

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

Réf. /11

CENTRE UNIVERSITAIRE DE MILA
INSTITUT DES SCIENCES ET DE LA TECHNOLOGIE

Mémoire de fin d'étude
Présenté pour l'obtention du diplôme de

Licence Académique

Domaine: **Mathématiques et Informatique**
Filière : **Informatique**
Spécialité: **Informatique Générale**

Thème

Développement d'une application
client/serveur pour la gestion de la bibliothèque
du centre universitaire de Mila
(Gestion des emprunts)

Présenté par :

- Bouali Meriem.
- Boulmaiz Amina.

Sous la direction de :

- Mr.Boukhechem Nadir.

Année universitaire 2010-2011

Remerciements

*Après avoir terminé ce mémoire de fin d'étude, nous Réservons ces lignes
Pour exprimer nos remerciements les plus sincères à notre dieu
Tout puissant de nous avoir donné la santé et la patience pour terminer
Ce travail Nous remercions tout d'abord*

ALLAH

Nous tenons à exprimer nos remerciements avec un grand plaisir

Et un grand respect à notre encadreur Mr Boukhechem Nadir,

Ses conseils, Sa disponibilité et ses encouragements qui nous ont

Permis de réaliser ce travail dans les meilleures conditions.

*Nous adressons aussi nos reconnaissances à tous les professeurs qui depuis
Quelques années leurs conseils et leurs connaissances qui nous ont bien servis.*

*Nous voudrions aussi exprimer notre gratitude envers tous ceux qui nous ont
accordé.*

Leurs soutien, tant par leurs gentillesse que par leurs dévouement.

*Nous ne pouvons nommer ici tout les personnes qui de près ou de loin qui nous
ont aidé.*

Enfin nous tenons à dire combien le soutien quotidien de notre famille a été

*Important tout au long de ces quelques années, nous leur dois
beaucoup.*

Amina & Meriem



DEDICACE



A mon dieu qui nous a donné le courage pour terminer notre projet.

*A mes parents qui m'ont inculqué un esprit de combativité et de
Persévérance et qui m'ont toujours poussé et motivé dans mes études.*

Sans eux, certainement je ne serais pas à ce niveau.

*A ma trésors ma très chère grand mère qui sacrifier leur vie et m'a
Poussée a lavant pour réaliser mes rêves.*

*A mon grand père qui je souhaite il est dans le vaste paradis de mon
dieu je dédie ce Succès et j'espère que finalement j'atteindre leur but je
n'ai jamais oublié toi et*

Vous Reste toujours dans mon cœur et une pieuse pensée.

A ma sœur « Karima » pour ses encouragements incessants

*Je vous serai toujours reconnaissant pour l'assistance que tu as apportée
dans ma vie.*

*A mon frère « Moussa » et ma petite sœur « Nessrine », Vous occupez
une place particulière dans mon cœur. Je vous dédie ce travail en vous
souhaitant un avenir radieux, plein de bonheur et de succès*

*A tous mes amis, pour votre sincère amitié, votre soutien permanent me
remonte le moral et vos conseils m'incitent à relever les défis.*



AMINA



DEDICACE



Tout d'abord, je rends grâce à dieu tout puissante qui nous a aidés à terminer ce mémoire.

A ma très chère mère

*« Ouarda » qui n'a jamais cessé de m'encourager et de me soutenir. Je ne saurais exprimer ma gratitude et ma reconnaissance. Je vous souhaite une longue et heureuse vie, que dieu vous garde pour moi,
A ma sœur « Hassiba » qui m'a beaucoup aidée et soutenue et qui a toujours encourager,*

A mes très chers et adorables frères « Mohammed » et « Riad »

A mes très chères sœurs « Soumia », « Nawal » et son marie « Salah »,

A mes gentilles et sympathiques cousines Waeil, Khawla et Asma

A toute ma grande famille surtout la famille « Fennour » en Mila et la famille « Bougrira » en Jijel

A toutes mes amies Amina, Fatima.s, Nabila, Fatima.b, Aicha, Souad, Amina.t et Roukia qui m'on accompagné ces trois ans passées en CUM.

A tous ceux que j'aime et qui m'aiment A ceux qui me connaissent et croient en moi et que je n'ai pas cité Je vous dédie ce travail en vous

souhaitant un avenir radieux, plein de

Bonheur et de succès.



MERIE.M.

Résumé

Ce projet traite le développement d'une application client/serveur pour la gestion de la bibliothèque du centre universitaire de Mila. Il permet aux utilisateurs de l'application de faire toutes les tâches de façon facile et souple.

Pour développer et modéliser notre application on a utilisé le langage de modélisation universel UML. On a aussi choisit d'utiliser l'architecture clients serveur 3-tiers qui présente beaucoup d'avantages par rapport aux autres architectures. Pour l'implémentation on a utilisé le langage java avec l'IDE NetBeans et le SGBD MYSQL.

Les Mots clé

UML, Gestion de bibliothèque, base de donnés, Architecture client/serveur.

الملخص

هذا المشروع يعالج تطوير برنامج زبون/خادم (Client/serveur) لمكتبة المركز الجامعي لولاية ميلة, حيث يسمح لمختلف المستعملين بالقيام بمختلف الأعمال بطريقة سهلة و مرنة.
لأجل تطوير هذا البرنامج إستعملنا لغة UML و إستعملنا كذلك هندسة Client/serveur 3-tiers التي تتمتع بخصائص و فوائد كثيرة مقارنة بالهندسات الأخرى.
من أجل البرمجة إستعملنا لغة البرمجة Java ونظام تسيير قاعدة البيانات SGBD MYSQL.

الكلمات المفتاحية

UML, تسيير المكتبة, قاعدة البيانات, هندسة زبون/خادم (Client/serveur).

Sommaire

INTRODUCTION GENERALE	1
CHAPITRE I : GENERALITES	4
INTRODUCTION	4
1-LA MODELISATION AVEC UML	4
1.1- DEFINITION D'UML.....	4
1.2- LES DIAGRAMMES D'UML.....	4
1.2.1- Diagramme de cas d'utilisation	5
1.2.1.1-Eléments de base des cas d'utilisation.....	5
1.2.1.2-Les relations de communication	6
1.2.2- Diagramme de classe	6
1.2.3- Diagramme de séquence.....	6
1.2.4- Diagramme de déploiement.....	8
1.2.5- Diagramme d'activité	8
2 - PRESENTATION DE L'ARCHITECTURE CLIENT/SERVEUR	9
2.1- DEFINITION	9
2.2- LES DIFFERENTS MODELES DE CLIENT/SERVEUR.....	10
2.3- QUELQUES ARCHITECTURES APPLICATIVES	10
CONCLUSION.....	12
CHAPITRE II : SPECIFICATION DES BESOINS.....	14
INTRODUCTION.....	14
1-PRESENTATION DE L'ORGANISME D'ACCUEIL.....	14
1.1- HISTORIQUE DE L'UNIVERSITE DE MILA	14

2-BIBLIOTHEQUE CENTRALE DE UNIVERSITE MILA.....	15
2.1-PRESENTATION DE LA BIBLIOTHEQUE	15
2.2- DESCRIPTION DU FONCTIONNEMENT DE LA BIBLIOTHEQUE	16
3-LES OBJECTIFS DU STAGE	17
4-IDENTIFIER LES ACTEURS	17
5-LES CAS D'UTILISATIONS DU SYSTEME.....	19
6-DESCRIPTION TEXTUELLE DU CAS D'UTILISATION.....	21
7-REALISATION DES DIAGRAMMES DE SEQUENCES	31
7.1- DIAGRAMMES DE SEQUENCES DU CAS D'UTILISATION AUTHENTIFIE	32
7.2- DIAGRAMMES DE SEQUENCES DU CAS D'UTILISATION GESTION D'OUVRAGE.....	33
7.2.1- diagrammes de séquences du cas d'utilisation d'ajouter ouvrage.....	34
7.2.2- diagrammes de séquences du cas d'utilisation de modifier ouvrage.....	35
7.2.3- diagrammes de séquences du cas d'utilisation d'ajouter un exemplaire	36
7.2.4- diagrammes de séquences du cas d'utilisation de modifier exemplaire.....	37
7.2.5- diagrammes de séquences du cas d'utilisation de supprimer un exemplaire.....	38
7.3- DIAGRAMMES DE SEQUENCES DU CAS D'UTILISATION GESTION DES ADHERENTS	39
7.3.1- diagrammes de séquences du cas d'utilisation d'ajouter un adhérent	40
7.3.2- diagrammes de séquences du cas d'utilisation de modifier un adhérent.....	41
7.3.3- diagrammes de séquences du cas d'utilisation de supprimer un adhérent.....	42
7.4- DIAGRAMMES DE SEQUENCES DU CAS D'UTILISATION GESTION DES EMPRUNTS	42
7.4.1- diagrammes de séquences du cas d'utilisation d'emprunter un exemplaire d'un ouvrage.....	43
7.4.2- diagrammes de séquences du cas d'utilisation de retourner un exemplaire d'un ouvrage.....	44
7.5- DIAGRAMMES DE SEQUENCES DE CAS D'UTILISATION DE RECHERCHER UN OUVRAGE....	44
7.6- DIAGRAMMES DE SEQUENCES DU CAS D'UTILISATION GESTION DE PENALITE.....	45
7.6.1- diagrammes de séquences du cas d'utilisation ajouté pénalité.....	45
7.6.2- diagrammes de séquences du cas d'utilisation supprimé pénalité.....	46
7.6.3- diagrammes de séquences du cas d'utilisation afficher liste des adhérents pénalisés	46
CONCLUSION.....	46

CHAPITRE III : ANALYSE.....	48
INTRODUCTION.....	48
1-ANALYSE DES DOMAINES.....	48
1.1-IDENTIFICATION LES CONCEPTS DES DOMAINES	48
1.2-AJOUTE LES ATTRIBUTS ET LES ASSOCIATIONS ENTRE CES CLASSES (MODELE DE DOMAINE)	49
2-REALISATION DE DIAGRAMME D'ACTIVITE DE CHAQUE CAS	
D'UTILISATION	50
2.1-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION AUTHENTIFIE.....	51
2.2-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION AJOUTER OUVRAGE.....	52
2.3-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION MODIFIER OUVRAGE	53
2.4-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION AJOUTER EXEMPLAIRE.....	54
2.5-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION MODIFIER EXEMPLAIRE.....	55
2.6-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION SUPPRIMER EXEMPLAIRE	56
2.7-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION AJOUTER ADHERENT.....	57
2.8-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION MODIFIER ADHERENT.....	58
2.9-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION SUPPRIMER ADHERENT	59
2.10-DIAGRAMMES D'ACTIVITE DU CAS D'UTILISATION EMPRUNTER EXEMPLAIRE.....	60
2.11-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION RETOURNER EXEMPLAIRE.....	61
2.12-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION RECHERCHER OUVRAGE.....	62
2.13-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION AJOUTER PENALITE.....	63
2.14-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION SUPPRIMER PENALITE	64
2.15-DIAGRAMME D'ACTIVITE DU CAS D'UTILISATION AFFICHER LISTE DES ADHERENTS PENALISES :	64
CONCLUSION.....	64
CHAPITRE IV : CONCEPTION.....	66
INTRODUCTION.....	66
1-DICTIONNAIRE DE DONNEES	66
1.1-CONCEVOIR LES CLASSES ET LES ATTRIBUTS	66
1.2-CONCEVOIR LES OPERATIONS.....	68

2-LES DIAGRAMMES DE SEQUENCE INTERACTION DE CHAQUE CAS	
D'UTILISATION	69
2.1-S' AUTHENTIFIER	70
2.2-AJOUTER OUVRAGE.....	71
2.3-MODIFIER OUVRAGE	72
2.4-AJOUTER EXEMPLAIRE	73
2.5-MODIFIER EXEMPLAIRE.....	74
2.6-SUPPRIMER EXEMPLAIRE	75
2.7-AJOUTER ADHERENT	76
2.8-MODIFIER ADHERENT	77
2.9-SUPPRIMER ADHERENT	78
2.10-EMPRUNTER EXEMPLAIRE.....	80
2.11- RETOURNER EXEMPLAIRE	81
2.12-RECHERCHER OUVRAGE.....	82
2.13-AJOUTER PENALITE	83
2.14-SUPPRIMER PENALITE	84
2.15-AFFICHER LISTE DES ADHERENTS PENALISES	85
3-DIAGRAMME DE CLASSES DE CONCEPTION	85
4 -LE MODELE LOGIQUE DE DONNEES.....	86
5- DIAGRAMME DE DEPLOIEMENT.....	87
CONCLUSION.....	88
CHAPITRE V: REALISATION.....	90
INTRODUCTION.....	90
1-ENVIRONNEMENT DE TRAVAIL	90
1.1-JAVA	90
1.2-NETBEANS	91
1.3-SGBDR(MYSQL).....	91
1.4-PHPMYADMIN	91
1.5-L'ARCHITECTURE 3 –TIERS	92
2- QUELQUE INTERFACE DE L'APPLICATION.....	93

CONCLUSION	97
CONCLUSION GENERALE	98
BIBLIOGRAPHIE	99

Liste des tableaux

Tableau II.1 : Fiche description textuelle du cas authentification.....	21
Tableau II.2 : Fiche description textuelle du cas Gestion d'ouvrage.....	24
Tableau II.3 : Fiche description textuelle du cas Gestion des adhérents.....	26
Tableau II.4 : Fiche description textuelle du cas Gestion des emprunts.....	28
Tableau II.5 : Fiche description textuelle du cas Rechercher ouvrage.....	29
Tableau II.6 : Fiche description textuelle du cas Gestion de pénalité.....	31
Tableau V.1 : Description détaillée des classes d'objets.....	67
Tableau V.2 : Liste et description des opérations.....	69

Liste des figures

Figure I.1 : Les différents éléments de diagramme de séquence.....	7
Figure I.2 : Système client/serveur.....	9
Figure I.3 : La répartition des différentes couches de l'architecture 3-tiers.....	11
Figure I. 4 : L'architecture client/serveur.....	12
Figure II.1 : La situation de CUM.....	14
Figure II.2 : La structure administrative du centre.....	15
Figure II.3 : Diagramme de cas d'utilisation.....	20
Figure II.4 : Diagramme de séquence du cas authentifier.....	32

Figure II.5: Diagramme de séquence du cas Gestion d'ouvrage.....	33
Figure II.6 : Diagramme de séquence du cas ajouter ouvrage.....	34
Figure II.7 : Diagramme de séquence du cas modifier ouvrage.....	35
Figure II.8: Diagramme de séquence du cas ajouté exemplaire.....	36
Figure II.9 : Diagramme de séquence du cas modifié exemplaire.....	37
Figure II.10 : Diagramme de séquence du cas supprimé exemplaire.....	38
Figure II.11 : Diagramme de séquence du cas Gestion des adhérents.....	39
Figure II.12 : Diagramme de séquence du cas ajouter un adhérent.....	40
Figure II.13 : Diagramme de séquence de cas modifié adhérent.....	41
Figure II.14 : Diagramme de séquence du cas supprimé adhérent.....	42
Figure II.15 : Diagramme de séquence de cas Gestion des emprunte.....	42
Figure II.16 : Diagramme de séquence du cas emprunté un exemplaire d'un ouvrage.....	43
Figure II.17 : Diagramme de séquence de cas retourné un exemplaire d'un ouvrage.....	44
Figure II.18 : Diagramme de séquence du cas Rechercher ouvrage.....	44
Figure II.19 : Diagramme de séquence du cas Gestion de pénalité.....	45
Figure II.20 : Diagramme de séquence du cas ajouté pénalité.....	45
Figure II.21 : Diagramme de séquence du cas supprimé pénalité.....	46
Figure II.22 : Diagramme de séquence du cas affiché liste des adhérents pénalisés.....	46
Figure III.1 : Diagramme de modèle de domaine.....	50
Figure III.2 : Diagramme d'activité du cas authentifie.....	51
Figure III.3 : Diagramme d'activité du cas ajouter ouvrage.....	52
Figure III.4 : Diagramme d'activité du cas modifier ouvrage.....	53

Figure III.5 : Diagramme d'activité du cas ajouté exemplaire.....	54
Figure III.6 : Diagramme d'activité du cas modifié exemplaire.....	55
Figure III.7 : Diagramme d'activité du cas supprimé exemplaire.....	56
Figure III.8 : Diagramme d'activité du cas ajouté adhérent.....	57
Figure III.9 : Diagramme d'activité du cas modifié adhérent.....	58
Figure III.10 : Diagramme d'activité du cas supprimé adhérent.....	59
Figure III.11 : Diagramme d'activité du cas emprunté exemplaire.....	60
Figure III.12 : Diagramme d'activité du cas retourné exemplaire.....	61
Figure III.13 : Diagramme d'activité du cas Rechercher ouvrage.....	62
Figure III.14 : Diagramme d'activité du cas ajouté pénalité.....	63
Figure III.15 : Diagramme d'activité du cas supprimé pénalité.....	64
Figure III.16 : Diagramme d'activité du cas afficher la liste des adhérents pénalisés....	65
Figure IV.1 : Diagramme de séquence d'interaction du cas authentifier.....	70
Figure IV.2 : Diagramme de séquence d'interaction du cas ajouté ouvrage.....	71
Figure IV.3 : Diagramme de séquence d'interaction du cas modifié ouvrage.....	72
Figure IV.4: Diagramme de séquence d'interaction du cas ajouté exemplaire.....	73
Figure IV.5 : Diagramme de séquence d'interaction du cas modifié exemplaire.....	74
Figure IV.6 : Diagramme de séquence d'interaction du cas supprimé exemplaire.....	75
Figure IV.7: Diagramme de séquence d'interaction du cas ajouté adhérent.....	76
Figure IV.8 : Diagramme de séquence d'interaction du cas modifié adhérent.....	77
Figure IV.9 : Diagramme de séquence d'interaction du cas supprimé adhérent.....	78

Figure IV.10 : Diagramme de séquence d'interaction du cas vérification.....	79
Figure IV.11 : Diagramme de séquence d'interaction du cas emprunté exemplaire.....	80
Figure IV.12: Diagramme de séquence d'interaction du cas retourné exemplaire.....	81
Figure IV.13 : Diagramme de séquence d'interaction du cas rechercher ouvrage.....	82
Figure IV.14 : Diagramme de séquence d'interaction du cas ajouté pénalité.....	83
Figure IV.15 : Diagramme de séquence d'interaction du cas supprimé pénalité.....	84
Figure IV.16 : Diagramme de séquence d'interaction du cas afficher liste des adhérents pénalisés.....	85
Figure IV.17 : Diagramme de class conception.....	86
Figure IV.18 : Diagramme de déploiement.....	88
Figure V.1: Interface d'accueil.....	93
Figure V.2: L'interface de menu principale.....	94
Figure V.3 : L'interface de gestion d'ouvrage	94
Figure V.4 : L'interface d'ouvrage.....	95
Figure V.5 : L'interface d'exemplaire.....	95
Figure V.6 : L'interface de gestion d'adhérent.....	96
Figure V.7 : L'interface de gestion d'emprunt.....	96
Figure V.8 : L'interface de gestion de recherche.....	97
Figure V.9 : L'interface de gestion de pénalité.....	97

Introduction générale

Durant ces dernières années l'informatique s'est imposée d'une manière très impressionnante dans les entreprises, cela est dû à son apport extraordinaire dans le domaine de gestion des bases de données. En effet, l'informatique désigne l'automatisation du traitement de l'information par un système concret « machine ». On entend également par « l'informatique » l'ensemble des sciences et techniques en rapport avec le traitement de l'information.

L'informatique est de plus en plus utilisée dans tous les domaines d'activités y compris celui de la gestion des bibliothèques auquel nous rattacherons d'ailleurs notre étude, et cela pour une meilleure gestion des différents traitements exigés par cette activité de gestion des bibliothèques.

Toutes les universités ont une bibliothèque centrale souvent chargée de coordonner le réseau de ses diverses bibliothèques universitaires, tout en exerçant sa mission traditionnelle d'acquisition, de conservation et de mise à disposition des documents.

Notre projet consiste à développer une application client/serveur pour la gestion de la bibliothèque au niveau de la bibliothèque du centre universitaire de Mila, nous nous intéressons exactement à la gestion des emprunts.

Le langage de modélisation qu'on a utilisé est UML ces diagrammes sont largement utilisés dans chaque étape et phases de développement. Pour l'implémentation, le choix du langage de programmation a été dicté par le type de l'application qui devrait être réalisée d'une part, et être accessible via le réseau Intranet de l'université d'autre part. Ainsi, le choix s'est porté sur le langage de programmation java. La base de données est implémentée avec le SGBD MySQL.

Nous avons organisé ce mémoire de la façon suivante :

Dans le premier chapitre intitulé « Généralités » nous présentons la méthode et les outils qui nous ont servi à l'accomplissement de notre projet.

Introduction générale

Dans le deuxième chapitre «Spécification des besoins » Nous commençons à approfondir la compréhension du contexte du système par la collecte des informations et nous déterminons les cas d'utilisation du système.

Dans le troisième chapitre «Analyse » nous analysons le système, et nous déterminons ses différentes classes.

Dans le quatrième chapitre « conception» on présente la conception de notre système d'information.

La réalisation et l'implémentation de notre application fera l'objet du cinquième chapitre « Réalisation» dans lequel nous illustrerons les différentes parties de l'application à savoir la base de données et les différentes interfaces qui permettent l'accès à celle-ci.

Enfin, nous terminerons ce mémoire par une conclusion générale.

Chapitre I

Généralités

Introduction

1- La modélisation avec UML.

2- Présentation de l'architecture client/serveur.

Conclusion.

CHAPITRE I : GENERALITES

Introduction

Dans le cadre de ce chapitre, nous allons définir quelques généralités portant sur la méthode et les outils utilisés pour réaliser notre projet. Nous allons présenter le langage de modélisation unifié UML (Unifié Modelling Language), définir la démarche générique du processus de développements logiciel qui l'accompagne, et présenter l'architecture client/serveur.

1-La modélisation avec UML

Pour programmer une application, il ne convient pas de se lancer directement dans l'écriture du code, il faut d'abord organiser ses idées, les documenter, puis organiser la réalisation en définissant les modules et étapes de la réalisation, C'est cette démarche antérieure à l'écriture que l'on appelle modélisation.

1.1- Définition d'UML

UML (*unified Modelling language*) est un langage de modélisation qui se base sur une sémantique précise et sur des notations graphique, il est indépendant du domaine d'application et du langage d'implémentation. UML est ensemble de graphismes définis pour représenter les objets, leurs types et leurs interactions, leurs évolutions et leurs assemblages. Donc il représente une grande richesse et diversité. Ce qui facilite la comparaison et l'évolution des solutions. .

1.2- Les diagrammes d'UML

UML définit plusieurs sortes de diagrammes pour représenter les différents points de vue de modélisation. Un diagramme donne à l'utilisateur un moyen de visualiser et de manipuler des éléments de modélisation. On distingue trois types de vue :

- Vue fonctionnelle : diagramme de cas d'utilisation(DCU).

- Vue statique ou structurelle : permet de représenter le système physique :
 - ❖ Diagramme d'objet (DOB).
 - ❖ Diagramme de classe (DCL).
 - ❖ Diagramme de composant.
 - ❖ Diagramme de déploiement.
- Vue dynamique ou comportementale :
 - ❖ Diagramme de séquence (DCE).
 - ❖ Diagramme de collaboration.
 - ❖ Diagramme d'état – transition (DET).
 - ❖ Diagramme d'activité (DAC).

1.2.1-Diagramme de cas d'utilisation

Ce diagramme est destiné à représenter les besoins des utilisateurs par rapport au système. Il constitue un des diagrammes les plus importants dans l'analyse d'un système.

1.2.1.1- Eléments de base des cas d'utilisation

Les acteurs

Généralement les acteurs sont les utilisateurs du système, il constitue les éléments externes au système.

Les acteurs peuvent être : des humains ou des logiciels, on distingue :

- *Les acteurs primaires* : ceux par le quel le système produit un résultat observable.
- *Les acteurs secondaires* : ils sont généralement sollicités pour des informations complémentaires.

Cas d'utilisation

Représente un ensemble de séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier.

1.2.1.2- Les relations de communication

- **Les relations entre acteurs** : La seule relation entre acteur est la relation de *généralisation*. Quand un acteur fils hérite d'un acteur père, il hérite en réalité de toutes les associations du père.
- **Les relations entre cas d'utilisation** :

> **Relation d'inclusion** : Une relation d'inclusion d'un cas d'utilisation A par rapport à un cas d'utilisation B signifie qu'une instance de A contient le comportement décrit dans B.

> **Relation d'extension** : Une relation d'extension d'un cas d'utilisation A par rapport à un cas d'utilisation B signifie qu'une instance de A peut être étendue par le comportement décrit dans B.

> **Relation de généralisation** : Les cas d'utilisation descendants héritent de la description de leurs parents communs. Chacun d'entre eux peut néanmoins comprendre des interactions spécifiques supplémentaires.

1.2.2- Diagramme de classe

Le diagramme de classes a pour objectif de décrire la structure des entités manipulées par les utilisateurs, le diagramme de classe décrit la structure interne du système. On définit dans ce diagramme les classes du système et les relations entre ces classes.

· **Une classe** : Représente la description abstraite d'un ensemble d'objets possédant les mêmes caractéristiques.

· **Un objet**: Est une entité possédant une identité et encapsulant un état et un comportement. Un objet est une instance (ou occurrence) d'une classe.

1.2.3- Diagramme de séquence

Ce diagramme permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets. Les différents éléments du diagramme de séquence sont décrits dans la figure suivante :

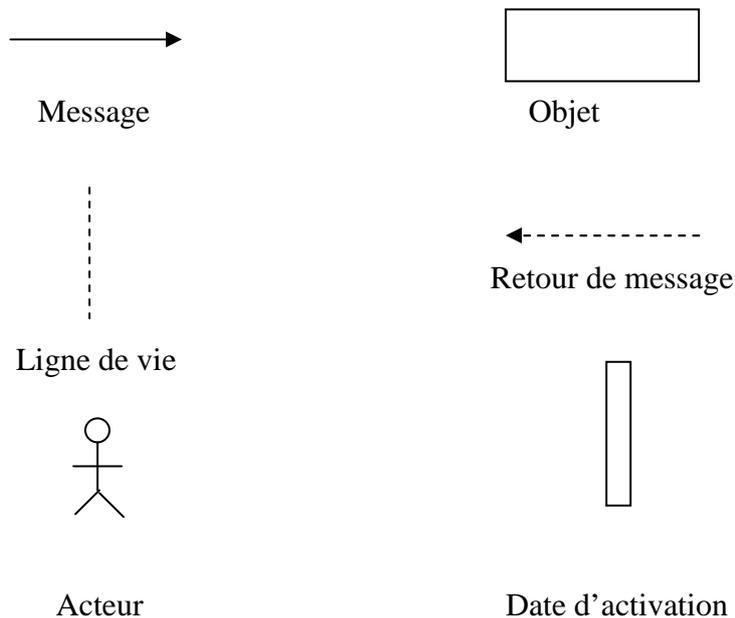


Figure I.1 : Les différents éléments de diagramme de séquence.

- **Ligne de vie** : Représente l'ensemble des opérations exécutées par un objet.
- **Message**: Un message est une transmission d'information unidirectionnelle entre deux objets, l'objet émetteur et l'objet récepteur. Dans un diagramme de séquence, deux types de messages peuvent être distingués :
 - > **Message synchrone** : Dans ce cas l'émetteur reste en attente de la réponse à son message avant de poursuivre ses actions.
 - > **Message asynchrone** : Dans ce cas, l'émetteur n'attend pas la réponse à son message, il poursuit l'exécution de ses opérations.

1.2.4- Diagramme de déploiement

- Les diagrammes de déploiement montrent la disposition physique du matériel qui compose le système et la répartition des composants du système sur ce matériel.
- Les ressources matérielles sont représentées sous forme de nœuds. ces nœuds sont connectés entre eux, à l'aide d'un support de communication.

1.2.5-Diagramme d'activité :

- Un diagramme d'activité permet de représenter le comportement interne d'une méthode (la réalisation d'une opération) ou d'un cas d'utilisation.
- Le diagramme d'activité est constitué de deux éléments fondamentaux :
 - Les activités qui décrivent un traitement.
 - Les transitions qui connectent les activités entre elles et déterminent la prochaine activité à déclencher si une activité est terminée
- Les activités sont reliées par des transitions automatiques.
- UML est seulement un langage de modélisation, il ne propose pas une démarche de modélisation explicitant et encadrant toutes les étapes d'un projet (de la compréhension des besoins à la production du code de l'application).

Pour réaliser notre projet on utilise une démarche qui définit une séquence d'étapes, partiellement ordonnées, dont l'objectif est de produire un logiciel de qualité qui répond aux besoins des utilisateurs dans des temps prévisibles, ces étapes sont :

- ❖ Phase de Spécification et présentation des besoins : cet étape permet de représenter les différents besoins des utilisateurs du système sur un diagramme de cas d'utilisation et aussi représente les différentes interactions entre les objets sur un diagramme de séquence système.

- ❖ Phase d'analyse : cette étape permet de détailler plus les besoins des utilisateurs du système. on définit dans cette étape le modèle de domaine et le diagramme d'activité.

- ❖ Phase de conception : cet étape permet de représenté la structure interne du système, on utilise le diagramme de séquence d'interaction, le diagramme de classe de conception et le diagramme de déploiement.

