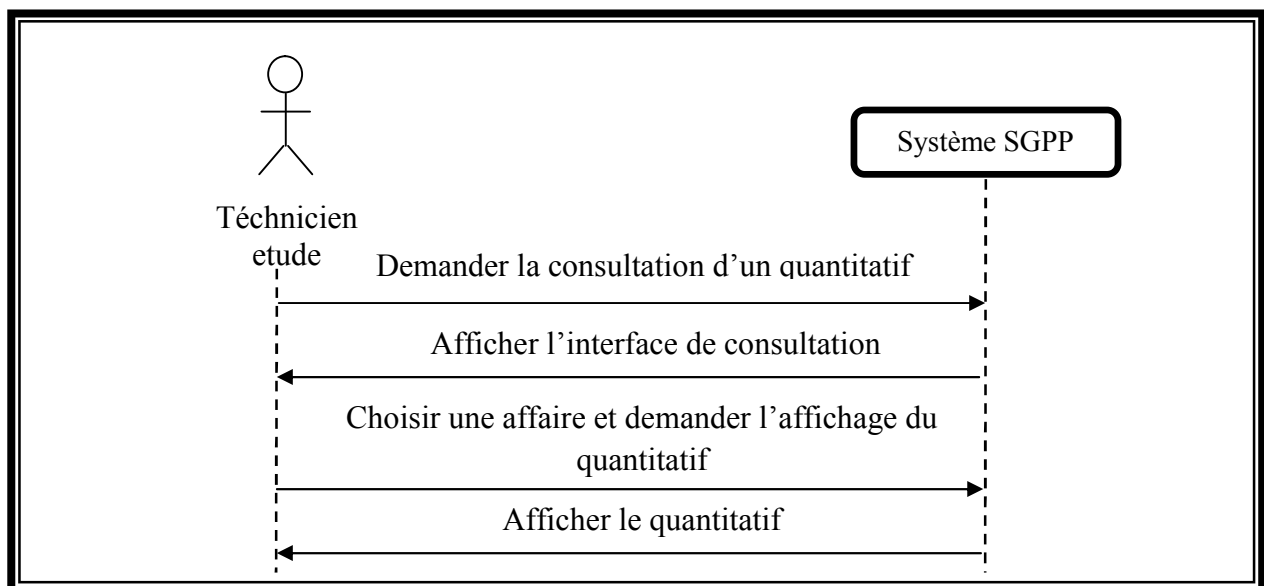


Figure 3.13: Diagramme de séquence «Consulter quantitatif»

➤ Diagramme de séquence «Ajouter Entreprise»

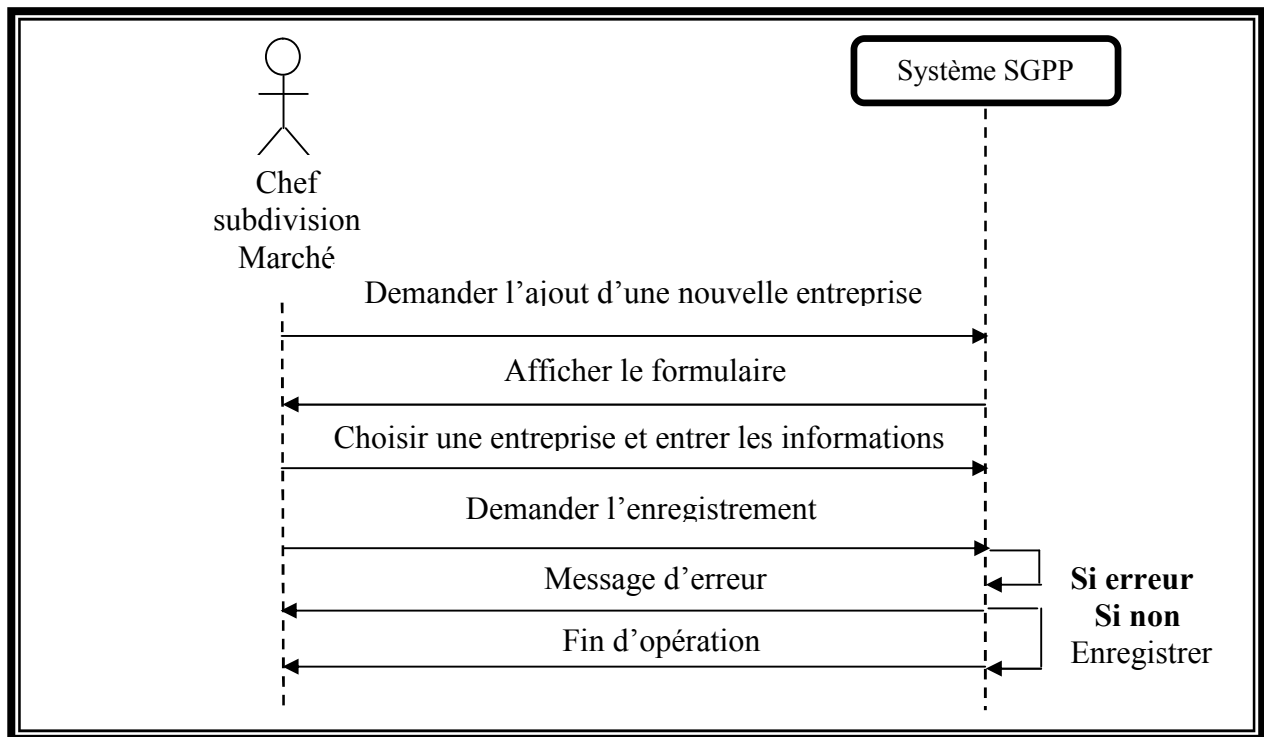


Figure 3.14: Diagramme de séquence «Ajouter Entreprise»

➤ Diagramme de séquence «Modifier Entreprise»

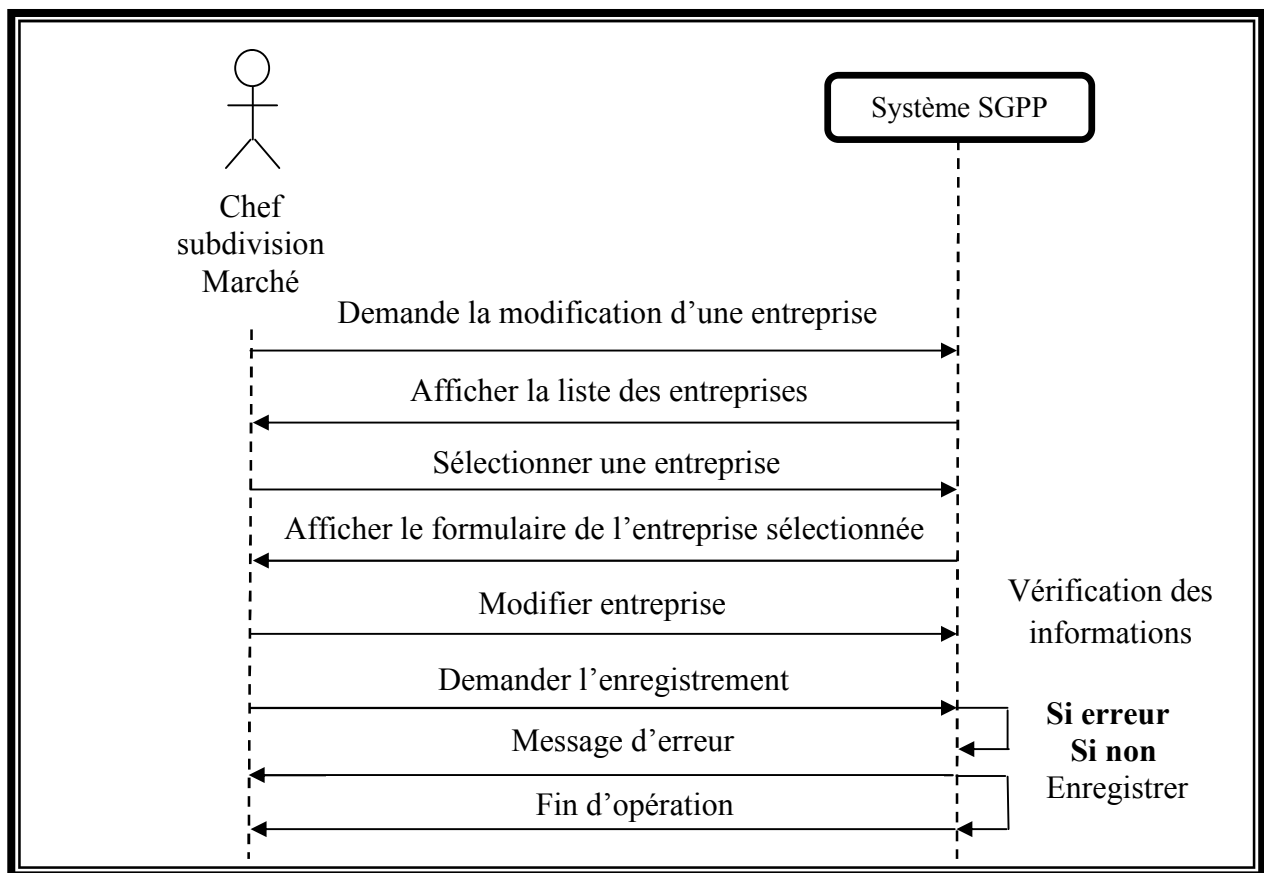


Figure 3.15: Diagramme de séquence «Modifier Entreprise»

➤ Diagramme de séquence «Supprimer Entreprise»

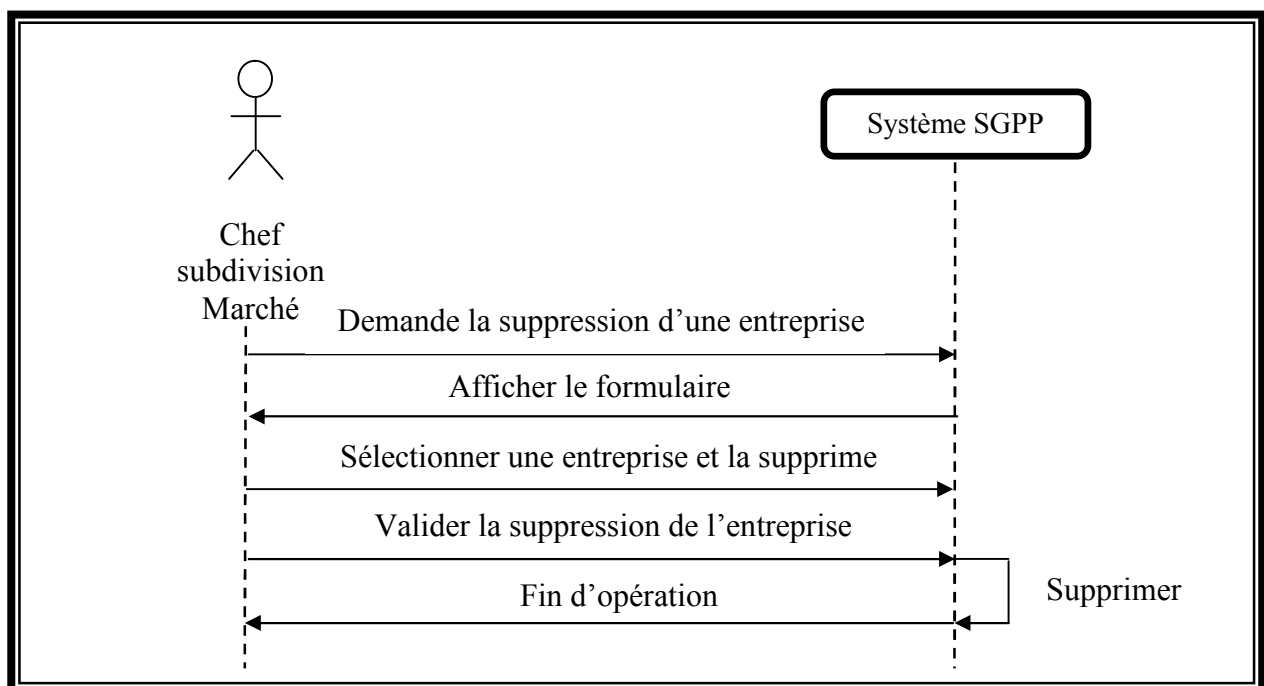


Figure 3.16: Diagramme de séquence «Supprimer Entreprise»

➤ Diagramme de séquence «Etablir Contrat»

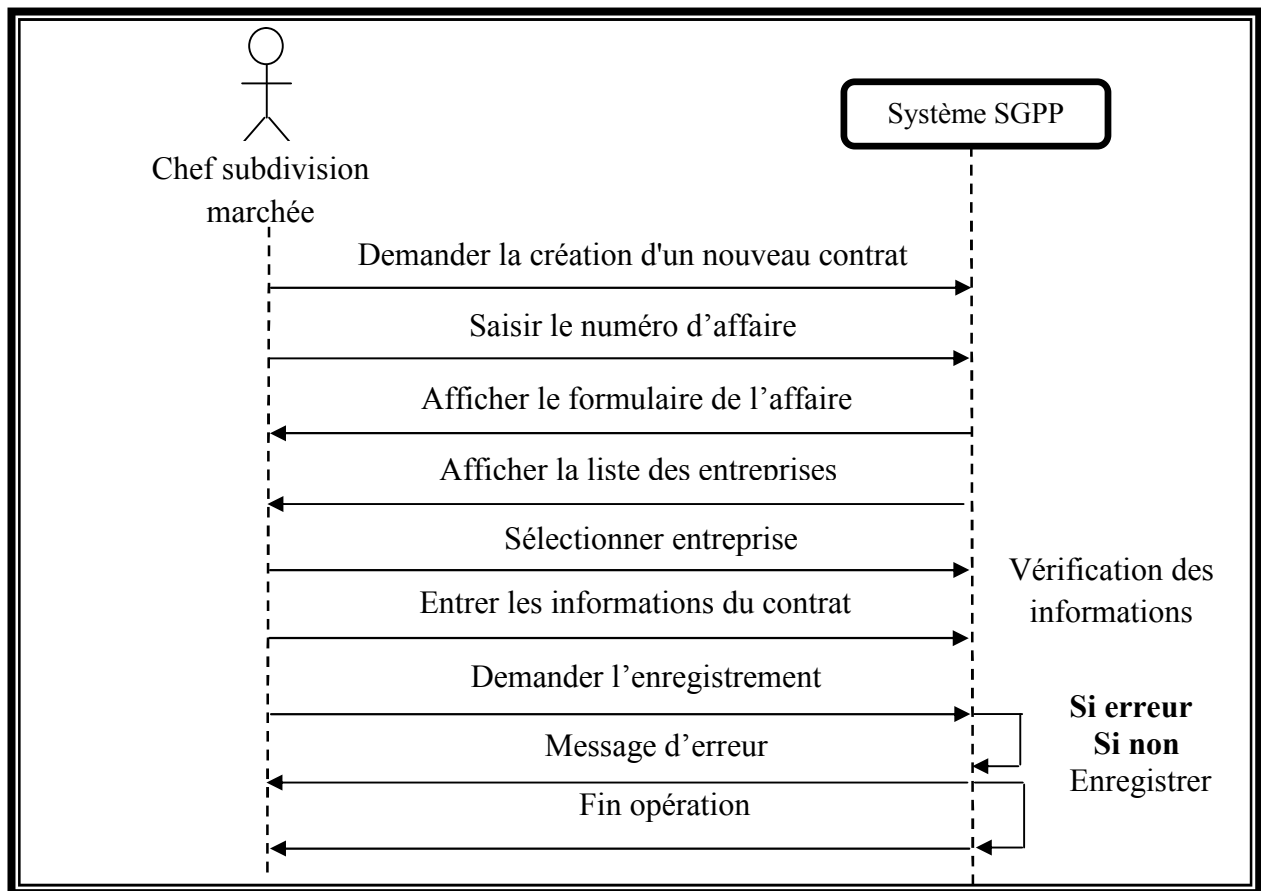


Figure 3.17 : Diagramme de séquence «Etablir Contrat»

➤ Diagramme de séquence «Ajouter Contrat»

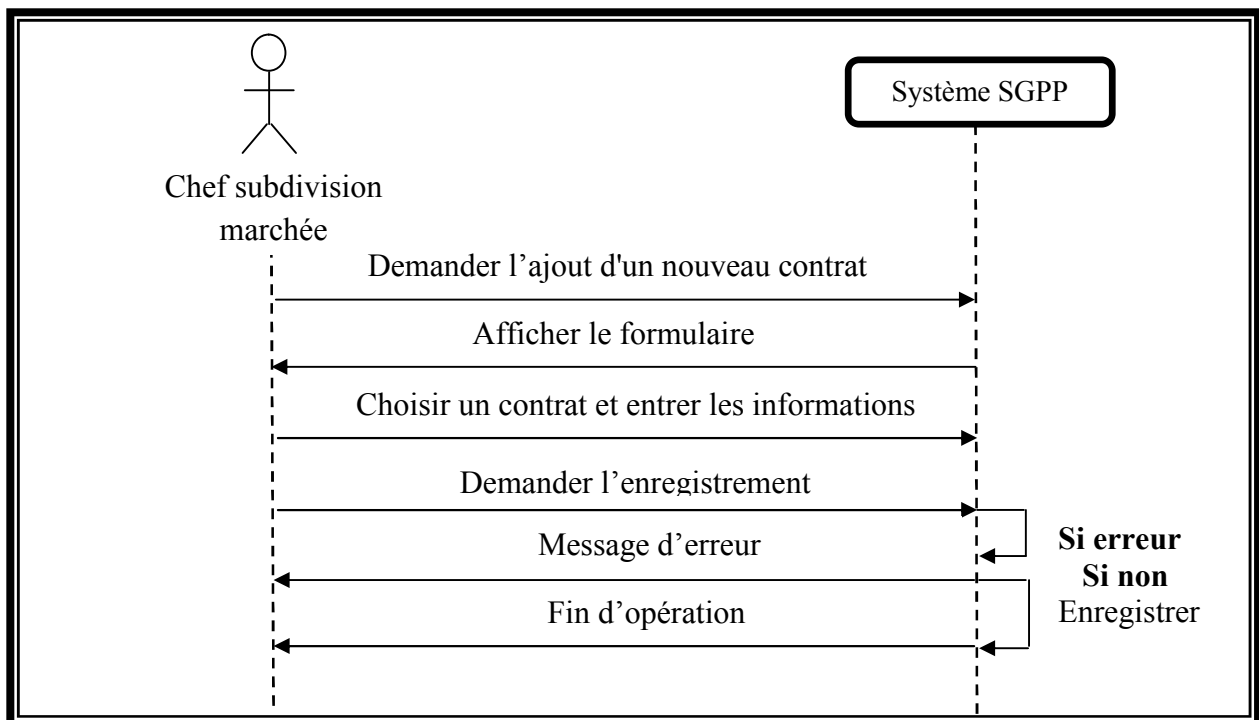


Figure 3.18 : Diagramme de séquence «Ajouter Contrat»

➤ Diagramme de séquence «Modifier Contrat»

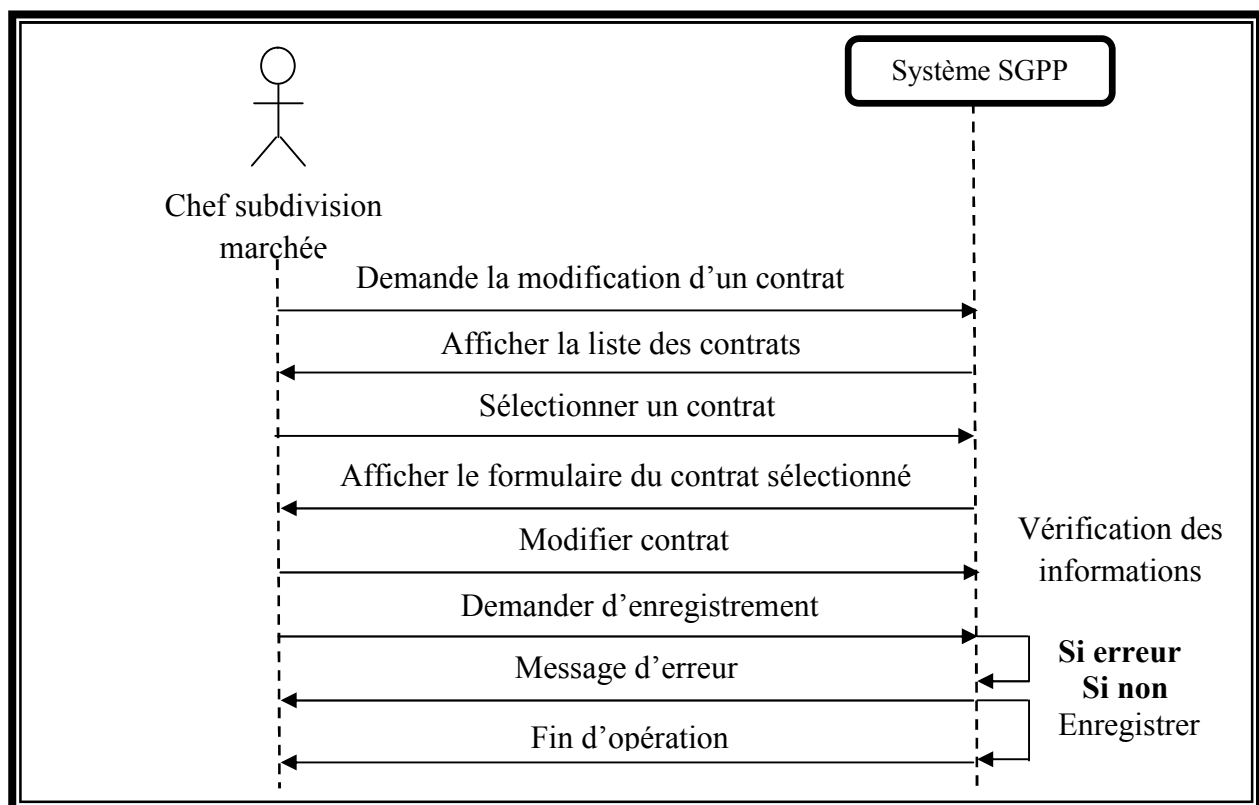


Figure 3.19 : Diagramme de séquence «Modifier Contrat»

➤ Diagramme de séquence «Supprimer Contrat»

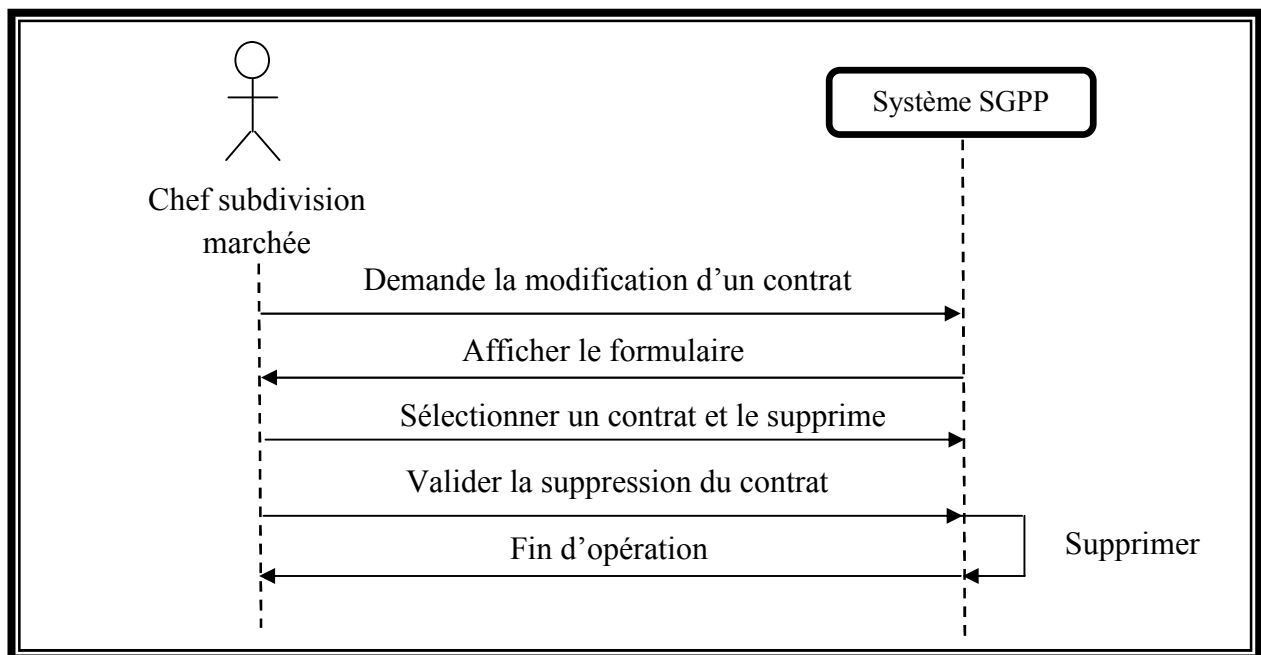


Figure 3.20 : Diagramme de séquence «Supprimer Contrat»