2- Présentation de l'architecture client/serveur

Ces vingt dernières années ont vues une évolution majeure des systèmes d'information, à s'avoir le passage d'une architecture centraliser à travers de grosses machine vers une architecture distribuée basée sur l'utilisation de serveurs et de postes clients grâce à l'utilisation des PC et des réseaux.

2.1-Définition

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

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

Le dialogue entre les applications peut se résumer par :

- le client demande un service au serveur.

- le serveur réalise ce service et renvoie le résultat au client.

Un système client/serveur fonctionne selon le schéma suivant:

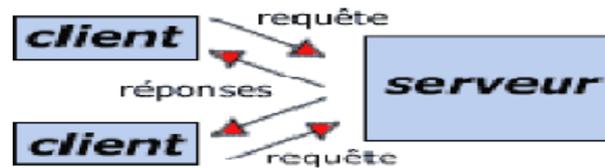


Figure I.2 : Système client/serveur.

2.2-Les différents modèles de client/serveur

- Le client -serveur de donnée

Dans ce cas, le serveur assure des tâches de gestion, stockage et de traitement de données.

C'est le cas le plus connu des architectures client/serveur, il est utilisé par tous les grands SGBD :

La base de données avec tous ses outils (sauvegarde,...) est installée sur un poste serveur.

Sur les clients, un logiciel d'accès est installé et permet d'accéder à la base de données du serveur.

Tous les traitements sur les données sont effectués sur le serveur qui renvoie les informations demandées (requête SQL) par le client.

- Client-serveur de présentation

Dans ce cas la présentation des pages affichées par le client est intégralement prise en charge par le serveur. Cette organisation présente l'inconvénient de générer un fort trafic réseau

- Client-serveur de traitement

Dans ce cas le serveur effectue des traitements à la demande du client. Il peut s'agir de traitement particulière sur des données, de vérification de formulaires de saisie...etc.

2.3 Quelques architectures applicatives

- Architecture à 1-tiers

Dans une approche d'application de type 1-tiers, les trois couches sont fortement et intimement liées, et s'exécutent sur la même machine. Dans ce cas, on ne peut pas parler d'architecture client-serveur mais d'informatique centralisée.

➤ Architecture 2 tiers

Dans une architecture 2 tiers, encore appelé client-serveur de premier génération ou client-serveur de données, le poste client se contente de déléguer la gestion des données à un service spécialisé. Le cas typique de cette architecture est une application de gestion fonctionnant sous Windows ou Linux et exploitant un SGBD centralisé.

Ce type d'application permet de tirer partie de la puissance des ordinateurs déployés en réseaux pour fournir à l'utilisateur une interface riche, tout en garantissant la cohérence des données qui restent gérées de façon centralisée.

➤ Architecture 3 tiers

Dans cette architecture un poste client simple communiquant avec le serveur par le biais d'un protocole de communication, les données sont toujours gérées de façon centralisée, La présentation est prise en charge par le post client et la logique applicative est prise en charge par un serveur intermédiaire.

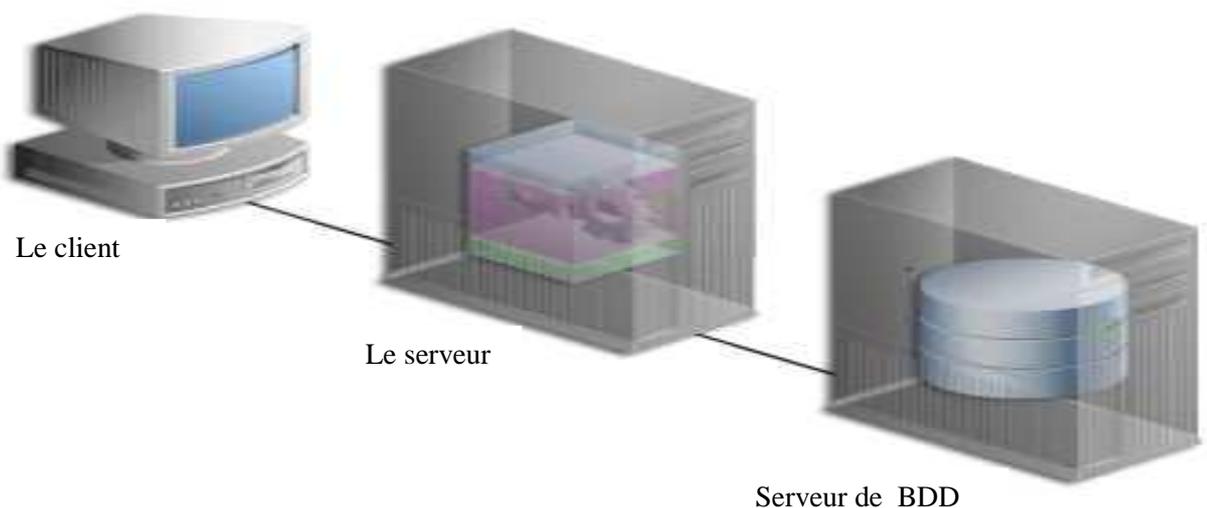


Figure I.3 : La répartition des différentes couches de l'architecture 3-tiers.

➤ Avantages de cette architecture

- Unicité de l'information : pour un site web dynamique qui vend des articles par exemple, certains articles du site sont stockés dans une base de données sur le serveur. De cette manière, les informations restent identiques. Chaque utilisateur accède aux mêmes informations.
- Meilleure sécurité : Lors de la connexion un PC client ne voit que le serveur, et non les autres PC clients. De même, les serveurs sont en général très sécurisés contre les attaques de pirates.
- Meilleure fiabilité : Le seul maillon faible du système est le serveur mais en générale il a une grande tolérance en pannes.
- Facilité d'évolution : Une architecture client/serveur est évolutive car il est très facile de rajouter ou d'enlever des clients, et même des serveurs.

➤ Inconvénients de cette architecture

Un coût d'exploitation élevé (bande passante, câbles, ordinateurs puissants).

L'architecture client/serveur est décrite dans la figure suivante :

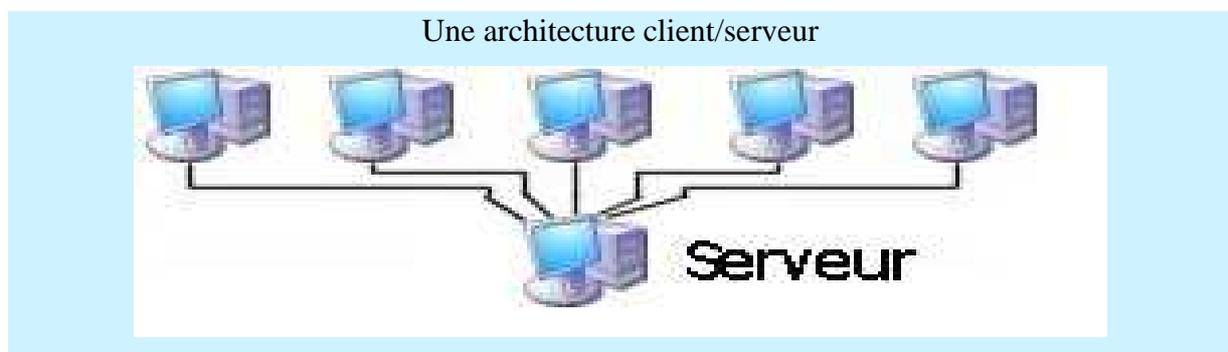


Figure I.4 : L'architecture client/serveur.

Conclusion

Dans ce chapitre nous avons présenté les différents diagrammes du langage UML. UML utilise la représentation graphique qui permet d'exprimer clairement les besoins et la structure interne statique et dynamique de future système difficilement et un bon commentaire permet d'enrichir une figure, nous avons aussi présenté l'architecture client/serveur. Cette architecture a de nombreux avantages par rapport aux autres architectures notamment la sécurité et la flexibilité. Dans le chapitre suivant nous allons spécifier les besoins de notre application.

Chapitre II

Spécification des besoins

Introduction

1- Présentation de l'organisme.

2- Bibliothèque centrale de université Mila.

3- Les objectifs du stage.

4- Identification des acteurs.

5- Les cas d'utilisations du système.

6- Description textuelle du cas d'utilisation.

7- Réalisation des diagrammes de séquences.

Conclusion

CHAPITRE II : SPECIFICATION DES BESOINS

Introduction

La spécification des besoins est l'une des étapes les plus importantes dans le développement des applications. Dans cette étape nous présentons l'organisme d'accueil, et nous exprimons les besoins attendus du futur système à développer.

1-Présentation de l'organisme d'accueil

La gestion des bibliothèques au sein de l'Université de Mila est une opération rigoureuse, qui mérite d'être perfectionnée et analysée soigneusement. Mais avant d'essayer de porter une solution informatique pour ce processus, on va dans cette partie présenter le centre universitaire de Mila et présenter la bibliothèque du centre.

1.1- Historique de l'université de Mila

Le centre universitaire de Mila est un établissement public située à 5KM du centre de la wilaya de Mila sur axe routier (route nationale n°79 Mila /Alger allant vers les villes de Zaghia et de Ferjioua.



Figure II.1 : La situation de CUM.

Le centre universitaire de Mila propose durant l'année 2010-2011 une formation supérieure dans les domaines suivants :

- Mathématiques et informatique.
- Science de la nature et de la vie.
- Science Economiques, Commerciales et Sciences de Gestion.
- Langues et Littérature Arabe.
- Anglais.

La structure administrative du centre est décrite dans la figure suivante :

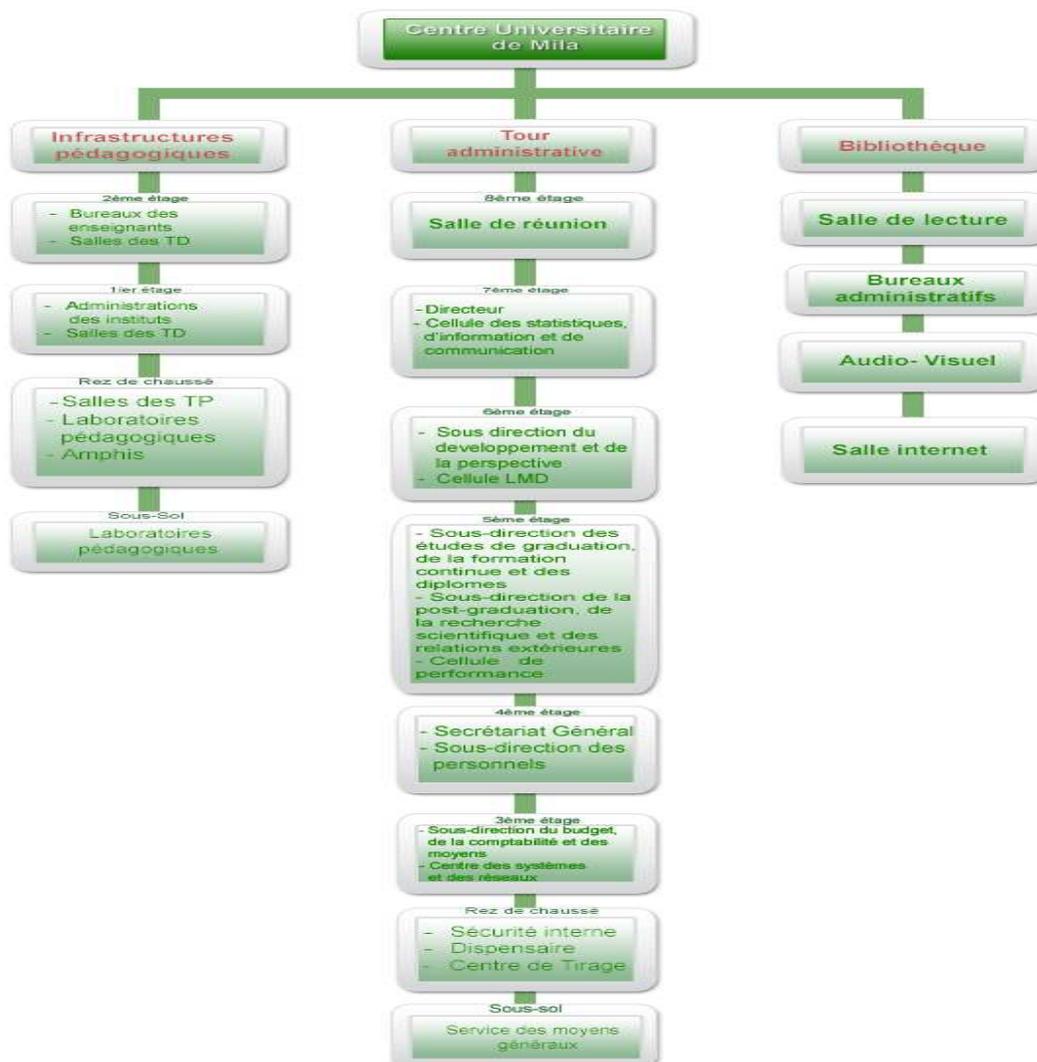


Figure II.2 : La structure administrative du centre.

2-Bibliothèque centrale de université Mila

2.1-Présentation de la bibliothèque

L'ouvert de la bibliothèque du centre coïncide avec l'ouverture officielle de l'année universitaire 2008-2009 à Mila soit le 18 Octobre 2008, sur une infrastructure d'une

superficie de 3394,40 mètres carrés et une capacité de lecteurs estimée à 500 places. Le fond documentaire comprend les diverses disciplines universitaires (sciences économiques, mathématiques et informatique, la littérature et langue arabe et la langue anglaise).

2.2- Description du fonctionnement de la bibliothèque

La bibliothèque se compose des deux sections : la section science économique et mathématique et informatique, et la section littérature et langue anglaise, Chaque adhérent de la bibliothèque possède une carte qui contient ses informations (code, nom, prénom, ...etc.). Un adhérent a le droit d'emprunter un seul exemplaire d'un ouvrage à la fois pendant une semaine. La bibliothèque a pour tâches principales:

- ❖ La gestion des ouvrages universitaires (ouvrages).
- ❖ La gestion des adhérents.
- ❖ La gestion des emprunts.
- ❖ La gestion des pénalités.
- ❖ La gestion de la recherche sur les ouvrages.

- a) La Gestion des ouvrages : elle consiste à ajouter et modifier des ouvrages ou bien ajouter, modifier et supprimer des exemplaires d'un ouvrage. il existe plusieurs types d'ouvrages on peut citer les livres, les mémoires de fin d'étude (prochainement).
- b) La Gestion des adhérents : elle consiste à ajouter, supprimer et modifier les informations relatives à un adhérent. un adhérent peut être un étudiant ou un enseignant. Un adhérent peut consulter un ouvrage sur place (c'est -à- dire dans la salle de lecture) ou l'emprunter.
- c) La Gestion des emprunts : Permet à un adhérent d'emprunter un exemplaire d'un ouvrage. Pour réaliser cette opération on suit les étapes suivant :
 - L'adhérent qui veut emprunter un exemplaire d'un ouvrage présente sa carte de la bibliothèque et indique le code de l'exemplaire qu'il veut emprunter.
 - L'agent de prêt vérifie l'existence et la disponibilité de l'exemplaire demandé avant de donner l'exemplaire à l'adhérent.

- L'agent de prêt enregistre le cod de l'exemplaire et saisie la carte de l'adhérent.

d) La Gestion de pénalité : cette gestion consiste à ajouter, supprimer, et de chercher les adhérents qui sont pénalisés. La pénalisation des adhérents peut être l'interdiction d'emprunts pendant un certain temps, ce délai dépendra de la durée de dépassement de la date de retour de l'exemplaire.

3-Les objectifs du stage

Notre objectif est de réaliser une application client/serveur pour la gestion des emprunts d'ouvrage dans la bibliothèque du centre universitaire Mila et cela pour :

- ❖ Automatiser les tâches qui sont réalisé manuellement.
- ❖ Facilite la recherche et l'accès aux informations (rechercher ouvrage, vérifier la disponibilité de livreetc.).
- ❖ Stockage des informations sur des supports informatiques ce qui assurera leur sécurité.

4-Identifier les acteurs

Dans cette étape, nous allons identifier les différents acteurs qui interagiront directement avec le système étudié, l'étude préliminaire des besoins fonctionnels a révèlè la présence de deux types d'acteurs : Le bibliothécaire et l'utilisateur simple,

On peut défini un acteur générale qui est l'utilisateur de l'application.

L'utilisateur: C'est Le bibliothécaire (l'administrateur de la bibliothèque ou l'agent de prêt) ou bien l'utilisateur simple. Chaqu'un peut interagir avec le système.

Le premier acteur qui est le bibliothécaire peut interagie après authentification et l'utilisateur simple peut interagir sans authentification.

L'administrateur peut faire toutes les tâches de la gestion et l'agent de prêt peut faire seulement la Gestion des emprunts, la Gestion de pénalité et la Recherche des ouvrages.

L'utilisateur simple de l'application peut faire juste la Recherche des ouvrages.

- L'administrateur: cet acteur peut :

- ❖ Gérer les MAJ (la Mise à Jour) des ouvrages et des exemplaires dans le catalogue de la bibliothèque:
 - Ajouter un ouvrage (livre, mémoire ...etc.)
 - Modifier les informations d'un ouvrage (livre, mémoire...etc.).
 - Ajouter un exemplaire d'un ouvrage.
 - Modifier un exemplaire d'un ouvrage (Modifier le numéro de L'exemplaire).
 - Retirer un exemplaire d'un ouvrage.
- ❖ Gérer les MAJ des adhérents :
 - Ajouter un adhérent.
 - Modifier les informations d'un adhérent.
 - Supprimer un adhérent.
- ❖ Rechercher les ouvrages.
- ❖ Gérer les MAJ de pénalité :
 - Pénaliser un adhérent.
 - Consulter la liste des adhérents pénalisés.
 - Supprimer un adhérent de la liste des adhérents pénalisés.
- L'agent de prêt : cet acteur peut :
 - ❖ Emprunter un exemplaire d'un ouvrage a un adhérent.
 - ❖ Retourner un exemplaire d'un ouvrage emprunté par un adhérent.
 - ❖ Rechercher les exemplaires d'un ouvrage et vérifier leur disponibilité.

- ❖ Gérer les MAJ de pénalité :

- Pénalisé un adhérent.
- Consulter la liste des adhérent pénalisés.
- Supprimer un adhérent de la liste des adhérents pénalisés.

- L'utilisateur simple: cet acteur peut :

- ❖ Rechercher les ouvrage(les livres, les mémoires).

5-Les cas d'utilisations du système

Les principales fonctionnalités de l'application à concevoir sont érigées autour des besoins et exigences des différents acteurs. Elles sont illustrées dans le diagramme de cas d'utilisation on distingue les cas d'utilisations suivant :

- **Authentification** : Permet à un acteur de s'authentifier avant d'accéder à l'application.
- **Gestion d'ouvrage** : Permet à l'administrateur d'ajouter un nouvel ouvrage ou de modifier les renseignements d'un ouvrage et de faire une mise à jour sur Les exemplaires des ouvrages (ajouter, supprimer ou modifier).
- **Gestion des adhérents**: Permet à l'administrateur d'ajouter un nouvel adhérent ou de modifier les renseignements d'un adhérent ou de le supprimer.
- **Gestion des emprunts** : Permet au bibliothécaire (l'administrateur ou l'agent de prêt) d'emprunter ou de retourner un exemplaire d'un ouvrage à un adhérent.
- **Rechercher ouvrage** : Permet à l'utilisateur (l'administrateur ou l'agent de prêt ou bien l'utilisateur simple) de rechercher avec un critère de recherche (par titre, par code, par auteur) n'importe quel ouvrage dans le catalogue des ouvrages et d'afficher tout les informations nécessaires de cet ouvrage.
- **Gestion de pénalité** : Permet au bibliothécaire (l'administrateur ou l'agent de prêt) de pénalisé un adhérent et de vérifier les adhérentes pénalités.

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

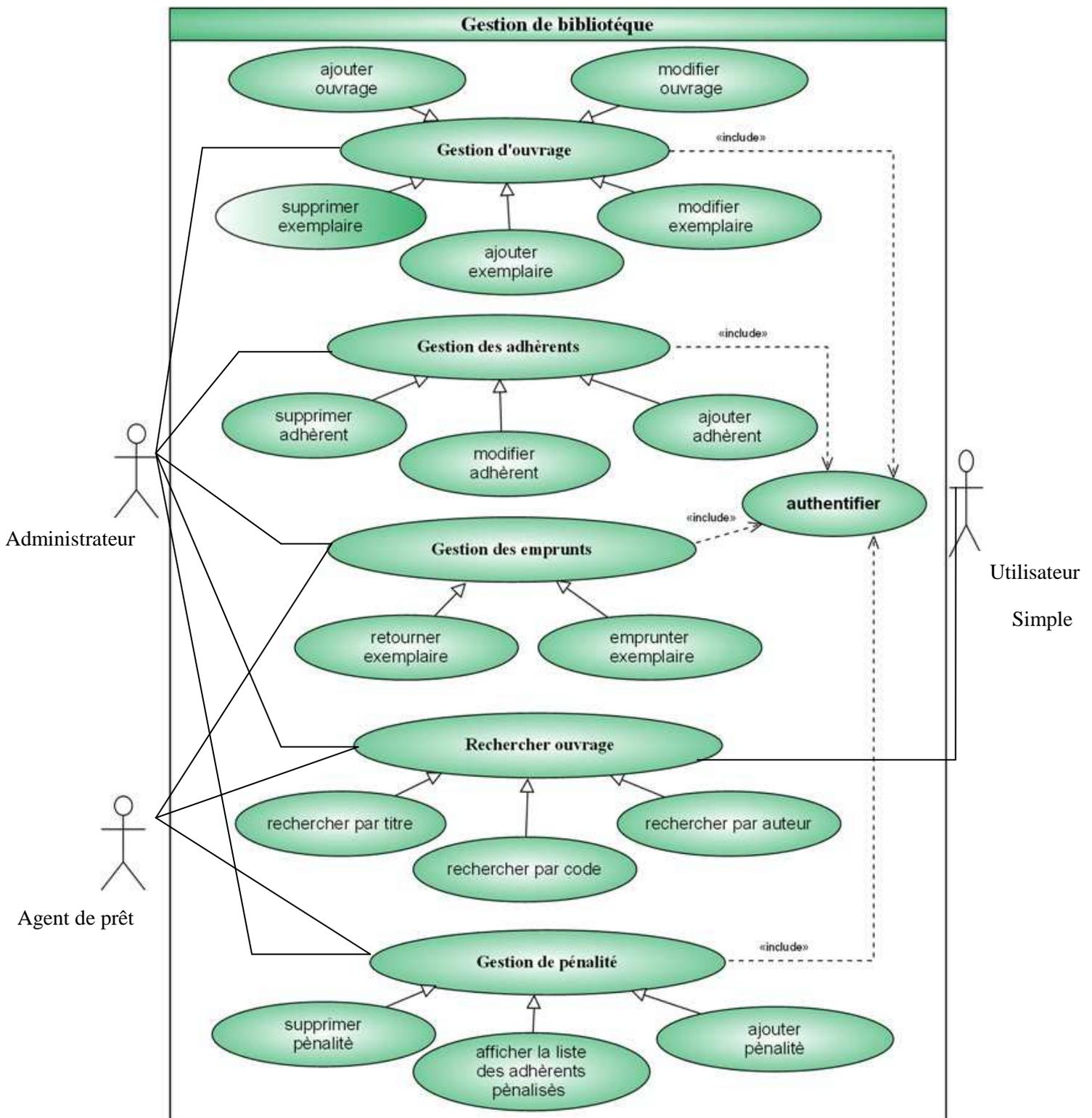


Figure II.3 : Diagramme de cas d'utilisation.

6-Description textuelle du cas d'utilisation

S'authentification
<p><u>Sommaire d'identification :</u></p> <ul style="list-style-type: none">◆ Titre : s'authentifier.◆ Bute : authentifier un utilisateur.◆ Acteur : l'administrateur, l'agent de prêt. <p><u>Description de l'enchaînement :</u></p> <ul style="list-style-type: none">❖ Enchaînement : ce cas d'utilisation commence lorsque le bibliothécaire (l'administrateur ou l'agent de prêt) veut accéder au système.<ul style="list-style-type: none">- Le système affiche un formulaire qui contient deux champs pour saisir le nom d'utilisateur et le mot de passe- l'administrateur ou l'agent de prêt saisir les informations demander.- Le système effectuer une vérification.<ul style="list-style-type: none">[Si erreur] le système afficher un message d'erreur.[Si non] le système affiche le menu principal du système. <p><u>Post condition :</u></p> <ul style="list-style-type: none">- fin normal : L'administrateur ou l'agent de prêt est authentifié.-annulation : l'opération est annulée

Tableau II.1 : Fiche description textuelle du cas authentification.

Gestion d'ouvrage

Sommaire d'identification :

- ◆ **Titre** : Gestion d'ouvrage.
- ◆ **Bute** : faire l'ajout et la modification d'un ouvrage, et aussi permet de faire l'ajout, la modification et la suppression d'un exemplaire de chaque ouvrage.
- ◆ **Acteur** : l'administrateur.

Description de l'enchaînement :

- ❖ Pré condition : l'administrateur est authentifié.
- ❖ Enchaînement : Ce cas d'utilisation commence lorsque l'administrateur clique sur le bouton Gestion d'ouvrage.
 - l'administrateur demande Gestion d'ouvrage.

Cas ajouter ouvrage

- Le système affiche un formulaire qui contient une liste de 5 opérations et demande à l'administrateur de choisir l'opération qu'il veut exécuter.
- l'administrateur choisit « ajouter ouvrage » et saisie le code de l'ouvrage.
- Si le code de l'ouvrage n'existe pas alors le système affiche un autre formulaire pour saisir les autres informations de l'ouvrage.
- l'administrateur saisie tout les informations demander par le système et demande leurs sauvegarde.
- Le système vérifie la cohérence des informations saisies.

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

[Sinon] le système enregistre les informations de l'ouvrage dans la base et affiche un message de succès.

- Ce cas se terminer lorsque l'ajout est terminé avec succès ou bien annulé.

Cas modifier ouvrage

- Le système affiche un formulaire qui contient une liste de 5 opérations et demande à l'administrateur de choisir l'opération qu'il veut exécuter.
- l'administrateur choisit « modifier ouvrage » et saisie le code de l'ouvrage.
- Si le code de l'ouvrage existe alors le système affiche un autre formulaire qui contient toutes les informations de l'ouvrage.
- l'administrateur effectuer la modification sur ses informations et demande leur sauvegarde.
- Le système vérifie la cohérence des informations modifié.

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

[Sinon] le système enregistre les informations de l'ouvrage qui est modifié dans la base et affiche un message de succès.

- Ce cas se termine lorsque la modification est terminée avec succès ou bien annulée.

Cas ajouter exemplaire

- Le système affiche un formulaire qui contient une liste de 5 opérations et demande à l'administrateur de choisir l'opération qu'il veut exécuter.
- L'administrateur choisit « ajouter exemplaire » et saisit le code de l'ouvrage.
- Si le code de l'ouvrage existe alors le système affiche un autre formulaire pour saisir les autres informations de l'exemplaire.
- L'administrateur saisit toutes les informations demandées par le système et demande leur sauvegarde.
- Le système vérifie la cohérence des informations saisies.

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

[Sinon] le système enregistre les informations de l'exemplaire dans la base et affiche un message de succès.

- Ce cas se termine lorsque l'ajout est terminé avec succès ou bien annulé.

Cas modifier exemplaire

- Le système affiche un formulaire qui contient une liste de 5 opérations et demande à l'administrateur de choisir l'opération qu'il veut exécuter.
- L'administrateur choisit « modifier exemplaire » et saisit le code de l'ouvrage.
- Si le code de l'ouvrage existe alors le système affiche un autre formulaire pour saisir les informations de l'exemplaire.
- L'administrateur modifie les informations de l'exemplaire et demande leur sauvegarde.
- Le système vérifie la cohérence des informations modifiées.

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

[Sinon] le système enregistre les informations de l'exemplaire modifié dans la base et affiche un message de succès.

- Ce cas se termine lorsque la modification est terminée avec succès ou bien annulée.

Cas supprimer exemplaire

- Le système affiche un formulaire qui contient une liste de 5 opérations et demande à l'administrateur de choisir l'opération qu'il veut exécuter.
- L'administrateur choisit « supprimer un exemplaire » et saisie le code de l'ouvrage.
- Si le code de l'ouvrage existe alors le système affiche un autre formulaire pour saisir le numéro de l'exemplaire supprimé.
- L'administrateur saisie le numéro exemplaire et valide la suppression.
- Le système vérifie l'existence de l'exemplaire.

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

[Sinon] le système supprime l'exemplaire de la base et affiche un message de succès.

- Ce cas se termine lorsque la suppression est terminée avec succès ou bien annulée.

Post condition :

- Fin normal : un ouvrage est ajouté ou modifié ou bien un exemplaire d'un ouvrage est ajouté, modifié ou bien supprimé.
- Annulation : l'opération effectuée est annulée.

Tableau II.2 : Fiche description textuelle du cas Gestion d'ouvrage.

Gestion des adhérents

Sommaire d'identification :

- ◆ **Titre** : Gestion des adhérents.
- ◆ **Bute** : faire l'ajout, la modification et la suppression des adhérents.
- ◆ **Acteur** : l'administrateur.

Description de l'enchaînement :

- ❖ Pré condition : l'administrateur est authentifié.
- ❖ Enchaînement : ce cas d'utilisation commence lorsque l'administrateur clique sur le bouton Gestion des adhérents.
 - L'administrateur demande Gestion des adhérents.

Cas ajouter adhérent

- Le système affiche un formulaire contient une liste de 3 opérations et demande à l'administrateur de choisir l'opération qu'il veut exécuter.
- l'administrateur choisit « ajouter adhérent » et saisie le code de l'adhérent.
- Si le code de l'adhérent n'existe pas alors le système affiche un autre formulaire pour saisie les informations de l'adhérent.
- l'administrateur saisie les informations de l'adhérent et demande leurs sauvegarde.
- Le système vérifie la cohérence des informations saisies.

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

[Sinon] le système enregistre les informations de l'adhérent dans la base et affiche un message de succès.

- Ce cas se terminer lorsque l'ajout est terminé avec succès ou bien annulé.

Cas modifier adhérent

- Le système affiche un formulaire qui contient une liste de 3 opérations et demande à l'administrateur de choisir l'opération qu'il veut exécuter.
- l'administrateur choisit « modifier adhérent » et saisie le code de l'adhérent.
- Si le code de l'adhérent existe alors le système affiche un autre formulaire qui contient toute les informations de l'adhérent.
- l'administrateur effectuer la modification sur ses informations et demande leur sauvegarde.

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

[Sinon] le système enregistre les informations de l'adhérent qui sont modifie dans la

- Le système vérifie la cohérence des informations modifiées.

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

[Sinon] le système enregistre les informations de l'adhérent qui sont modifiées dans la base et affiche un message de succès.

- Ce cas se termine lorsque la modification est terminée avec succès ou bien annulée.

Cas supprimer adhérent

- Le système affiche un formulaire qui contient une liste de 3 opérations et demande à l'administrateur de choisir l'opération qu'il veut exécuter.
- L'administrateur choisit « supprimer adhérent » et saisit le code de l'adhérent.
- Si le code de l'adhérent existe alors le système affiche un autre formulaire qui contient toutes les informations de l'adhérent.
- L'administrateur valide la suppression.

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

[Sinon] le système supprime l'adhérent de la base et affiche un message de succès.

- Ce cas se termine lorsque la suppression est terminée avec succès ou bien annulée.

Post condition :

- Fin normal : un adhérent est ajouté ou modifié ou bien supprimé.
- Annulation : l'opération effectuée est annulée.

Tableau II.3: Fiche description textuelle du cas Gestion des adhérents.

Gestion des emprunts

Sommaire d'identification :

- ◆ **Titre** : Gestion des emprunts.
- ◆ **Bute** : faire l'emprunt et le retour des exemplaires des ouvrages.
- ◆ **Acteur** : Le bibliothécaire (l'administrateur ou l'agent de prêt).

Description de l'enchaînement :

- ❖ Pré condition : Le bibliothécaire (l'administrateur ou l'agent de prêt) est authentifié.
- ❖ Enchaînement : ce cas d'utilisation commence lorsque Le bibliothécaire (l'administrateur ou l'agent de prêt) clique sur le bouton Gestion des emprunts.
 - Le bibliothécaire (l'administrateur ou l'agent de prêt) demande la Gestion des emprunts.

Cas emprunter un exemplaire d'un ouvrage

- Le système affiche un formulaire qui contient une liste de 2 opérations et demande au bibliothécaire (l'administrateur ou l'agent de prêt) de choisir l'opération qu'il veut exécuter.
- Le bibliothécaire (l'administrateur ou l'agent de prêt) choisit « emprunter un exemplaire » et saisie les informations demandé par le système (code ouvrage, n_exemplaire, cod adhérent).
- Le bibliothécaire (l'administrateur ou l'agent de prêt) valide l'emprunt.
- Le système vérifie la disponibilité de l'exemplaire et le droit de l'adhérent.

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

[Sinon] le système enregistre l'emprunt de l'exemplaire dans la base et affiche un message de succès.

- Ce cas se terminer lorsque l'emprunt est terminé avec succès ou bien annulé.

Cas retourner un exemplaire d'un ouvrage

- Le système affiche un formulaire qui contient une liste de 2 opérations et demande au bibliothécaire (l'administrateur ou l'agent de prêt) de choisir l'opération qu'il veut exécuter.

- Le bibliothécaire (l'administrateur ou l'agent de prêt) choisit « retourner un exemplaire » et saisie les informations demandé par le système.
- Le bibliothécaire (l'administrateur ou l'agent de prêt) valide le retour.
- Le système vérifie la cohérence des informations saisies (l'exemplaire a été déjà emprunté).

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

[Sinon] le système enregistre le retour de l'exemplaire dans la base et affiche un message de succès.

- Ce cas se terminer lorsque le retour est terminé avec succès ou bien annulé.

Post condition :

- Fin normal : un nouvel exemplaire d'un ouvrage est emprunté ou bien retourné.
- Annulation : l'opération effectuer est annulée.

Tableau II.4: Fiche description textuelle du cas Gestion des emprunts.

Rechercher un ouvrage
<p><u>Sommaire d'identification :</u></p> <ul style="list-style-type: none">◆ Titre : Rechercher ouvrage.◆ Bute : faire la recherche des ouvrages.◆ Acteur : L'utilisateur (l'administrateur, l'agent de prêt ou l'utilisateur simple). <p><u>Description de l'enchaînement :</u></p> <ul style="list-style-type: none">❖ Enchaînement : ce cas d'utilisation commence lorsque L'utilisateur (l'administrateur, l'agent de prêt ou l'utilisateur simple) clique sur le bouton Rechercher ouvrage.<ul style="list-style-type: none">- L'utilisateur (l'administrateur, l'agent de prêt ou l'utilisateur simple) demande rechercher ouvrage.

- Le système affiche un formulaire qui contient une liste de 3opérations et demander au bibliothécaire (l'administrateur ou l'agent de prêt) de choisir l'opération qu'il veut exécuter.
- Le bibliothécaire (l'administrateur ou l'agent de prêt) choisit « ajouter pénalité » et saisie les informations demander par le système (cod adhérent, cause, date fin pénalité).
- Le bibliothécaire (l'administrateur ou l'agent de prêt) demande la sauvegarde des informations.
- Le système vérifie la cohérence des informations saisies.

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

[Sinon] le système ajoute l'adhérent dans la liste des adhérents pénalisés dans la base et affiche un message de succès.

- Ce cas se terminer lorsque l'ajout est terminé avec succès ou bien annulé.

Cas supprimer pénalité

- Le système affiche un formulaire qui contient une liste de 3opérations et demande au bibliothécaire (l'administrateur ou l'agent de prêt) de choisir l'opération qu'il veut exécuter.
- Le bibliothécaire (l'administrateur ou l'agent de prêt) choisit « supprimer pénalité d'un adhérent » et saisie le code de l'adhérent.
- Si le code de l'adhérent est existe alors le système affiche dans le formulaire la cause de la pénalité.
- Le bibliothécaire (l'administrateur ou l'agent de prêt) valide la suppression.

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

[Sinon] le système supprimé la pénalité de l'adhérent de la base et affiche un message de succès.

- Ce cas se terminer lorsque la suppression est terminé avec succès ou bien annulé.

Cas afficher liste des adhérents pénalisés

- Le système affiche un formulaire qui contient une liste de 3 opérations et demande au bibliothécaire (l'administrateur ou l'agent de prêt) de choisir l'opération qu'il veut exécuter.
- Le bibliothécaire (l'administrateur ou l'agent de prêt) choisit « afficher la liste des adhérentes pénalités ».
- Ce cas se termine lorsque l'affichage est terminé avec succès ou bien annulé.

Post condition :

- Fin normal : un adhérent est pénalisé ou supprimé de la liste des Adhérents pénalisés ou bien affiché avec la liste des adhérents Pénalisés.
- Annulation : l'opération est annulée.

Tableau II.6 : Fiche description textuelle du cas Gestion de pénalité.

7-Réalisation des diagrammes de séquences

Ce diagramme permet de représenter les interactions et les différents scénarios entre les objets c'est – à dire qu'on y met l'accent sur la chronologie des échanges de message.

7.1- diagrammes de séquences du cas d'utilisation authentifié

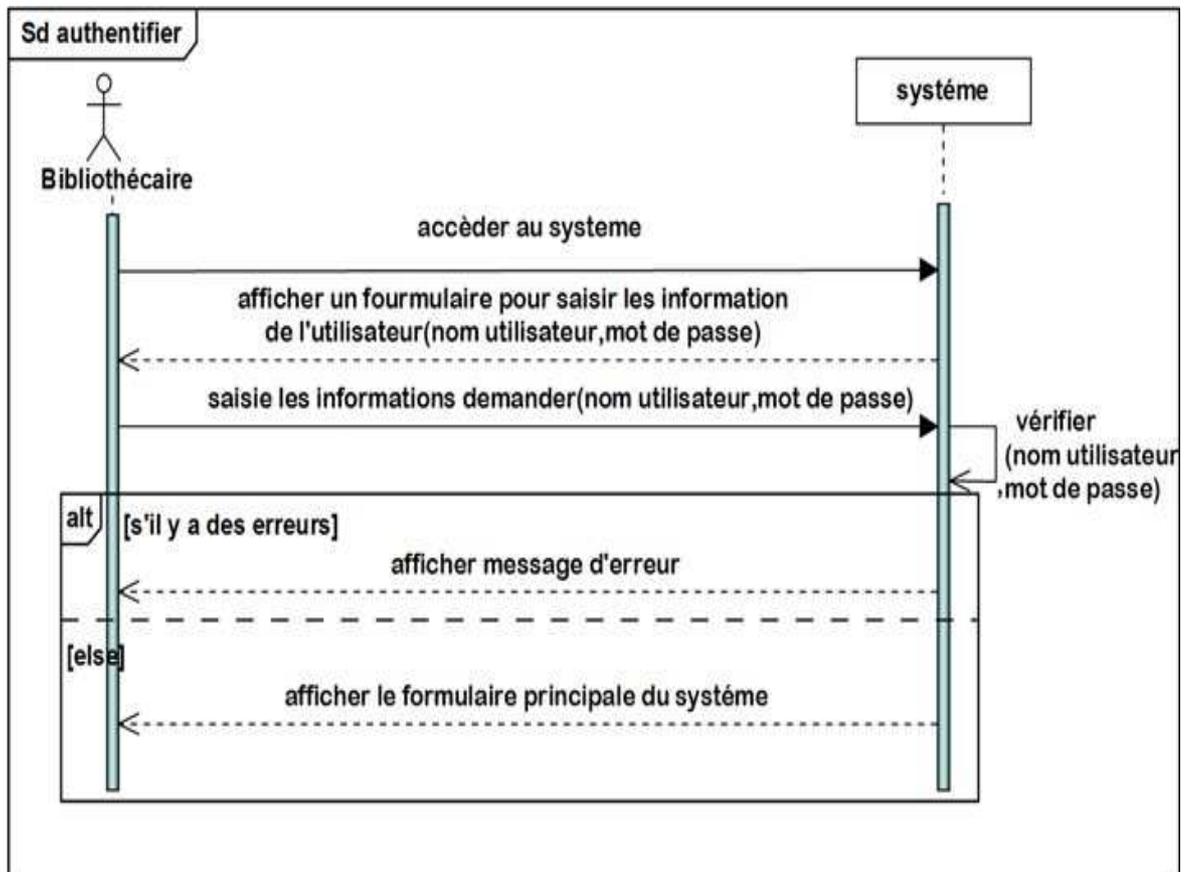


Figure II.4 : Diagramme de séquence du cas authentifié.

7.2- diagrammes de séquences du cas d'utilisation Gestion d'ouvrage

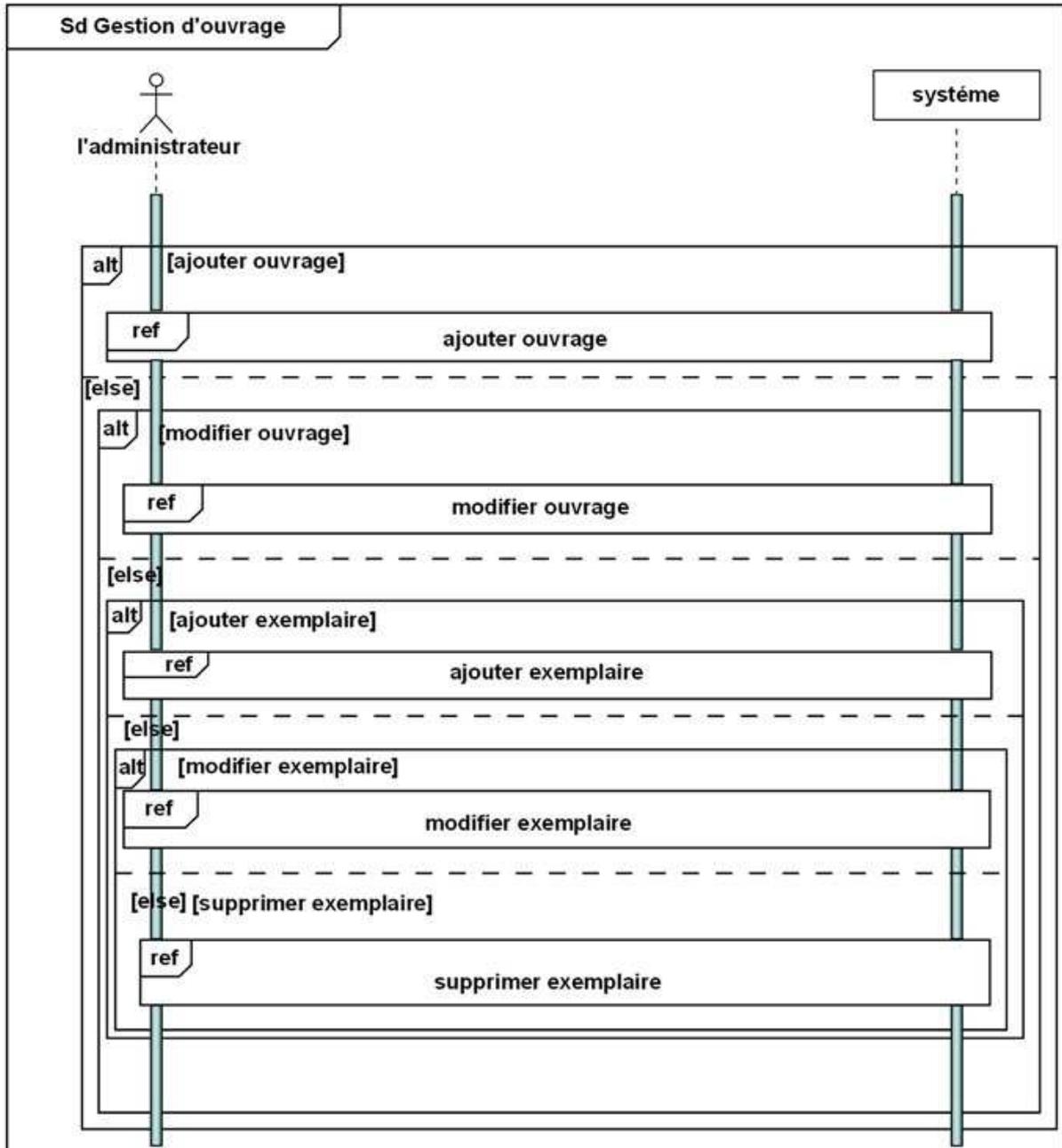


Figure II.5: Diagramme de séquence du cas Gestion d'ouvrage.

7.2.1- diagrammes de séquences du cas d'utilisation ajouter ouvrage

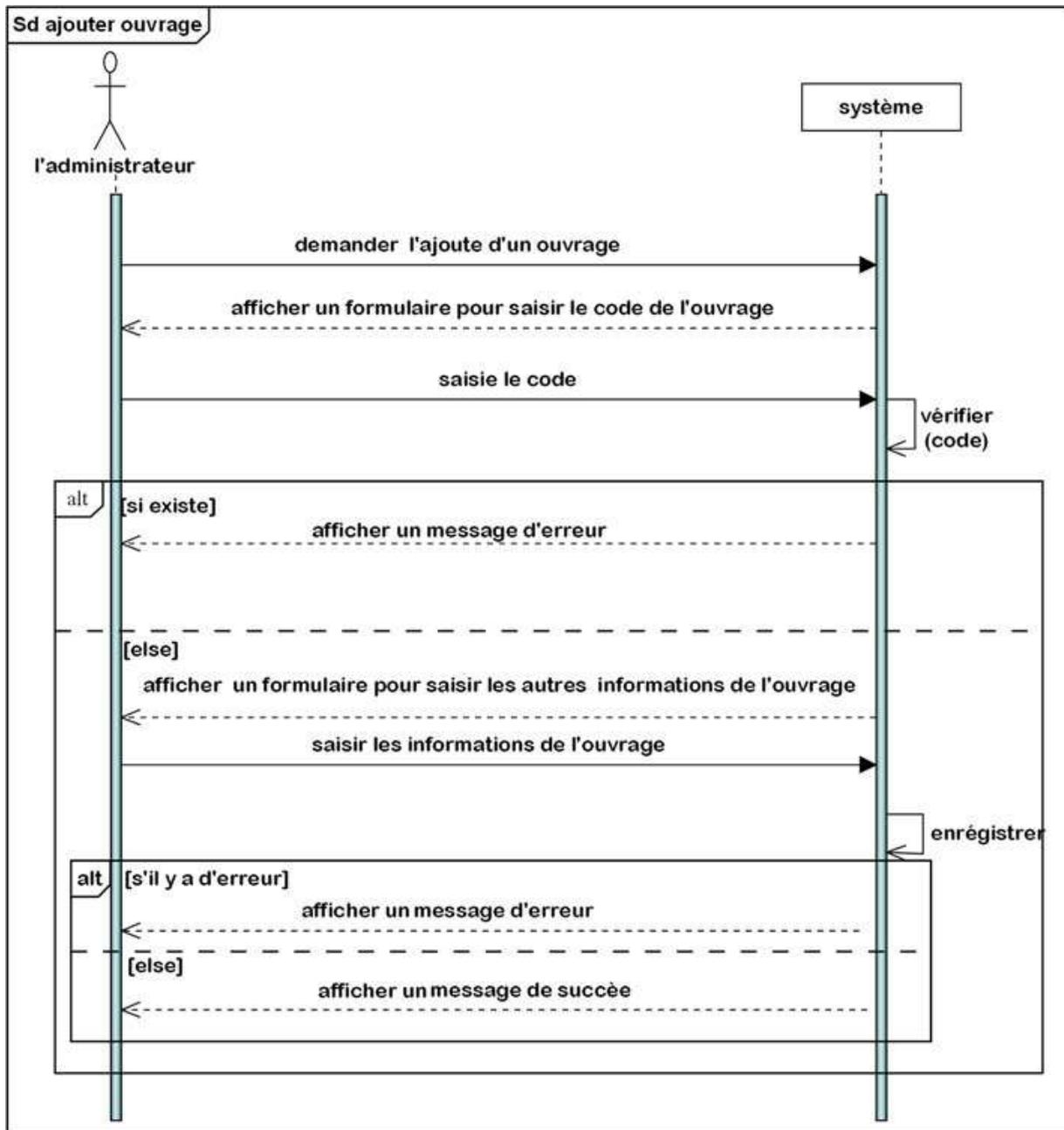


Figure II.6 : Diagramme de séquence du cas ajouter ouvrage.

7.2.2- diagrammes de séquences du cas d'utilisation modifier ouvrage

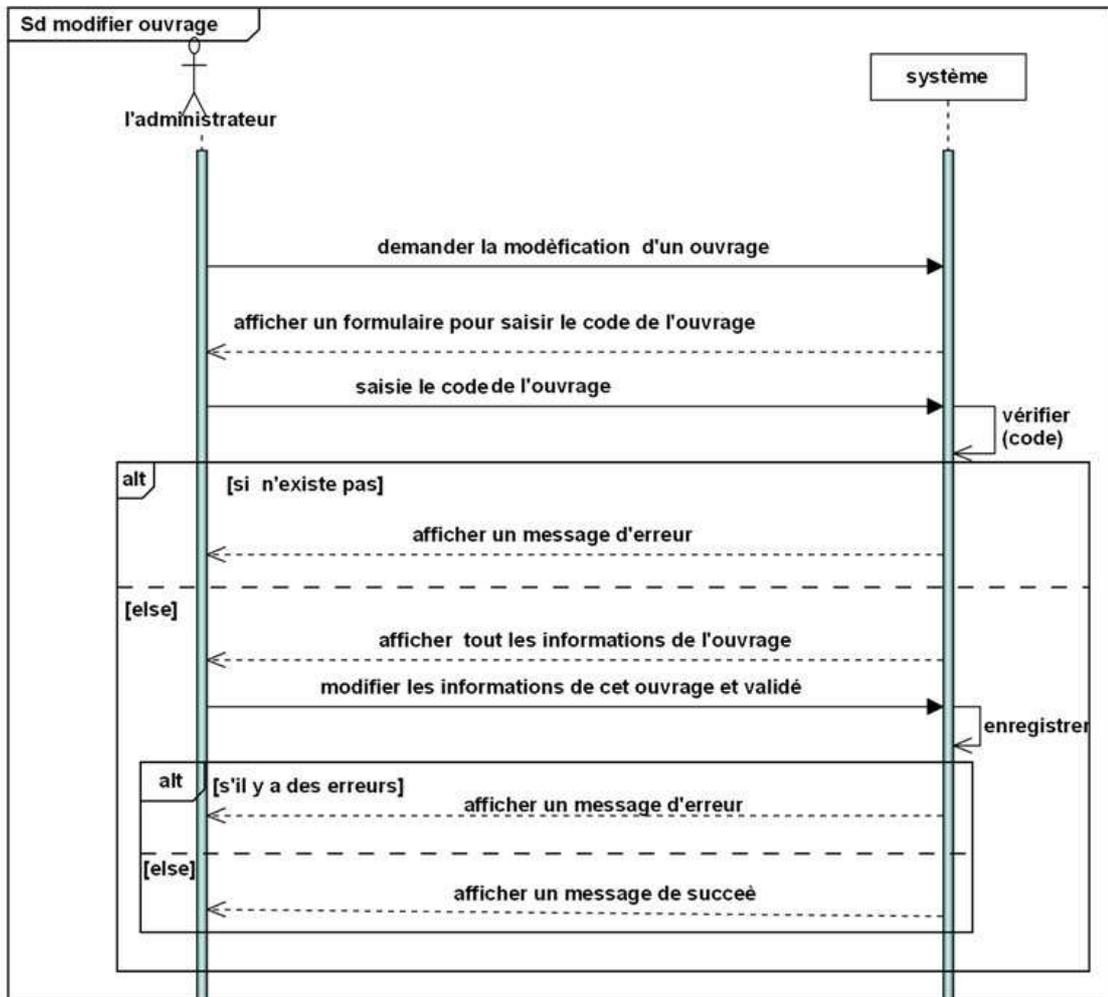


Figure II.7 : Diagramme de séquence du cas modifier ouvrage.

7.2.3- diagrammes de séquences du cas d'utilisation ajouter un exemplaire

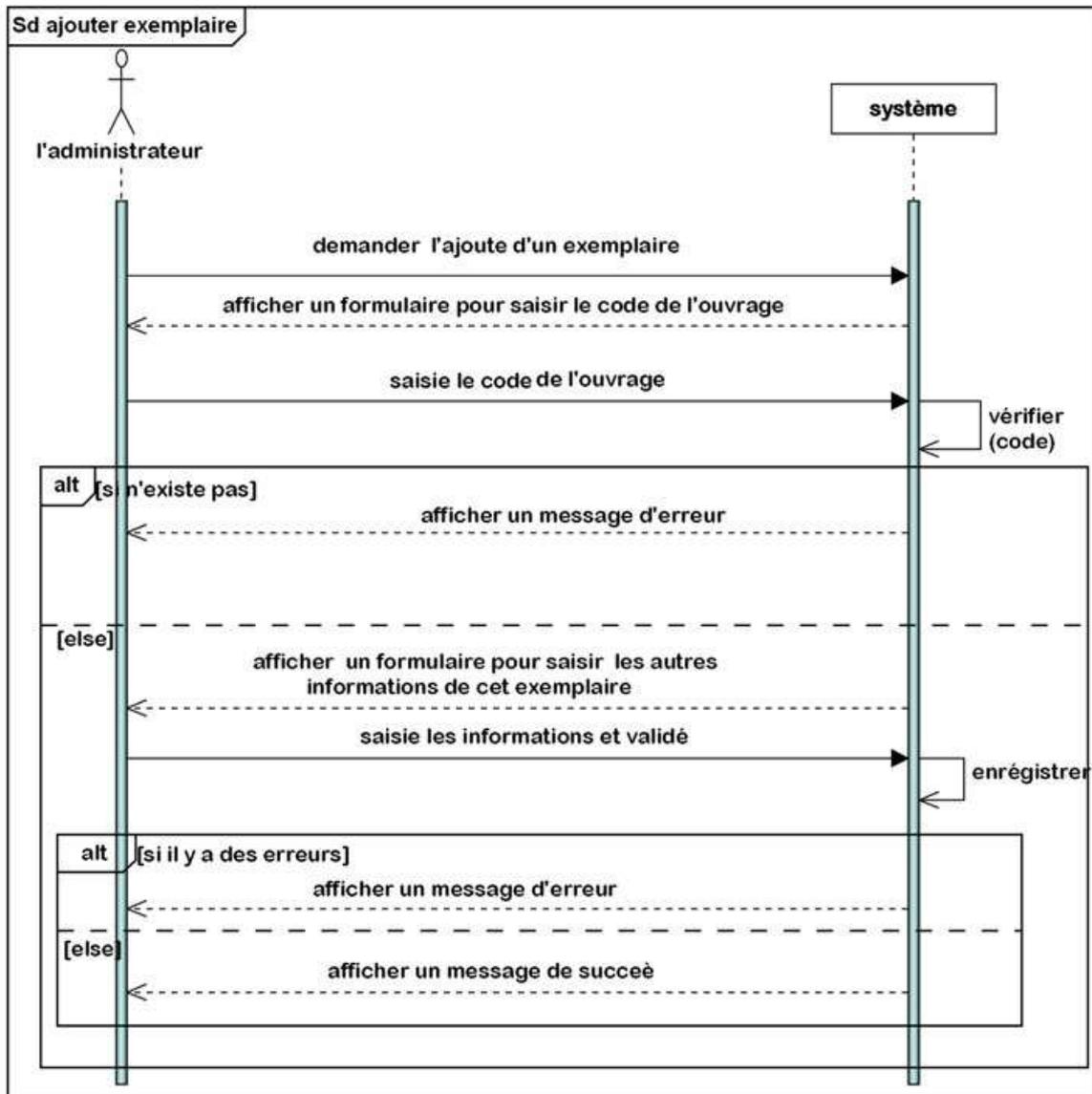


Figure II.8 : Diagramme de séquence du cas ajouté exemplaire.

7.2.4- diagrammes de séquences du cas d'utilisation modifier exemplaire

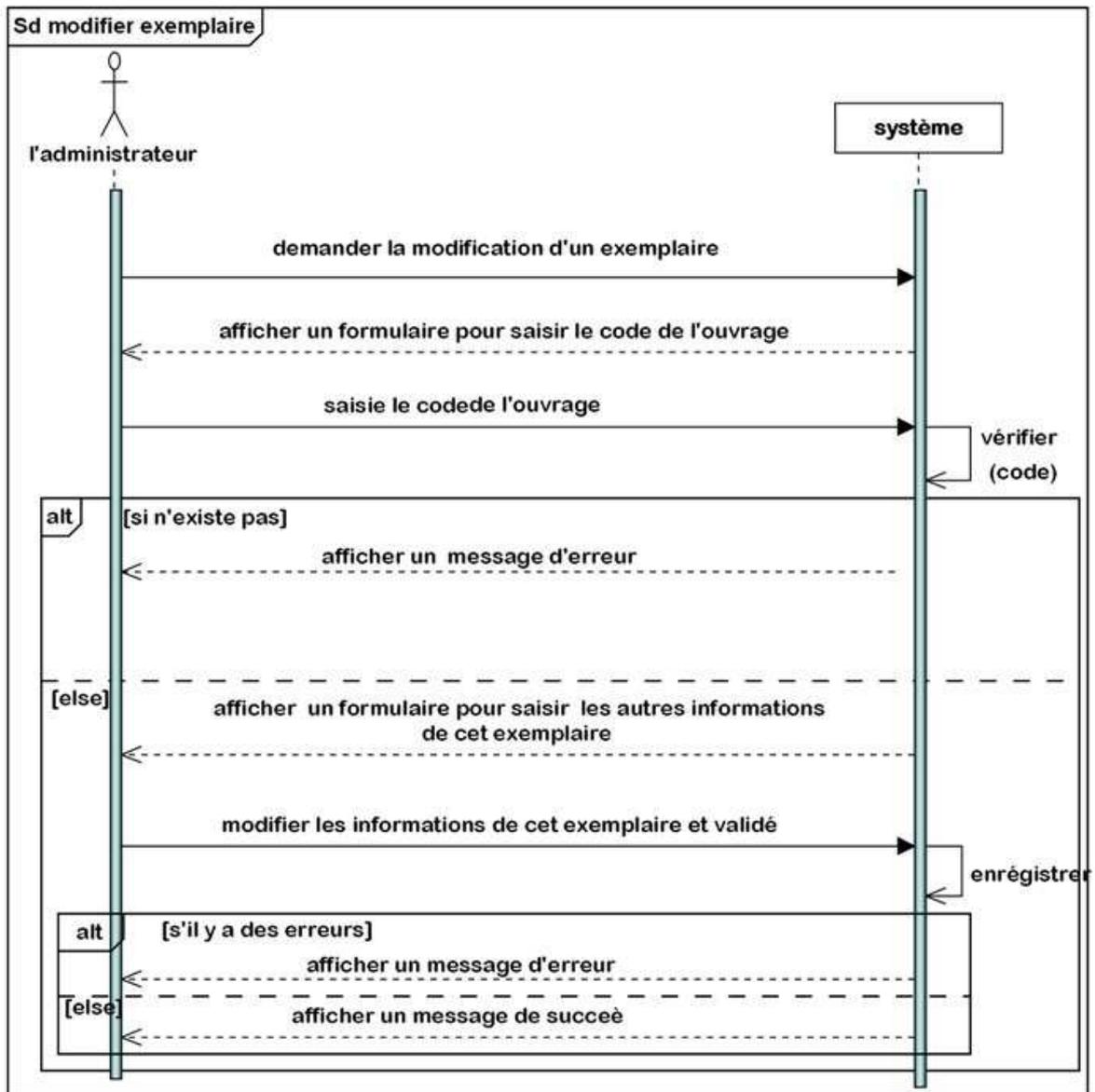


Figure II.9 : Diagramme de séquence du cas modifié exemplaire.