➤ Diagramme de séquence : « Valider contrat»

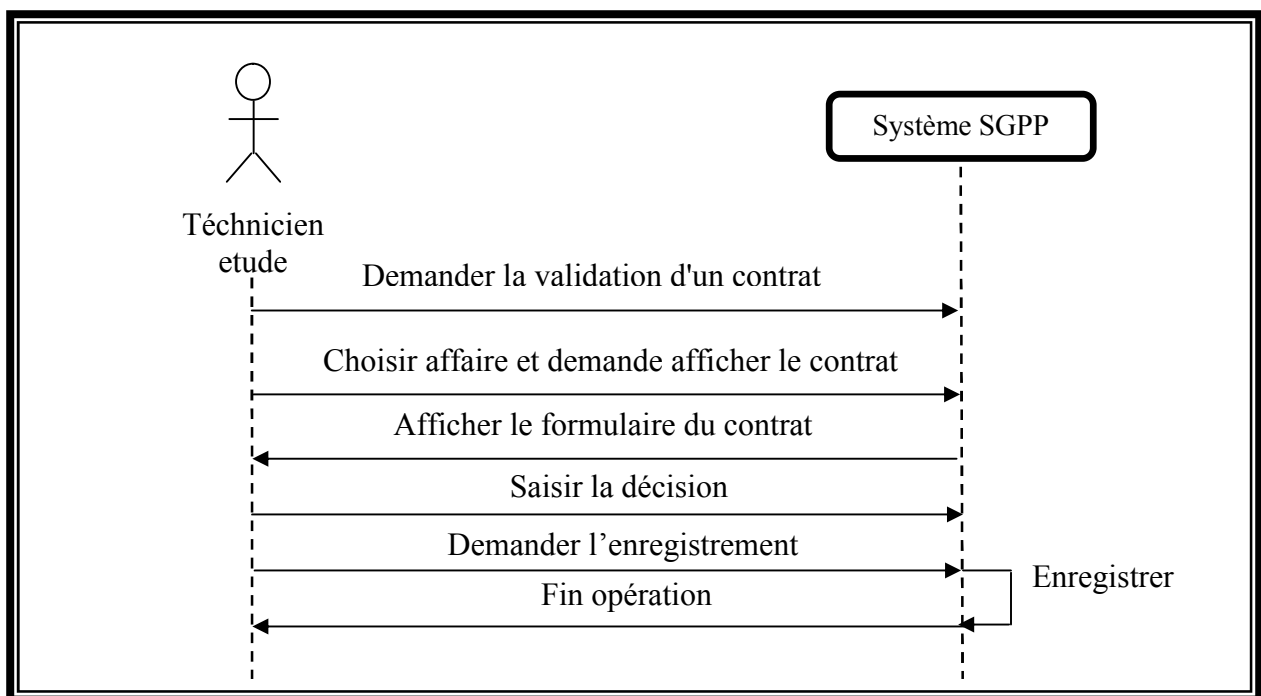


Figure 3.21 : Diagramme de séquence «Valider contrat»

2.6. Identification des classes candidates :

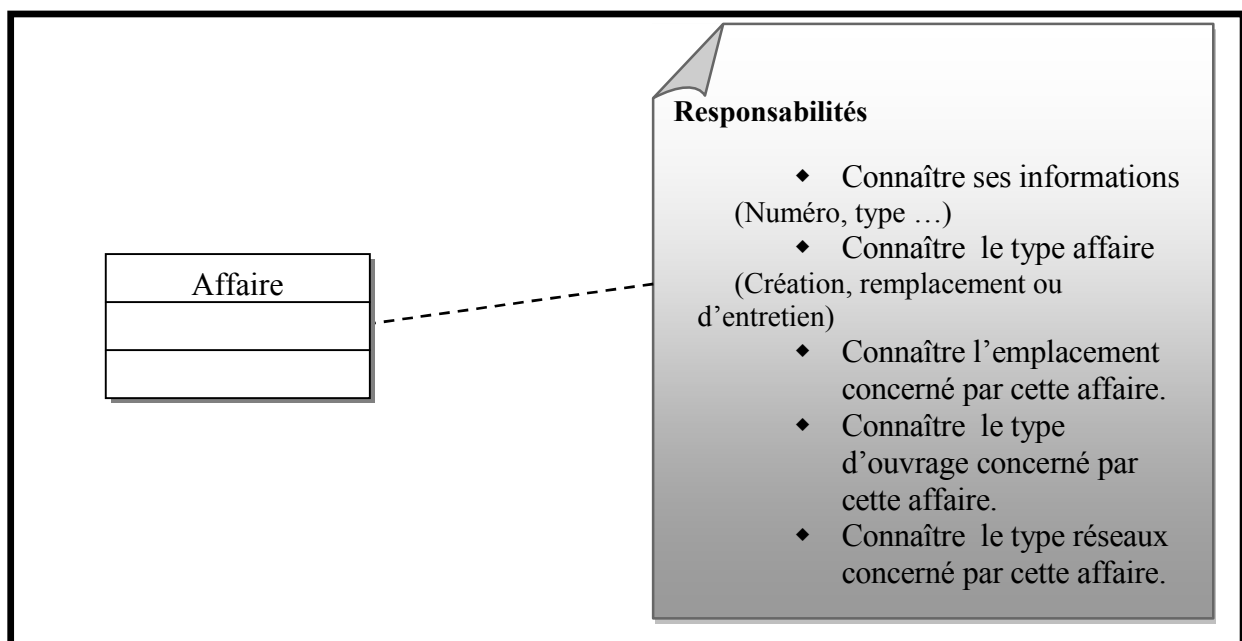
2.6.1. La liste des classes candidates du système SGPP :

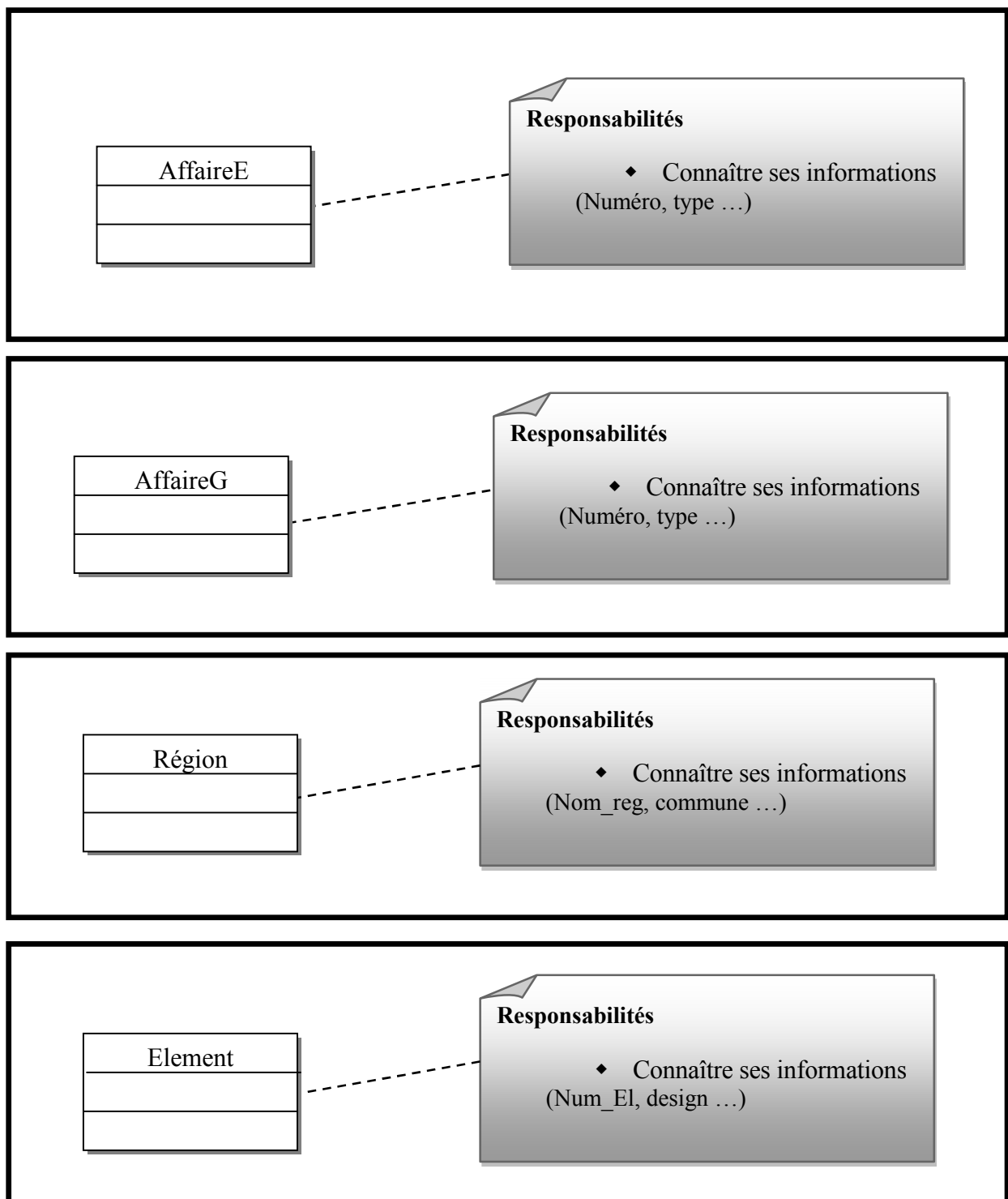
Liste des classes candidates
Affaire. Affaire Electricité (AffaireE) Affaire Gaz (AffaireG). Région. Elément. Réseau (Type_res). Entreprise. Contrat. Employé.

Tableau 3.4: liste des classes candidates du SGPP.

2.6.2. Responsabilités des classes :

Une responsabilité est une sorte de contrat, ou d'obligation, pour une classe. Elle se place à un niveau d'abstraction plus élevé que les attributs ou les opérations. En fait, on peut dire que les attributs, les opérations, et les associations représentent les propriétés élémentaires qui contribueront à remplir les responsabilités de la classe. [2]





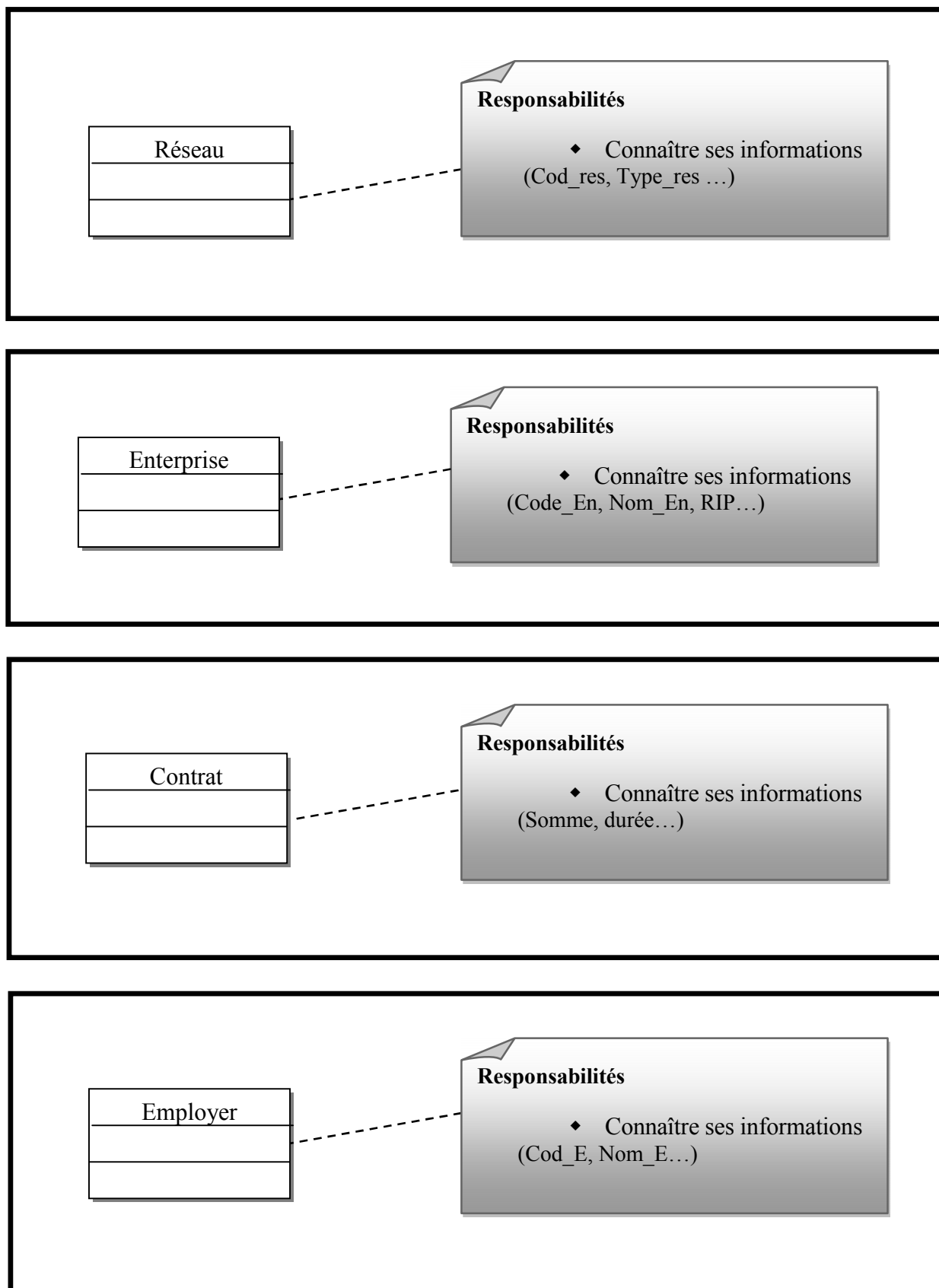


Tableau 3.5: Responsabilité des classes candidates.

2.6.3. Diagramme de classes participantes :

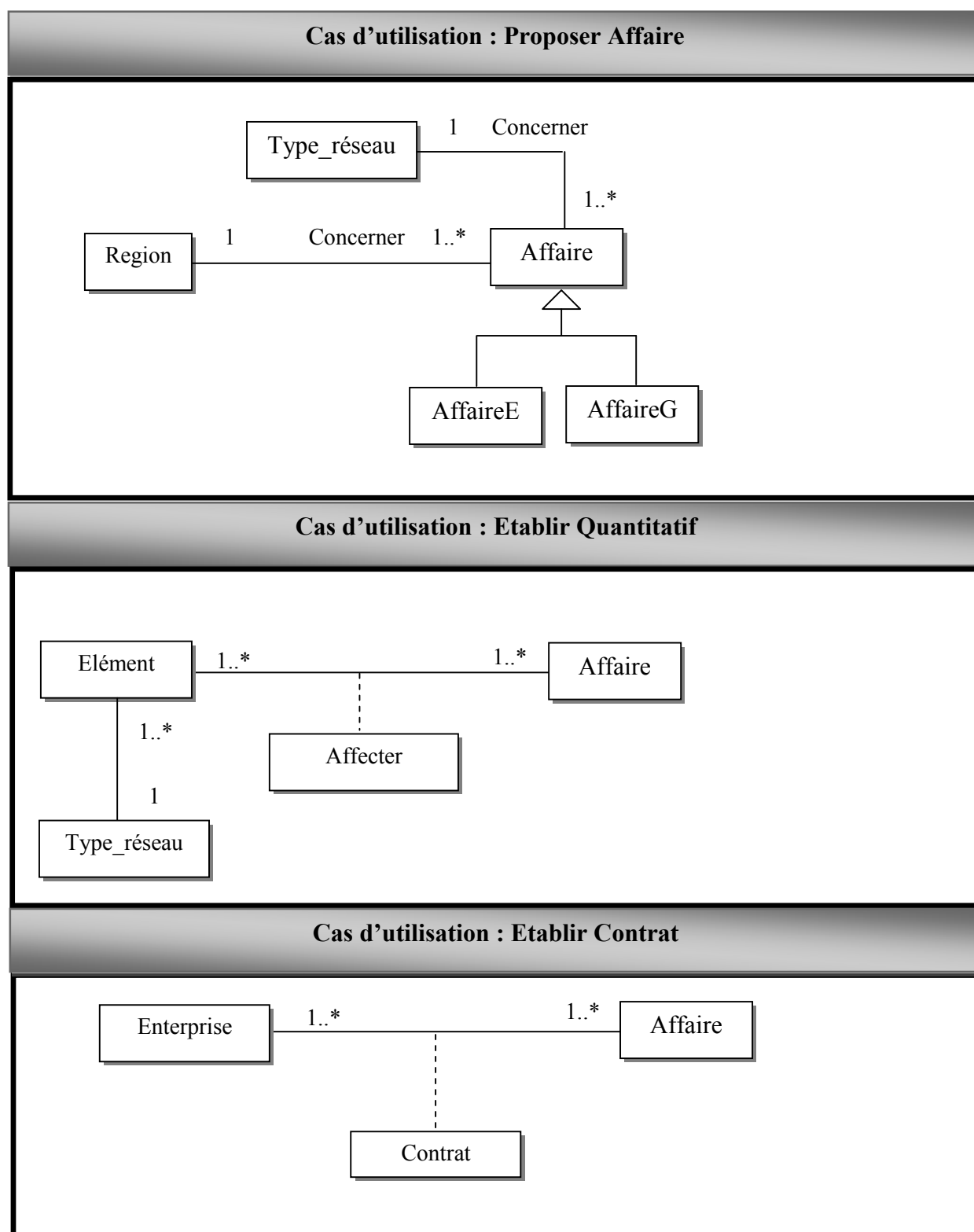


Tableau 3.6 : Diagrammes des classes participantes des cas d'utilisation.

3. Capture des besoins techniques :

La capture des besoins techniques est la première étape de la branche droite du cycle en Y, elle couvre par complémentarité avec celle des besoins fonctionnels, toutes les contraintes qui ne traitent ni de la description du métier des utilisateurs, ni de la description applicative, elle est primordiale pour la conception de l'architecture.

Les étapes à suivre sont décrites dans ce qui suit :

3.1. Spécification technique du point de vue matériel :

Les choix stratégiques de développement impliquent les contraintes relatives à la configuration du réseau matériel, elles sont de nature géographique, organisationnelle, et technique.

3.1.1. Style d'architecture utilisée :

La configuration géographique du système SGPP impose le développement d'une solution client/serveur à deux niveaux:

- Un niveau département pour les informations du bureau partagées entre les postes de travail.
- Un niveau local pour l'application à déployer sur les postes de travail.

3.1.2. Configuration matérielle du système SGPP:

La configuration matérielle de notre système est schématisée par le diagramme de déploiement suivant :

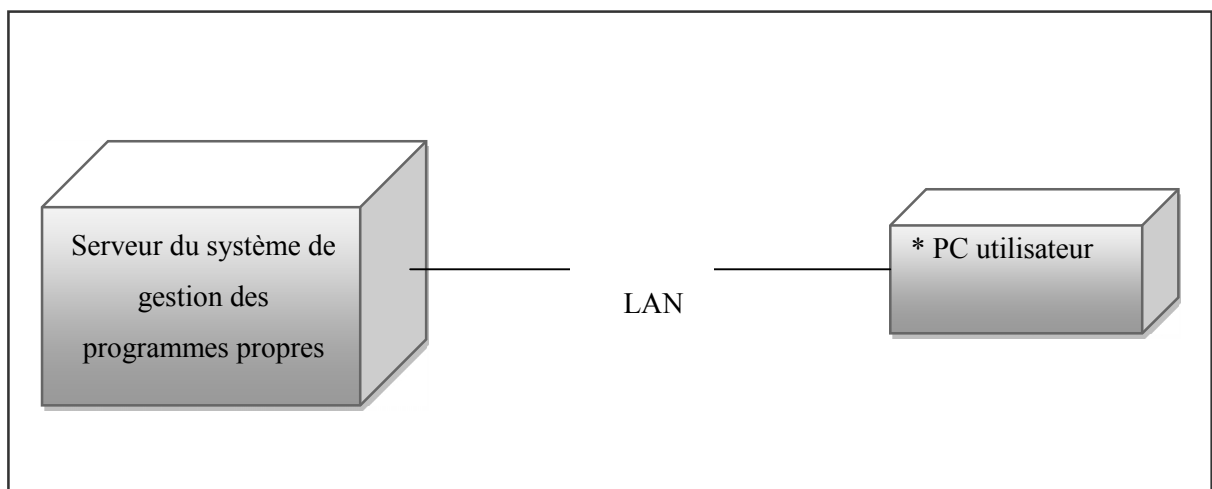


Figure3.22 : Configuration matérielle du système de gestion des programmes propres.

3.2. Spécification d'architecture:

3.2.1. Style d'architecture en tiers:

Le style d'architecture 2-tiers correspond à la configuration la plus simple d'un système client/serveur, dans ce cas, il incombe aux clients de gérer l'interface utilisateur et les processus d'exploitation. Les serveurs ont pour responsabilité de traiter le stockage de données.

Ce type d'architecture est parfaitement adapté aux systèmes départementaux.

3.2.2. Modèle préliminaire des composants du système:

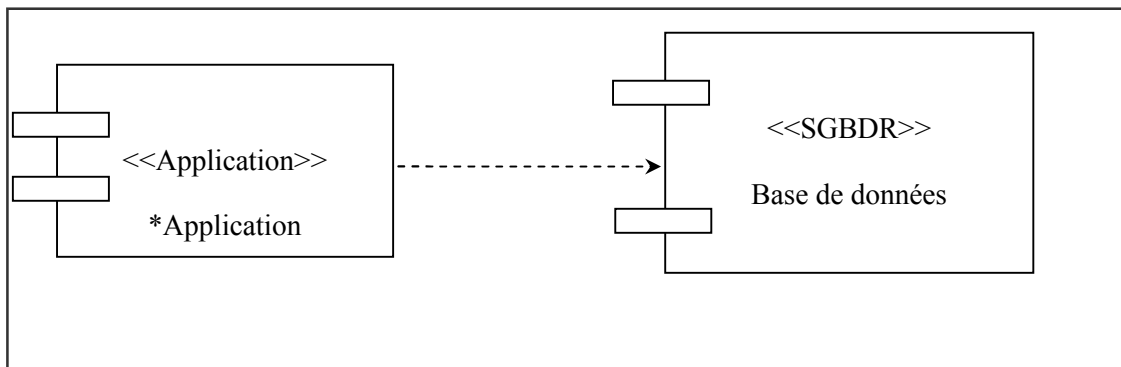


Figure 3.23: Modèle préliminaire des composants du SGPP.

3.3. Élaboration du modèle de spécification logicielle :

Une fois la spécification technique est exprimée, on s'intéresse aux fonctionnalités propres du système technique en procédant à une spécification logicielle. Dans ce cas, on utilise les cas d'utilisation de manière différente que pour la spécification fonctionnelle.

C'est pourquoi nous introduisons le concept d'exploitant et de cas d'utilisation technique.