7.2.5- diagrammes de séquences du cas d'utilisation supprimer un exemplaire

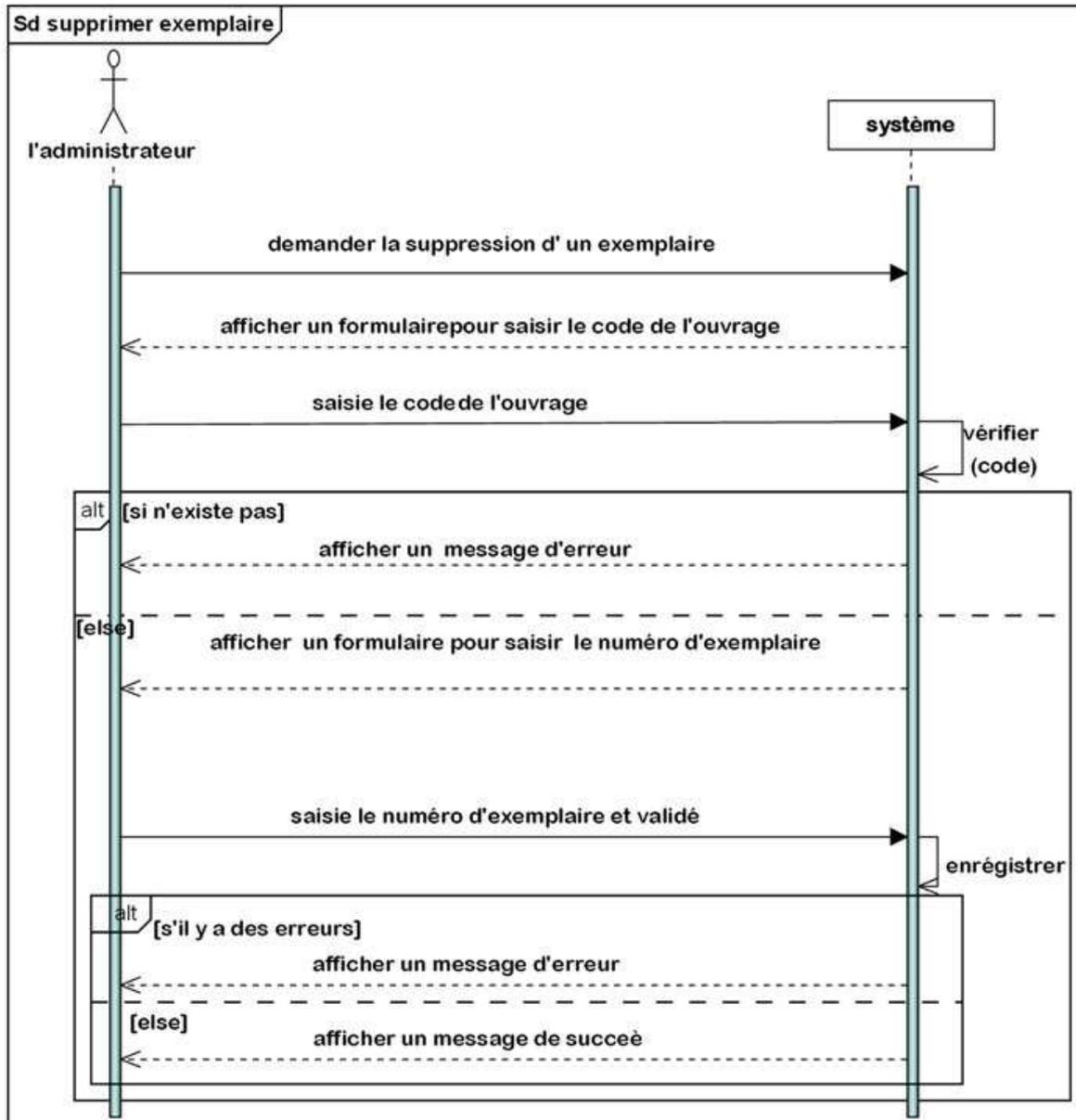


Figure II.10 : Diagramme de séquence du cas supprimé exemplaire.

7.3- diagrammes de séquences du cas d'utilisation Gestion des adhérents

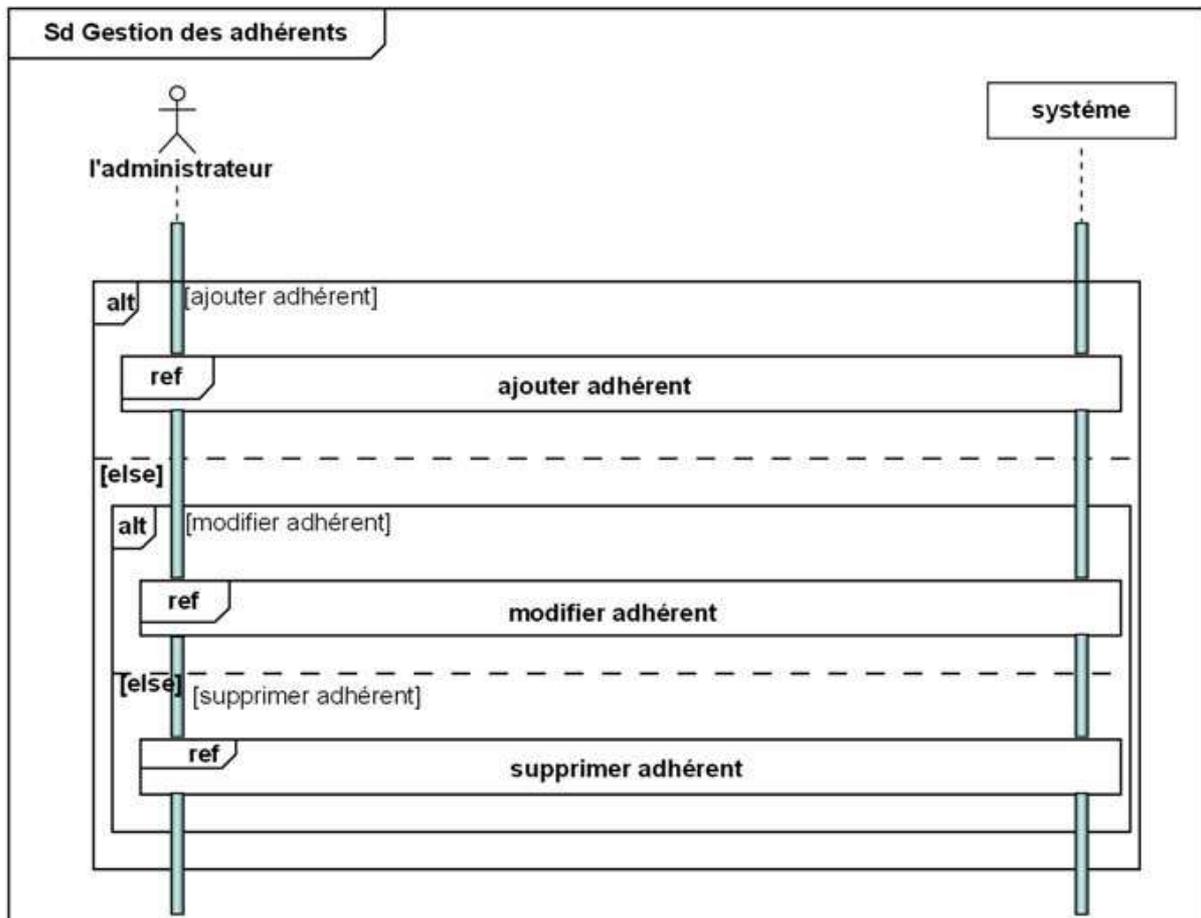


Figure II.11 : Diagramme de séquence du cas Gestion des adhérents.

7.3.1- diagrammes de séquences du cas d'utilisation ajouter un adhérent

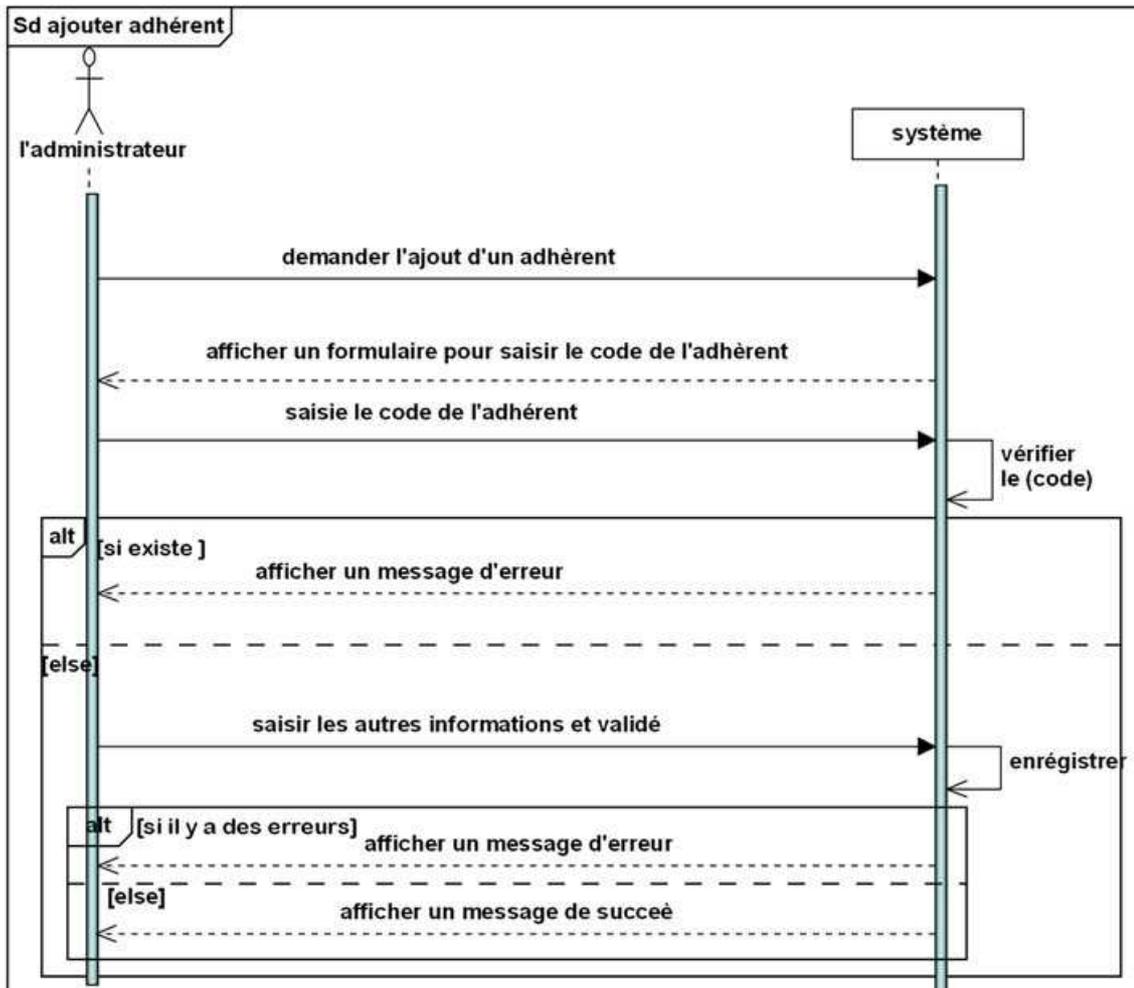


Figure II.12 : Diagramme de séquence du cas ajouter un adhérent.

7.3.2- diagrammes de séquences du cas d'utilisation modifier un adhérent

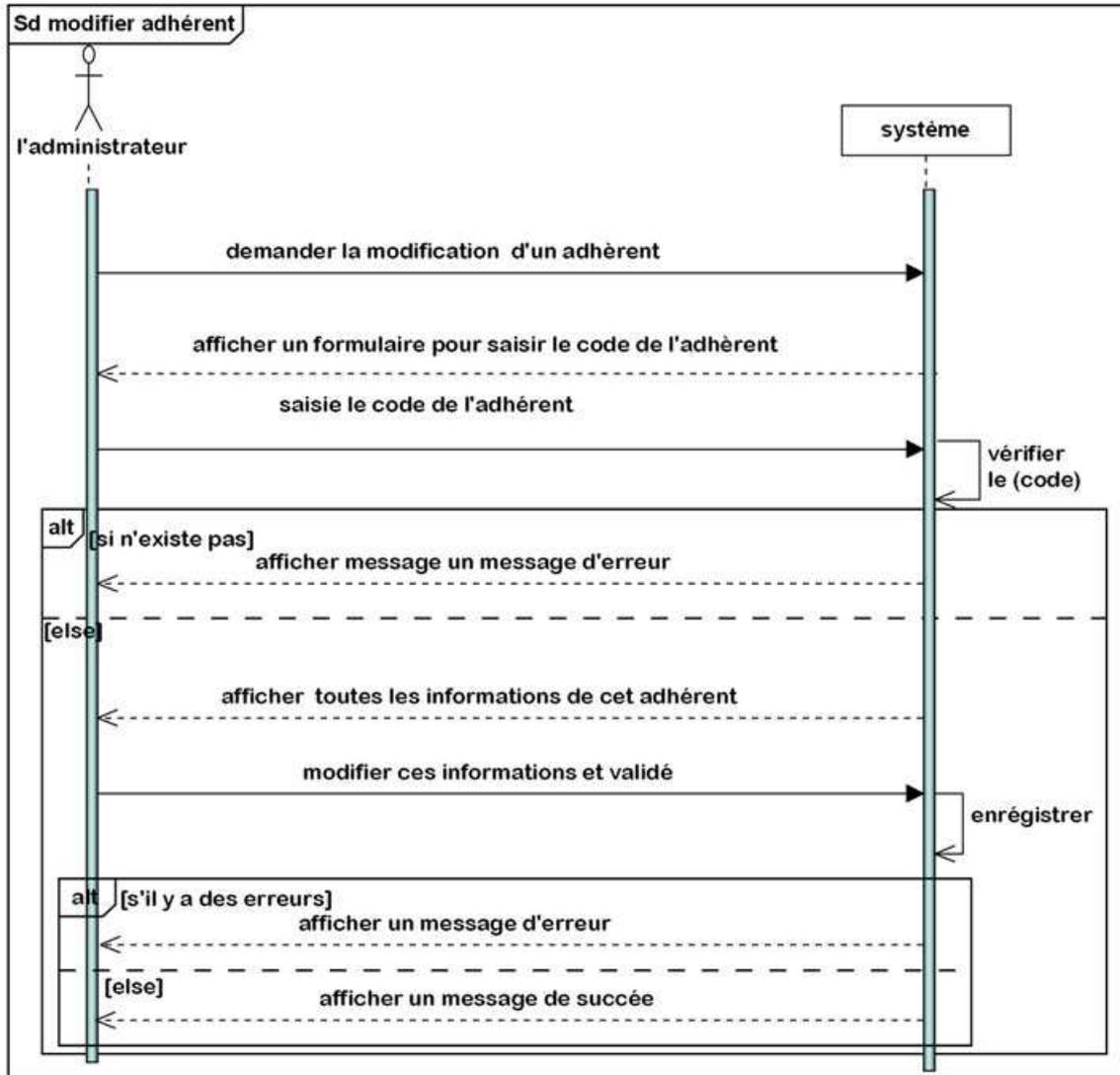


Figure II.13: Diagramme de séquence du cas modifié adhérent.

7.3.3- diagrammes de séquences du cas d'utilisation supprimer un adhérent

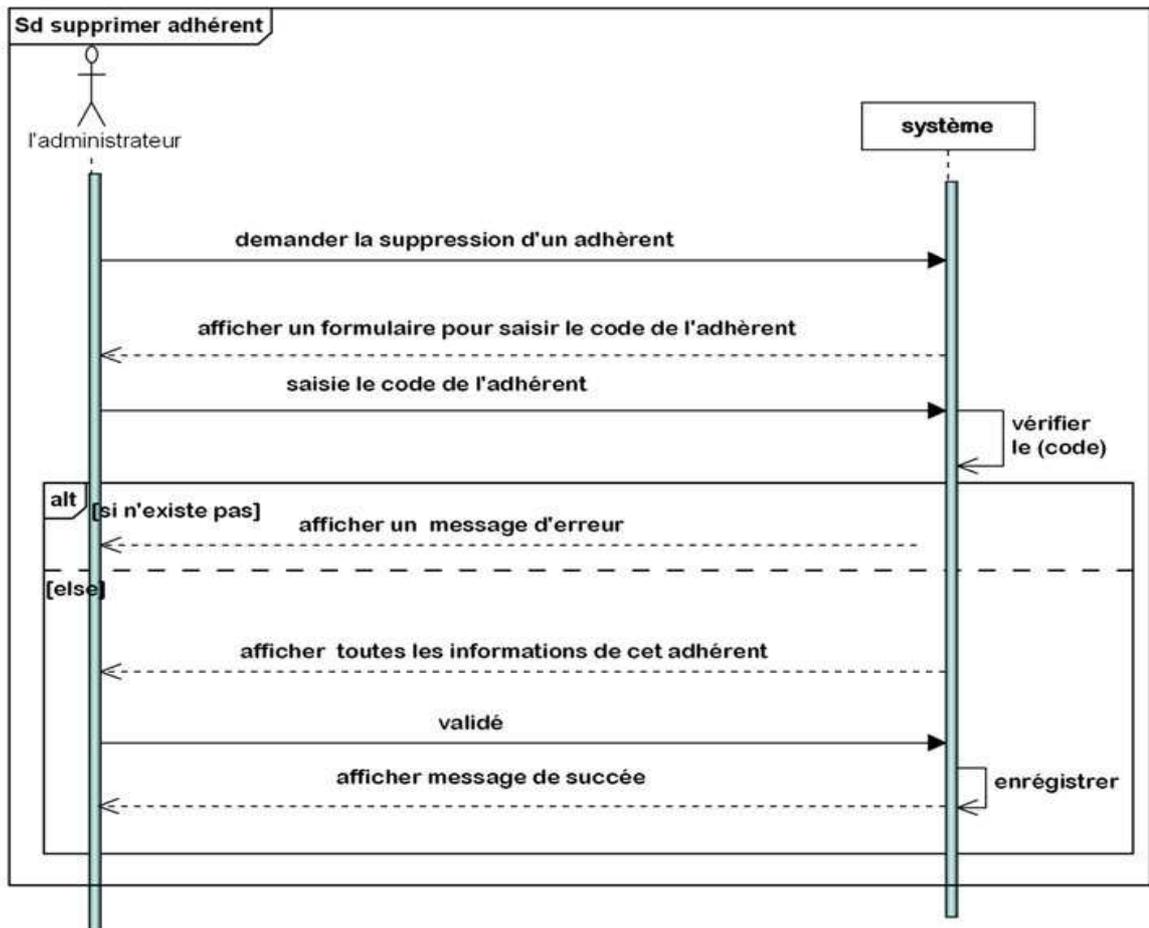


Figure II.14 : Diagramme de séquence du cas supprimé adhérent.

7.4- diagrammes de séquences du cas d'utilisation Gestion des emprunts

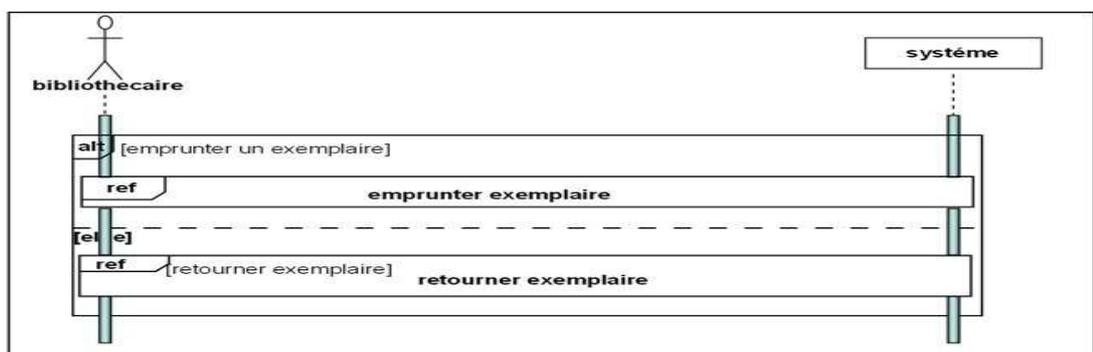


Figure II.15 : Diagramme de séquence du cas Gestion des emprunts.

7.4.1- diagrammes de séquences du cas d'utilisation emprunter un exemplaire d'un ouvrage

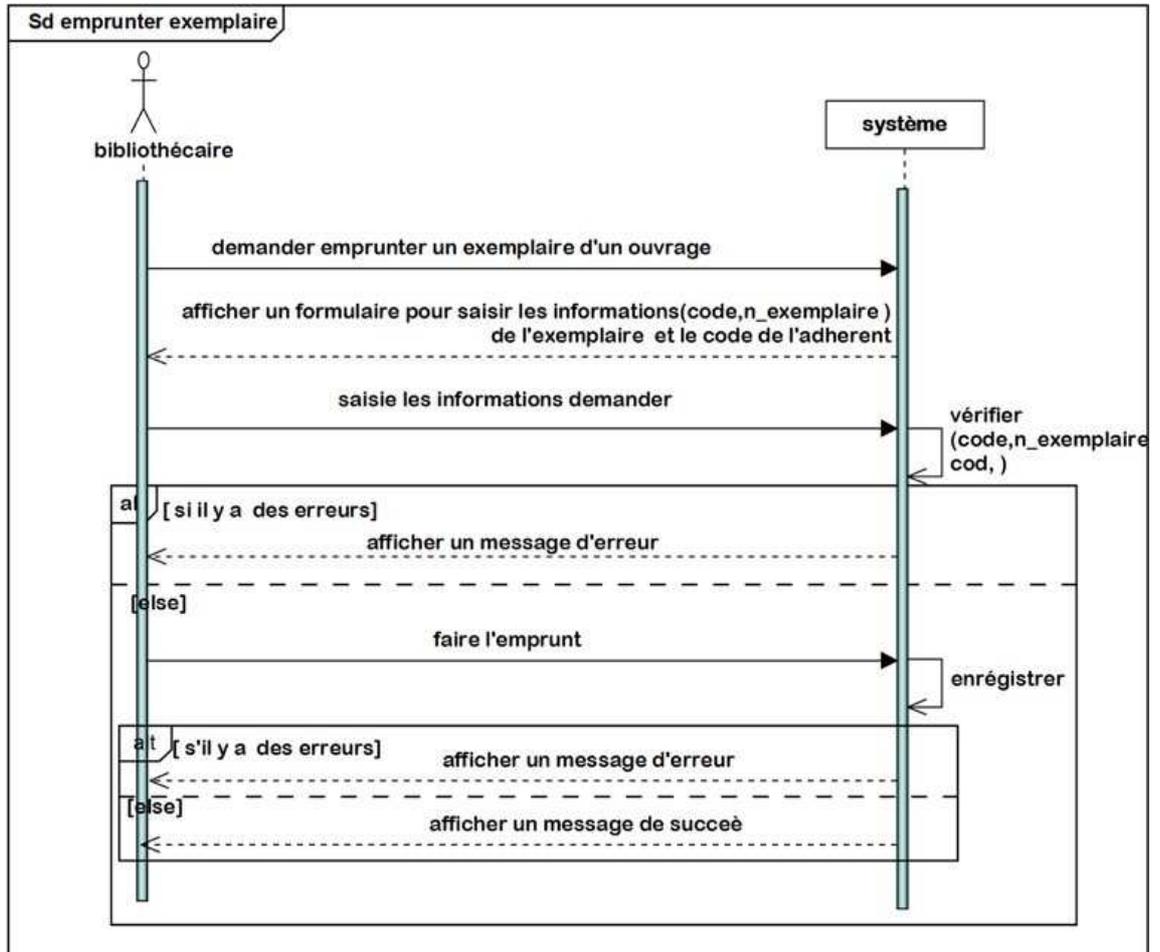


Figure II.16 : Diagramme de séquence du cas emprunté un exemplaire d'un ouvrage.

7.4.2- diagrammes de séquences du cas d'utilisation retourner un exemplaire d'un ouvrage

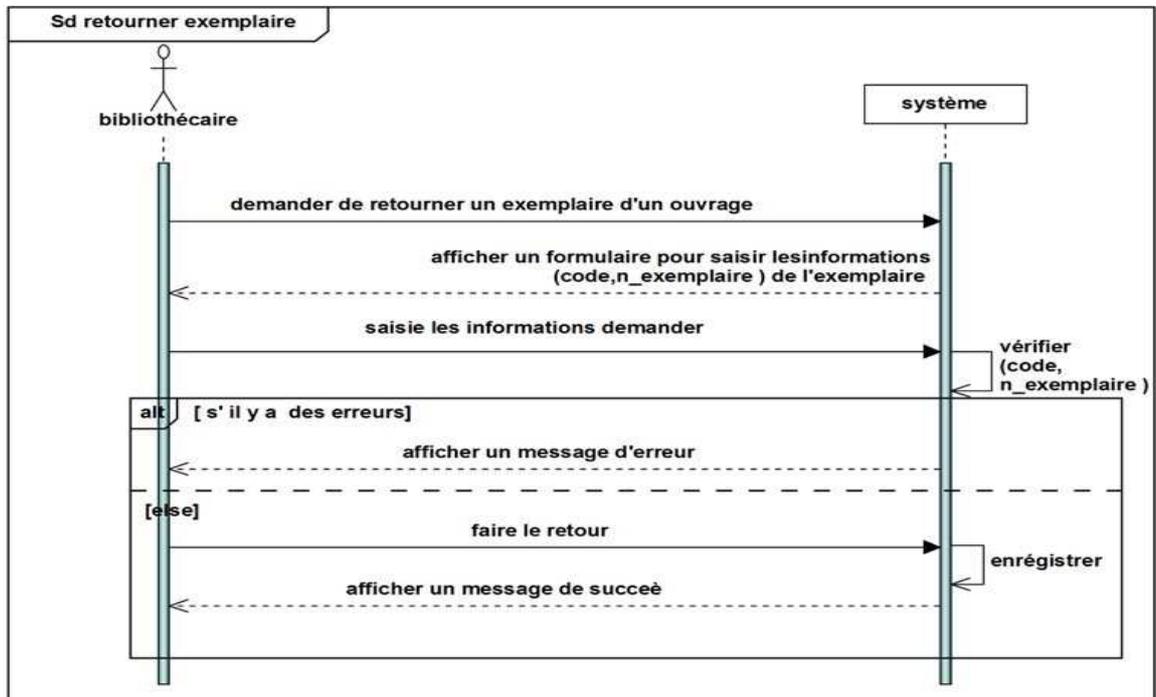


Figure II.17 : Diagramme de séquence du cas retourné un exemplaire d'un ouvrage.

7.5- diagrammes de séquences du cas d'utilisation Rechercher un ouvrage



Figure II.18 : Diagramme de séquence du cas Rechercher ouvrage.

7.6- diagrammes de séquences du cas d'utilisation Gestion de pénalité

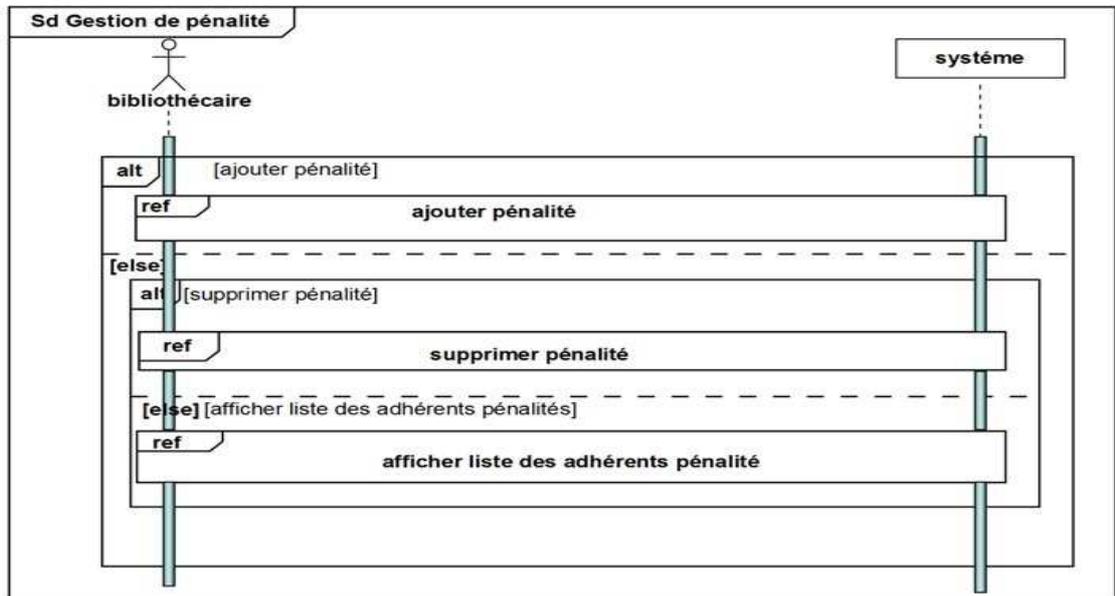


Figure II.19 : Diagramme de séquence du cas Gestion de pénalité.

7.6.1- diagrammes de séquences du cas d'utilisation ajouter pénalité

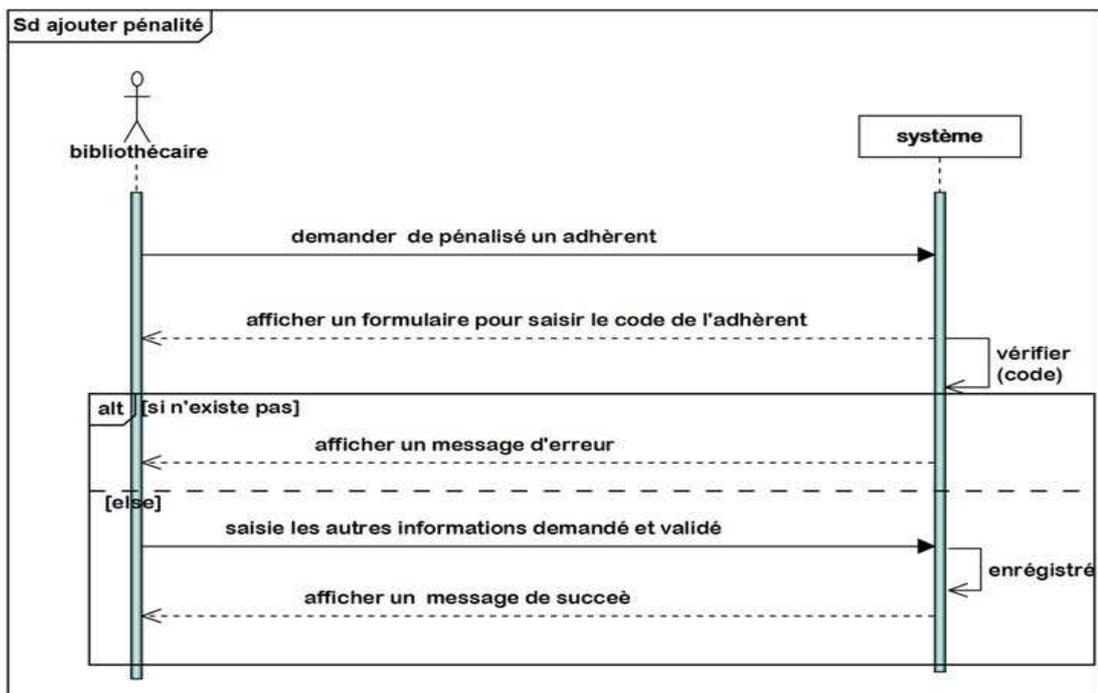


Figure II.20 : Diagramme de séquence du cas ajouté pénalité.

7.6.2- diagrammes de séquences du cas d'utilisation supprimer pénalité

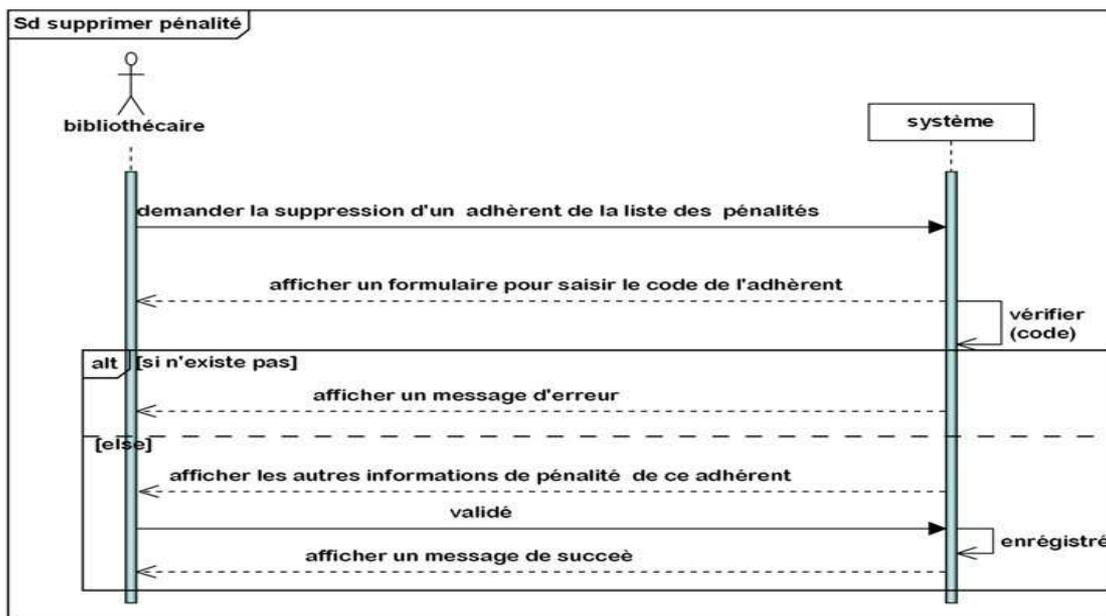


Figure II.21 : Diagramme de séquence du cas supprimé pénalité.

7.6.3- diagrammes de séquences du cas d'utilisation afficher liste des adhérents pénalisés

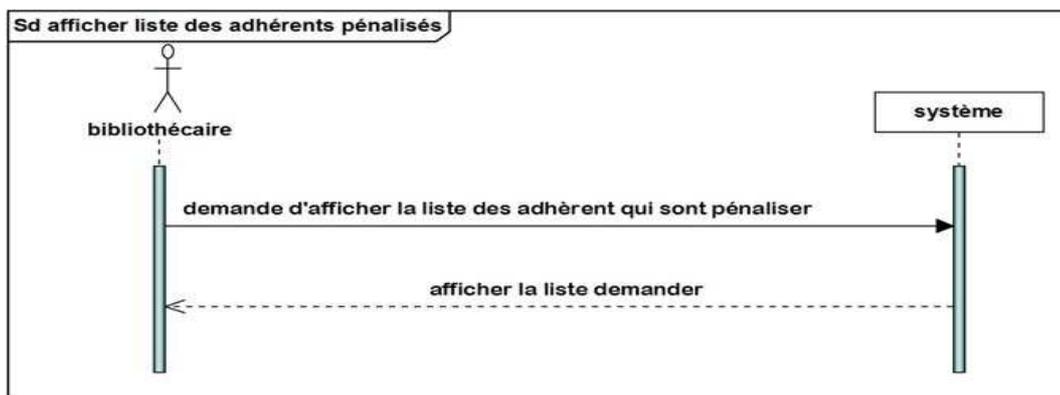


Figure II.22 : Diagramme de séquence du cas affiché liste des adhérents pénalisés.

Conclusion

Cette étude du système actuel nous a permis de déterminer les différents acteurs et leurs tâches, on peut maintenant procéder à la conception du nouveau système en commençant par l'étape d'analyse.

Chapitre III

Analyse

Introduction

1- Analyse des domaines.

2- Réalisation de diagramme d'activité de chaque cas d'utilisation.

Conclusion

CHAPITRE III : Analyse

Introduction

Dans cette phase nous allons préciser les besoins au quelles doit répondre notre système, pour cela on utilise le diagramme de classe (modèle de domaine) et le diagramme d'activité.

1-Analyse des domaines

La phase d'analyse du domaine permet d'élaborer la première version du diagramme de classe appelée modèle du domaine. Ce modèle doit définir les classes qui modélisent les entités ou concepts présents dans le domaine de l'application. On va dans cette étape :

- Identifier les concepts des domaines ou bien identifier les classes qui réalisent chaque cas d'utilisation.
- Ajouter les associations entre ces classes.
- Ajouter les attributs des classes.

1.1-Identification les concepts des domaines

a) S'authentifier

- ✓ Le bibliothécaire (l'administrateur ou l'agent de prêt).

b) Gestion d'ouvrage

- ✓ L'administrateur.
- ✓ Ouvrage.
- ✓ exemplaire (d'un ouvrage).

c) Gestion des adhérents

- ✓ L'administrateur.
- ✓ L'Adhérent.

d) Gestion des emprunts

- ✓ Le bibliothécaire (l'administrateur et l'agent de prêt).
- ✓ Ouvrage.

- ✓ L'adhérent.
- ✓ Exemplaire.
- ✓ Emprunt.
- ✓ Pénalité.

e) Rechercher ouvrage

- ✓ L'utilisateur (l'administrateur, l'agent de prêt et l'utilisateur simple).

- ✓ Ouvrage.
- ✓ Exemplaire.

f) Gestion de pénalité

- ✓ Le bibliothécaire (l'administrateur et l'agent de prêt).

- ✓ Adhérent.
- ✓ Pénalité.

1.2-Ajouté les attributs et les associations entre ces classes (modèle de domaine)

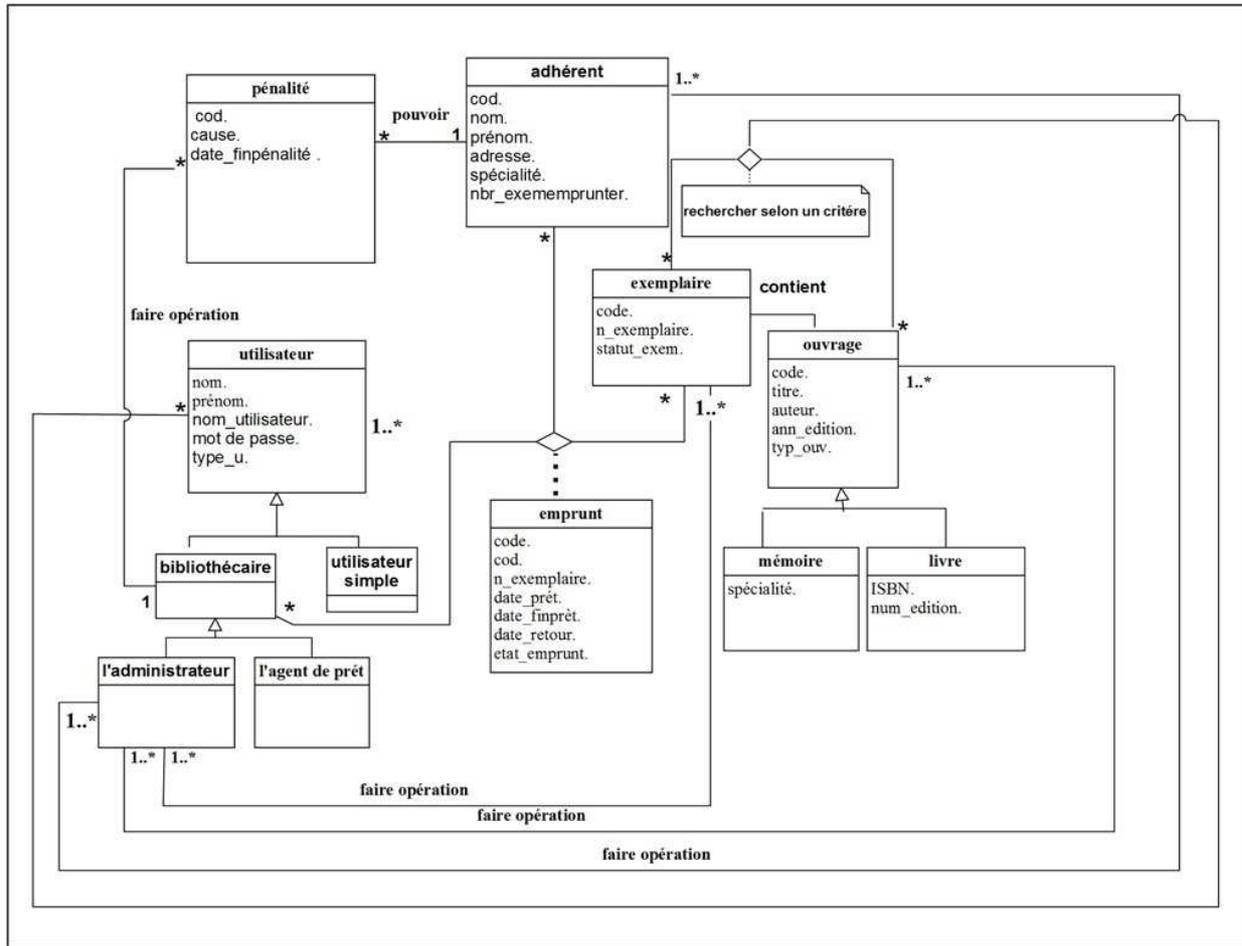


Figure III.1 : Diagramme de modèle de domaine.

2-Réalisation de diagramme d'activité de chaque cas d'utilisation

Ce diagramme permet de décrire l'enchaînement des activités propres à un cas d'utilisation et préciser comment un système se comporte au cours du temps.

2.1 diagramme d'activité du cas d'utilisation authentifié

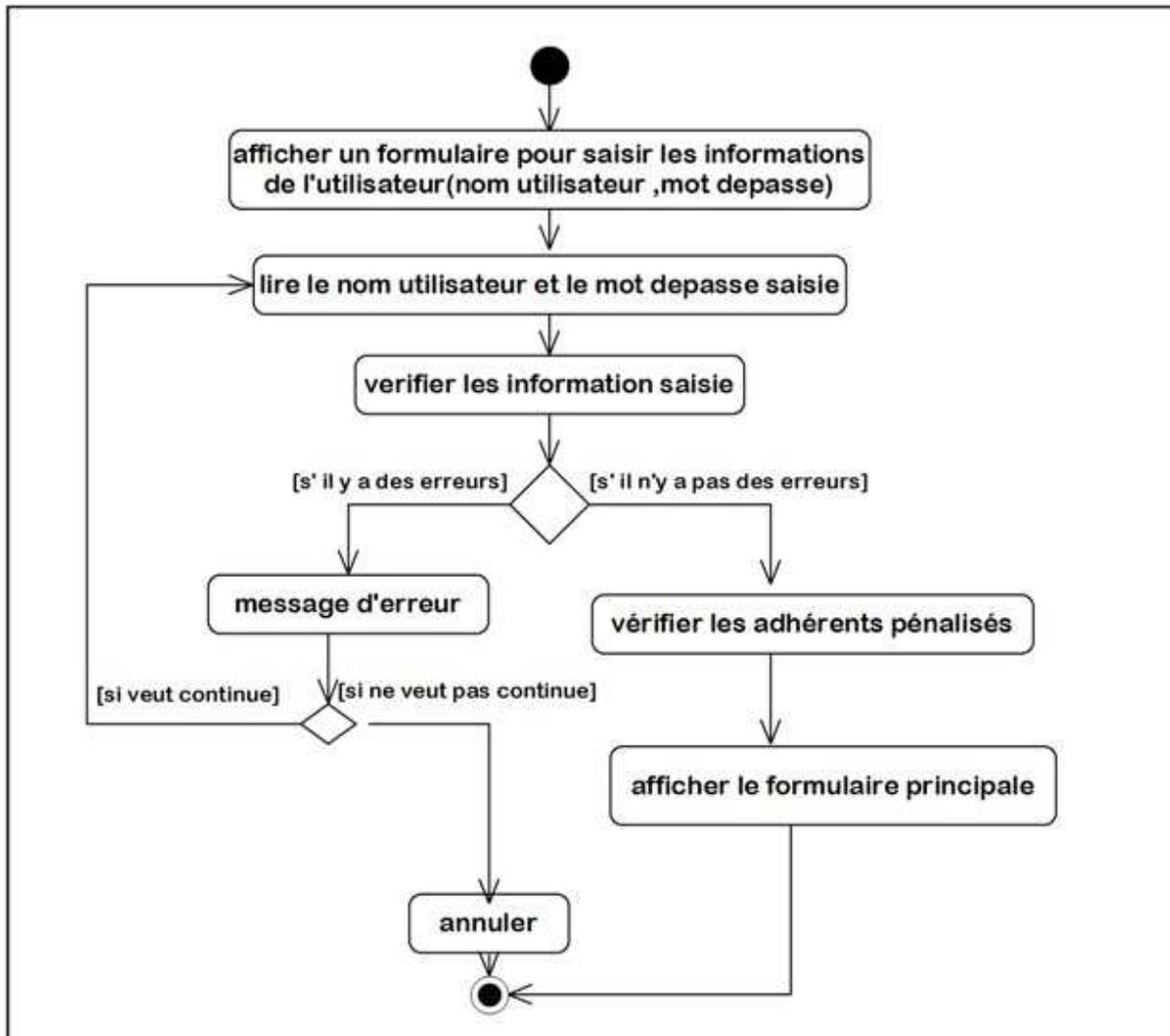


Figure III.2 : Diagramme d'activité du cas authentifié.

2.2-diagramme d'activité du cas d'utilisation ajouter ouvrage

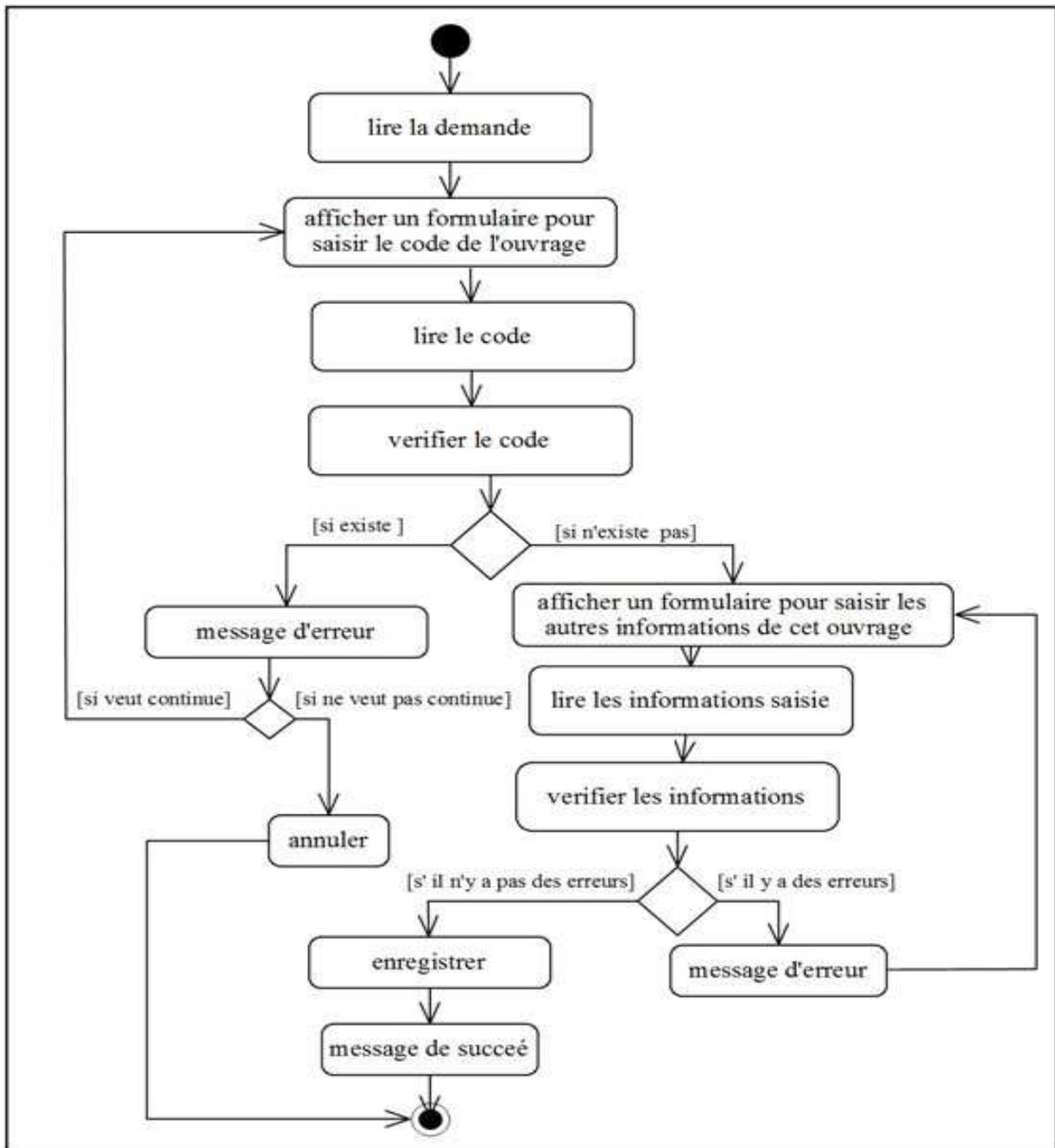


Figure III.3 : Diagramme d'activité du cas ajouter ouvrage.

2.3- diagramme d'activité du cas d'utilisation modifier ouvrage

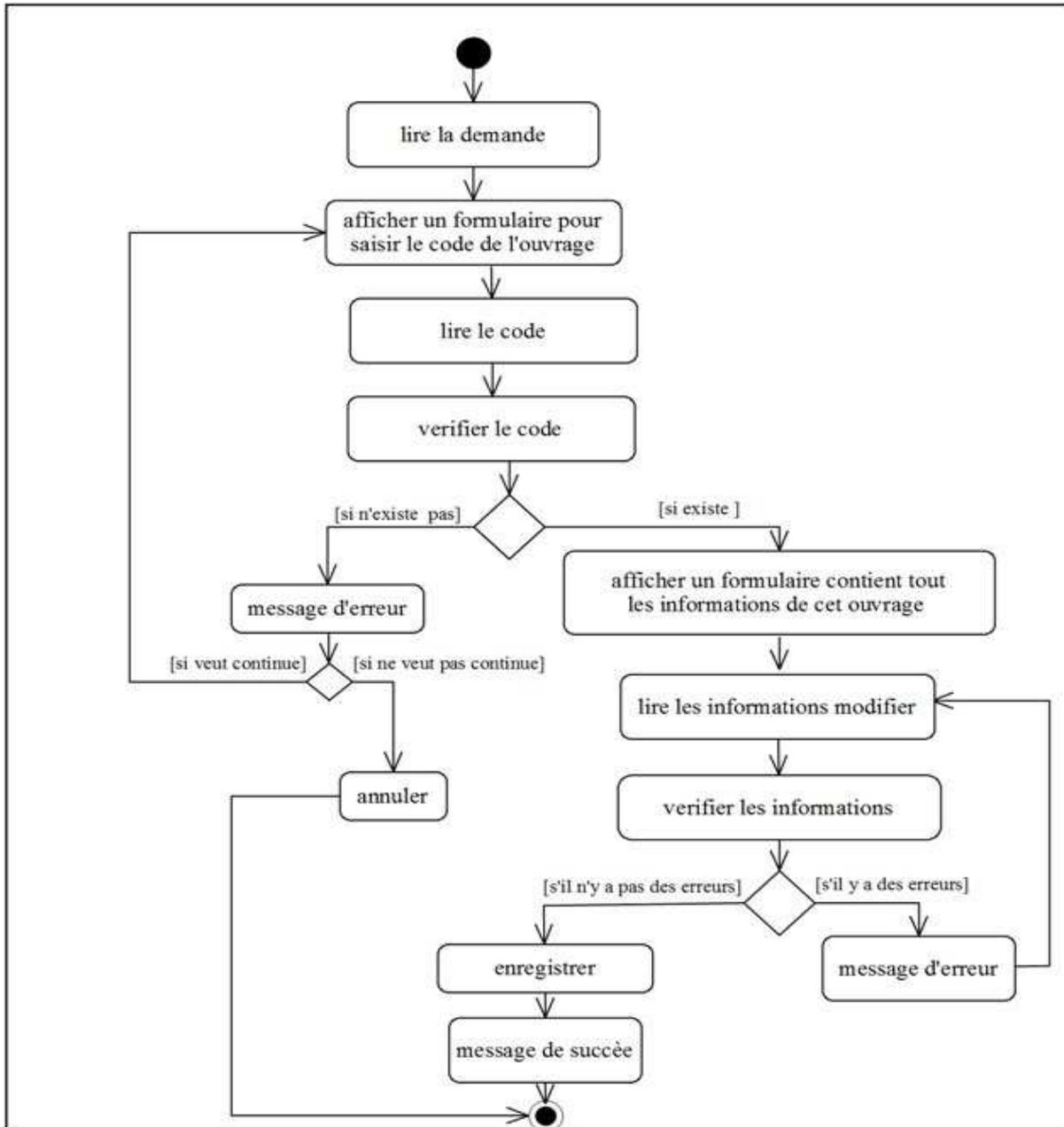


Figure III.4 : Diagramme d'activité du cas modifier ouvrage.

2.4 - diagramme d'activité du cas d'utilisation ajouter exemplaire

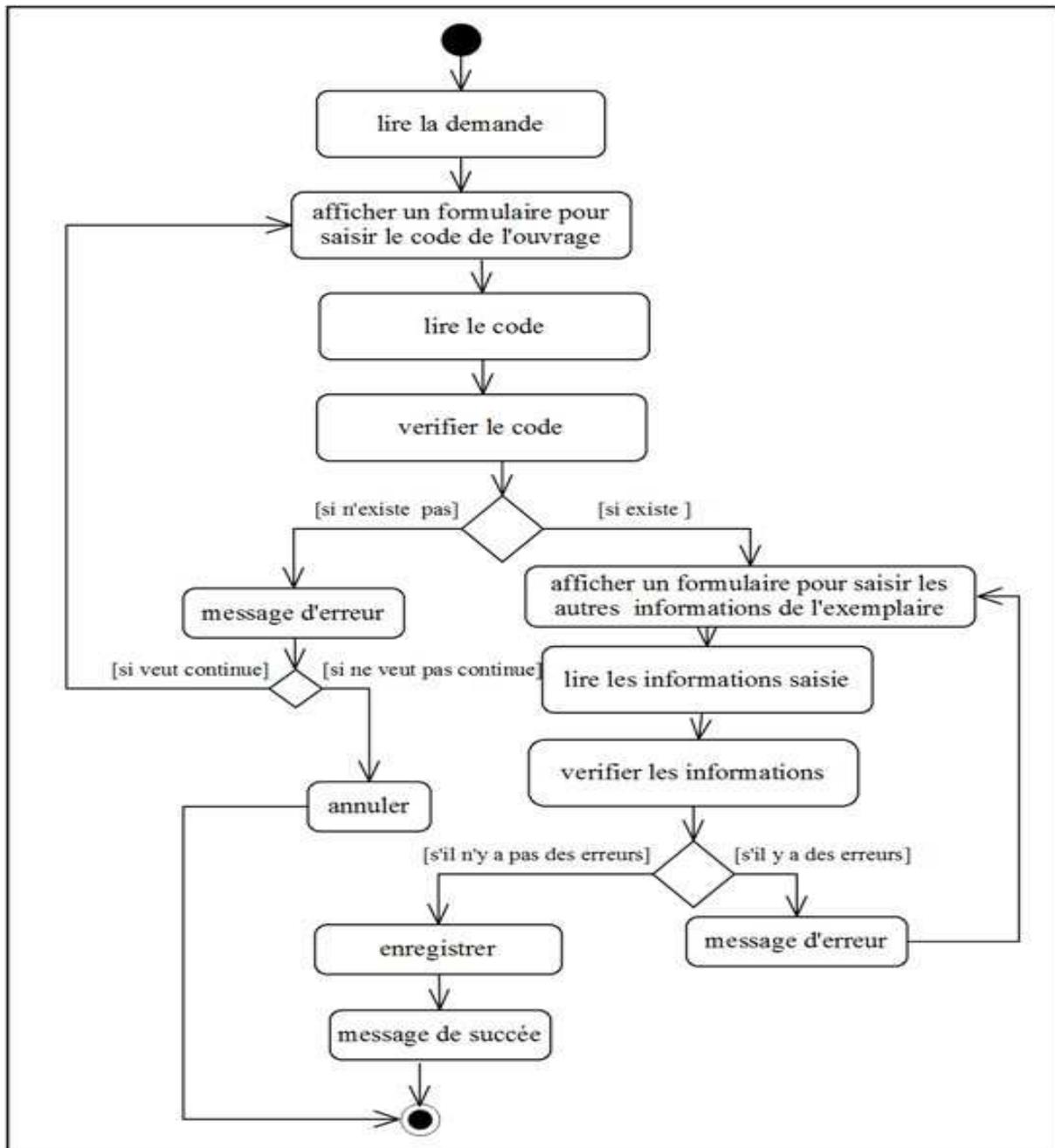


Figure III.5 : Diagramme d'activité du cas ajouté exemplaire.