3.3.1. Définitions :

- **Exploitant :** l'exploitant est un acteur au sens d'UML, si ce n'est qu'il ne bénéficie que des fonctionnalités techniques du système.

Tout système informatique possède au minimum un exploitant qui est «l'utilisateur du système». Il s'agit ici de l'utilisateur dans son sens le plus général, indépendamment des fonctions ou du métier qu'il réalise au travers de l'application.

- **Cas d'utilisation technique :** un cas d'utilisation technique est destiné à l'exploitant. C'est une séquence d'actions produisant une valeur ajoutée opérationnelle ou purement technique.

3.3.2. Modèle de spécification logicielle :

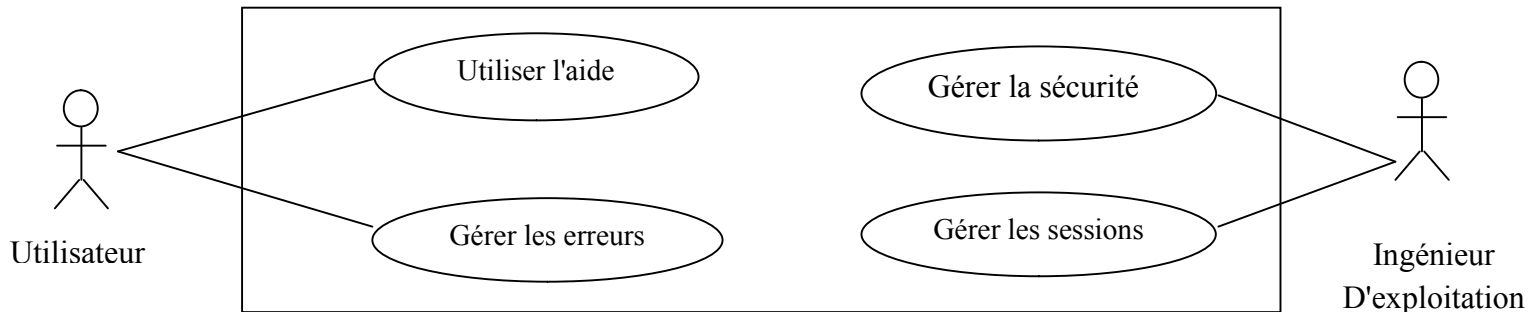


Figure3.24: Modèle de spécification logicielle.

3.3.3. Description textuelle des cas d'utilisation techniques:

Cas d'utilisation : Utiliser l'aide
Intention: Tous les utilisateurs du système peuvent utiliser l'aide contextuelle du système.
Action: offrir un module d'aide pour faire fonctionner le système.
Cas d'utilisation : Gérer les erreurs
Intention: gérer les erreurs de frappe, et de manipulation.
Action: spécifier l'erreur, puis la traiter.
Cas d'utilisation : Gérer la sécurité
Intention: tous les utilisateurs doivent être connectés, s'authentifier et reconnus par le système.
Action: vérification et autorisation d'accès au système permission d'exécuter des opérations dans le système.

Cas d'utilisation : Gérer les sessions
Intention: donner la possibilité aux utilisateurs de faire des opérations différentes.
Action: établir une connexion au serveur.

Tableau3.7: Description textuelle des cas d'utilisation techniques.

4. Conclusion :

La capture des besoins fonctionnels joue un rôle essentiel, d'une part, compléter les recueils initiaux des besoins opérés pendant l'étude préliminaire, et d'autre part, donner un point de vue du prochain chapitre concernant l'analyse afin d'identifier les classes candidates du modèle statique qui présentent une des approches orientées objets.

La capture des besoins techniques est une étape de prise en compte des contraintes techniques et logicielles.

CHAPITRE 4

ANALYSE

1. Introduction :

L'étape d'analyse est entamée suite à l'étape de capture des besoins fonctionnels, l'analyse cible le système futur plutôt que le système existant.

Ce chapitre, donc, traite la phase de l'analyse du SGPP selon l'ensemble des activités suivantes :

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

2. Découpage en catégories :

2.1. Les catégories :

Une catégorie consiste en un regroupement logique de classes à forte cohérence interne et faible couplage externe. [2]

Les catégories résultantes sont les suivantes :

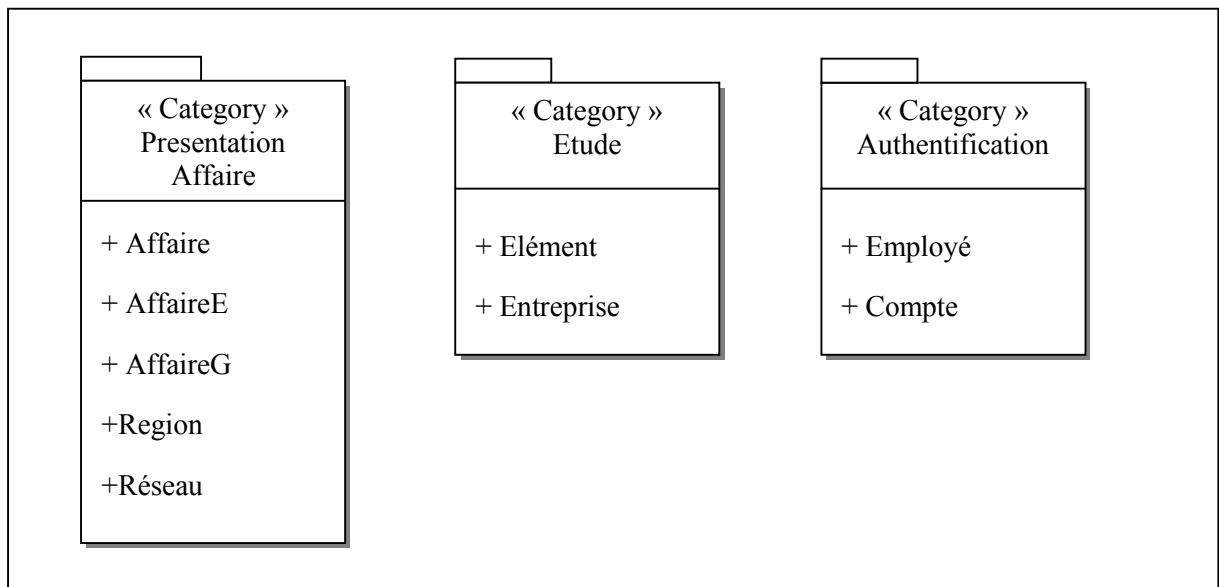


Figure 4.1 : Premier découpage en Catégories du SGPP

2.2. Élaboration des diagrammes de classes préliminaire par catégories :

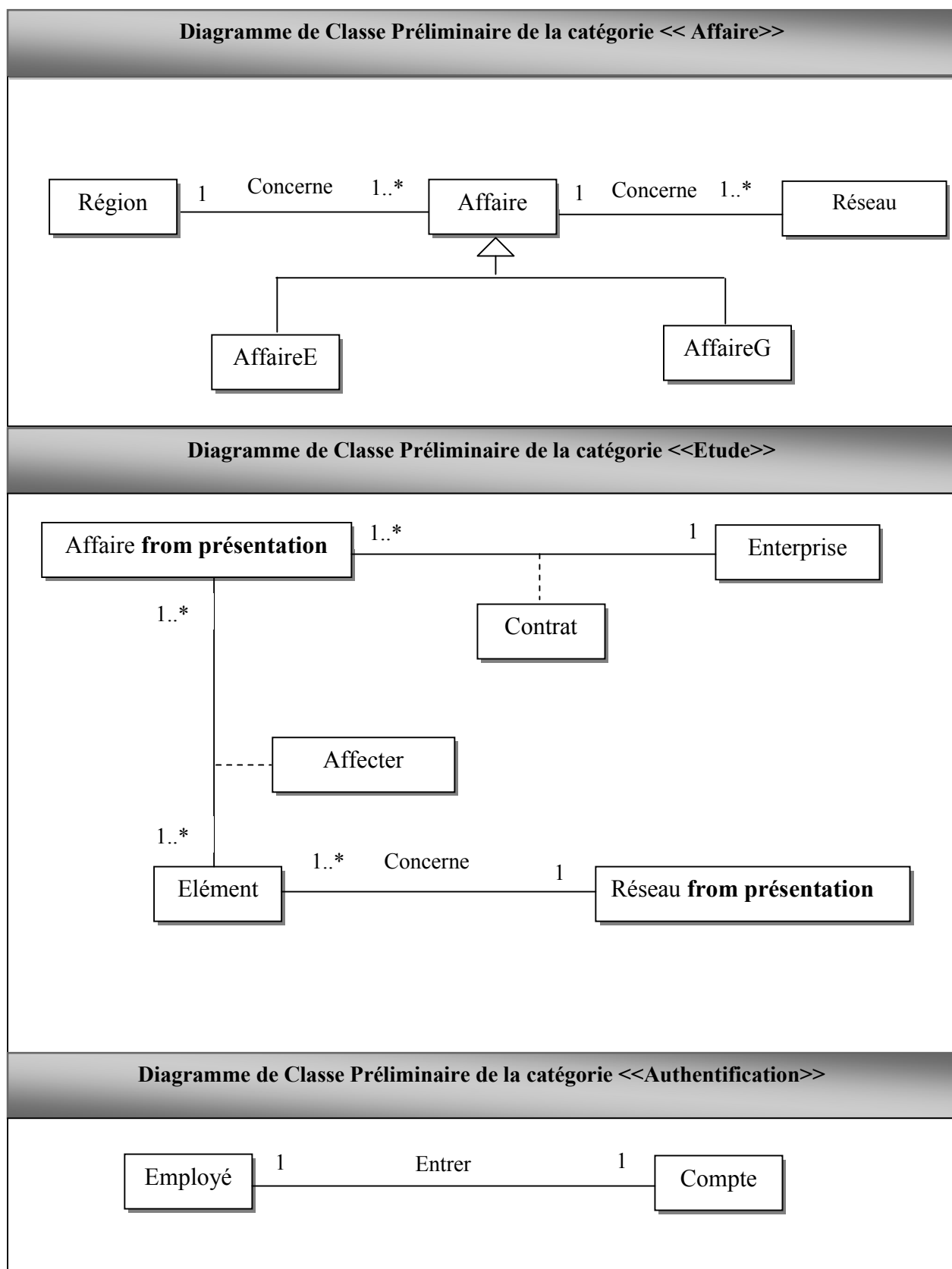


Tableau 4.1 : Diagramme de classes préliminaire par catégorie.

2.3. Modèle structurel d'analyse :

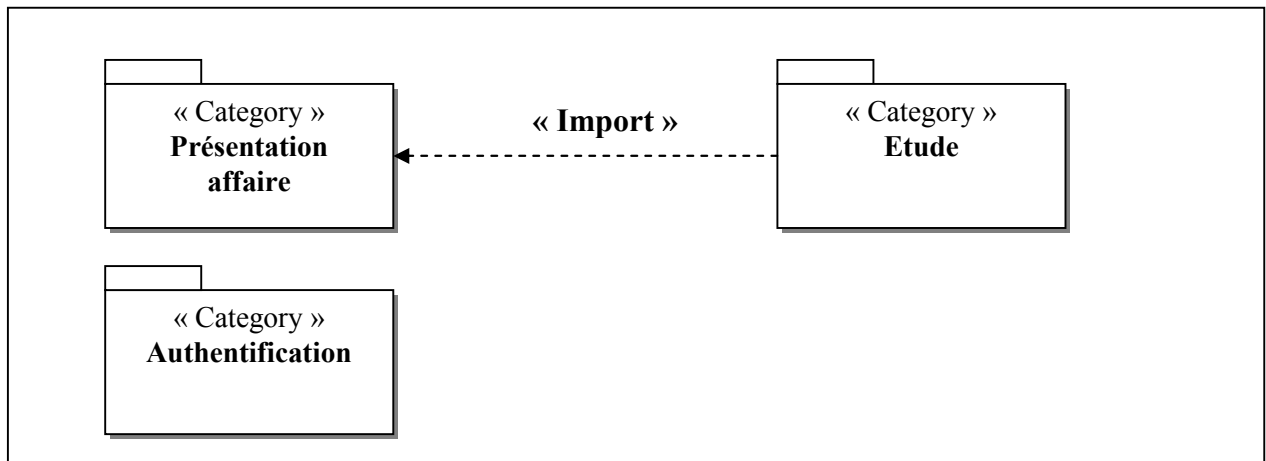


Figure 4.2 : Modèle structurel d'analyse du système SGPP

3. Développement du modèle statique :

Les diagrammes des classes établis, sommairement dans les diagrammes de classes participants du chapitre précédent, puis réorganisés lors du découpage en catégories (dans ce chapitre), vont être détaillés, complétés, et optimisés dans ce chapitre.

Nous allons suivre les opérations suivantes :

- Affiner les classes.
- Affiner les associations.
- Ajouter les attributs.
- Ajouter les opérations.

3.1. Diagrammes de classes détaillées par catégorie :

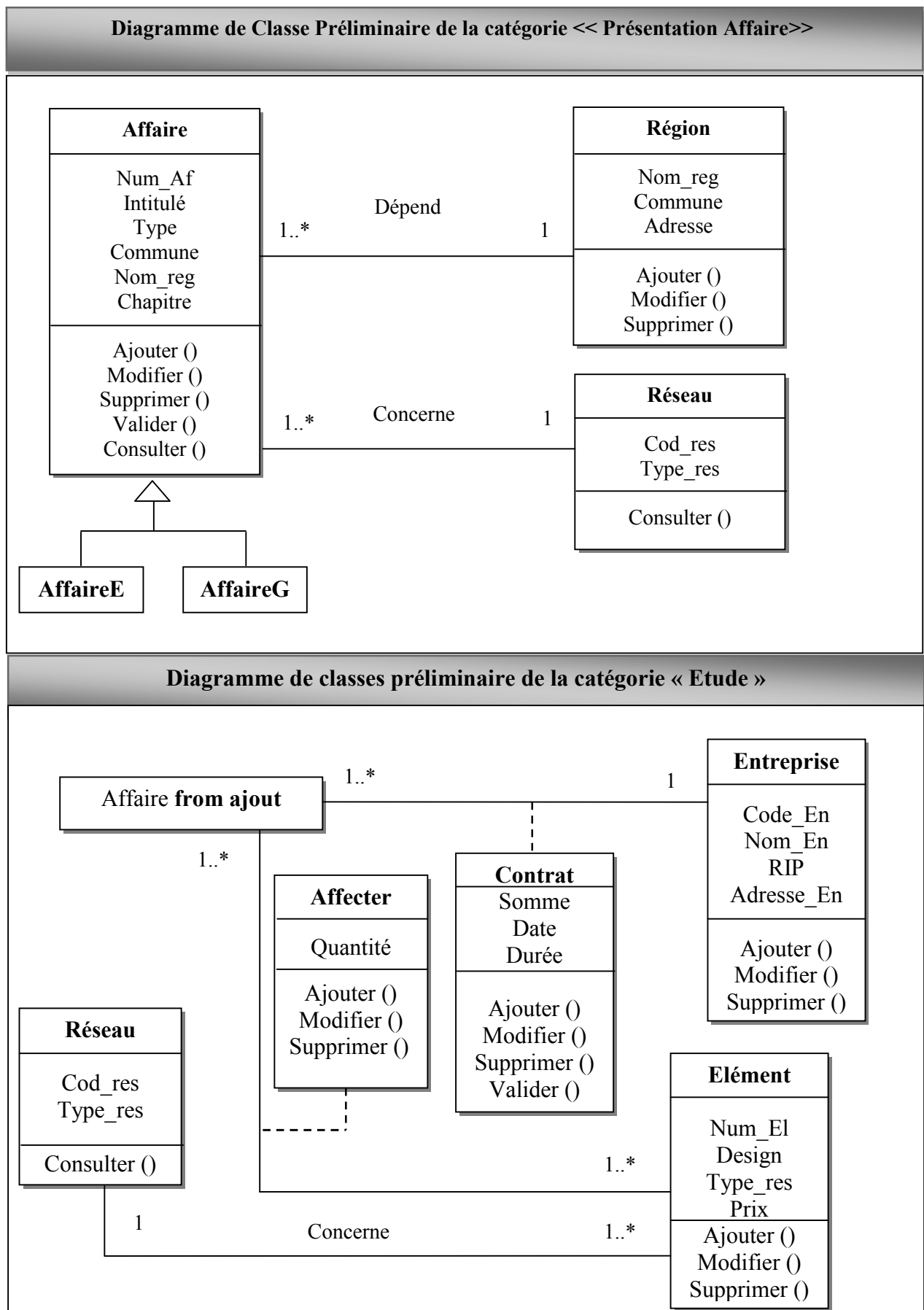


Tableau 4.2: Diagrammes de classes détaillées par catégories.

3.2. Diagramme de classe globale :

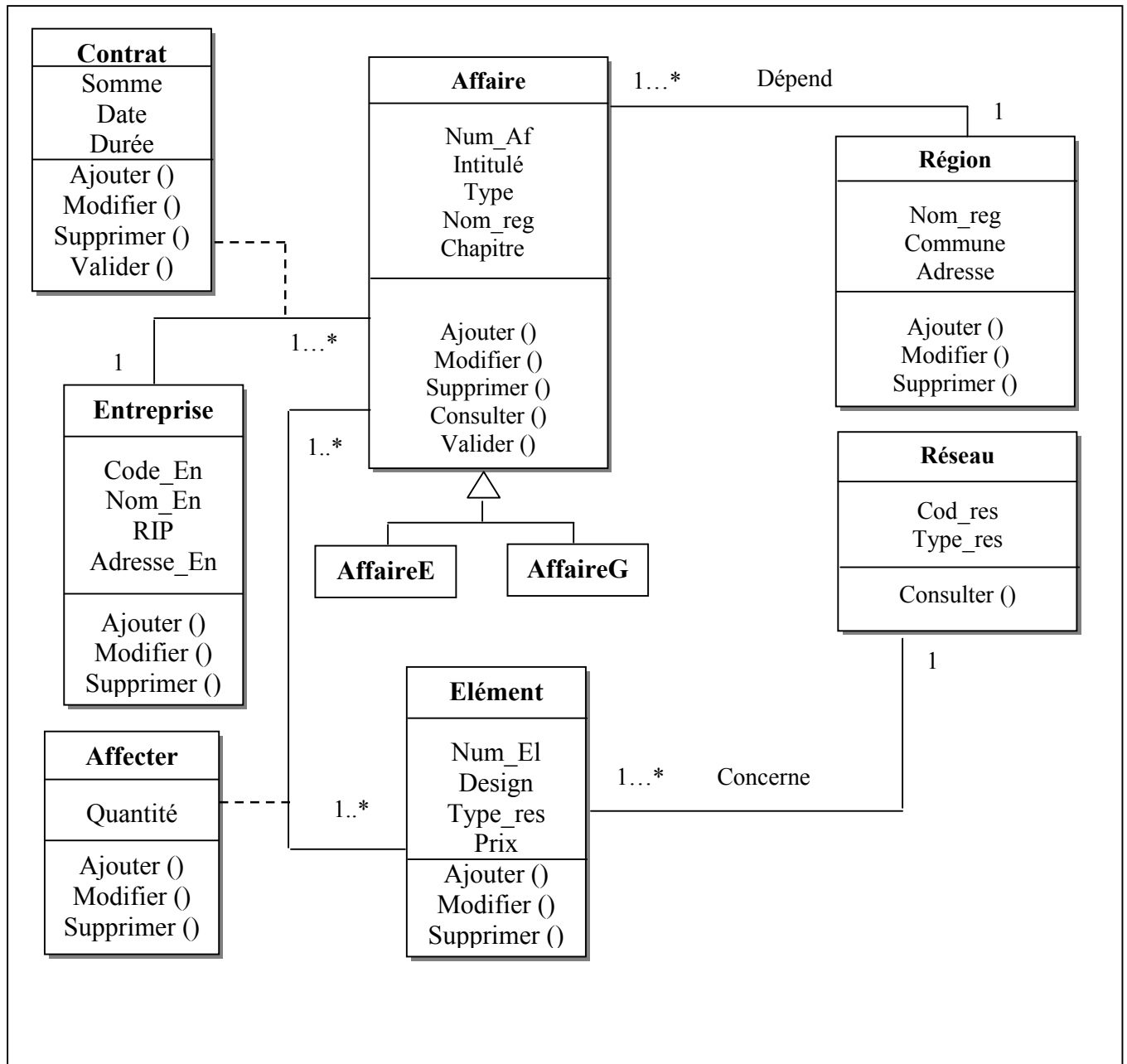


Tableau 4.3: Diagramme de classe globale.

4. Développement du modèle dynamique :

Le développement du modèle dynamique constitue la troisième activité de l'étape d'analyse. Il s'agit d'une activité itérative, fortement couplée avec l'activité de modélisation statique.