2.5- diagramme d'activité du cas d'utilisation modifier exemplaire

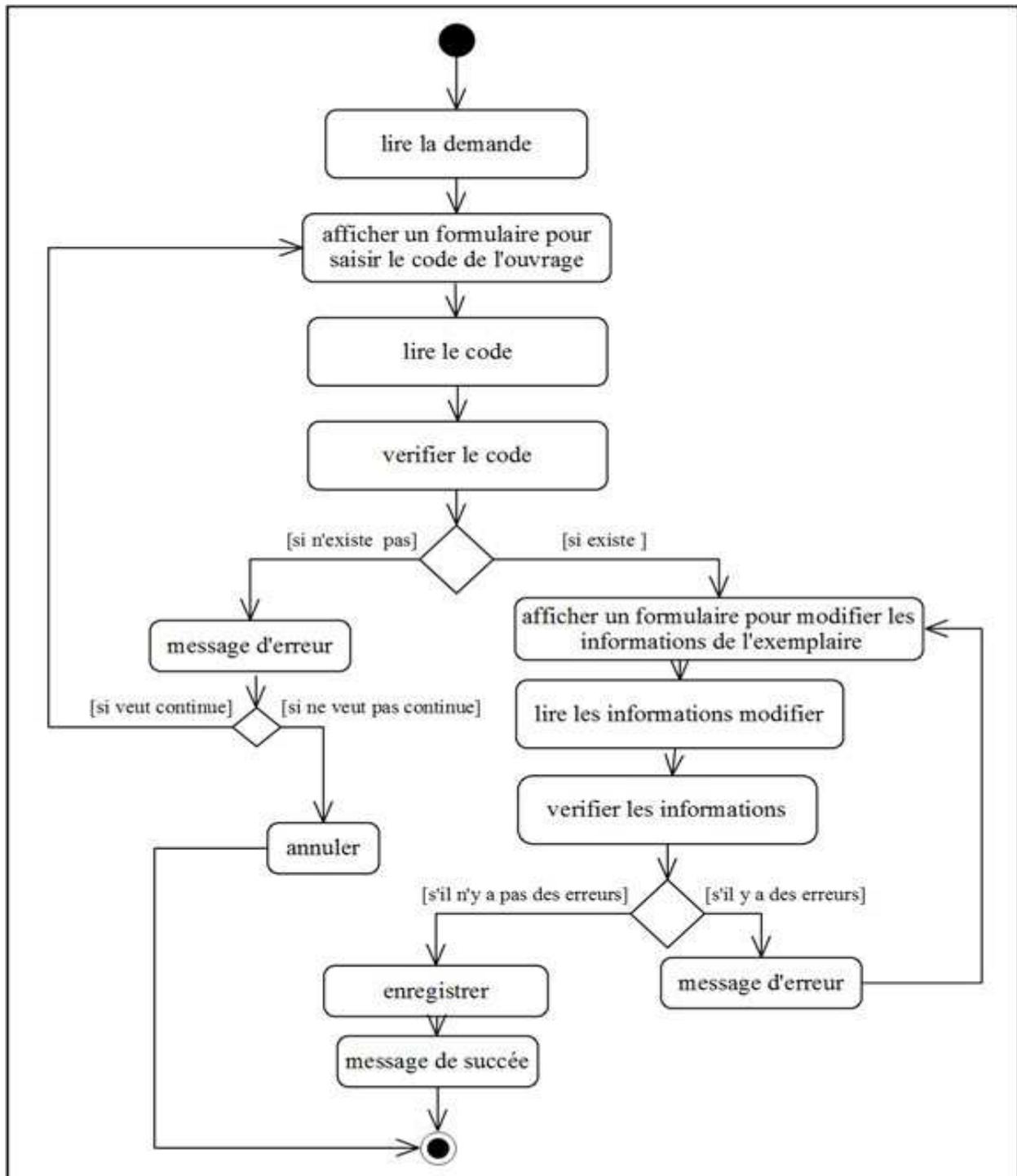


Figure III.6 : Diagramme d'activité du cas modifié exemplaire.

2.6- diagramme d'activité du cas d'utilisation supprimer exemplaire

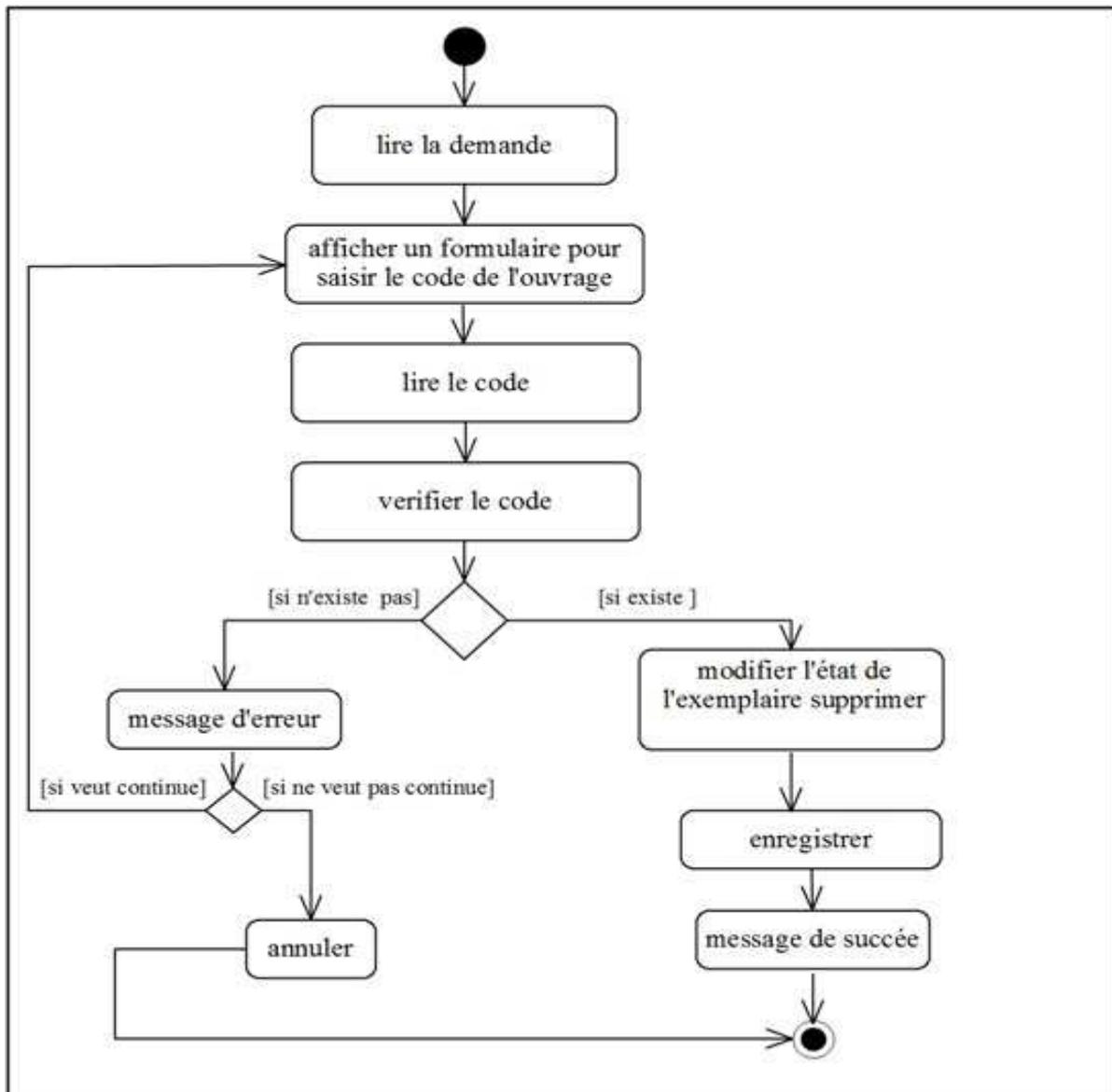


Figure III.7 : Diagramme d'activité du cas supprimé exemplaire.

2.7- diagramme d'activité du cas d'utilisation ajouter adhérent

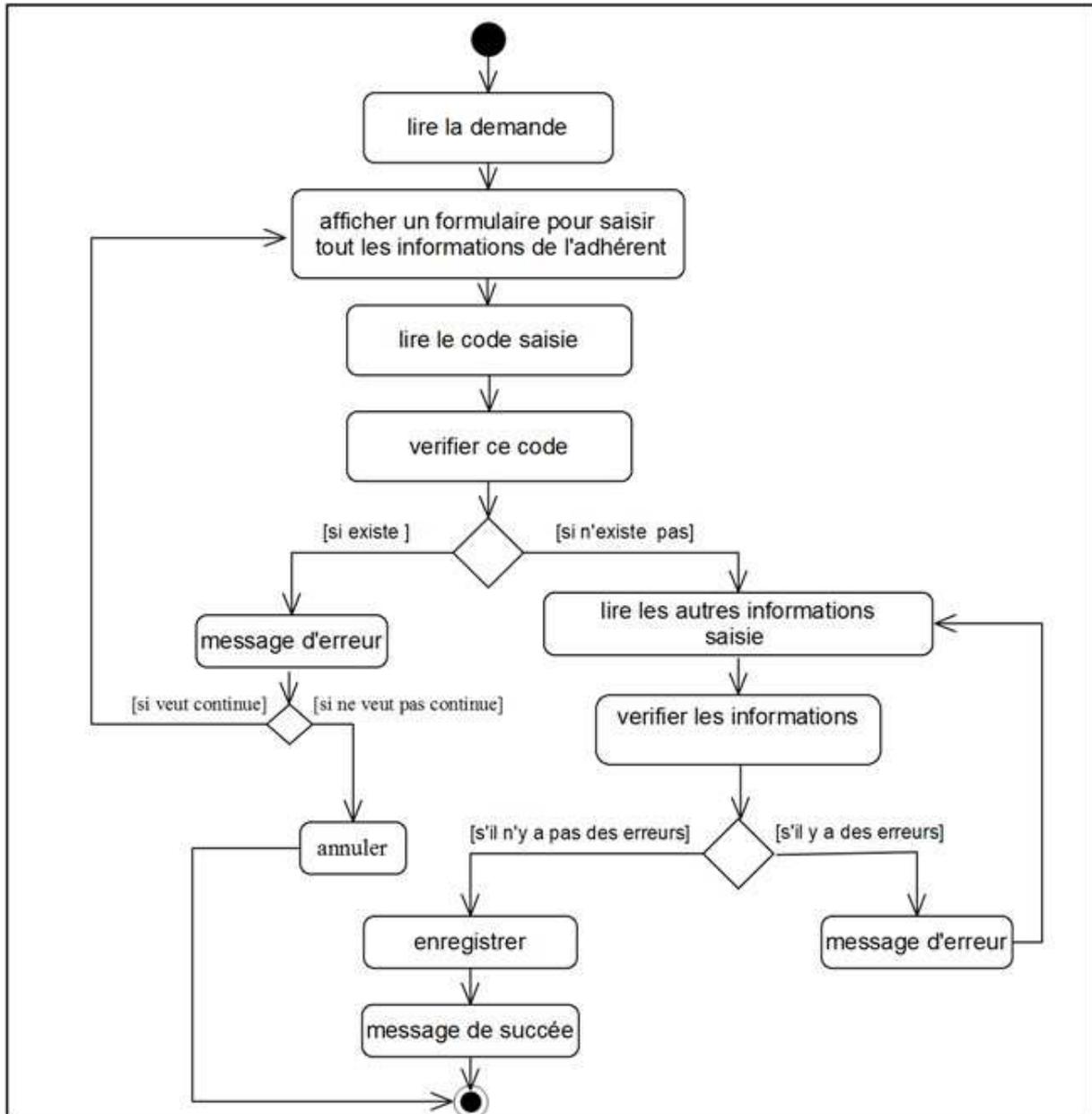


Figure III.8 : Diagramme d'activité du cas ajouté adhérent.

2.8- diagramme d'activité du cas d'utilisation modifier adhérent

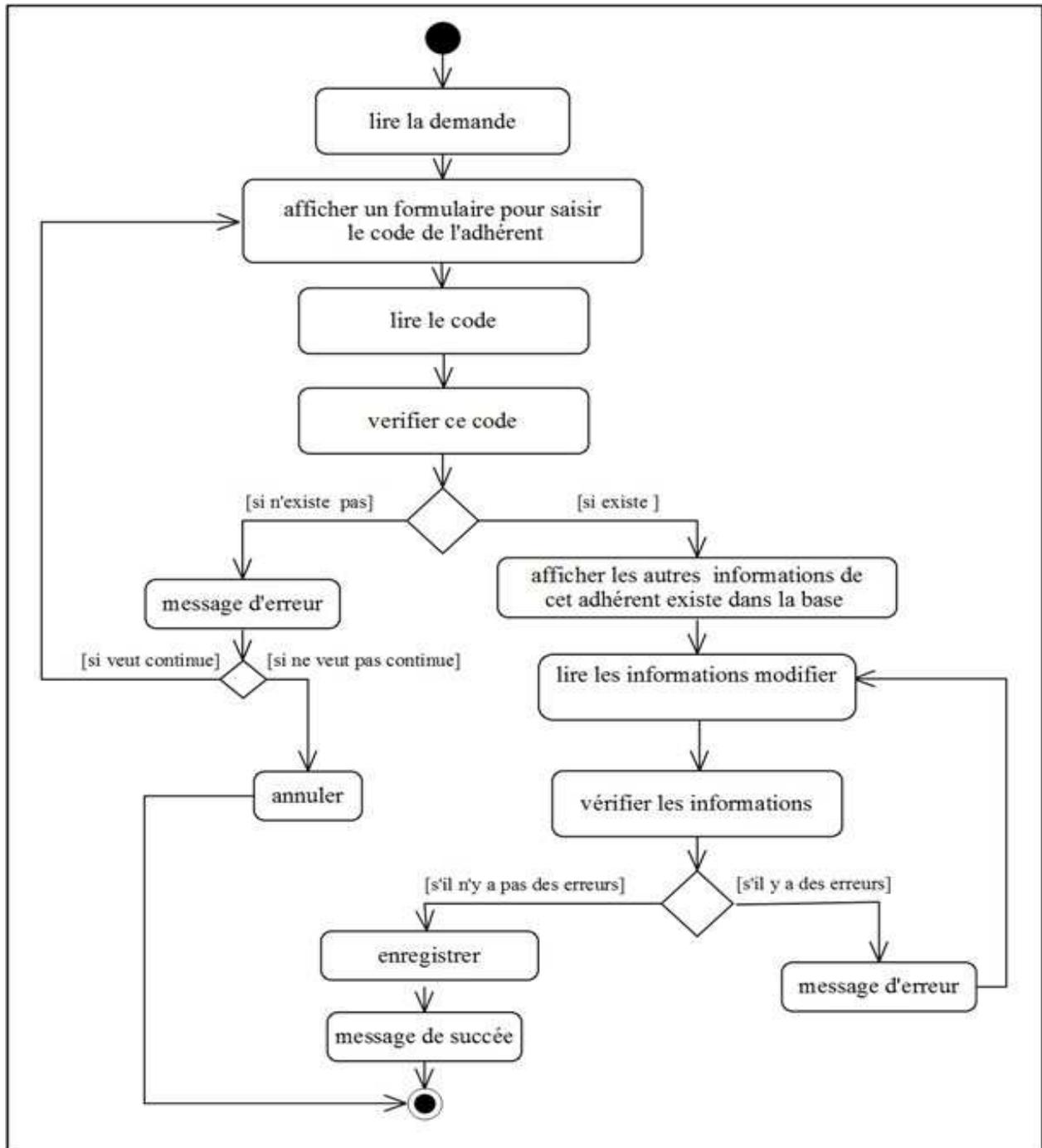


Figure III.9 : Diagramme d'activité du cas modifié adhérent.

2.9-diagramme d'activité du cas d'utilisation supprimer adhérent

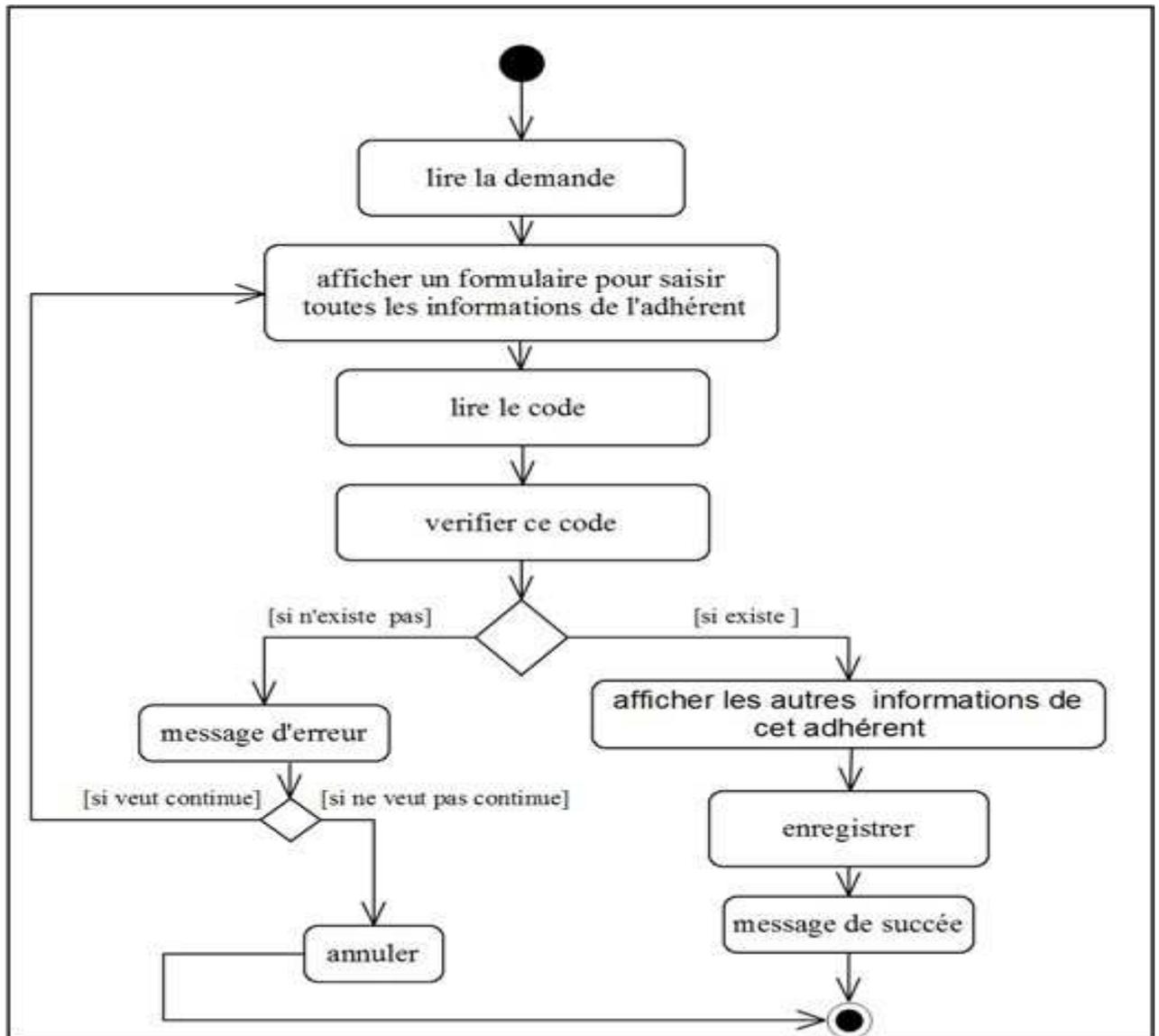


Figure III.10 : Diagramme d'activité du cas supprimé adhérent.

2.10- diagrammes d'activité du cas d'utilisation emprunter exemplaire

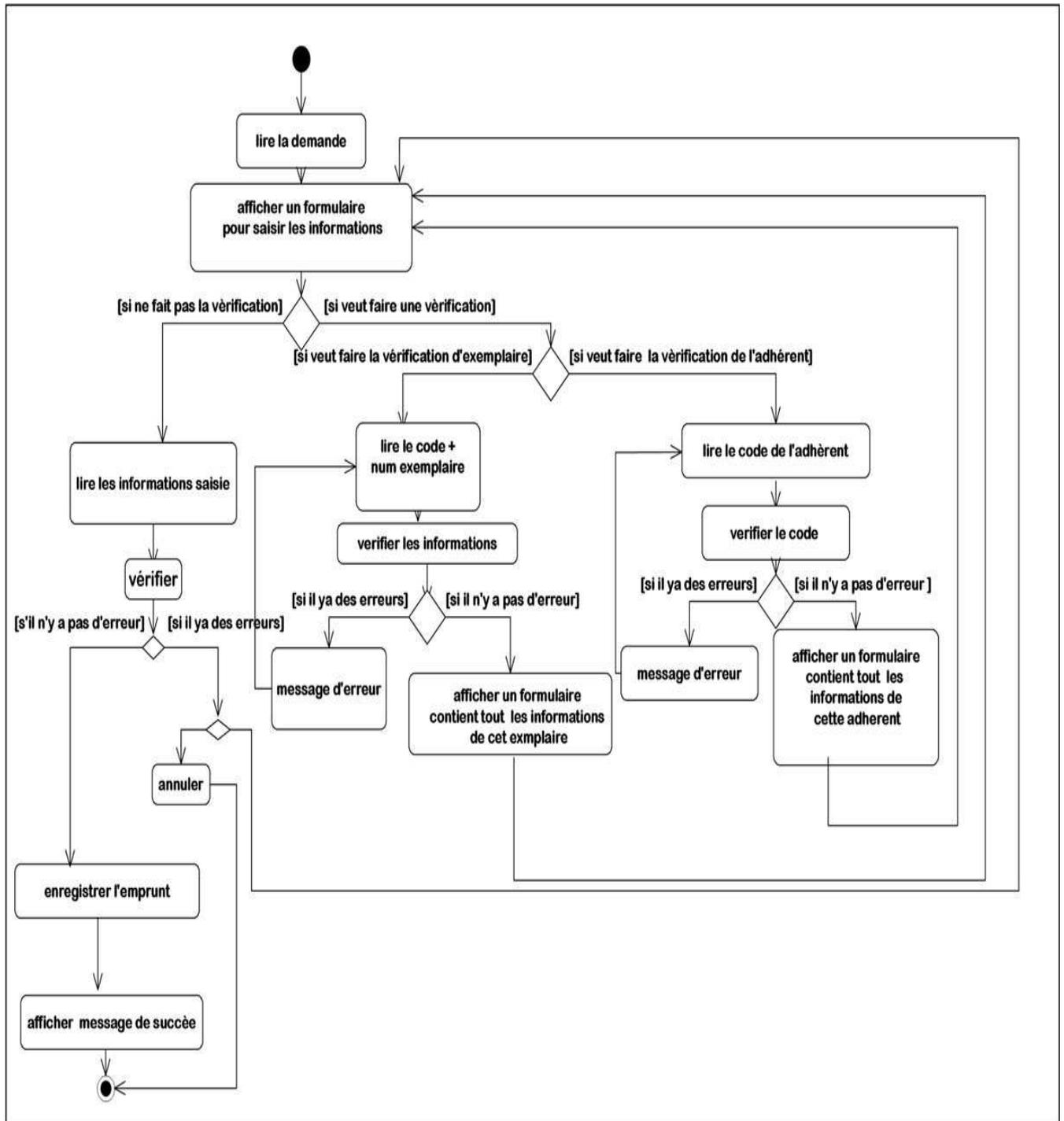


Figure III.11 : Diagramme d'activité du cas emprunté exemplaire.

2.11- diagramme d'activité du cas d'utilisation retourner exemplaire

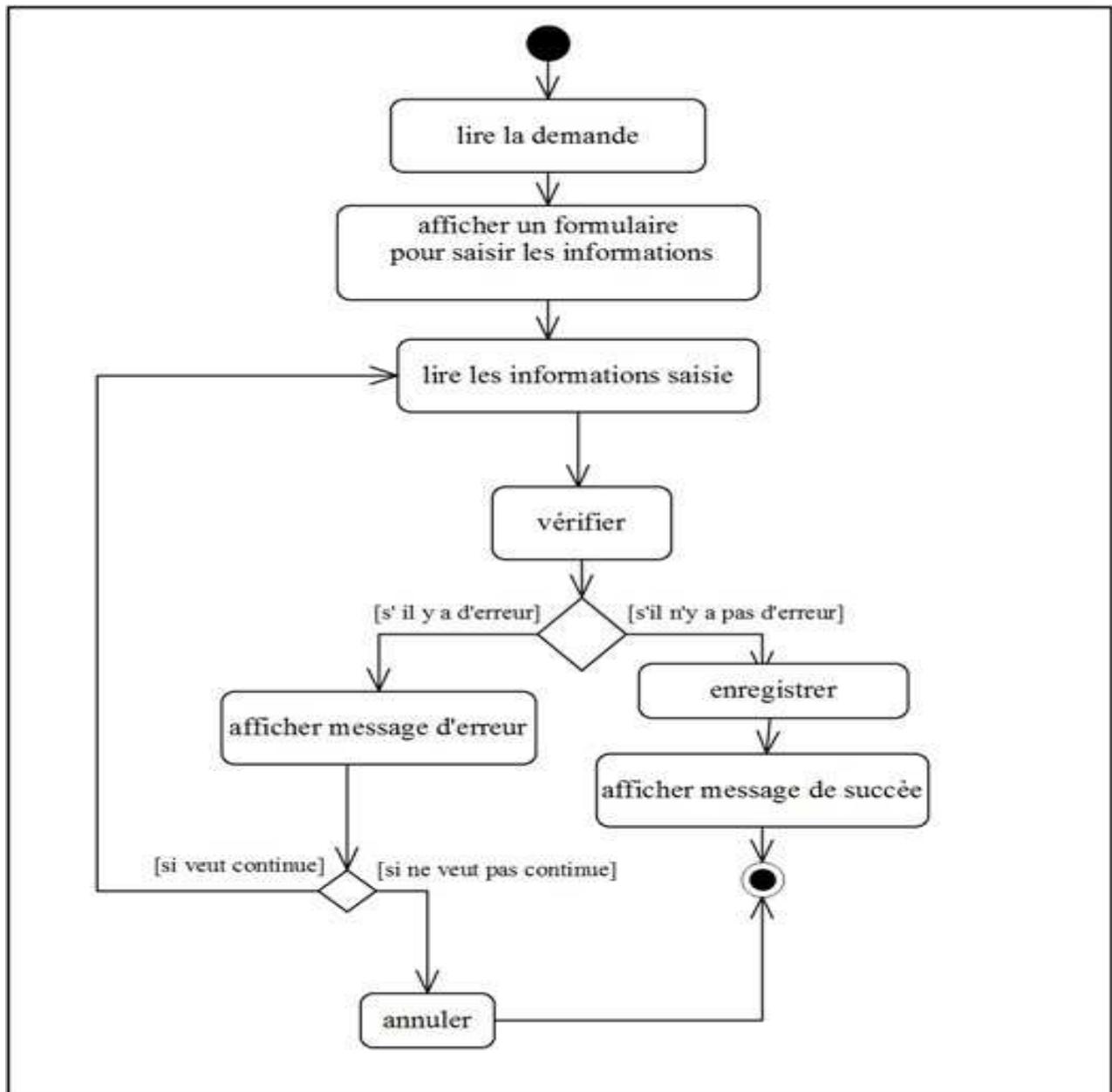


Figure III.12 : Diagramme d'activité du cas retourné exemplaire.

2.12- diagramme d'activité du cas d'utilisation rechercher ouvrage

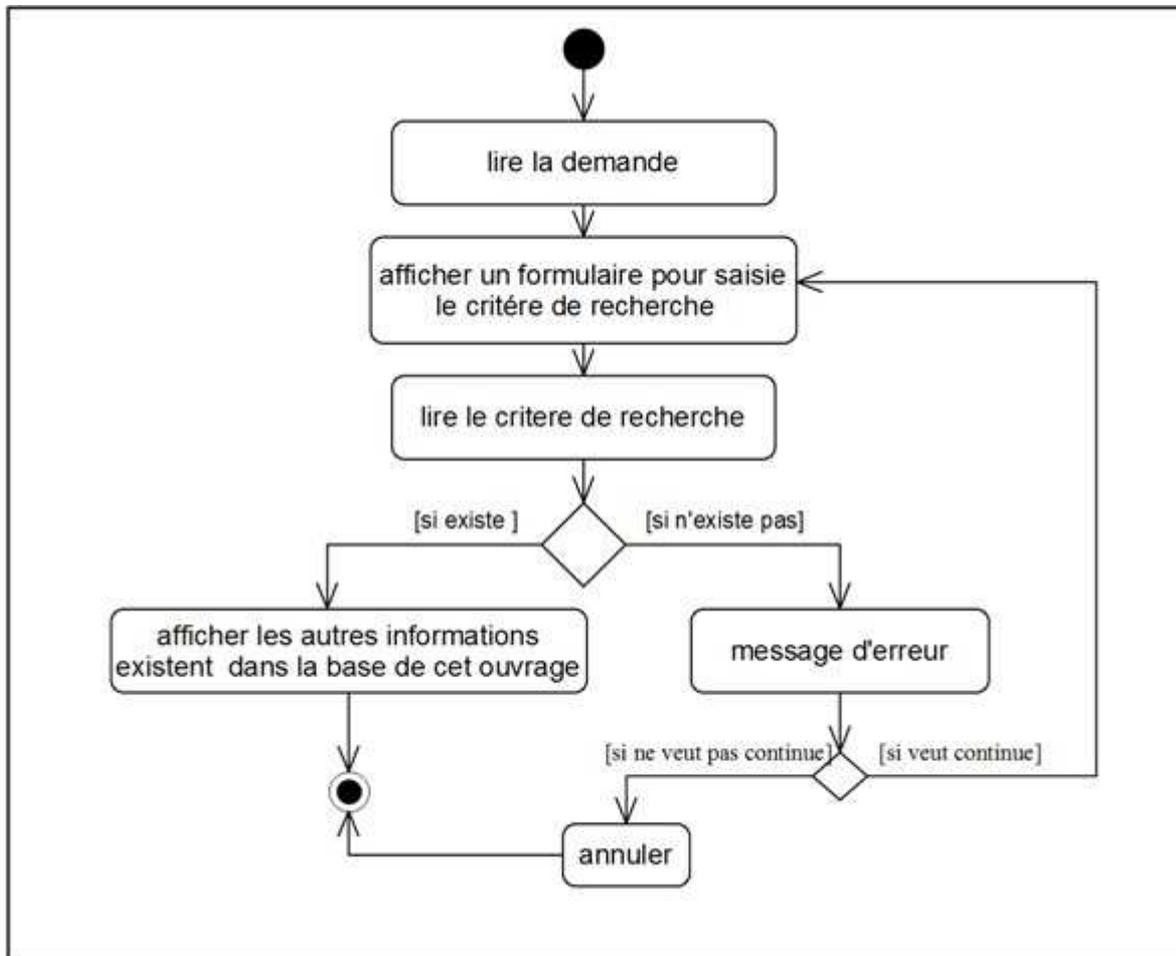


Figure III.13 : Diagramme d'activité du cas Rechercher ouvrage.

2.13- diagramme d'activité du cas d'utilisation ajouter pénalité

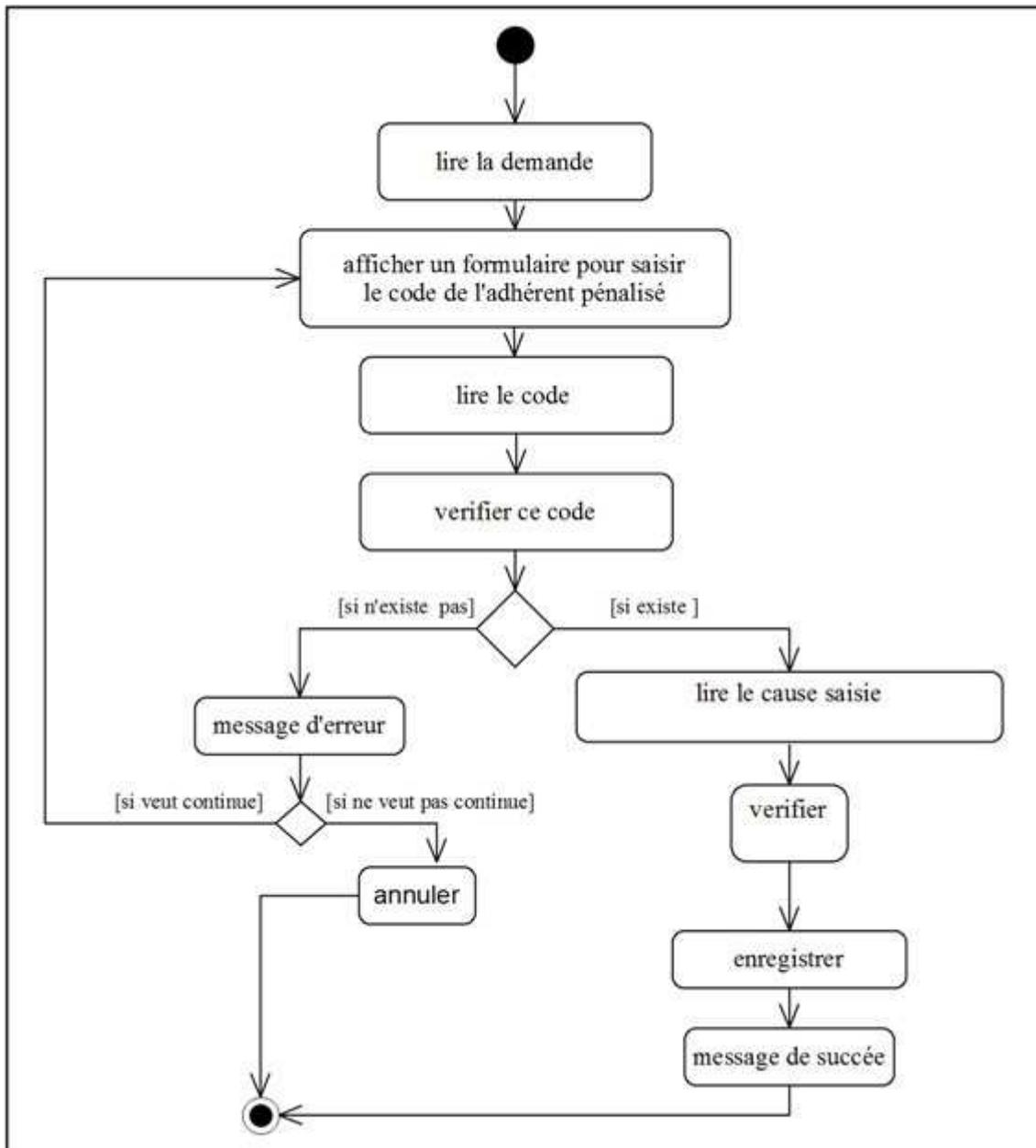


Figure III.14 : Diagramme d'activité du cas ajouté pénalité.

2.14- diagramme d'activité du cas d'utilisation supprimer pénalité

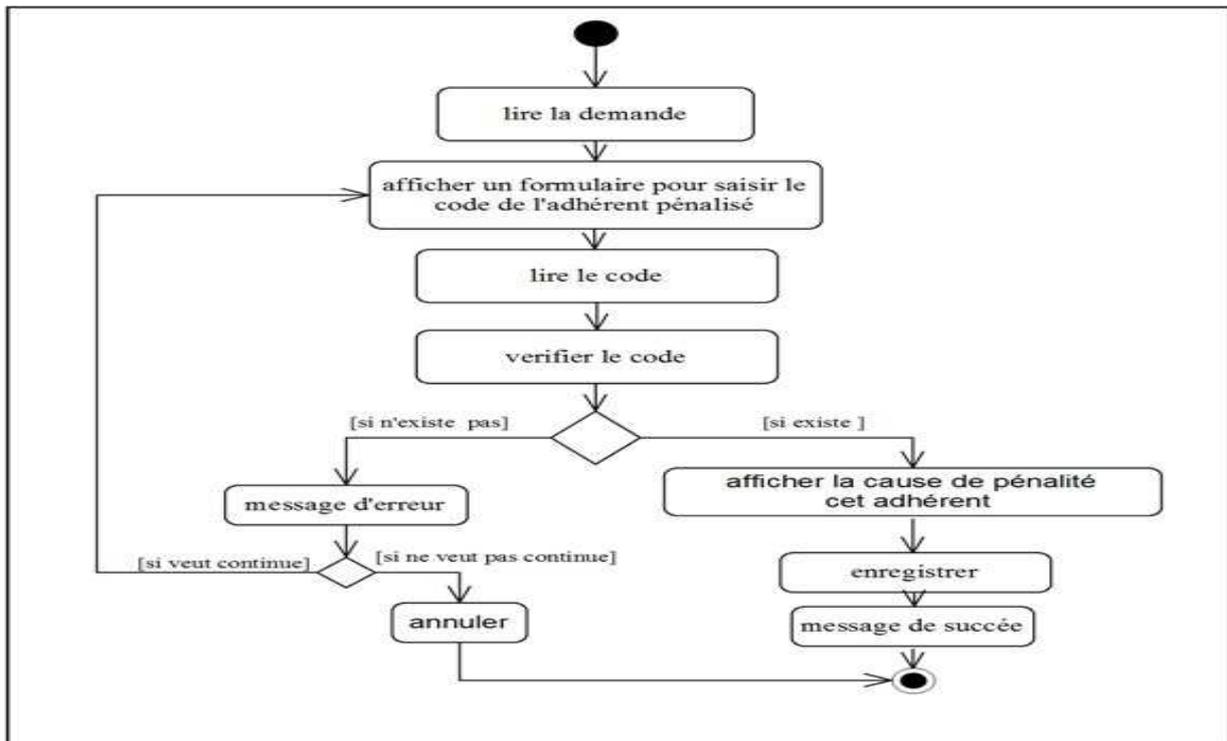


Figure III.15 : Diagramme d'activité du cas supprimé pénalité.

2.15- diagramme d'activité du cas d'utilisation afficher liste des adhérents pénalisés :

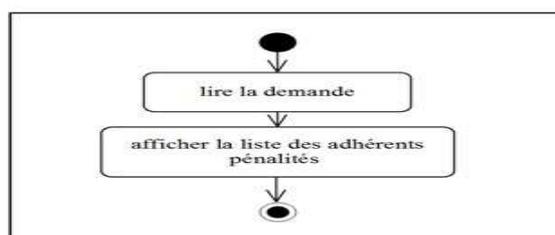


Figure III.16 : Diagramme d'activité du cas afficher la liste des adhérents pénalisés.

Conclusion

Dans cette étape nous avons exprimé clairement les objectifs attendus du futur système à concevoir, ainsi que l'analyse associée à chaque cas d'utilisation. Il faut noter que l'étape d'analyse est une activité utile qui va nous permettre d'introduire la prochaine étape «Conception », que nous allons détailler dans le chapitre suivant.

Chapitre IV

Conception

Introduction

3- Dictionnaire de données.

4- Les diagrammes de séquences de chaque cas d'utilisation.

5- Diagramme de classe de conception.

6- Le modèle logique de données.

7- Diagramme de déploiement.

Conclusion

Chapitre IV : Conception

Introduction

Dans ce chapitre on va décrire la façon dont le système va fonctionner. On va façonner le système et lui donner une forme et une architecture en utilisant le Diagramme de séquence d'interaction, le Diagramme de classe de conception et le Diagramme de déploiement.

1-Dictionnaire de données

1.1-Concevoir les classes et les attributs

Classe	Information	Code	Type
Utilisateur	le nom	Nom	Var char (25)
	le prénom	prénom	Var char (25)
	nom utilisateur	Nom_utilisateur	Var char (25)
	mot de passe	mot de passe	Var char (25)
	Type utilisateur	Typ_u	Var char (25)
Ouvrage	Identifiant de l'ouvrage	<u>code</u>	Var char (25)
	International Standard Book Number	ISBN	Var char (25)
	Le titre de l'ouvrage	titre	Var char (50)
	L'auteur de l'ouvrage	auteur	Var char (50)
	le numéro édition de l'ouvrage	Num_edition	Var char (50)
	Spécialité de l'ouvrage	Spécialité	Var char (50)
	Année édition de l'ouvrage	Ann_édition	Var char (50)

	Le type de l'ouvrage	Typ_ouv	Var char (50)
exemplaire	Identifiant de l'ouvrage	<u>code</u>	Var char (50)
	Statut de l'exemplaire	Statut_exem	Var char (50)
	Numéro de l'exemplaire	n_exemplaire	Var char (50)
Adhérent	Identifiant de l'adhérent	<u>cod</u>	Var char (50)
	Le nom	nom	Var char (50)
	Le prénom	Le prénom	Var char (50)
	L'adresse	adresse	Var char (50)
	La spécialité	spécialité	Var char (50)
	Nombre d'exemplaire qui peut emprunter	Nbr_exememprunter	INT (10)
Emprunt	Identifiant de l'ouvrage	code	Var char (50)
	Identifiant de l'adhérent	cod	Var char (50)
	Numéro de l'exemplaire	n_exemplaire	Var char (50)
	La date de prêt	Date_pret	Date
	La date de fin de prêt	Date_finprêt	Date
	La date de retour	Date-retour	Date
	Etat d'emprunt	Etat_emp	Var char (50)
Pénalité	Identifiant de l'adhérent	<u>cod</u>	Var char (50)
	La cause	cause	Var char (50)
	La date de fin pénalité	Date_finpenalite	Date

Tableau V.1 : Description détaillée des classes d' objets.

1.2-Concevoir les opérations

Classe	Méthode	Description
Ouvrage	Ajouter_ ouvrage ()	Ajoute un nouvel ouvrage.
	Modifier_ouvrage ()	Modifier un ouvrage.
	Vérifier_code ()	Vérifier l'existence de cet ouvrage par le code.
	afficher_info ()	Permet d'afficher tout les informations d'un ouvrage.
	Recherche_code ()	Permet d'afficher tout les informations d'un ouvrage et les exemplaires de cet ouvrage (rechercher par code).
	Recherche_auteur ()	Permet d'afficher tout les informations de l'ouvrage et les exemplaires de cet ouvrage (rechercher par l'auteur).
	Recherche_titre ()	Permet d'afficher tout les informations de l'ouvrage et les exemplaires de cet ouvrage (rechercher par titre).
Exemplaire	Ajouter_exemplaire ()	Ajoute un nouvel exemplaire.
	Modifier_exemplaire ()	Modifier un exemplaire.
	Supprimer_exemplaire ()	Supprimer un exemplaire.
	Vérifier_exepmlaire ()	Afficher les informations d'un exemplaire.

Adhérent	Ajouter_adhèrent ()	Ajouter un nouvel adhérent.
	Modèfier_adhèrent ()	Modifier un adhérent.
	Supprimer_adhèrent ()	Supprimer un adhérent.
	Vérifier _adhèrent ()	Afficher les informations d'un adhérent.
Emprunt	Emprunte ()	Permet d'emprunter un exemplaire d'un ouvrage.
	Retour ()	Permet de retourner un exemplaire d'un ouvrage.
Pénalité	Pénalité ()	Permet de pénalisé un adhérent.
	Supprimer_pénalité ()	Permet de supprimer un adhérent à la pénalité.
	Rechercher_pénalité ()	Permet d'afficher la liste des adhérents pénalisés.
Bibliothécaire	Authentifier ()	Permet d'authentifier un bibliothécaire.

Tableau V.2 : Liste et description des opérations.

2-Les Diagrammes de séquence interaction de chaque cas d'utilisation

Représente les objets qui interagissent dans le système pour réalisé les fonctions de l'application, le comportement de l'application repose sur les communications entres les objets (appelles de méthodes).

2.1- S'authentifier

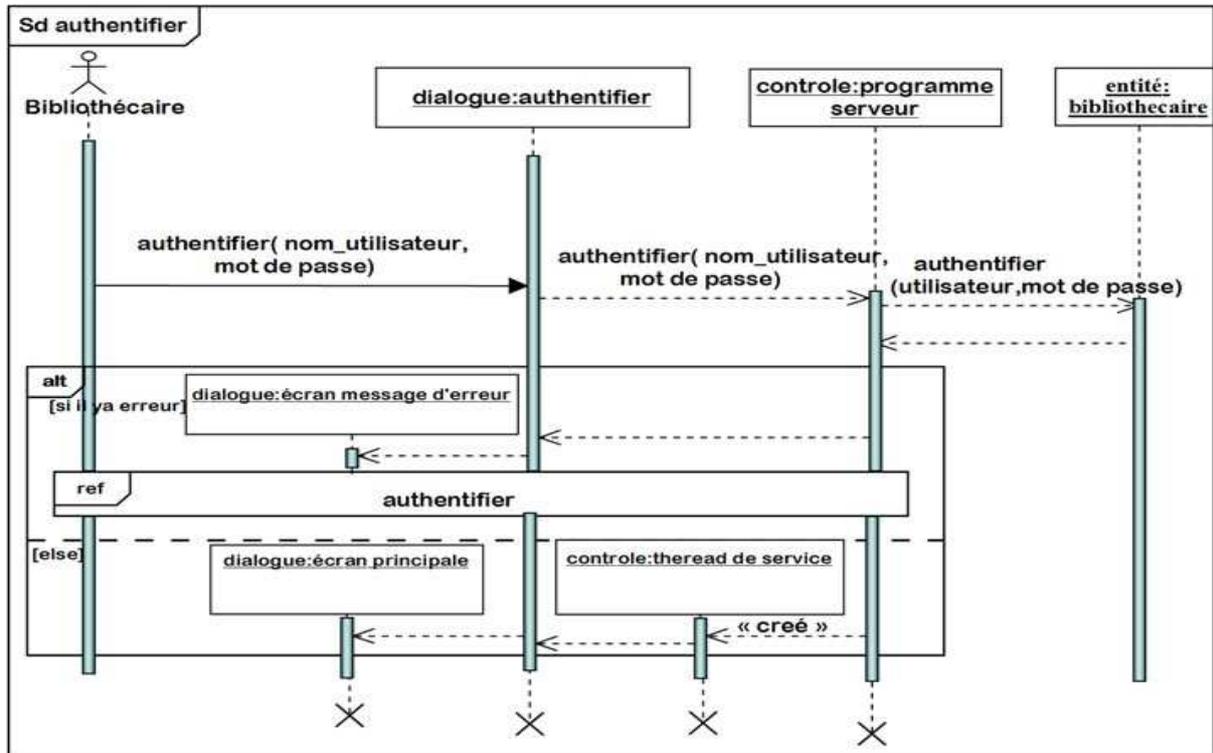


Figure IV.1 : Diagramme de séquence d'interaction du cas authentifier.

2.2-Ajouter ouvrage

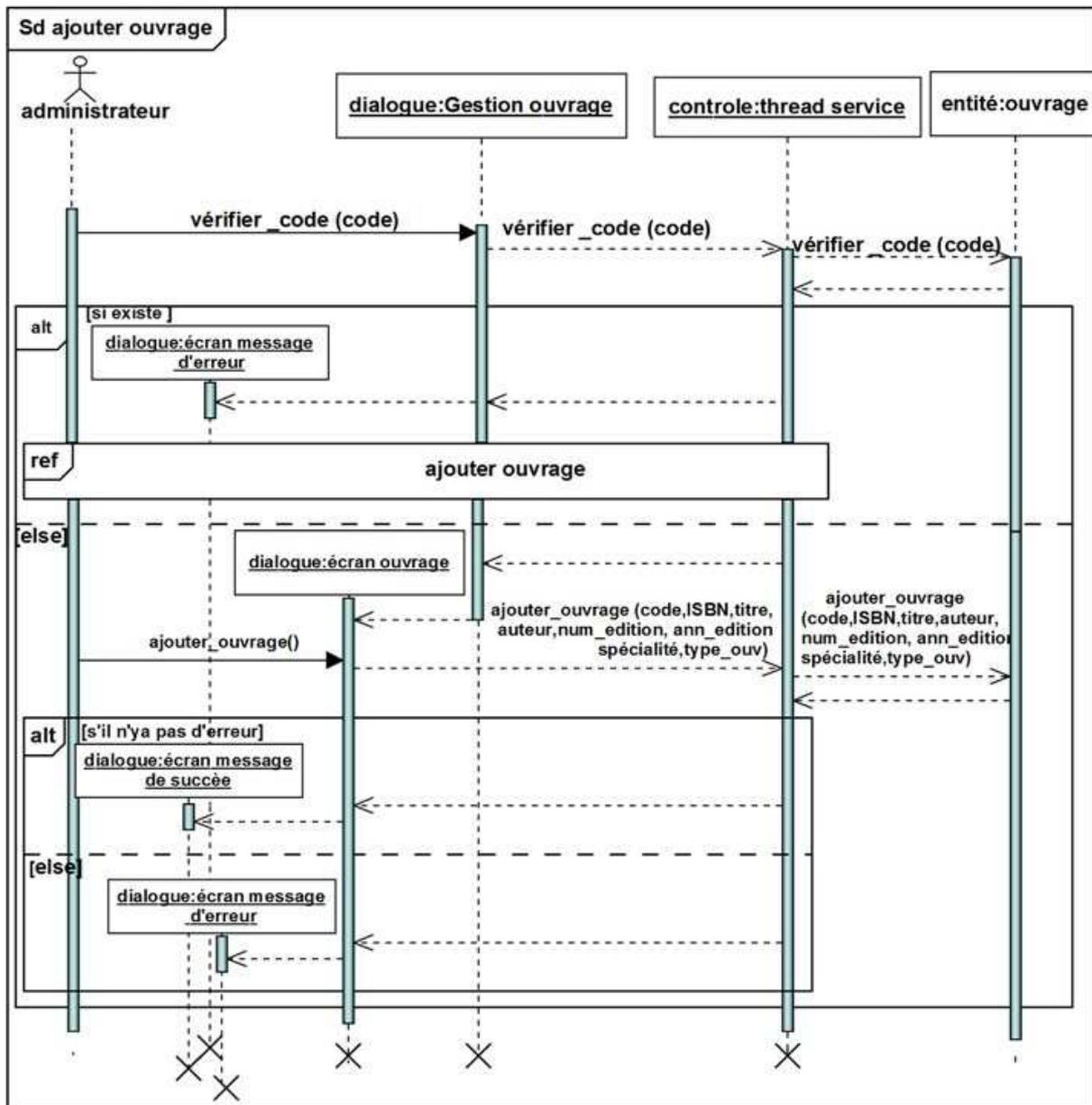


Figure IV.2 : Diagramme de séquence d'interaction du cas ajouté ouvrage.

2.3-Modifier ouvrage

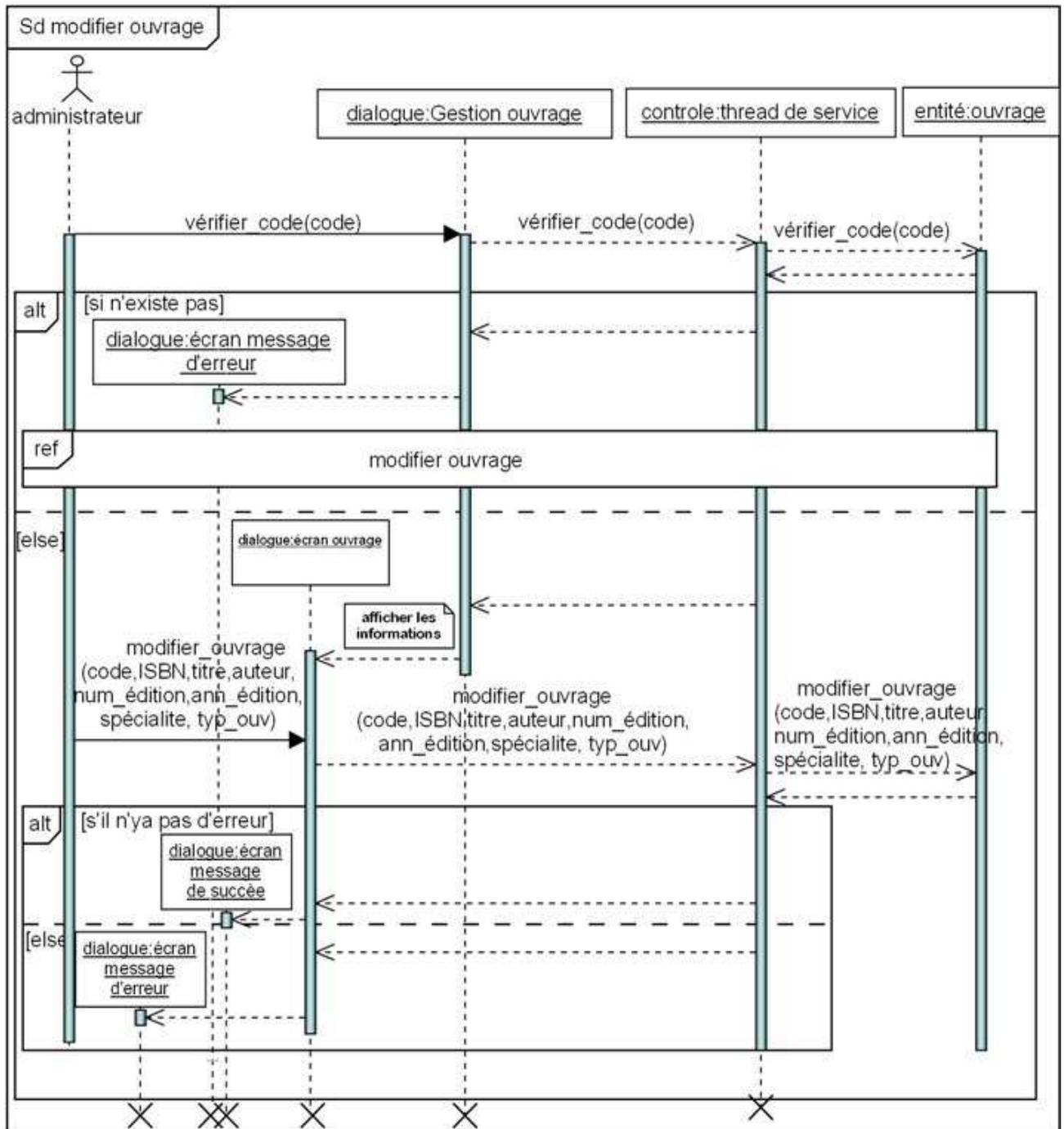


Figure IV.3 : Diagramme de séquence d'interaction du cas modifié ouvrage.

2.4-Ajouter exemplaire

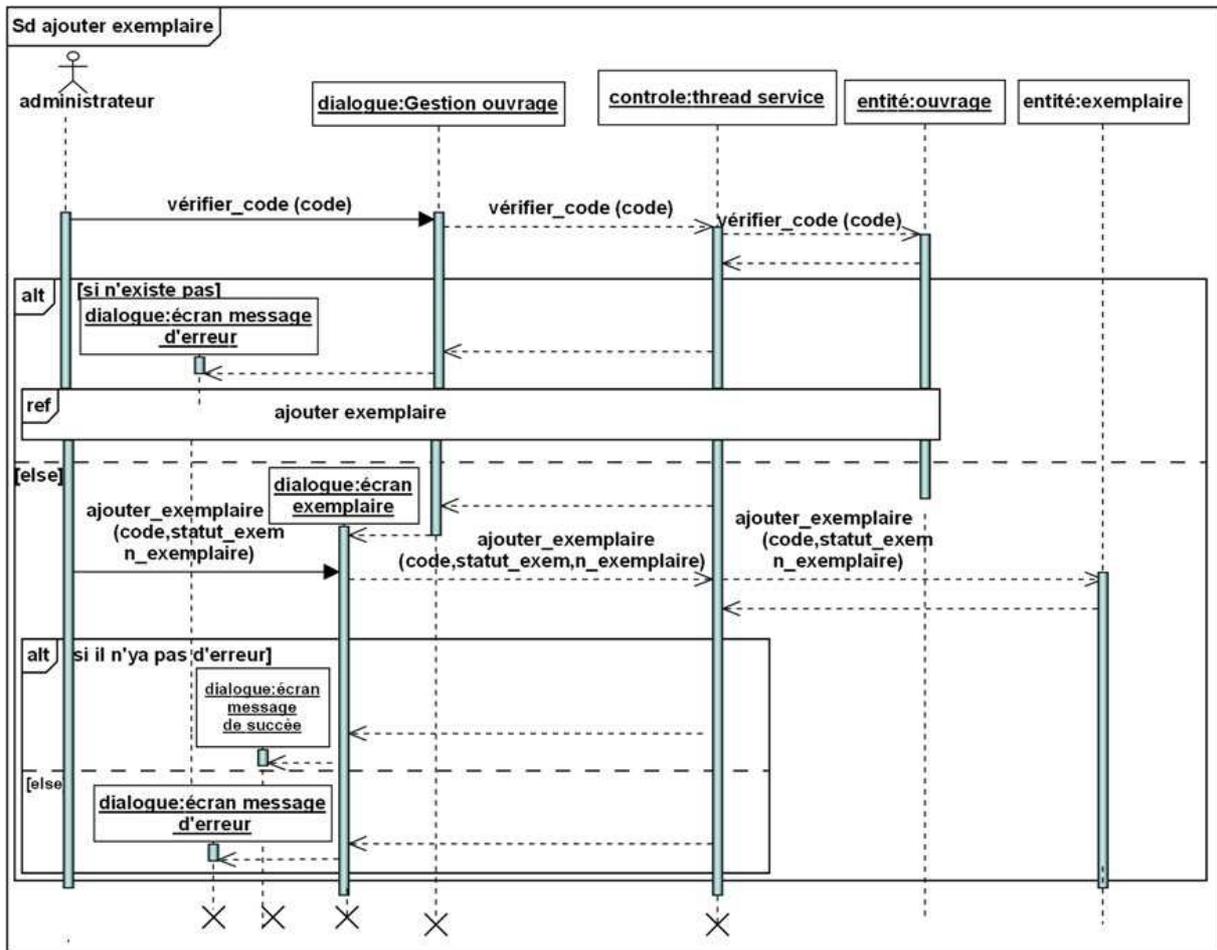


Figure IV.4: Diagramme de séquence d'interaction du cas ajouté exemplaire.