4.1. Diagramme de d'interaction :

4.1.1. Cas d'utilisation « Proposer Affaire »

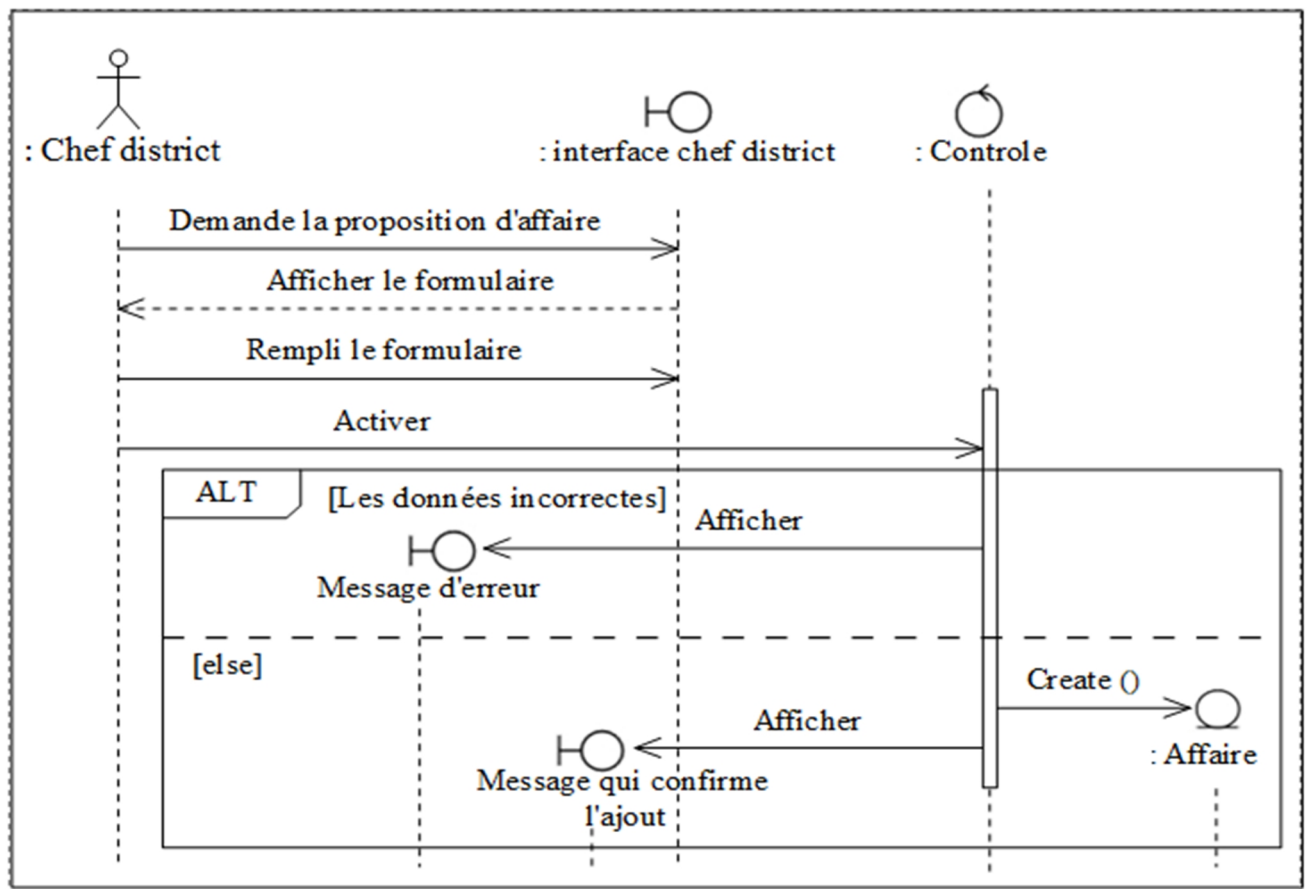


Figure 4.3 : Diagramme d'interaction « Proposer Affaire »

4.1.2. Cas d'utilisation « Modifier Affaire »

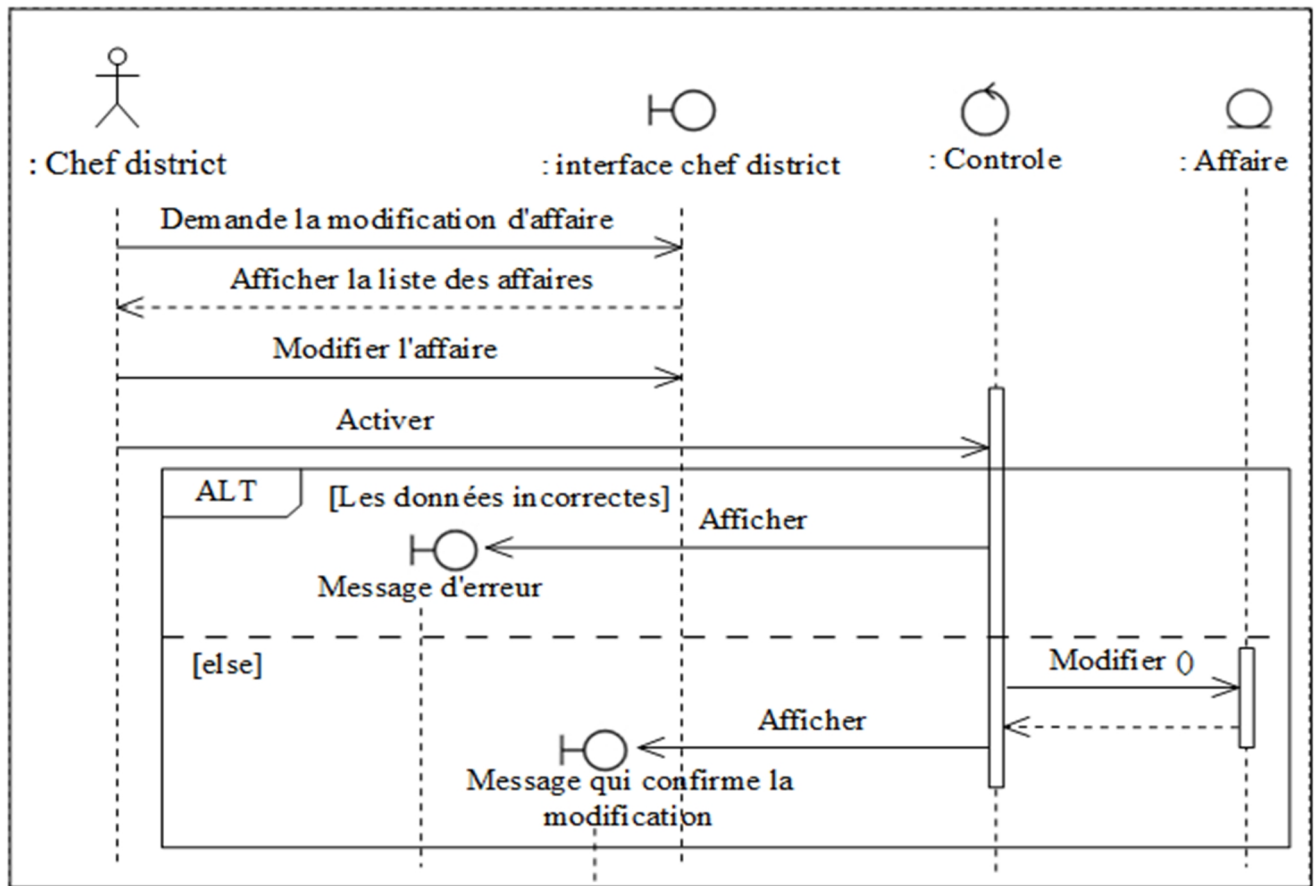


Figure 4.4 : Diagramme d'interaction « Modifier Affaire »

4.1.3. Cas d'utilisation « Supprimer Affaire »

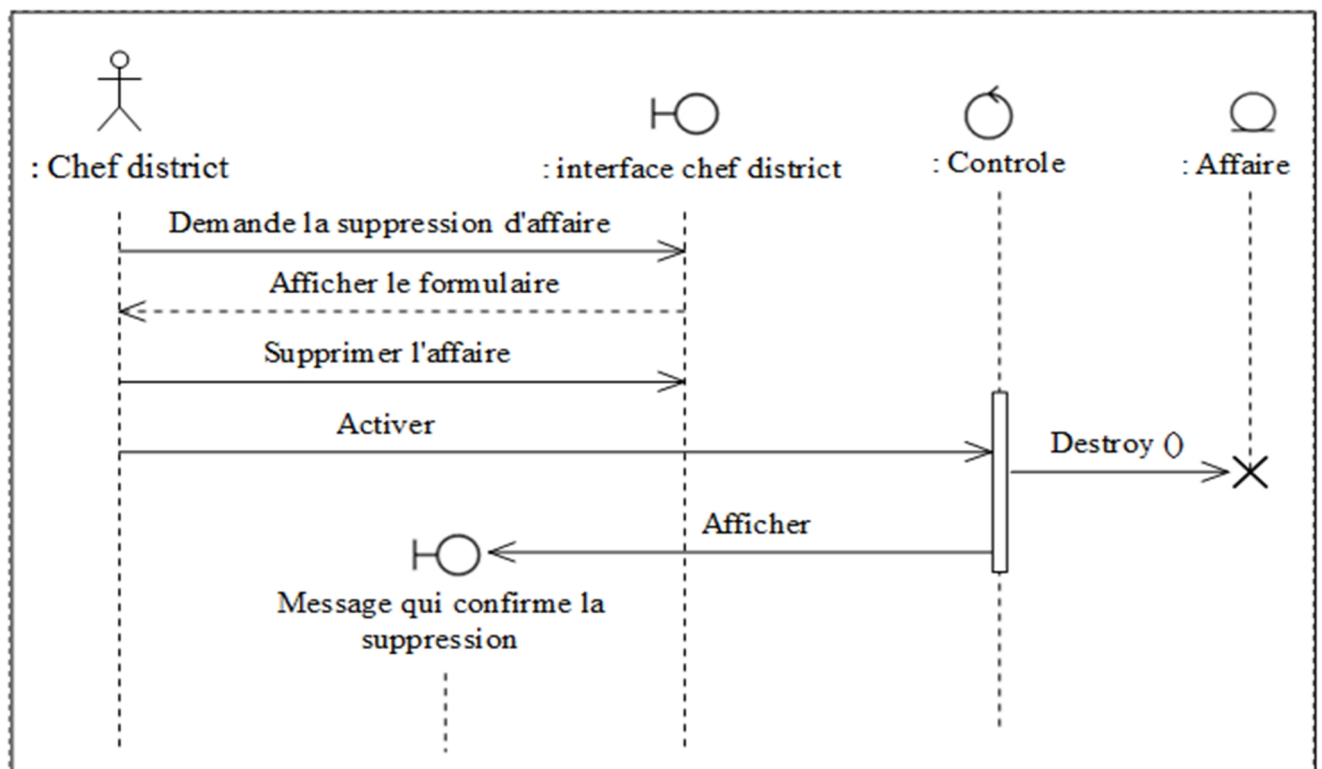


Figure 4.5 : Diagramme d'interaction « Supprimer Affaire »

4.1.4. Cas d'utilisation « Valider Affaire »

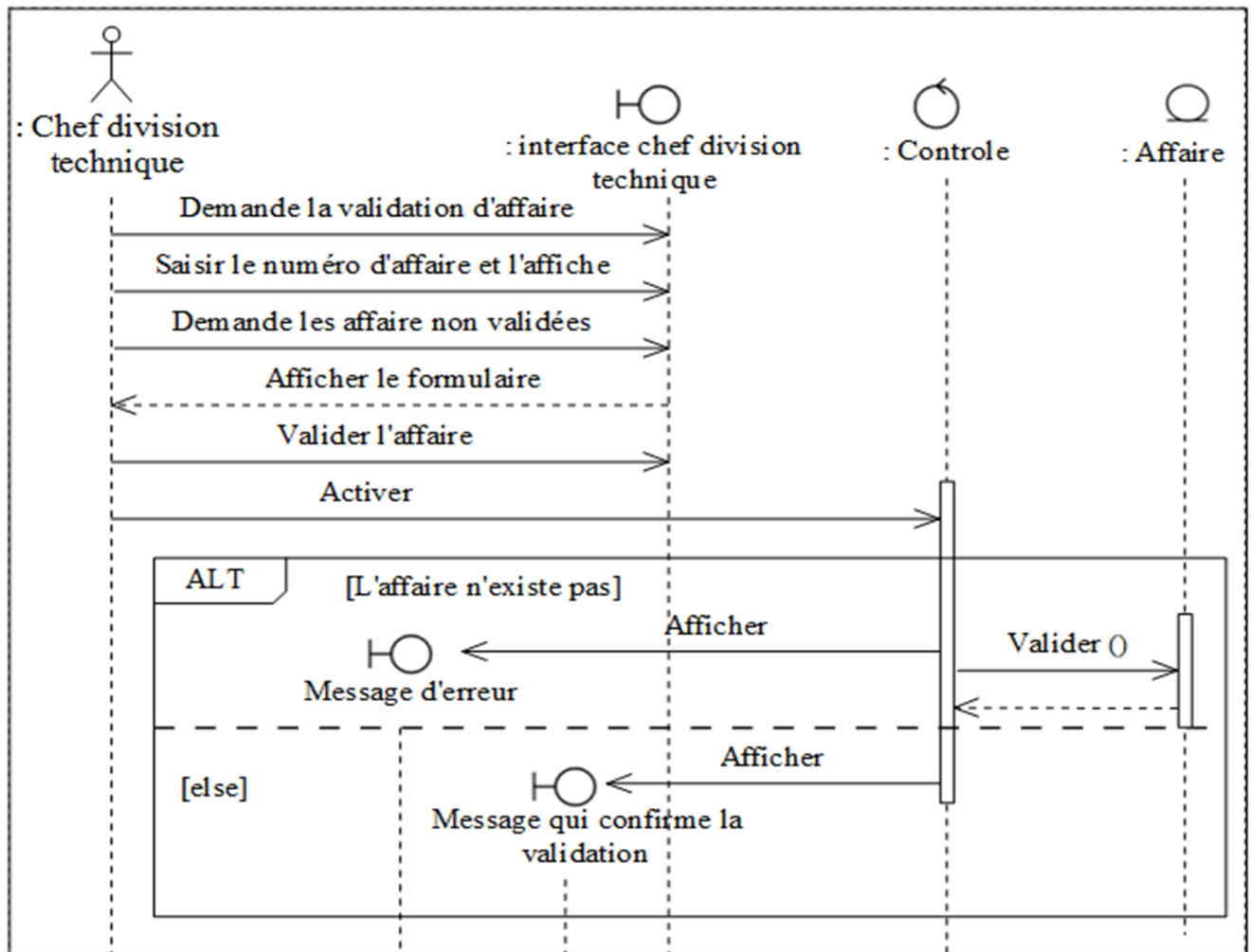


Figure 4.6 : Diagramme d'interaction « Valider Affaire »

4.1.5. Cas d'utilisation « Etablir Avant projet »

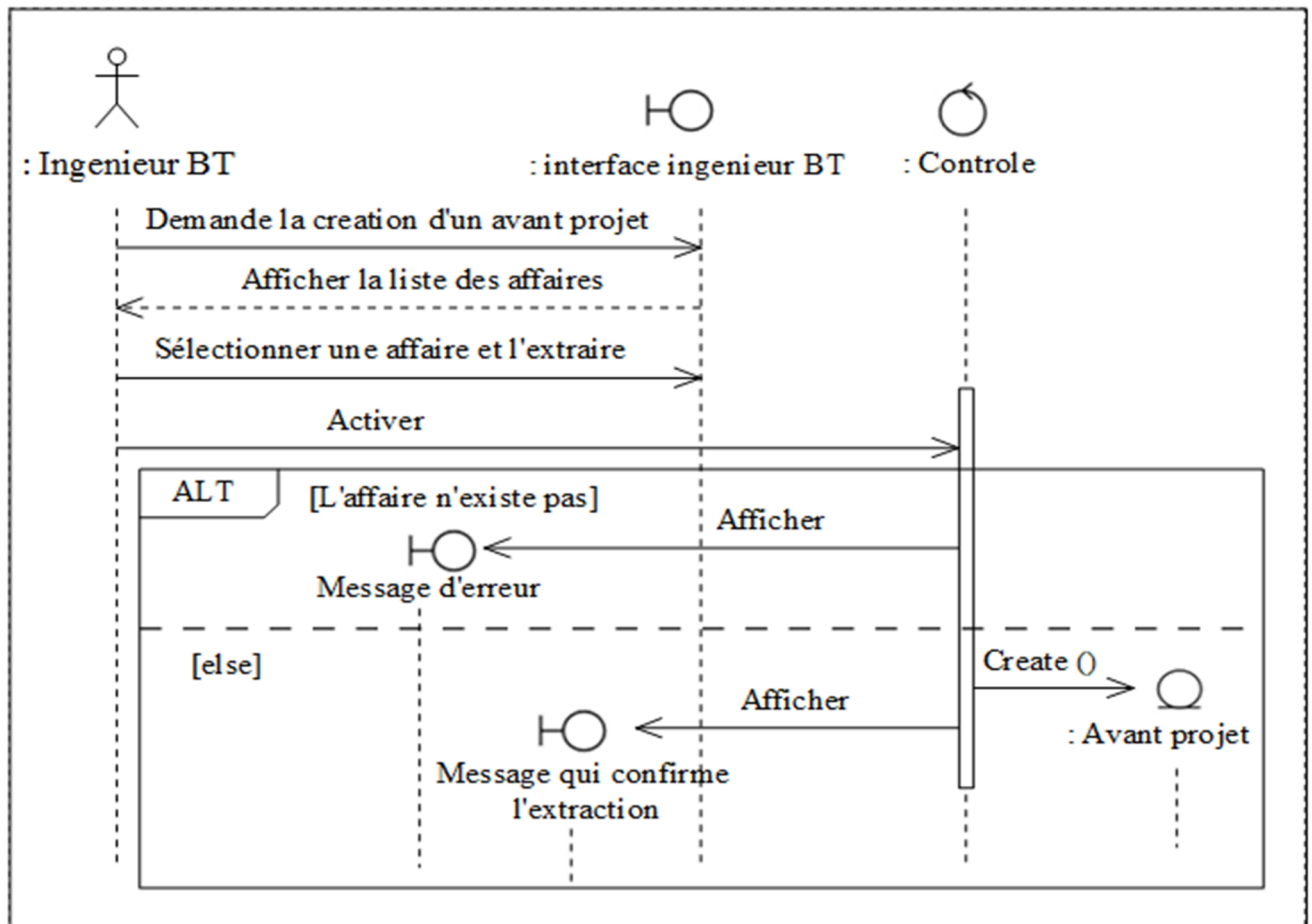


Figure 4.7 : Diagramme d'interaction «Etablir Avant projet »

4.1.6. Cas d'utilisation « Affecter Elément »

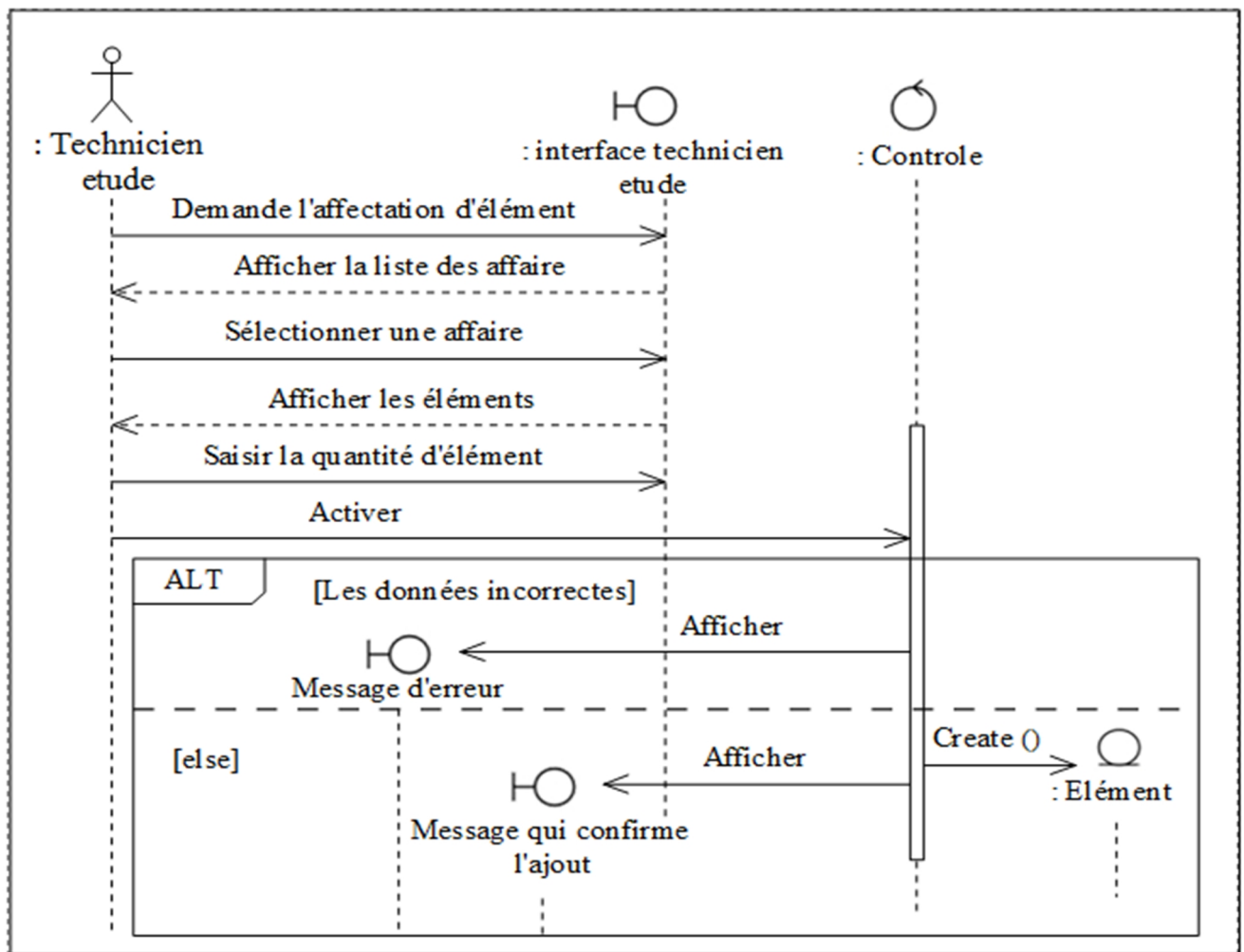


Figure 4.8 : Diagramme d'interaction «Affecter Elément»

4.1.7. Cas d'utilisation « Modifier Élément affecté »

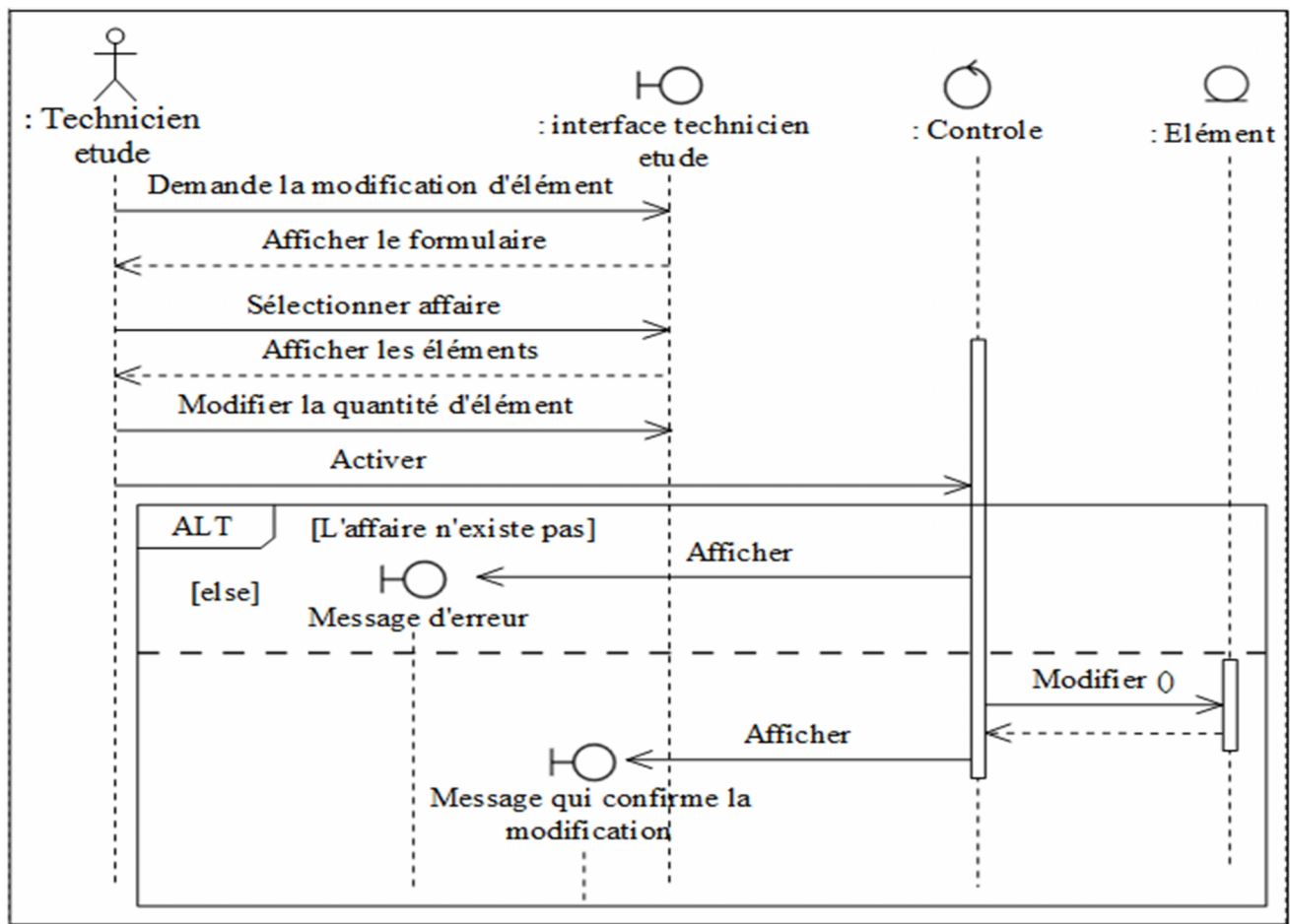


Figure 4.9 : Diagramme d'interaction «Modifier Élément affecter»

4.1.8. Cas d'utilisation « Supprimer Élément affecté »

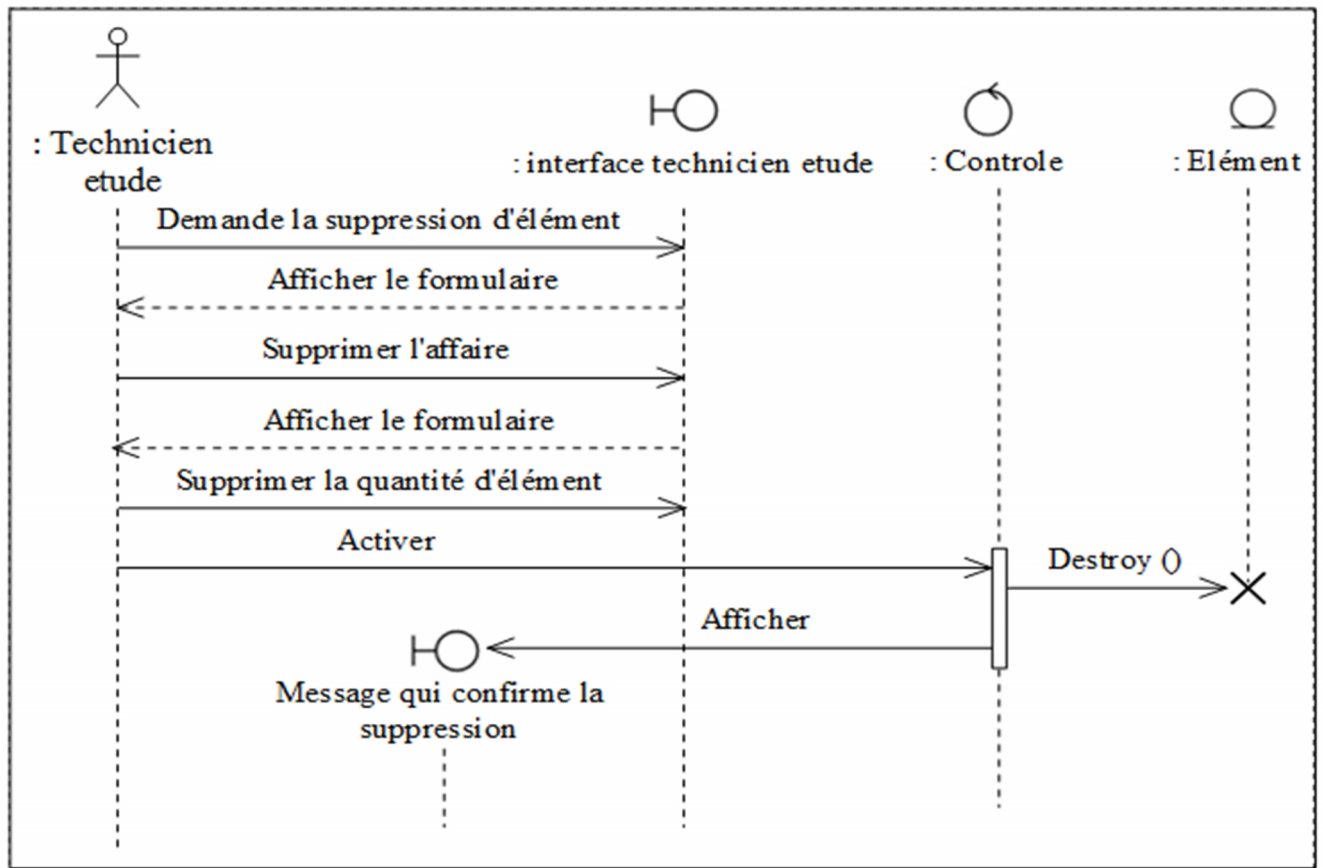


Figure 4.10 : Diagramme d'interaction «Supprimer Élément affecter»

4.1.9. Cas d'utilisation «Ajouter Élément»

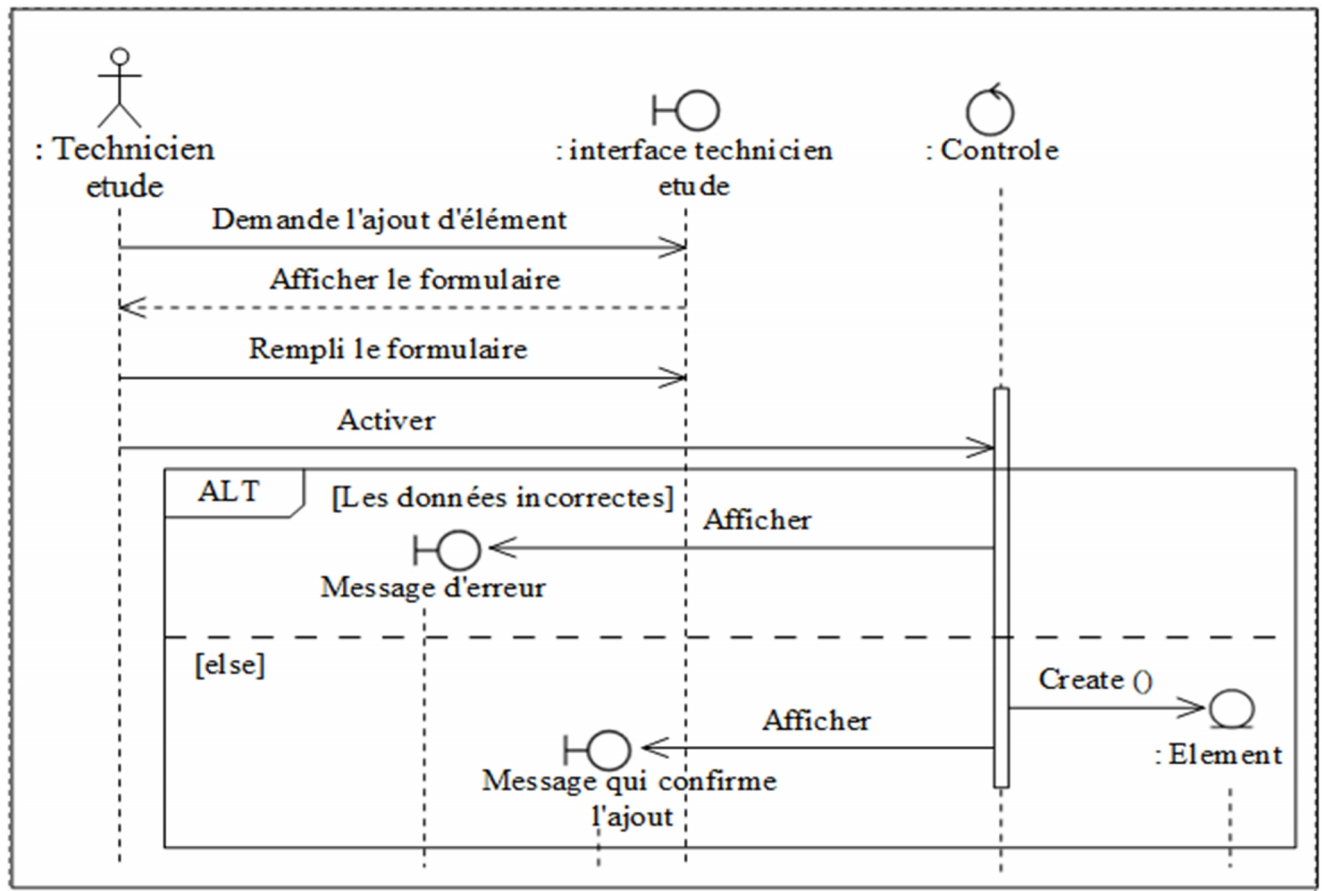


Figure 4.11 : Diagramme d'interaction «Ajouter Élément»

4.1.10. Cas d'utilisation «Modifier Élément»

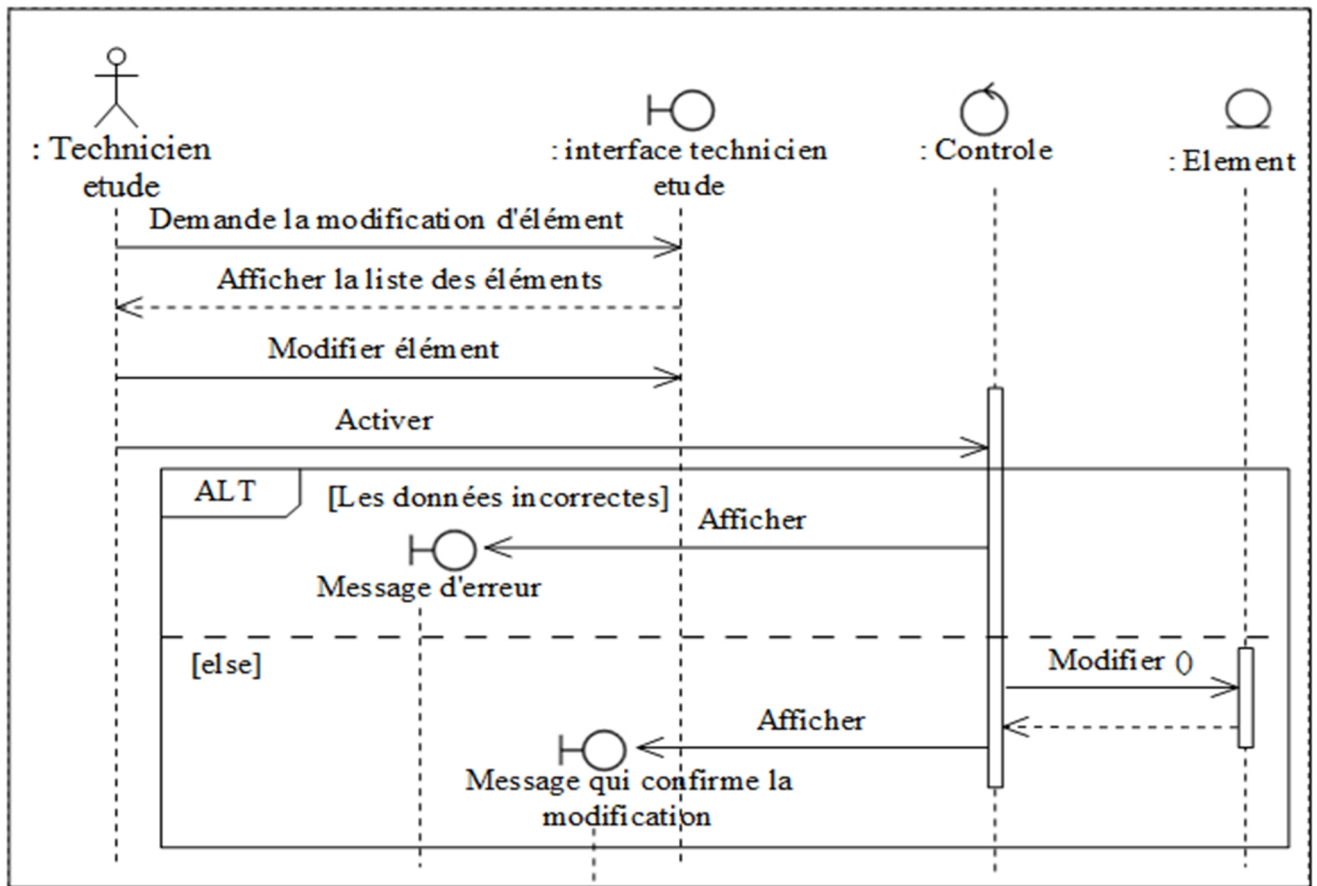


Figure 4.12 : Diagramme d'interaction «Modifier Élément»

4.1.11. Cas d'utilisation «Supprimer Élément»

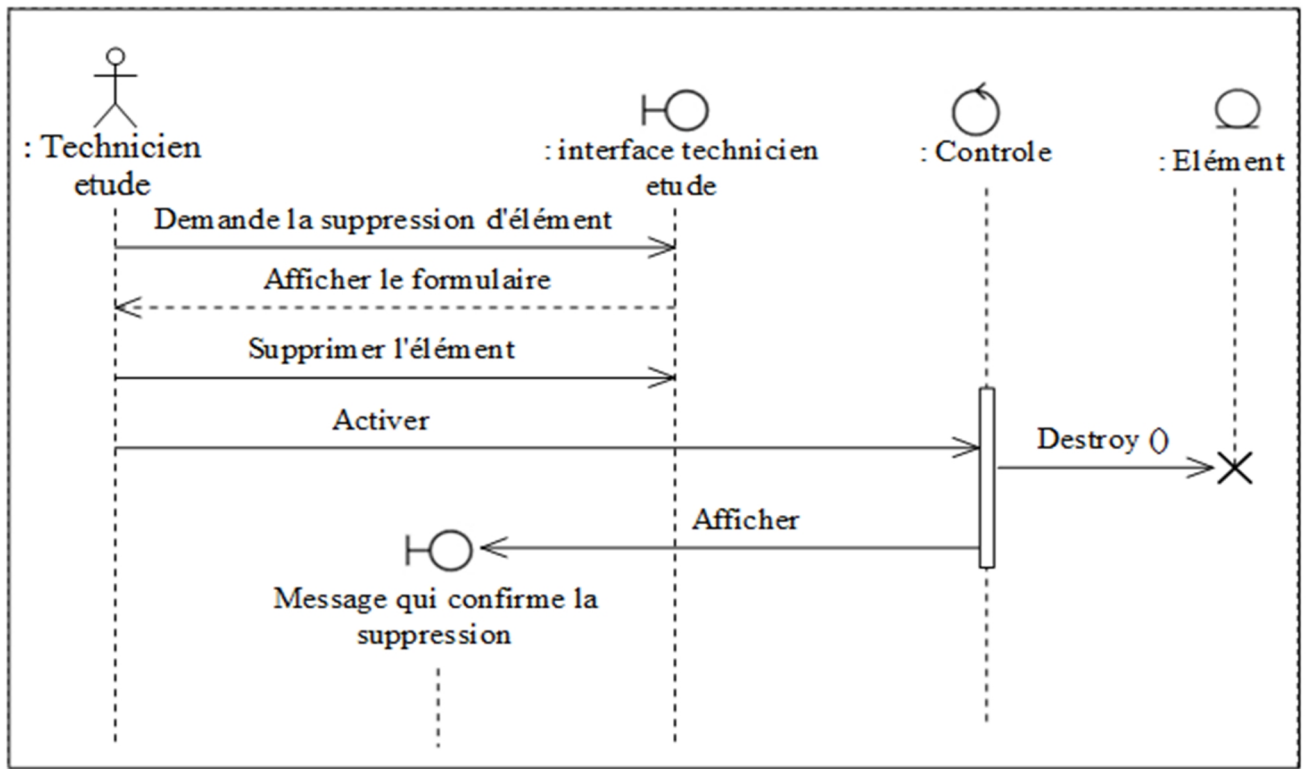


Figure 4.13 : Diagramme d'interaction «Supprimer Élément»

4.1.12. Cas d'utilisation « Consulter quantitatif»

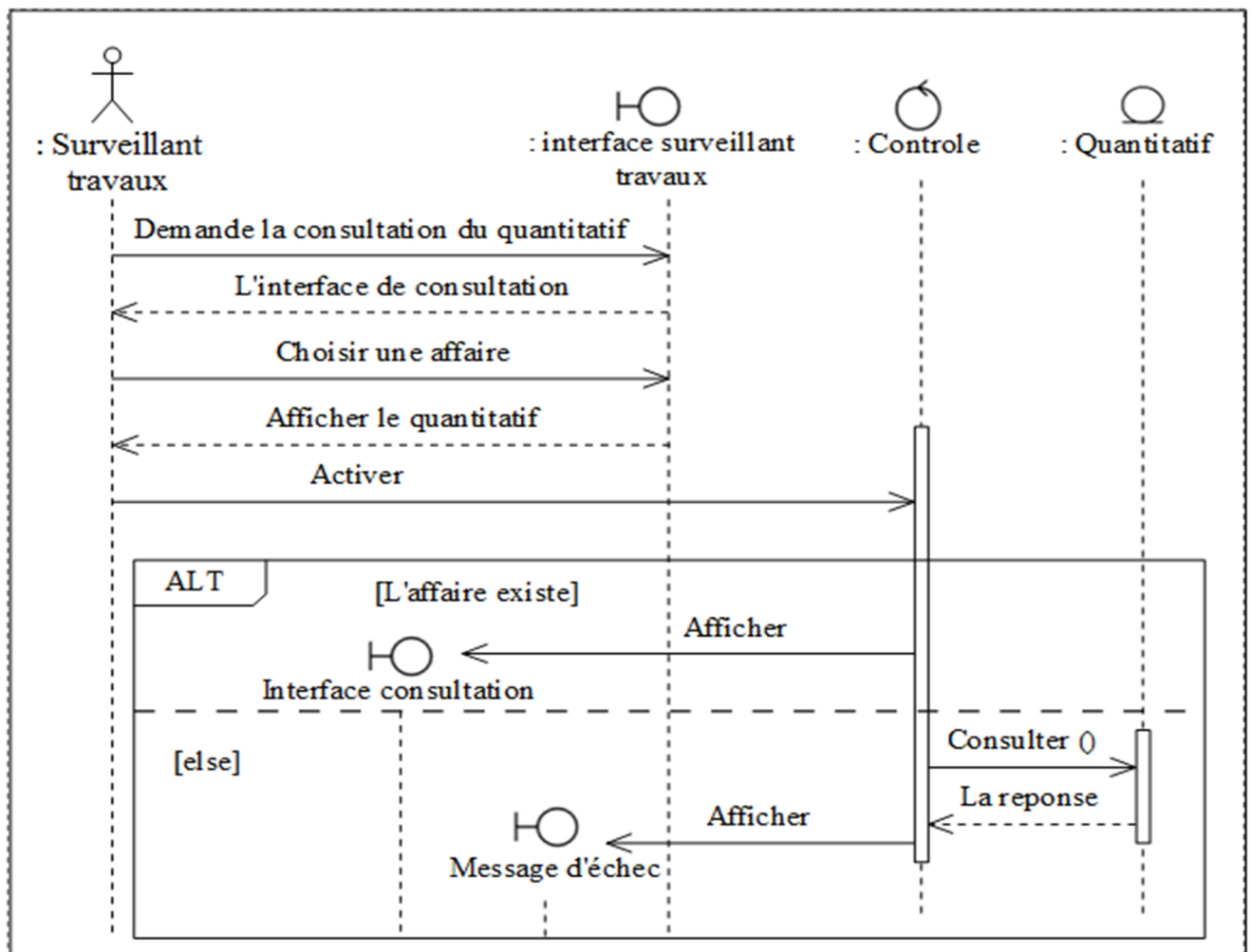


Figure 4.14 : Diagramme d'interaction «Consulter Quantitatif»

4.1.13. Cas d'utilisation « Ajouter Entreprise »

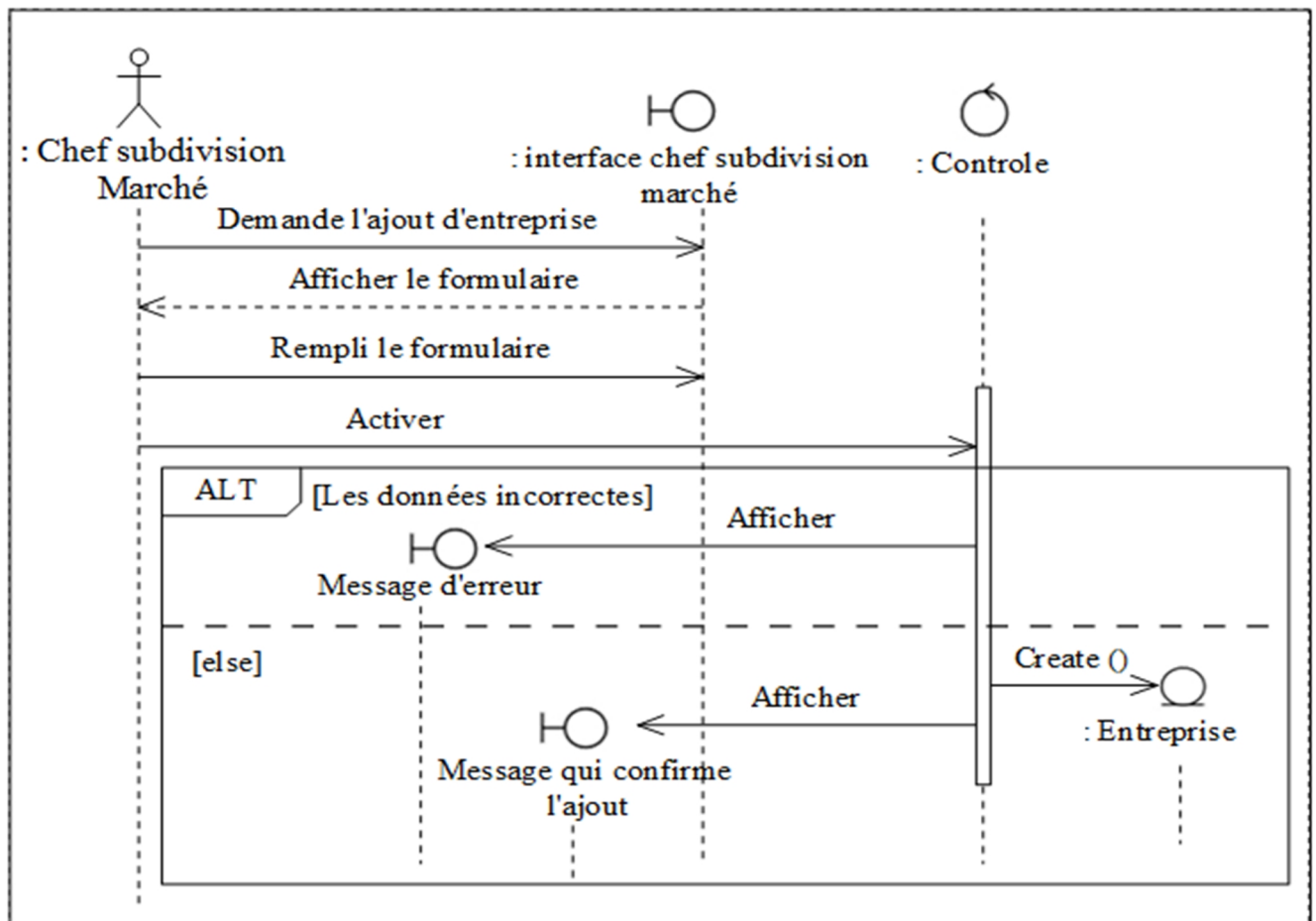


Figure 4.15 : Diagramme d'interaction «Ajouter Entreprise»

4.1.14. Cas d'utilisation « Modifier Entreprise »

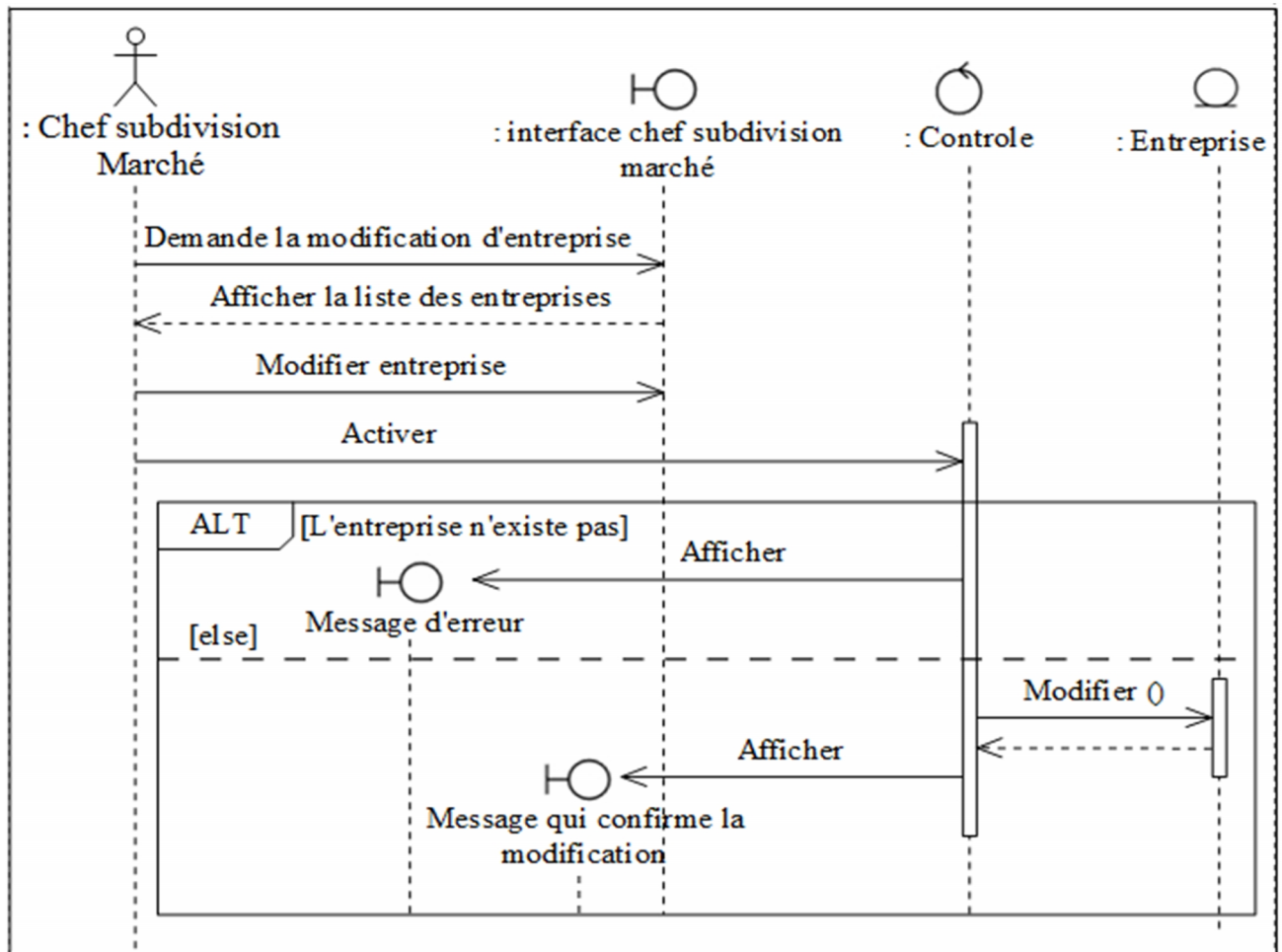


Figure 4.16 : Diagramme d'interaction «Modifier Entreprise»

4.1.15. Cas d'utilisation «Supprimer Entreprise»

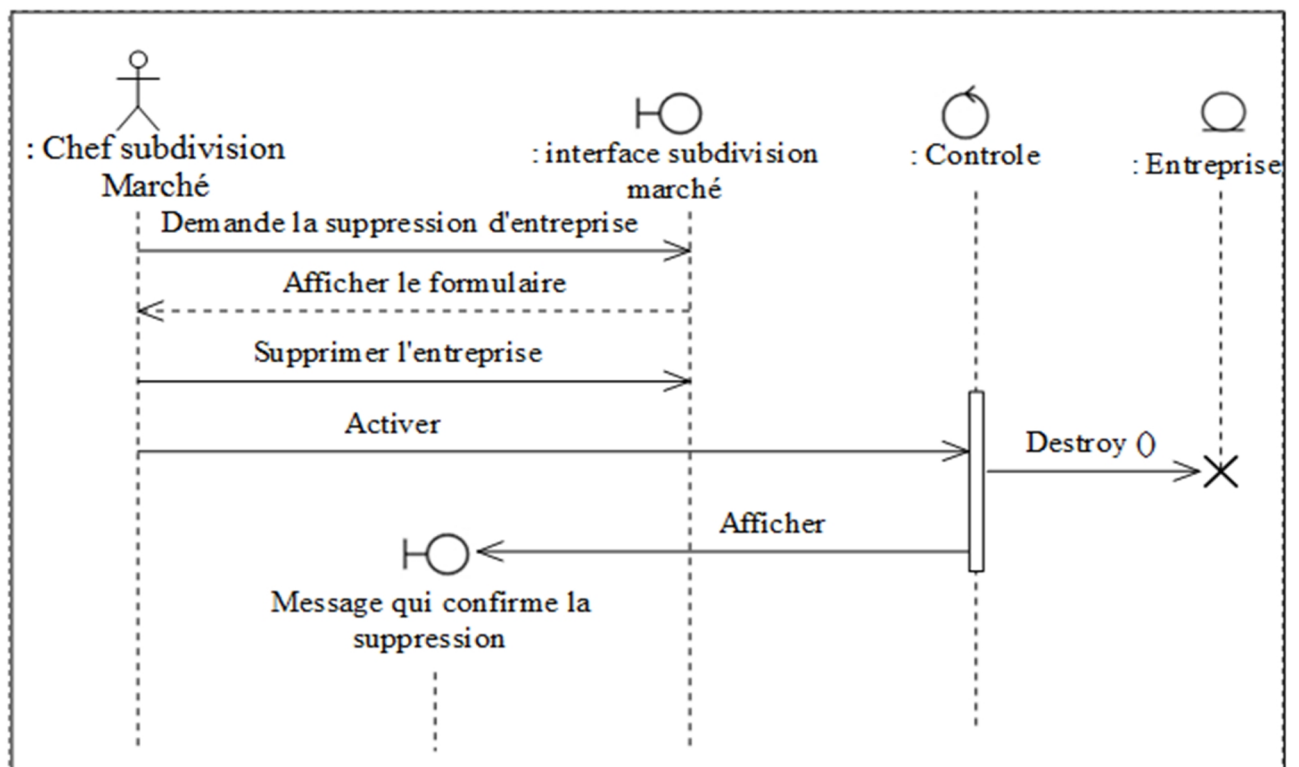


Figure 4.17 : Diagramme d'interaction «Supprimer Entreprise»

4.1.16. Cas d'utilisation «Etablir Contrat»

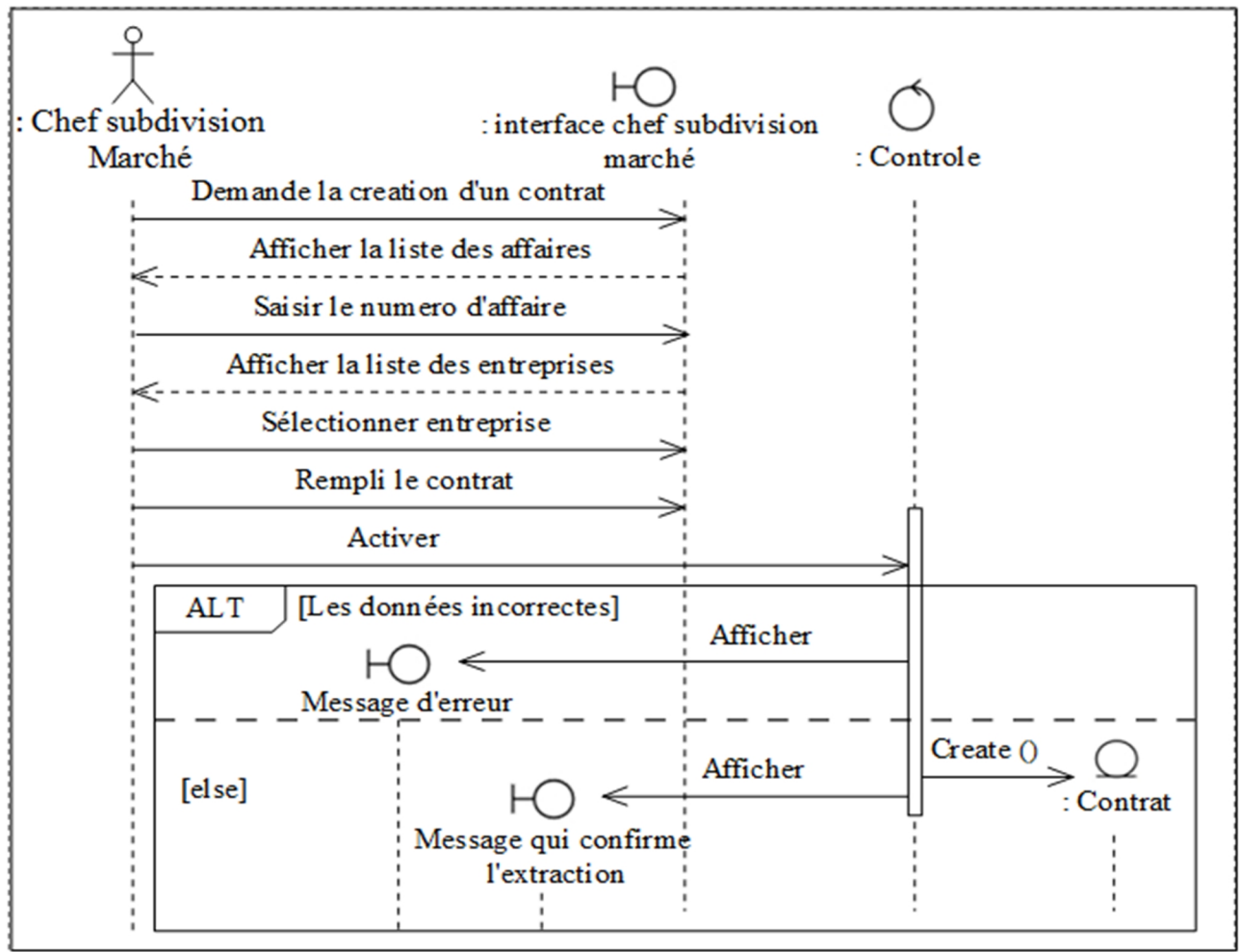


Figure 4.18 : Diagramme d'interaction «Etablir Contrat»

4.1.17. Cas d'utilisation «Ajouter Contrat»

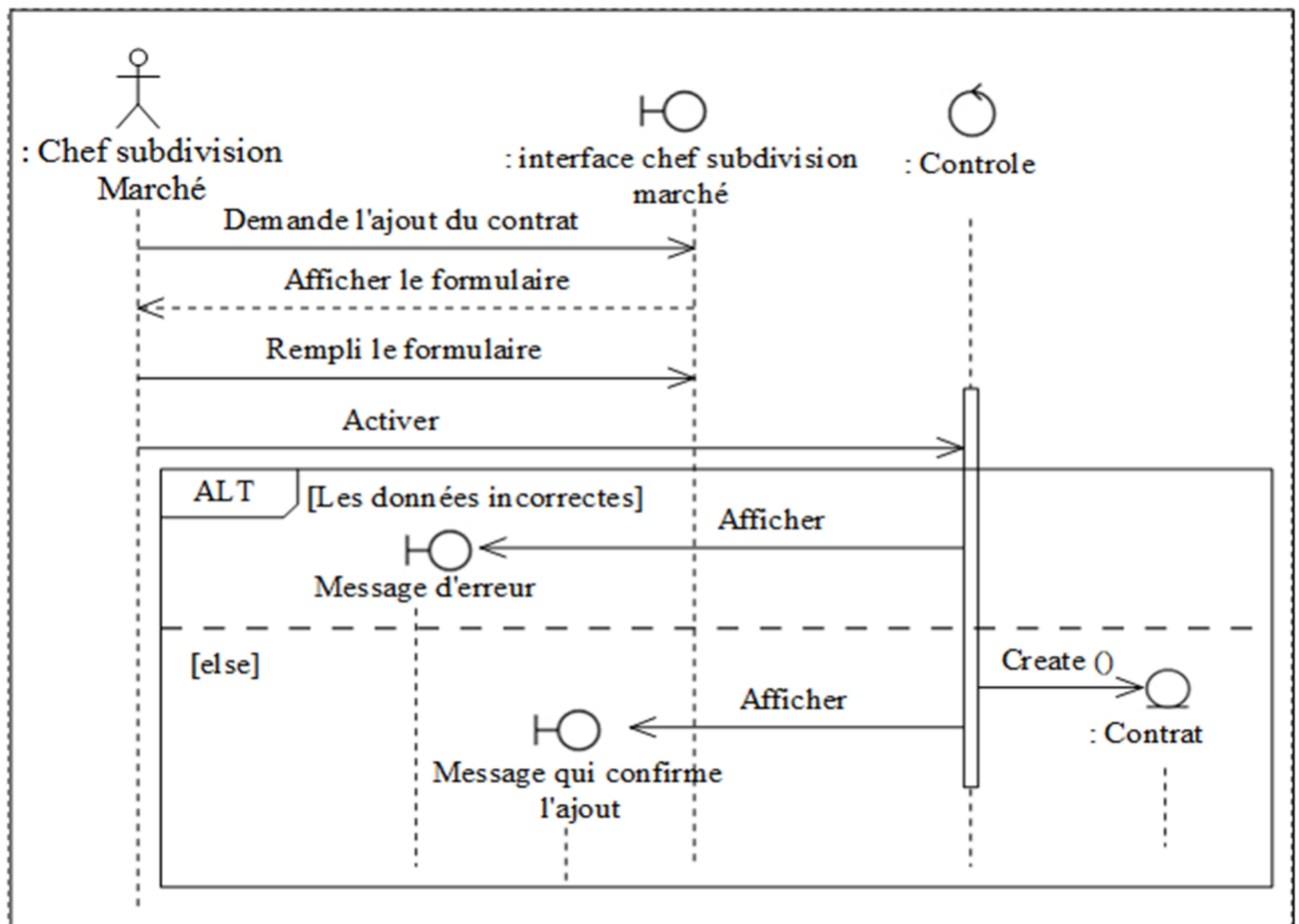


Figure 4.19 : Diagramme d'interaction «Ajouter Contrat»

4.1.18. Cas d'utilisation «Modifier Contrat»

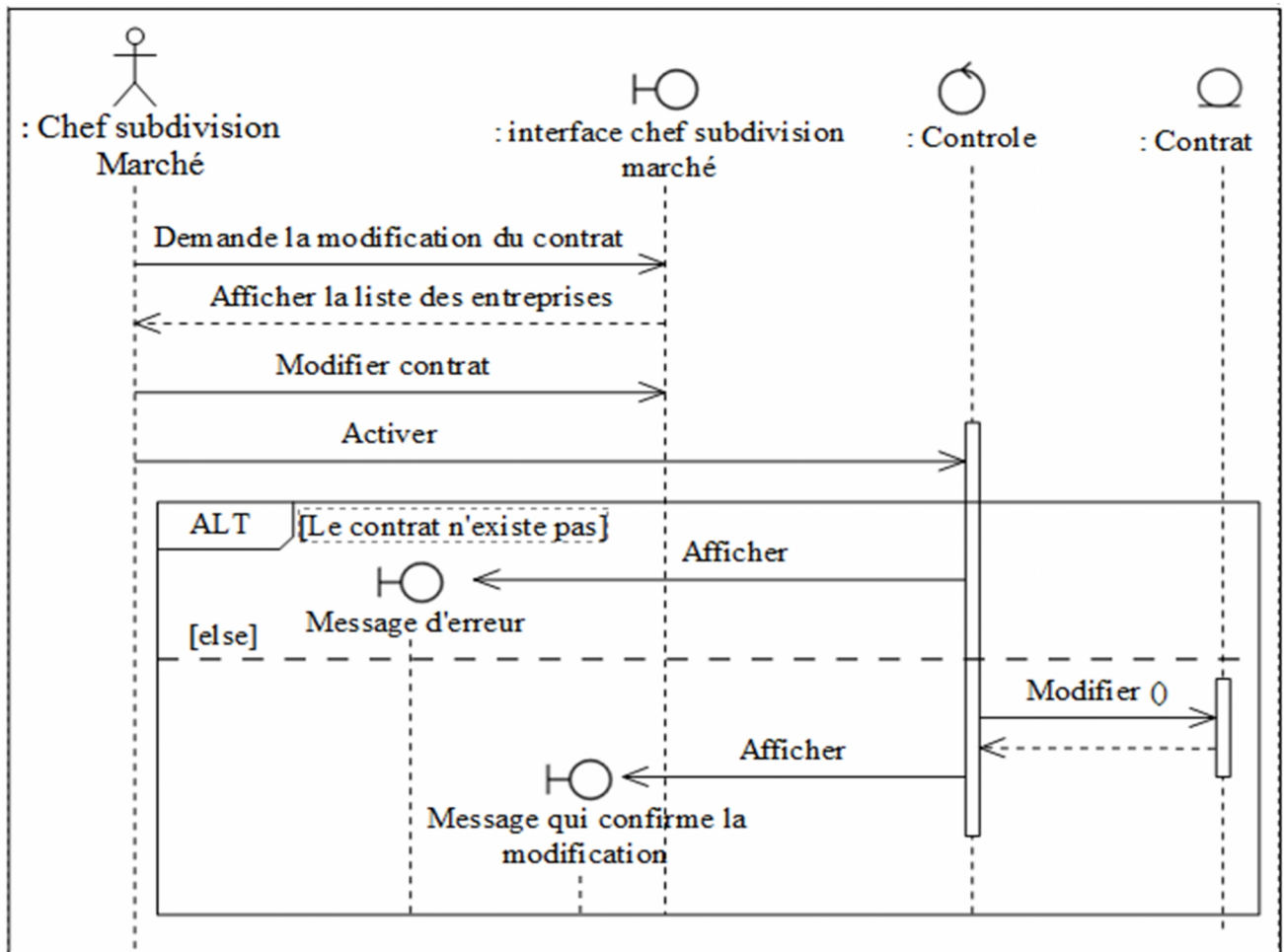


Figure 4.20 : Diagramme d'interaction «Modifier Contrat»

4.1.19. Cas d'utilisation «Supprimer Contrat»

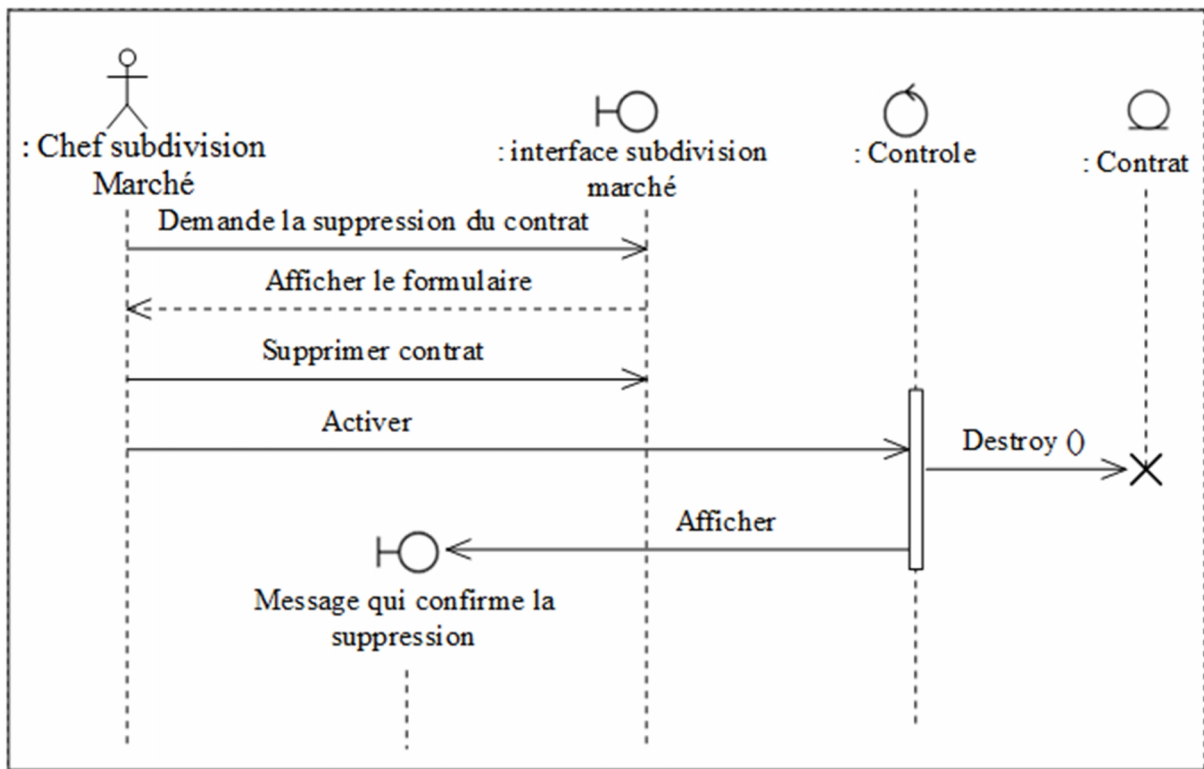


Figure 4.21 : Diagramme d'interaction «Supprimer Contrat»

4.1.20. Cas d'utilisation «Valider Contrat»

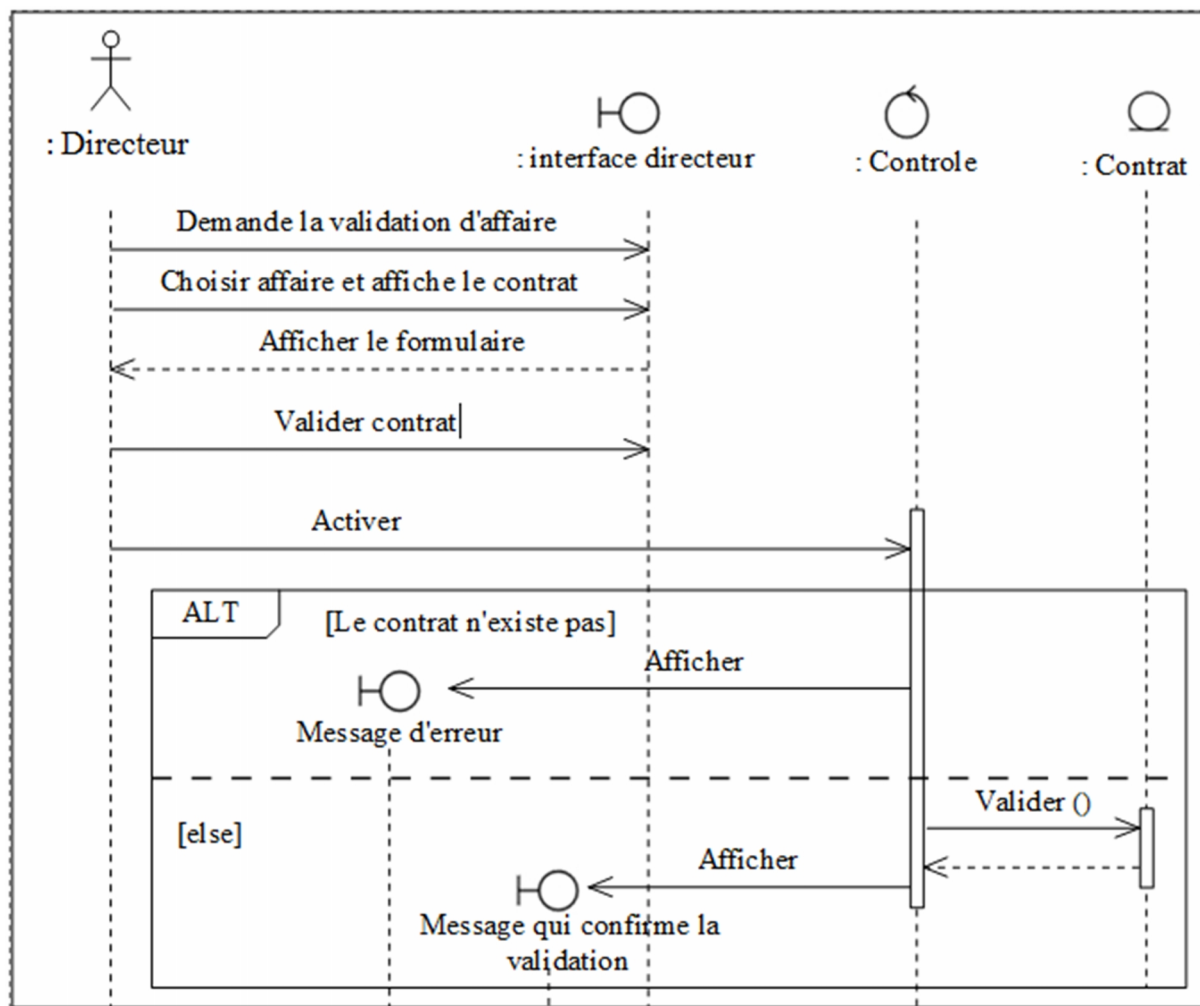


Figure 4.22 : Diagramme d'interaction «Valider Contrat»

5. Conclusion

Dans ce chapitre nous avons présenté l'analyse objet du système, les classes issues des besoins fonctionnels sont regroupées en catégories pour organiser le modèle structurel d'analyse. Ce modèle nécessite un travail d'analyse détaillé de la structure des classes. Celui-ci est considéré comme une base pour le développement du modèle dynamique.

Nous allons décrire dans le chapitre suivant la conception du système SGPP.

CHAPITRE 5

CONCEPTION

1. Introduction :

Nous allons étudier maintenant le rôle d'UML lors de l'étape de conception préliminaire. Les diagrammes UML servent ici plus particulièrement à construire et à documenter la façon dont le développement de la solution doit être organisé. La conception détaillée consiste à construire et à documenter précisément les classes, les interfaces, les tables et les méthodes qui constituent le codage de la solution.

2. Conception préliminaire :

La conception préliminaire est certainement l'étape la plus délicate du processus 2TUP, car elle représente le cœur ; C'est en effet à cette occasion que s'effectue la fusion des études fonctionnelles et techniques. [2]

2.1. Développement du modèle de déploiement :

Les modèles de déploiement et de configuration matérielle s'expriment tous deux à l'aide d'un diagramme de déploiement; cependant ils n'expriment pas tout à fait le même niveau de description.

- Le modèle de configuration matérielle a été utilisé pour exprimer les contraintes de mise en œuvre au niveau physique.
- Le modèle de déploiement considère plutôt chaque nœud comme un poste de travail, Il exprime la répartition physique des fonctions métiers du système et permet de justifier la localisation des bases de données et des environnements de travail.

Le modèle de déploiement du système SGPP est illustré par la figure suivante :

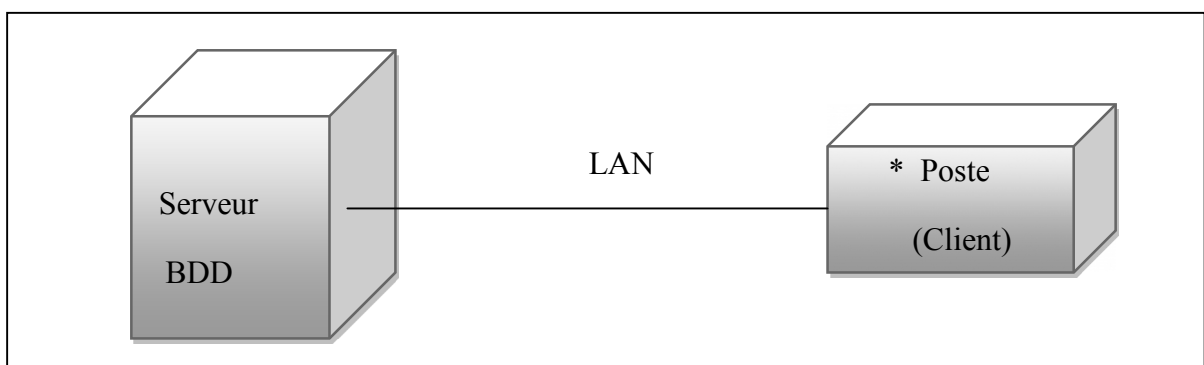


Figure5.1 : Le modèle de déploiement du système.

3. Conception Détaillée :

L'étape de conception détaillée est l'avant dernière étape avant le codage. A cette étape, on commence à raffiner les choix de conception en incluant les spécificités du langage ou environnement de développement.

Nous allons traiter la conception détaillée du système SGPP selon l'ensemble des activités suivantes :

- Le typage des attributs.
- La transformation des classes et des associations.
- L'identification des classes modifiées.
- Le passage du modèle objet au modèle relationnel.

3.1. Typage des attributs :

Le typage des attributs consiste principalement à définir les types des attributs identifiés en analyse.

- **Tables des classes d'objet:**

Symbole	Désignation
Varchar	Caractère variable
Char	Caractère
Date	Date
Int	Numérique
Float	Réal

Tableau 5.1: Table des classes d'objet.

Le typage des attributs des classes du système est illustré par le tableau suivant :

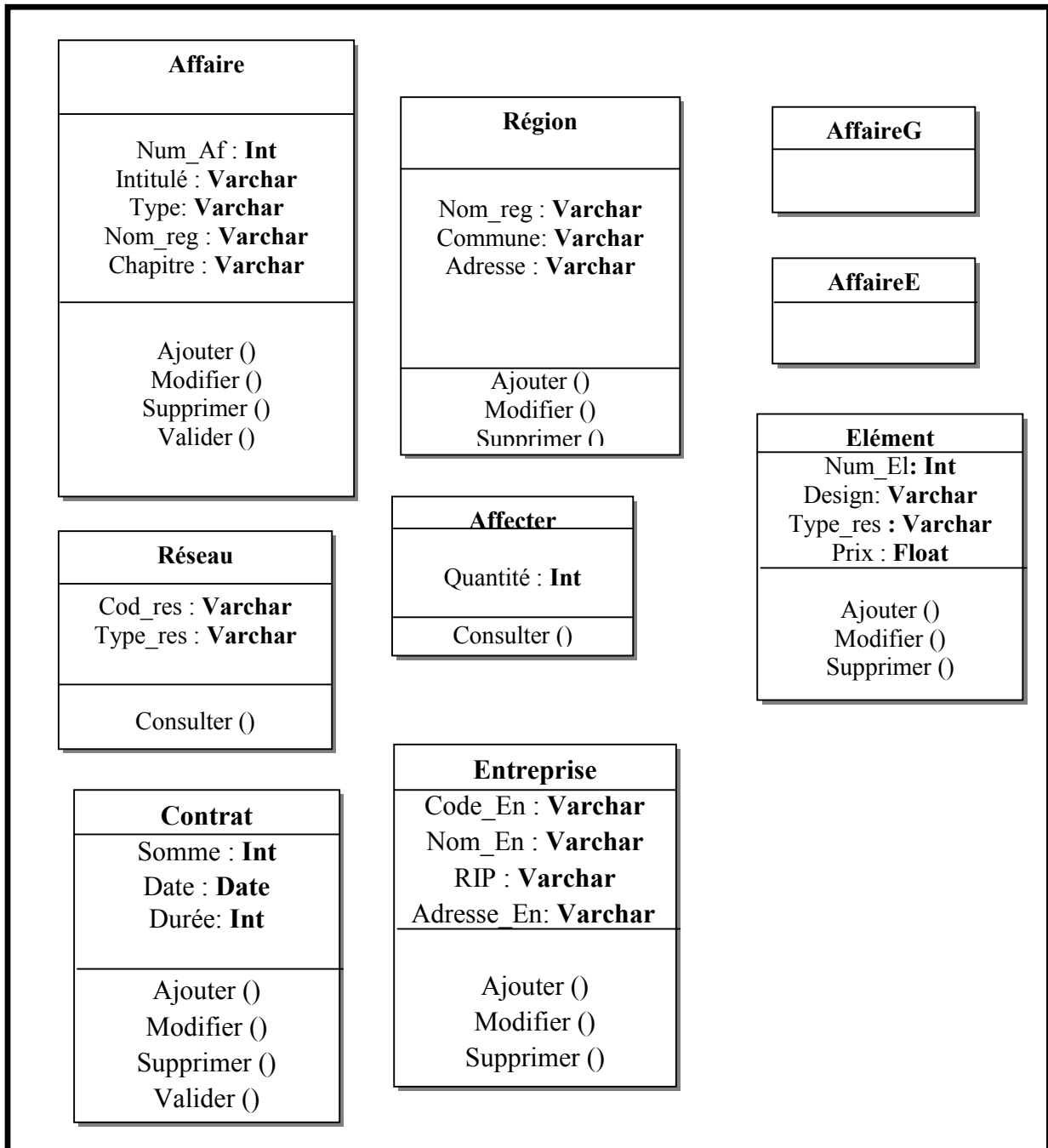


Tableau 5.2: Typage des attributs des classes du système SGPP.

3.2. Règles de transformation :

➤ Transformation des classes :

Chaque classe du diagramme UML devient une relation, les attributs de la classe deviennent attributs de la relation. Il faut choisir un attribut de la classe pouvant jouer le rôle d'identifiant (Clé primaire).

Si aucun attribut ne convient en tant qu'identifiant, il faut en ajouter un à la relation (les outils proposent l'ajout de tels attributs).

➤ Transformation des associations :

Les règles de transformation des associations dépendent des multiplicités maximales des associations. Nous distinguons trois familles d'associations :

✓ Association 1-* :

Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. On peut se rappeler cette règle de la manière suivante : la clé de la relation père migre dans la relation fils.

✓ Association *-* et classes-associations :

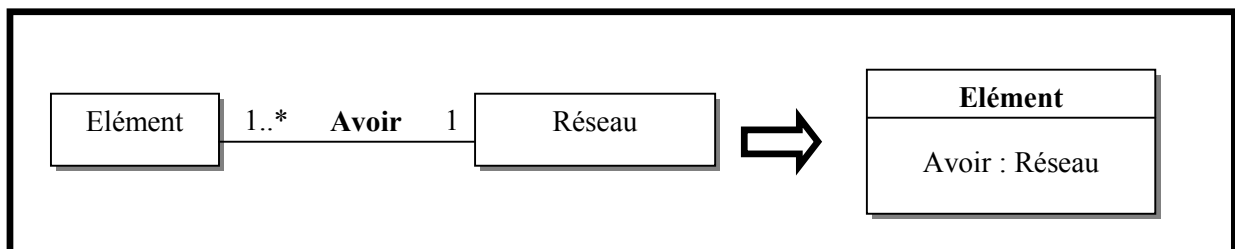
L'association/classe-association devient une relation. La clé primaire de cette relation est la concaténation des clés primaires des classes connectées à l'association. Chaque attribut devient clé étrangère. Les attributs de l'association /classe-association doivent être ajoutés à la nouvelle relation. Ces attributs ne sont ni clé primaire, ni clé étrangère.

✓ Association 1-1 :

Il faut ajouter un attribut de type clé étrangère dans la relation dérivée de la Classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de la classe connectée à l'association.

Cette règle permet d'éviter les valeurs NULL dans la base de données.

3.3. Traduction des associations :



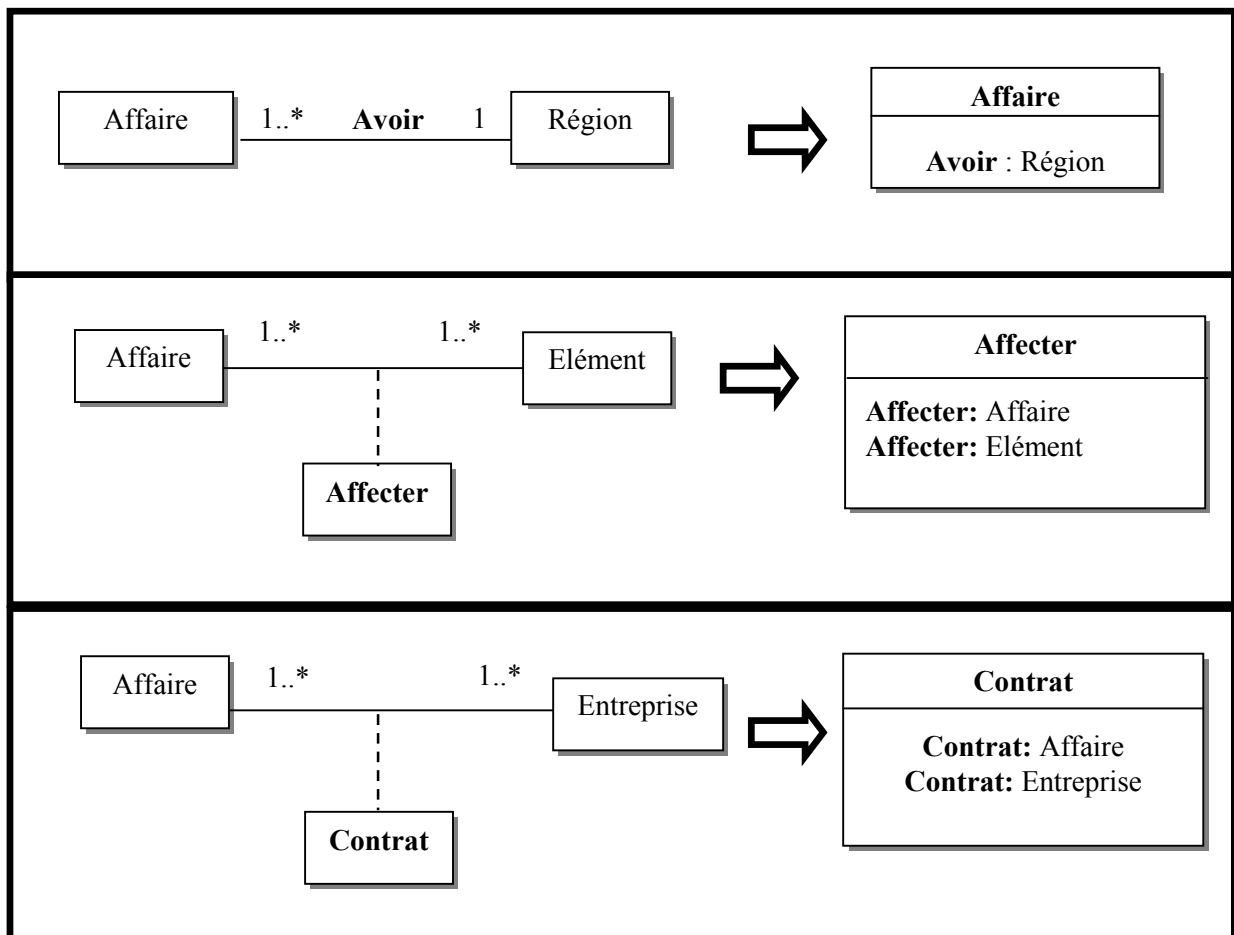
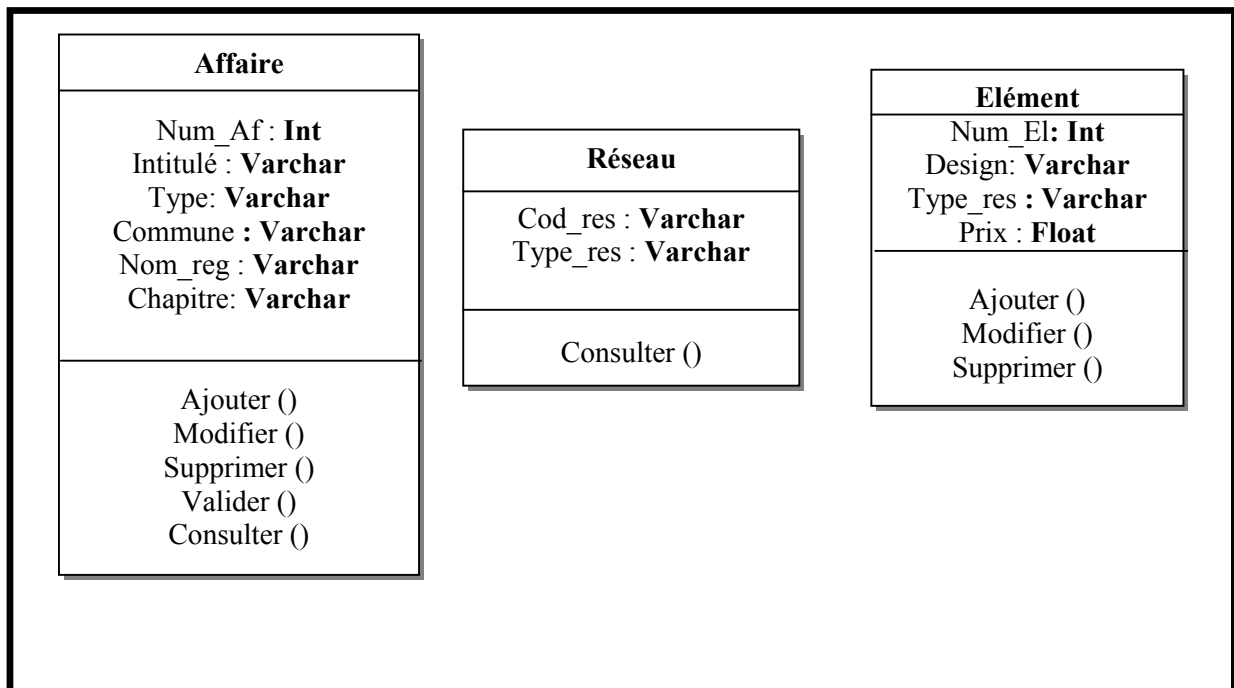


Tableau 5.3: Traduction des associations du système SGPP.

3.4. Les classes modifiées :



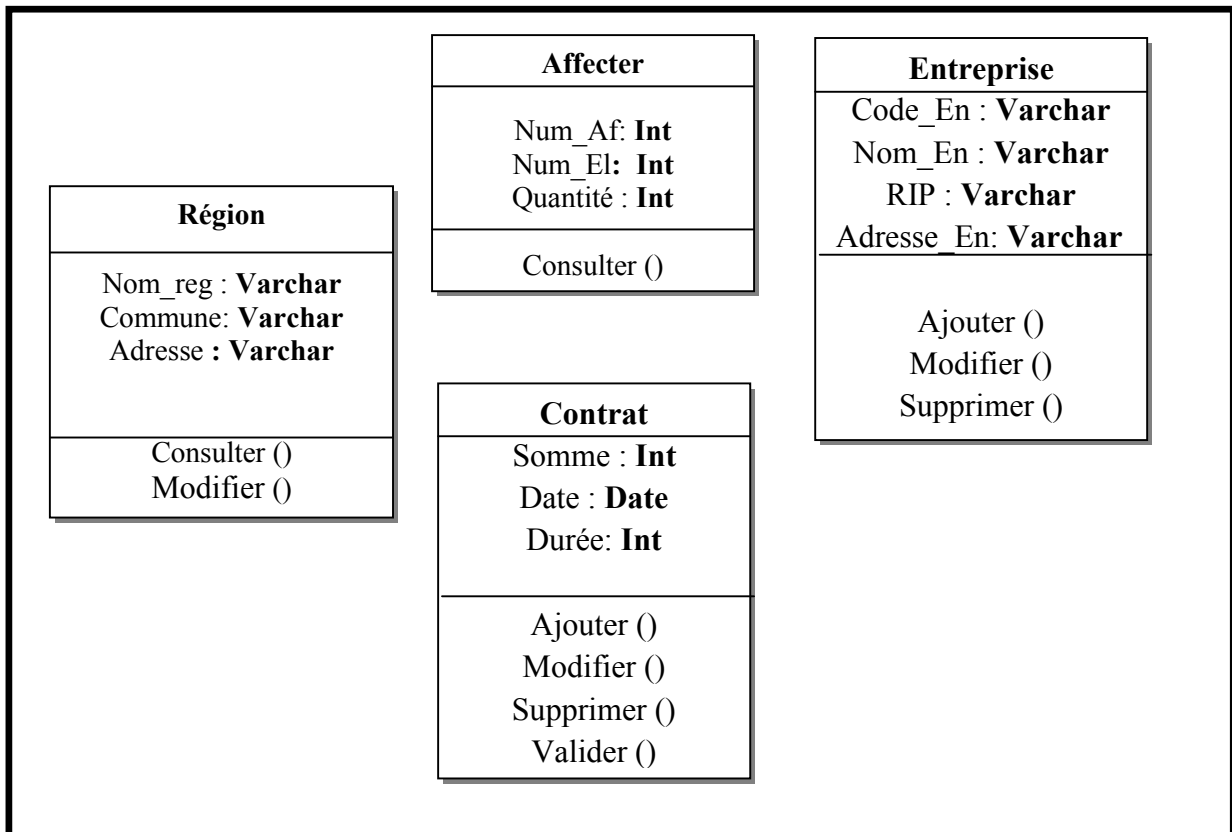


Tableau 5.4: les classes modifiées du système SGPP.

3.5.Passage du modèle objet au modèle relationnel:

- L'utilisation d'un SGBDR impose un changement de représentation entre la structure des classes et la structure des données relationnelles. Les deux structures ayant des analogies, les équivalences exprimées au tableau sont généralement utilisées.
- Une classe définit une structure de données à laquelle souscrivent des instances ; elle correspond donc à une table au modèle relationnel : chaque attribut, donne lieu à une colonne, chaque instance stocke ses données dans une ligne (*T-uplet*) et son OID sert de clé primaire.

Modèle objet	Modèle relationnel
Classe	Table
Attribut de type simple	Colonne
Attribut de type composé	Colonnes ou clé étrangère
Instance	T-uplet
OID	Clé primaire
Association	Clé étrangère ou table de liens
Héritage	Clé primaire identique sur plusieurs tables

Tableau 5.5: Equivalence entre les concepts objets et relationnels.

3.6.Modèle relationnel du système :

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

Relations	Attributs
Affaire	(<u>Num_Af</u> , Intitulé, Type, Nom_reg, Chapitre)
Elément	(<u>Num_El</u> , Design, Type_res, Prix)
Affecter	(<u>Num_Af</u> , <u>Num_El</u> , Quantité)
Entreprise	(<u>Code_En</u> , Nom_En, RIP, Adresse_En)
Contrat	(<u>Somme</u> , Date, Durée)
Région	(<u>Nom_reg</u> , Commune, Adresse)
Réseau	(<u>Cod_res</u> , Type_res)

Tableau 5.6: Table de la base de données

4.Conclusion

Dans ce chapitre nous avons étudié la conception du système de gestion des programmes propres (SGPP).La conception préliminaire nous a permis de spécifier la configuration matérielle de notre futur système. La conception détaillée nous a permis d'élaborer un modèle relationnel représentant les différentes tables de la base de données du système SGPP et de montrer les différentes jointures existantes entre ces différentes tables, donc il nous reste de passer à la phase de réalisation de notre futur système.

CHAPITRE 6

RÉALISATION

1. Introduction :

De l'analyse des besoins à la conception, le produit final sera représenté dans un logiciel, ce dernier est un ensemble d'interfaces homme/machine (IHM) réalisé par des outils de développement.

Dans ce chapitre, nous allons parler du produit final qu'on a construit en décrivant les étapes majeures suivies dans notre travail et quelques interfaces présentées dans le logiciel.

2. Présentation de l'environnement de programmation

2.1. Langage de programmation java

Le langage Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy, présenté officiellement le 23 mai 1995 au SunWorld.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications. Pour cela, divers plateformes et frameworks associés visent à guider, sinon garantir, cette portabilité des applications développées en Java [8].

2.2. NetBeans

NetBeans est un environnement de développement intégré (IDE) pour Java, placé en open source par Sun en juin 2000 sous licence CDDL (Common Développement and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, XML et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages web).

NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OSX et Open VMS.

NetBeans est lui-même développé en Java, ce qui peut le rendre assez lent et gourmand en ressources mémoires [9].

2.3. MYSQL

MySQL est l'un des systèmes de gestion de base de données (SGBD) les plus utilisés au monde, autant par le grand public (application Web principalement) que par les professionnels. MySQL est un SGBD Multi-utilisateurs, qu'il fonctionne sur de nombreux systèmes d'exploitation différents, incluant Linux, Mac OS X, Solaris, Windows 9x, NT, XP et Vista.

MySQL est très facile à administrer. Il prend en charge les API (Application Programming Interface) clients de nombreux langages de programmation (java... etc.) ce qui permet d'écrire facilement les programmes clients qui doivent accéder aux données d'une base MySQL.

3. La présentation de l'IHM

Une application interactive doit être évaluable d'après la phase de son développement, l'évaluation d'un système interactif se base sur un ensemble de critères, cet ensemble n'est pas universel (il n'y a pas des critères fixe ou exhaustifs). Dans notre travail nous avons respecté l'essentiel de ces aspects ; se sont les Aspects ergonomiques intervenant dans la conception des interfaces définies par **Bastien**.

3.1. Guidage : il s'agit de moyens mise en oeuvre pour conseiller, orienter, informer et guider l'utilisateur dans l'utilisation de l'application. Le guidage prend plusieurs formes :

- ✓ indication : il s'agit d'un moyen permettant à l'utilisateur d'atteindre son but
- ✓ groupement / distinction : il s'agit de regrouper ou séparer les éléments d'interface afin de faciliter leur utilisation (saisir, recherche,...), on peut regrouper les items en utilisant des frames, cadres ...
- ✓ feed back immédiate : l'utilisateur doit voir le fait de ses actions surtout les actions erronées immédiatement en utilisant pour ça des moyennes informatives d'avertissement ou d'erreur
- ✓ lisibilité : l'information présentée à l'utilisateur doit être visible, les messages doivent être explicites.

3.2. Charge de travail : Le critère charge de travail concerne l'ensemble des éléments de l'interface qui ont pour effet de réduire la charge perspective des utilisateurs, augmenter l'efficacité du dialogue. Des éléments importants pour ce critère :

- ✓ La brièveté : Présente les informations essentielles pour chaque action

- ✓ La densité de l'information : Limite la qualité d'informations traitées

3.3. Adaptabilité : L'adaptabilité d'un système concerne sa capacité à réagir selon le contexte et selon les besoins et préférences des utilisateurs. Ce critère concerne :

- ✓ La flexibilité : permettre à l'utilisateur d'adapter l'interface à ces besoins
- ✓ prise en compte de l'expérience des utilisateurs

3.4. Gestion des erreurs : Prévention et correction.

3.5. Homogénéité et cohérence : il faut que l'application avoir un même comportement et présentation toute au long de son utilisation surtout si elle est développée par plusieurs personnes. Les aspects suivants sont tenus en compte :

- ✓ même choix de couleurs pour les composants de même type
- ✓ Même libellés pour les boutons qu'ils effectuent les mêmes actions
- ✓ Même emplacement des éléments similaires

4. Quelques interfaces de l'application

4.1. Fenêtre d'authentification :

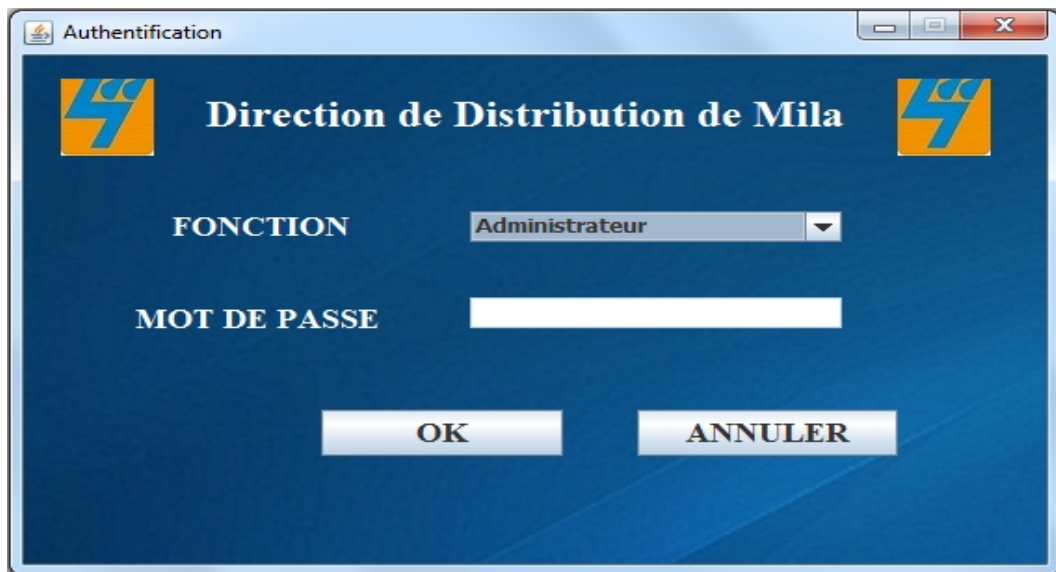


Figure 6.1 : Fenêtre « Authentification »

4.2. Fenêtre d'administrateur :

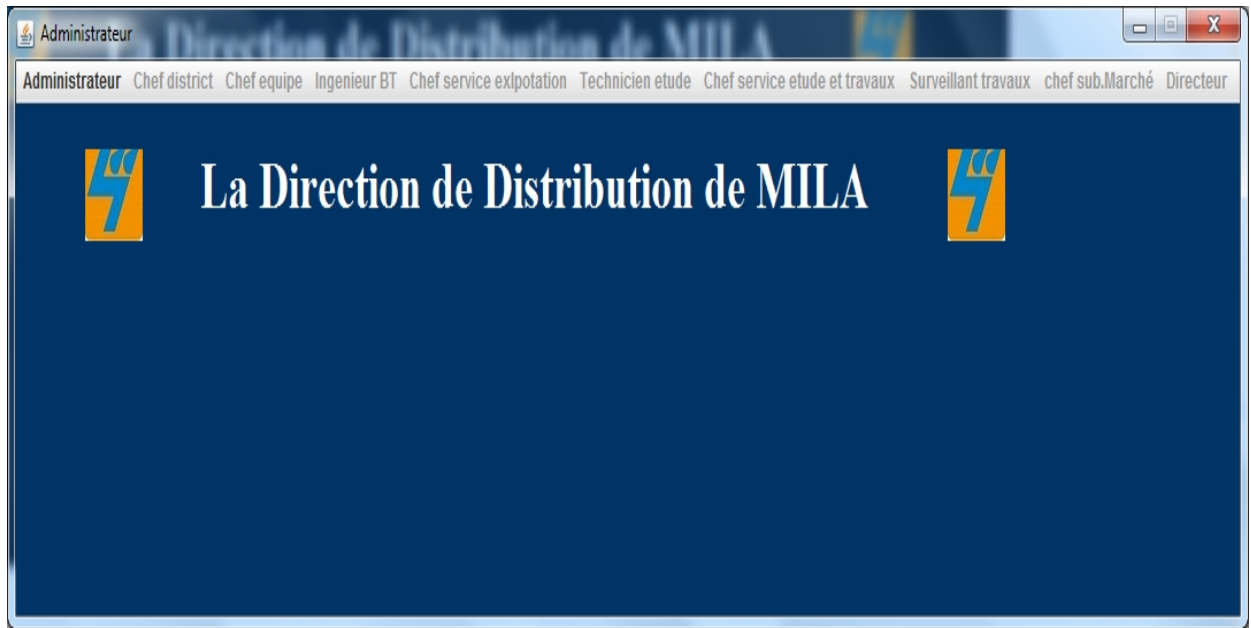


Figure 6.2 : Fenêtre « Administrateur »

4.3. Fenêtre de chef district :

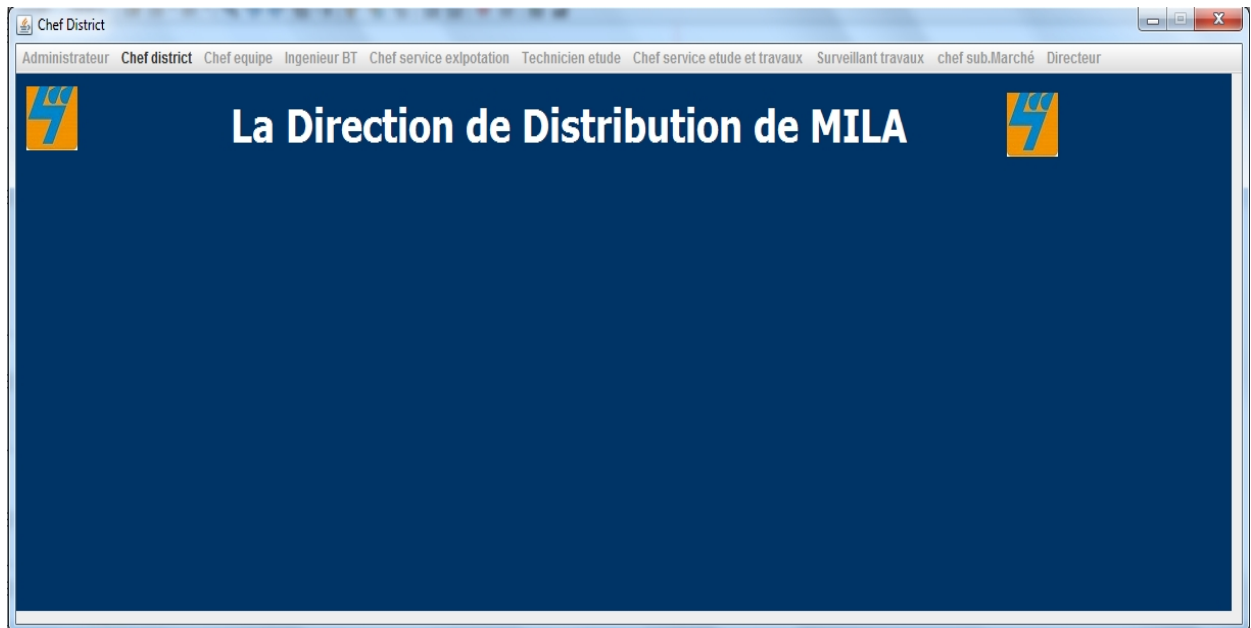


Figure 6.3 : Fenêtre « Chef district »

4.4. Fenêtre d'ajouter affaire :

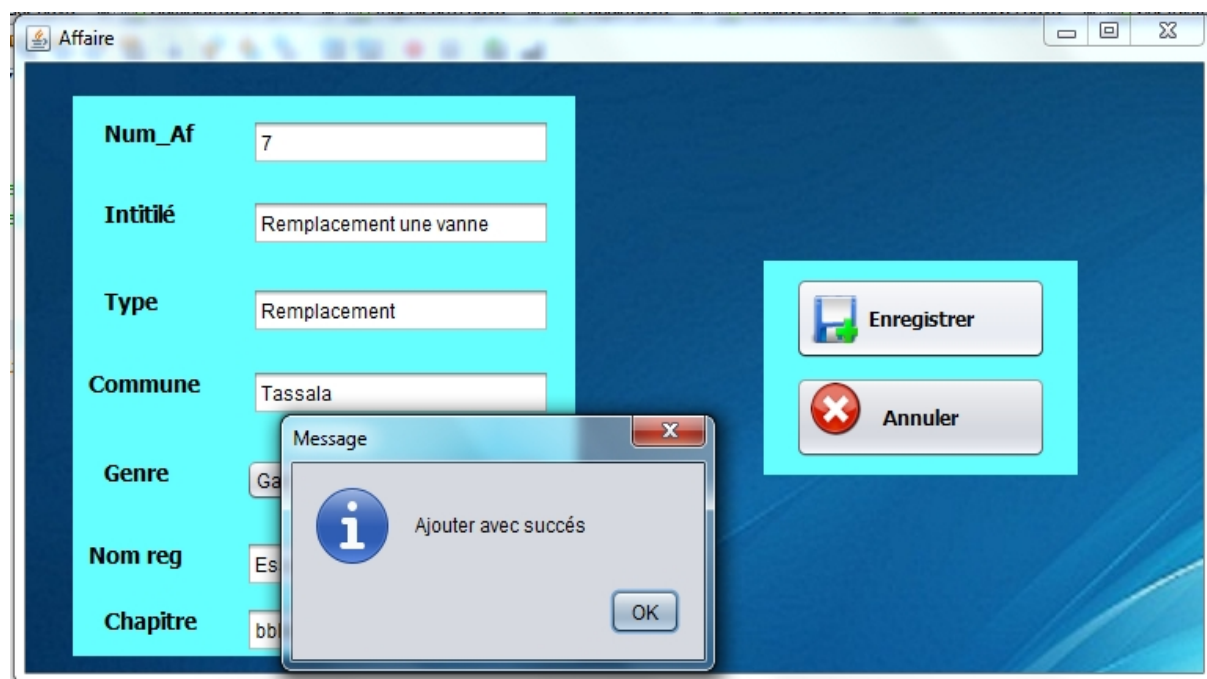


Figure 6.4 : Fenêtre « Ajouter Affaire »

4.5. Fenêtre de modifier affaire :

The screenshot displays a software window titled "Modifier affaire". At the top, there is a search section labeled "Recherche Affaire" with a text input field for "Num_Af" containing the value "7" and a button labeled "Afficher" with a green checkmark icon. Below this is a form with several fields: "Num_Af" (7), "Intitulé" (Remplacement une vanne), "Type" (Remplacement), "Commune" (Arres), "Genre", "Nom_reg", and "Chapitre". To the right of the form are two buttons: "Enregistrer" (with a floppy disk icon) and "Annuler" (with a red X icon). Overlaid on the bottom center of the window is a small "Message" dialog box with a blue information icon, the text "Modification avec succès", and an "OK" button.

Figure 6.5 : Fenêtre « Modifier Affaire »

4.6. Fenêtre de supprimer affaire :

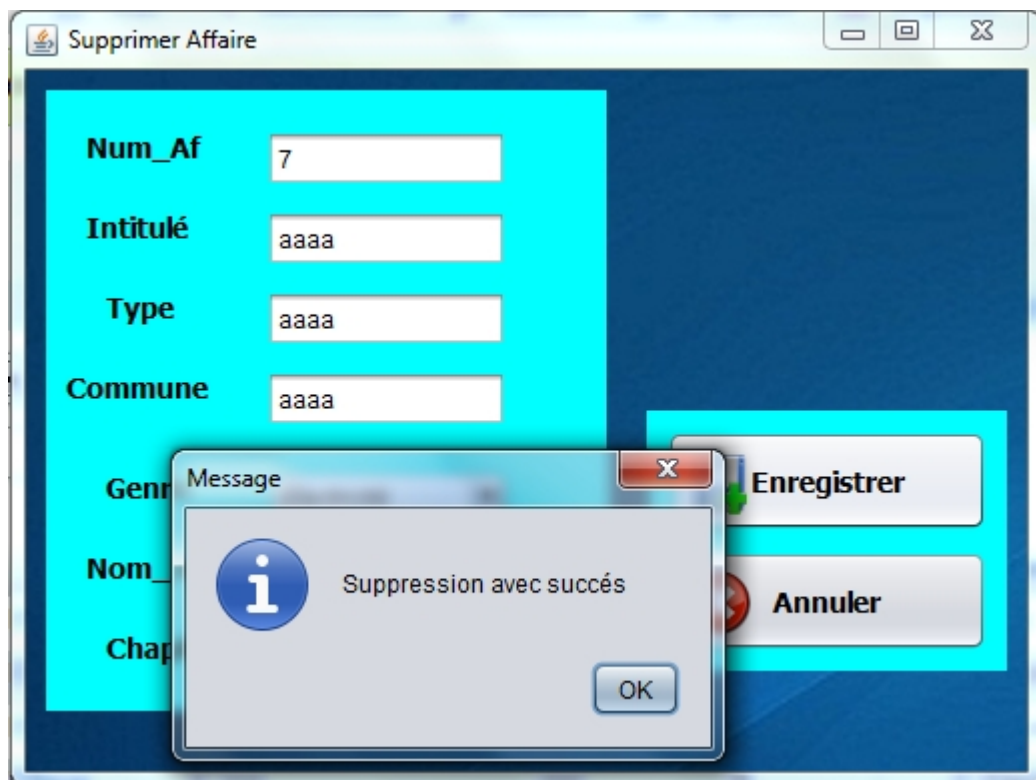


Figure 6.6 : Fenêtre « Supprimer Affaire »

4.7. Fenêtre de valider affaire :

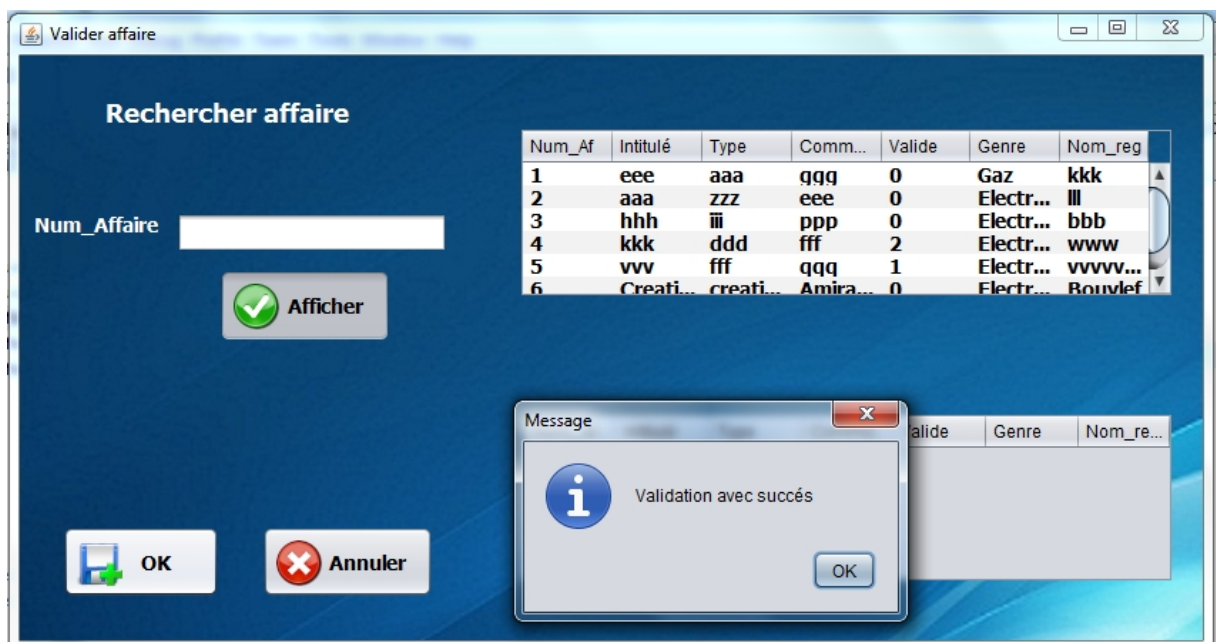


Figure 6.7 : Fenêtre « Valider Affaire »

4.8. Fenêtre d'établir quantitatif :

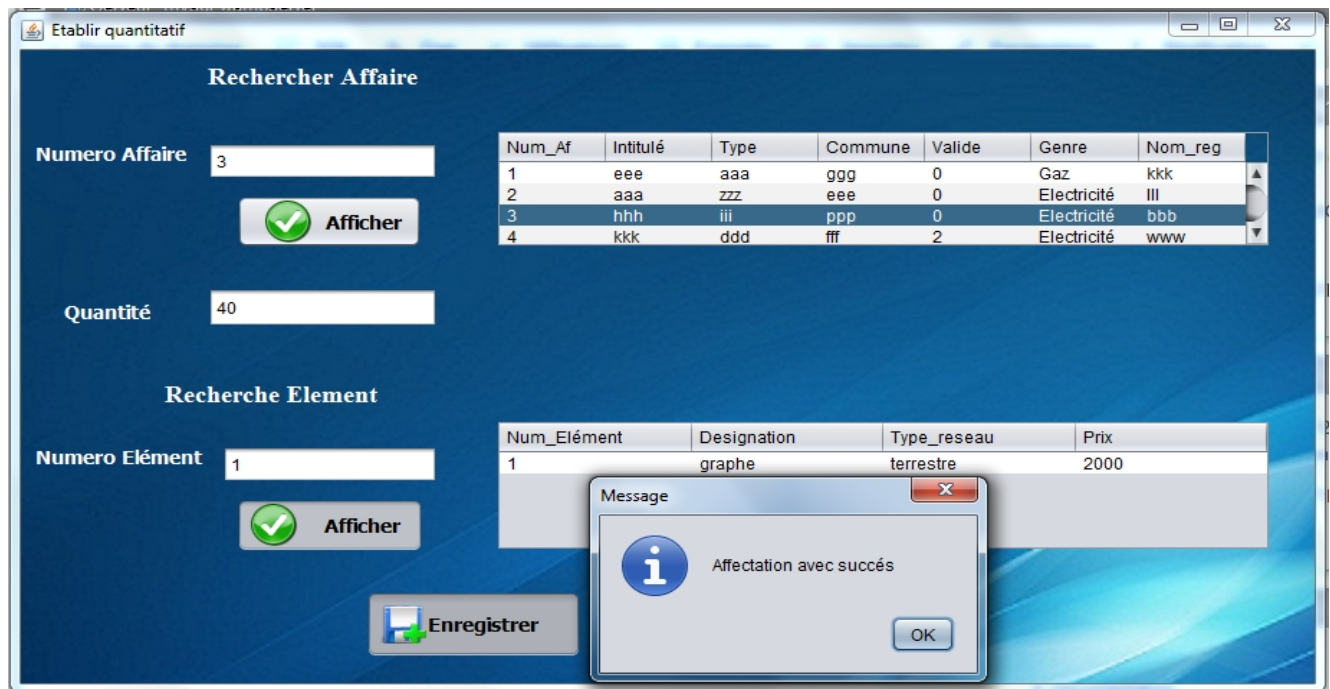


Figure 6.8 : Fenêtre « Etablir Quantitatif »

5. Conclusion

La réalisation d'un logiciel qui dépend de la complexité du problème à coder, comme elle dépend aussi des outils utilisés et des méthodes suivies.

A travers ce chapitre, nous avons présenté l'environnement de développement de notre application, et nous avons présenté quelques interfaces graphiques de notre logiciel que nous avons jugé les plus importantes.

CONCLUSTION GÉNÉRALE

CONCLUSION GÉNÉRALE

Le but visé, par ce projet, est de mettre en œuvre, au niveau de la direction de distribution de MILA, un système qui offre une grande souplesse pour l'acquisition, le stockage, l'organisation et la représentation des informations relatifs au gestion des programmes propres, d'une manière fiable, efficace et rapide en déployant notre application sur une architecture client/serveur.

La démarche que nous avons suivie pour atteindre ce but est basé sur l'utilisation du langage de modélisation UML et le processus de développement 2TUP, le modèle relationnel élaboré au niveau logique a été mise en œuvre au sein de l'ASGBR MySQL avec le langage de programmation Java.

Le stage que nous avons effectué au niveau de la direction de distribution a été pour nous une expérience très enrichissante et très intéressante, dans la mesure où il nous a permis de mettre en pratique et de développer les connaissances que nous avons acquises tout au long de notre cursus universitaire, et de tirer profit de l'expérience et des compétences des collaborateurs.

Nos perspectives, a court terme est de compléter ce travail de point de vu architecture logiciel, et a long terme de réaliser d'autre taches complémentaire au niveau de la direction.

BIBLIOGRAPHIQUE

- [1] Introduction en UML ; T.Penders édition OEM 2002.
- [2] Pascal Roques & Franck vallée, UML en action de l'analyse des besoins à la conception en Java, Eyrolles, 2000.
- [3] Transmission et réseau, Stéphane Lohier, Edition DUNOD 1999.
- [4] Pratique de MYSQL et PHP, Philippe Rigaux, Edition O'RELLY out 2001.
- [5] Technique des ordinateurs et des réseaux, Pierre-Alain GOUPILLE, [Edition juin 1999].
- [6] Intranet Mini Doc, Golden Benett, Edition 1996.
- [7] PUJOLLE des réseaux, Guy Pujolle, Edition EYROLLES 2003.
- [8] http://fr.wikipedia.org/wiki/Langage_de_programmation_Java
- [9] <http://www.techno-science.net/?onglet=glossaire&definition=5346>

RÉSUMÉ

Le projet présenté dans ce mémoire de fin d'études, consiste à la conception et la réalisation d'une application pour le suivi du programme propre au niveau de la direction de la distribution (Elec & Gaz) de MILA.

Nos objectifs consistent en l'automatisation du travail effectué au sein de différents services de la direction, ainsi que l'amélioration de la documentation, la communication et l'échange de l'information entre ces services.

Pour atteindre ces objectifs, nous avons mené une étude conceptuelle en suivant la démarche 2TUP, utilisé UML comme langage de modélisation, NetBeans comme IDE et MySQL comme système de gestion de base de données.

Enfin, notre application sera exploitée par les différents services de la direction de la distribution de Mila.

Mots clés : Système d'information, 2TUP, UML, JAVA, JDBC, MySQL.