2.5-Modifier exemplaire

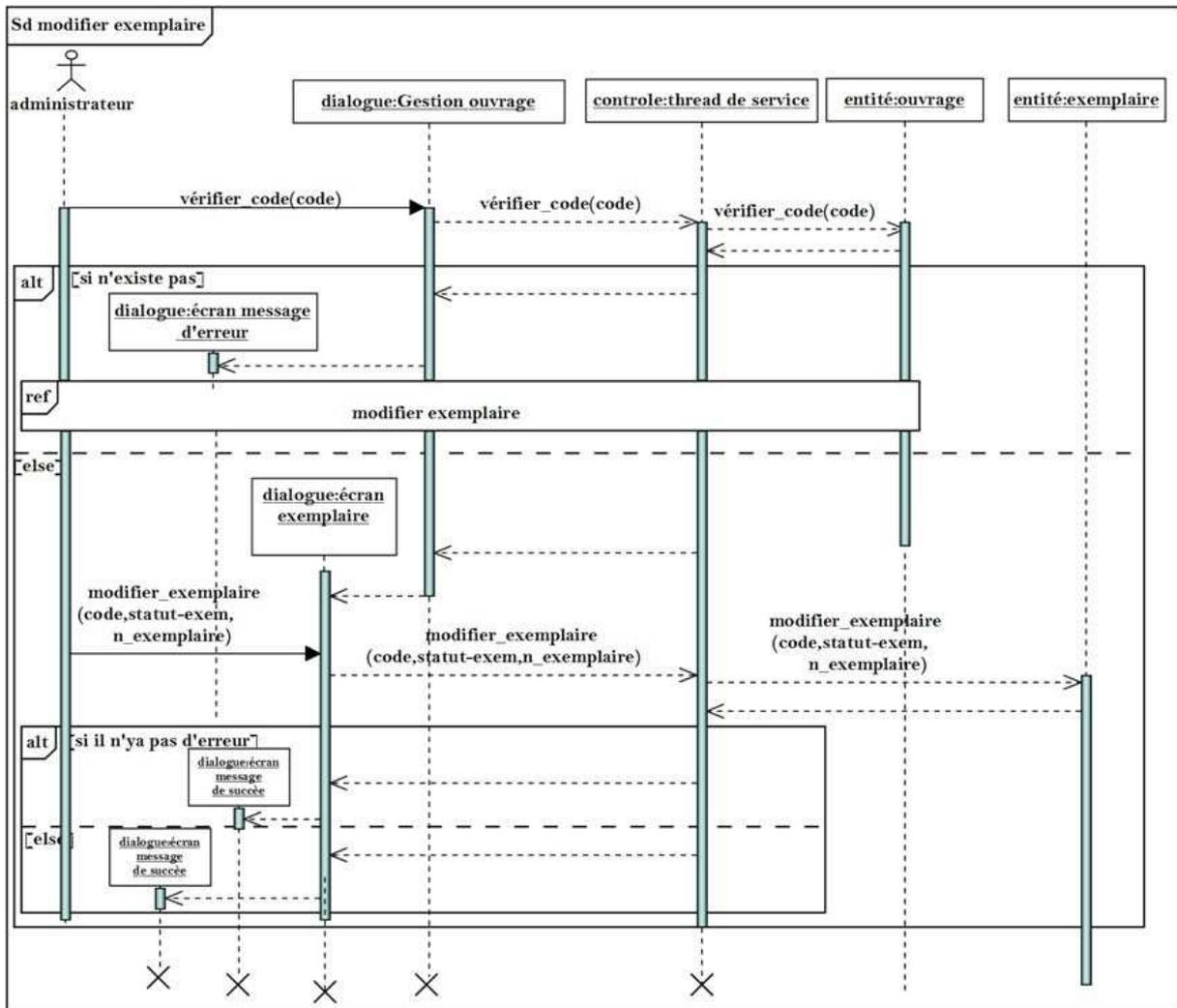


Figure IV.5 : Diagramme de séquence d'interaction du cas modifié exemplaire.

2.6-Supprimer exemplaire

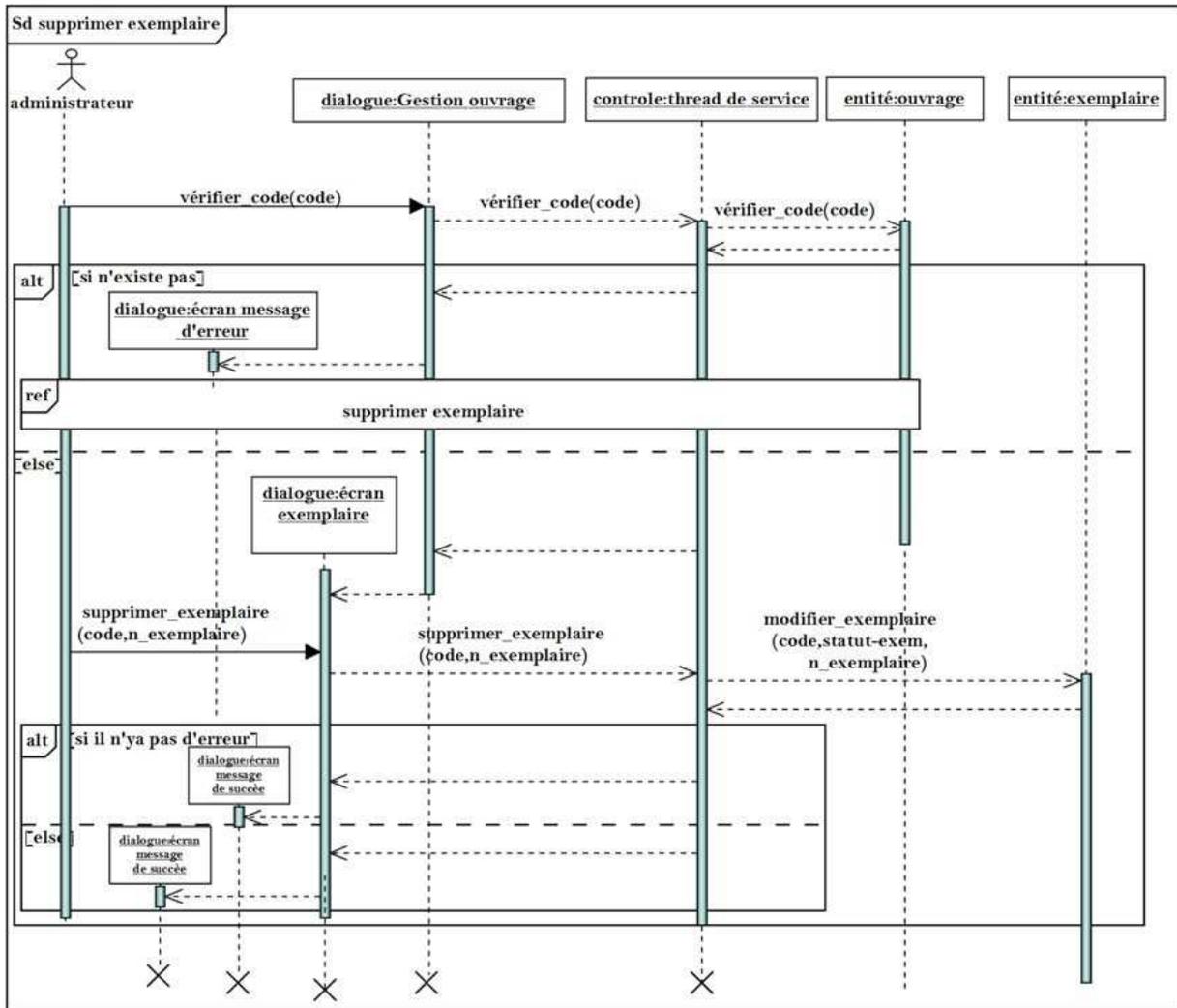


Figure IV.6 : Diagramme de séquence d'interaction du cas supprimé exemplaire.

2.7-Ajouter adhérent

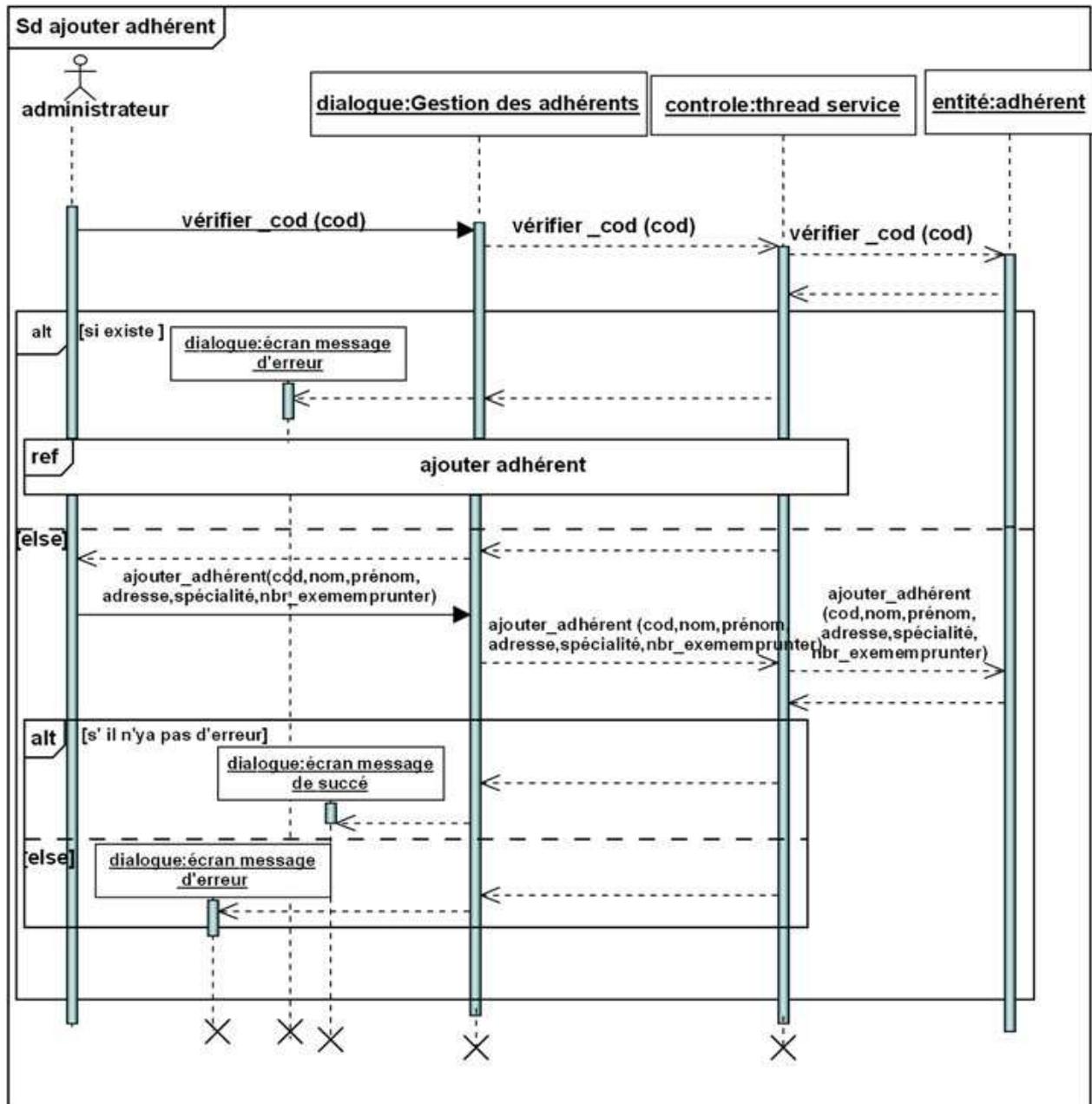


Figure IV.7: Diagramme de séquence d'interaction du cas ajouté adhérent.

2.8-Modifier adhérent

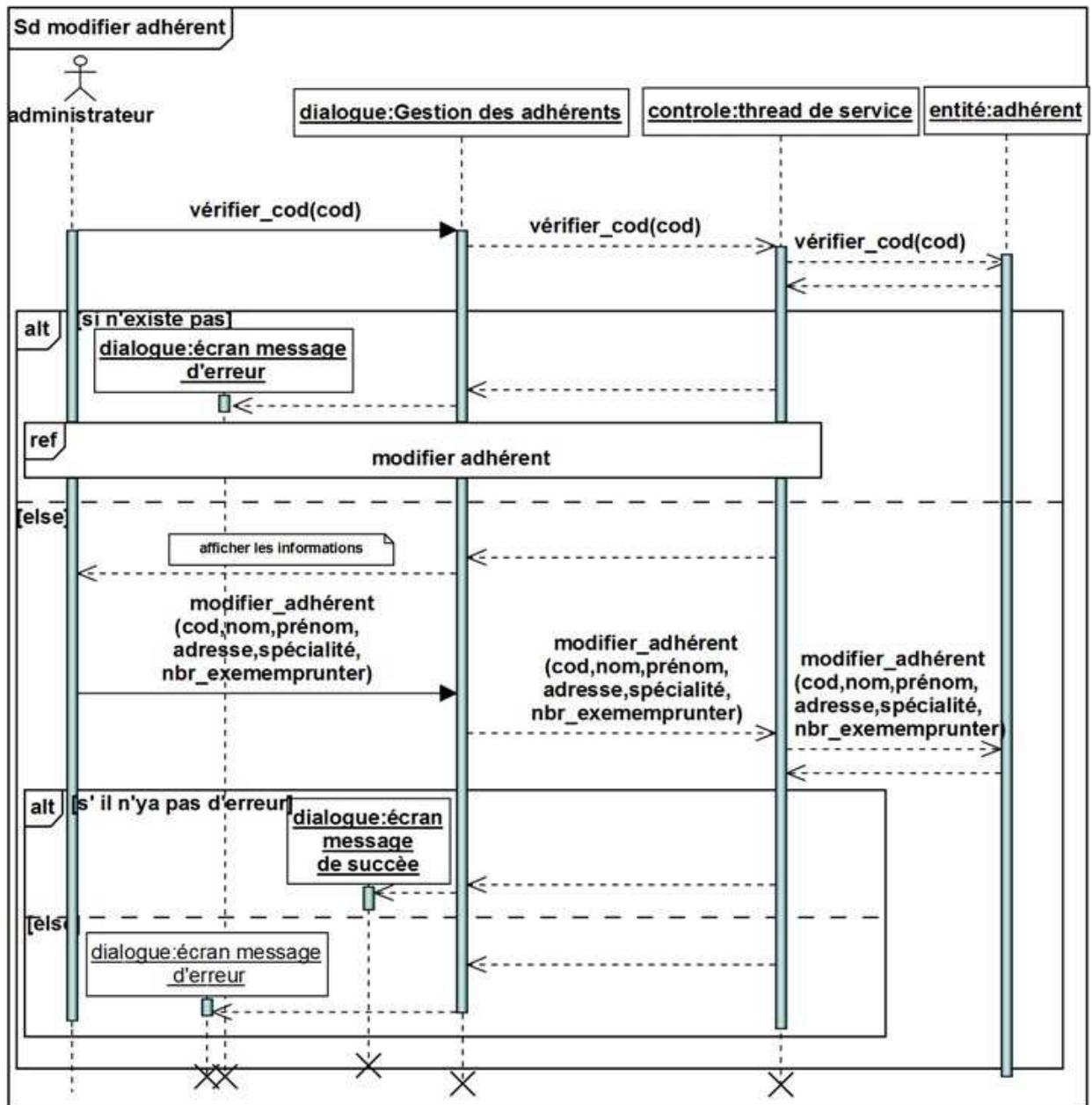


Figure IV.8 : Diagramme de séquence d'interaction du cas modifié adhérent.

2.9-Supprimer adhérent

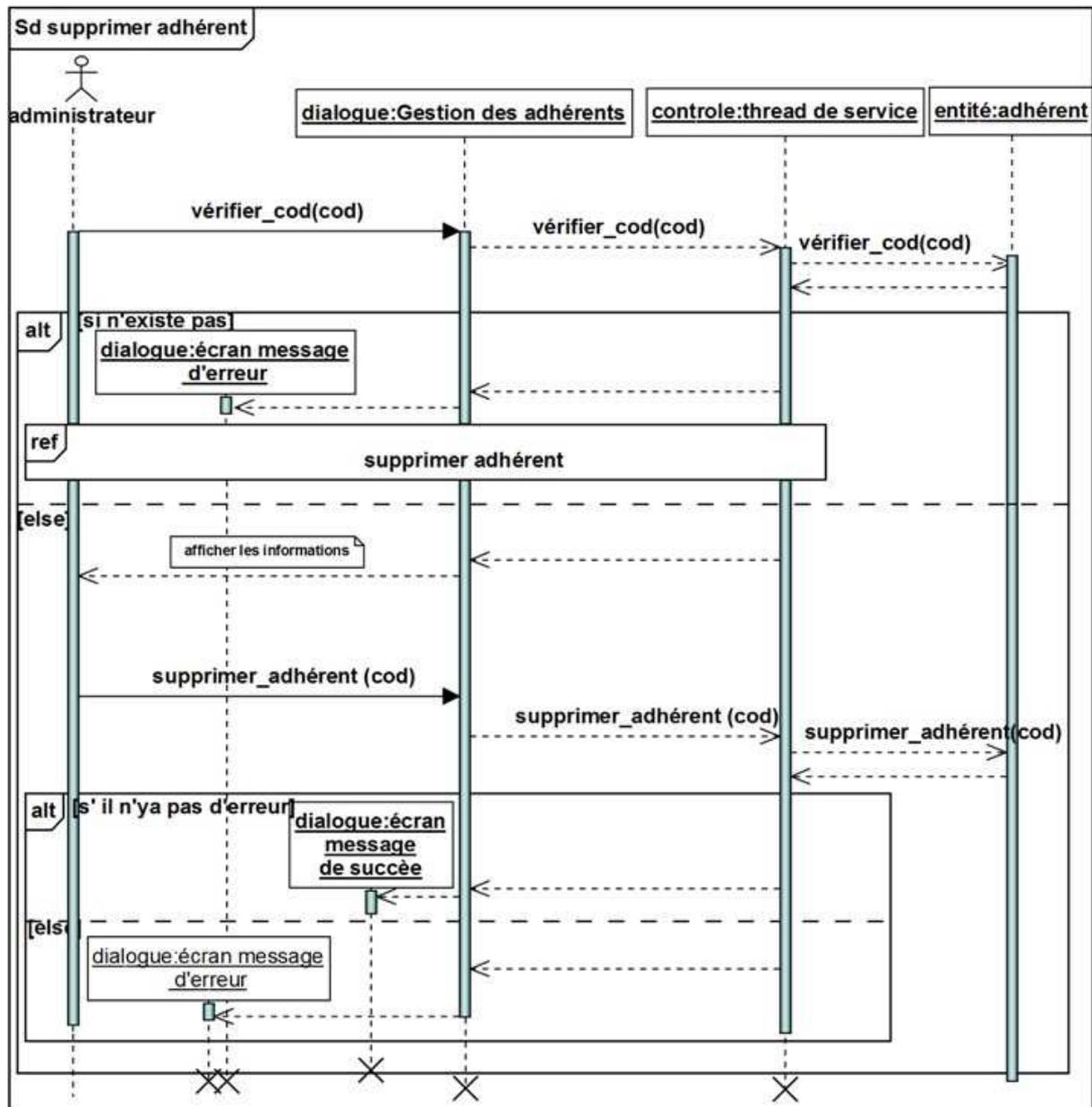


Figure IV.9 : Diagramme de séquence d’interaction du cas supprimé adhérent.

2.10-Emprunter exemplaire

Vérification

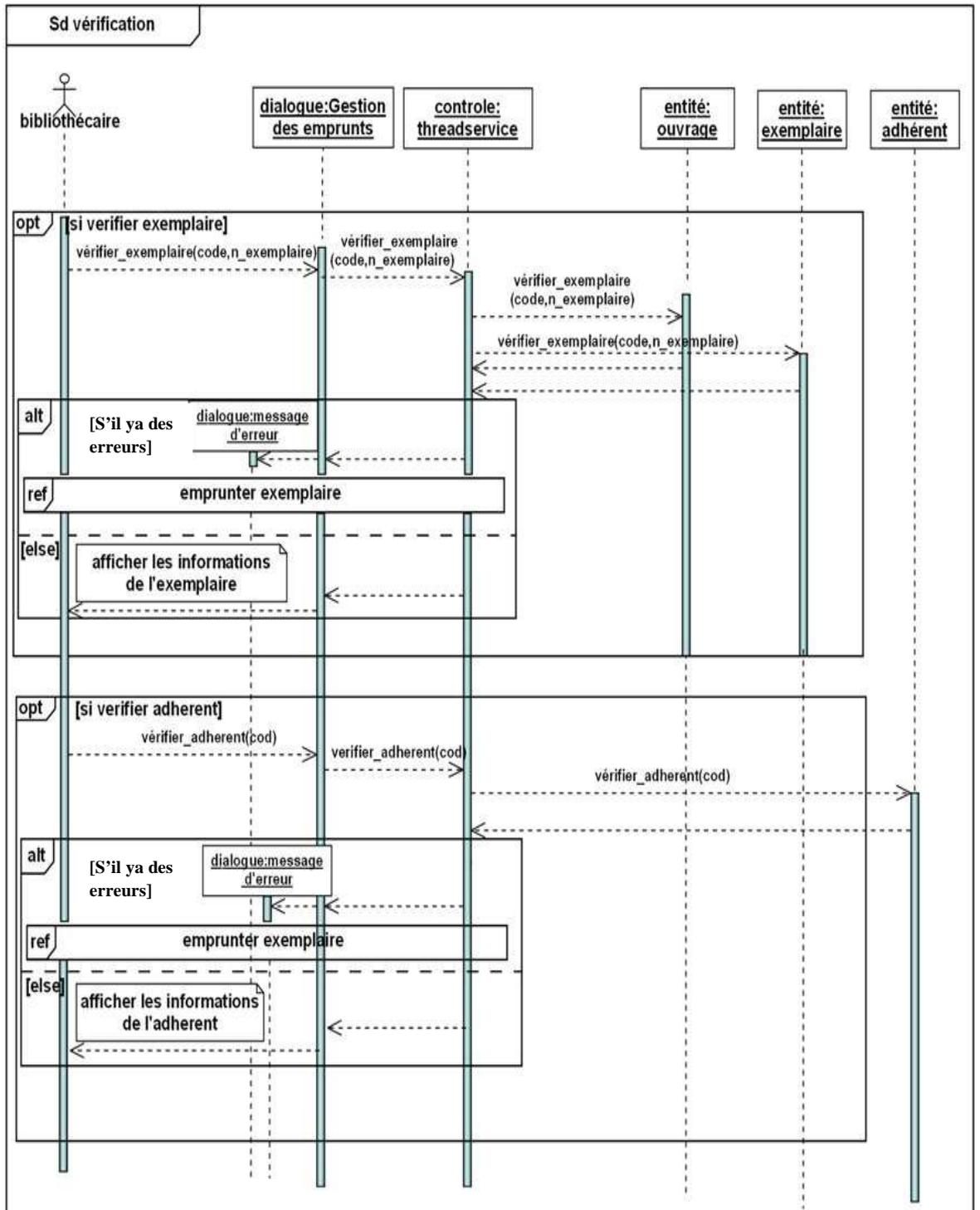


Figure IV.10 : Diagramme de séquence d'interaction du cas vérification.

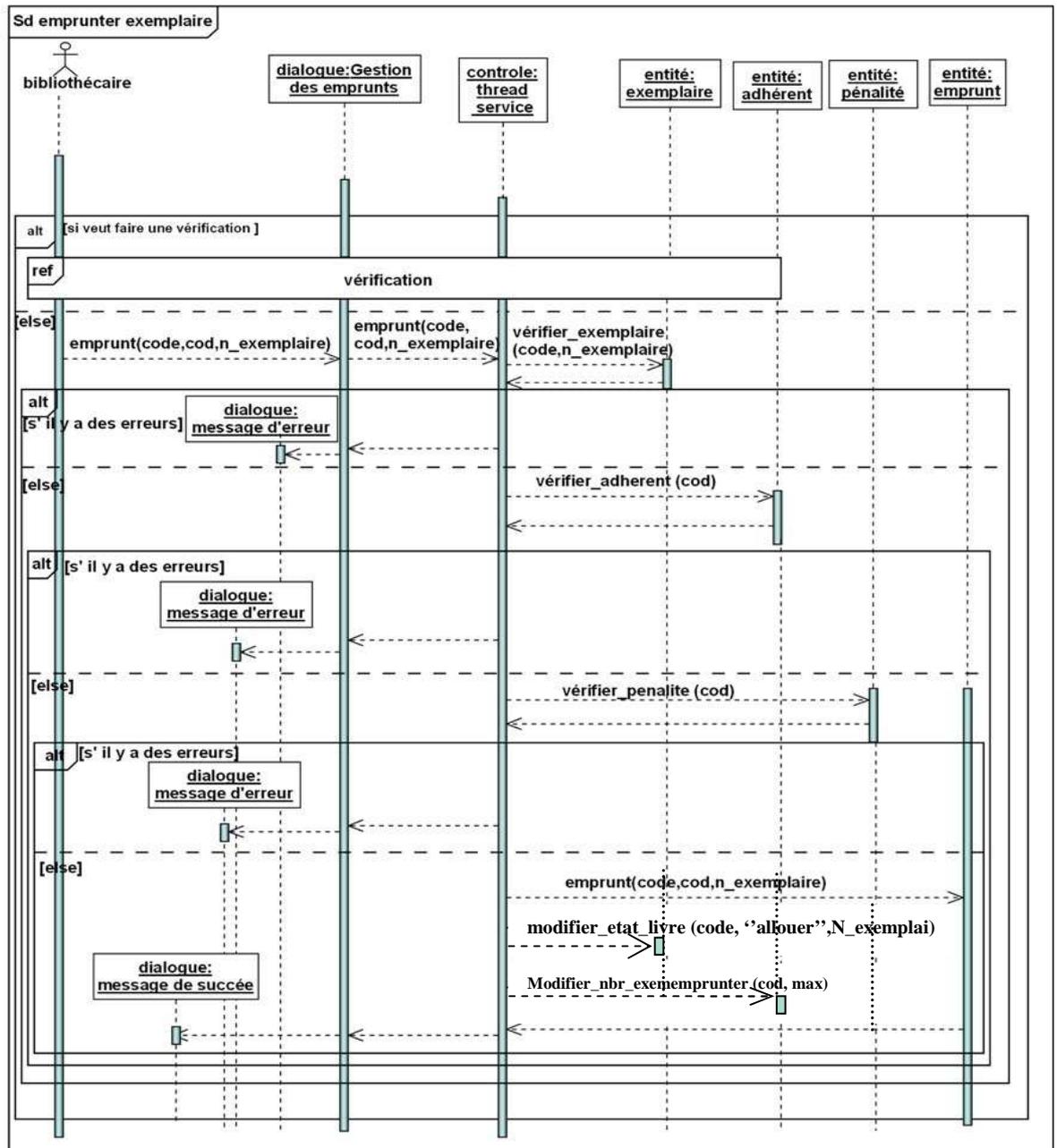


Figure IV.11 : Diagramme de séquence d’interaction du cas emprunté exemplaire.

2.11- retourner exemplaire

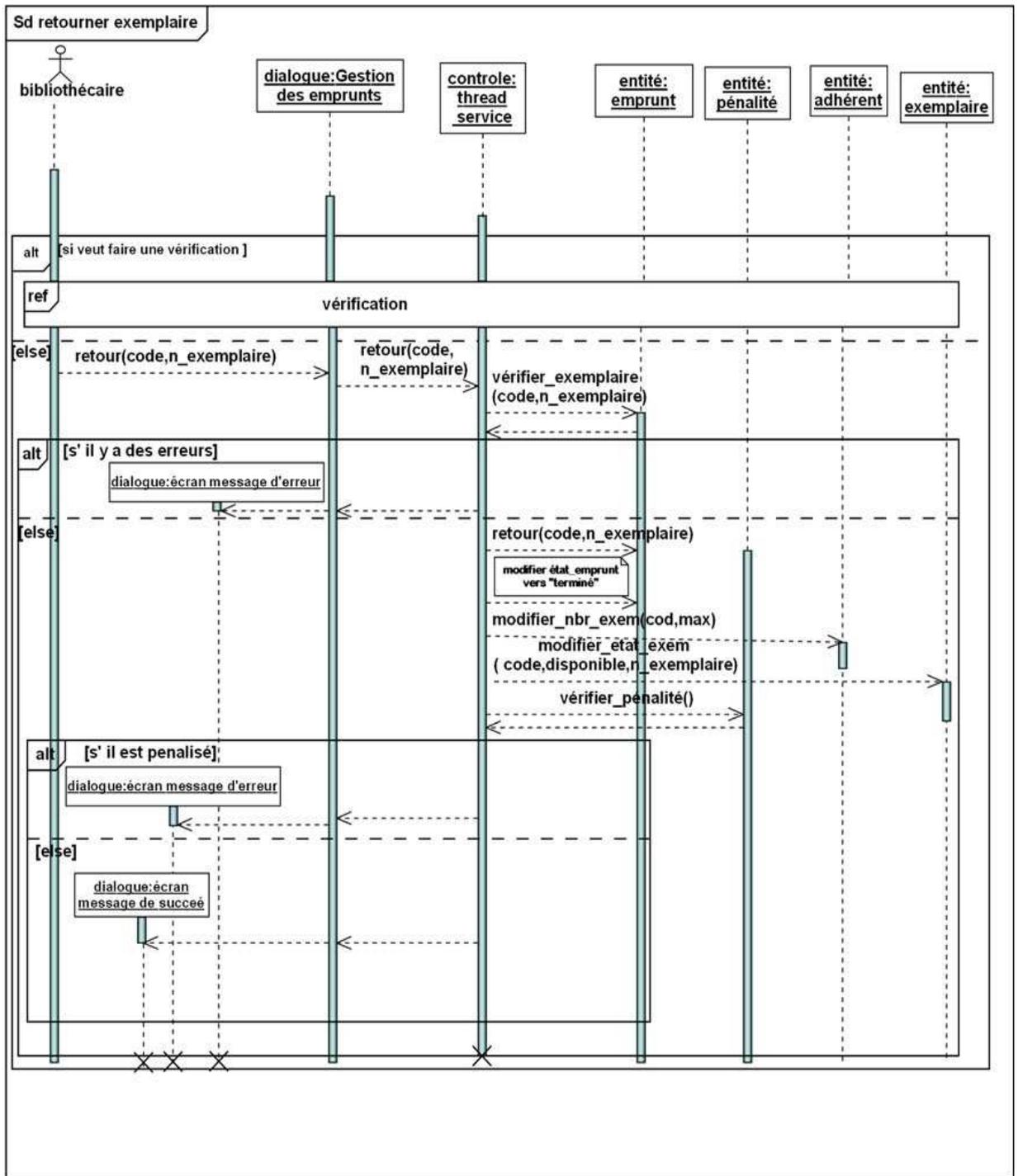


Figure IV.12: Diagramme de séquence d’interaction du cas retourné exemplaire.

2.12-Rechercher ouvrage

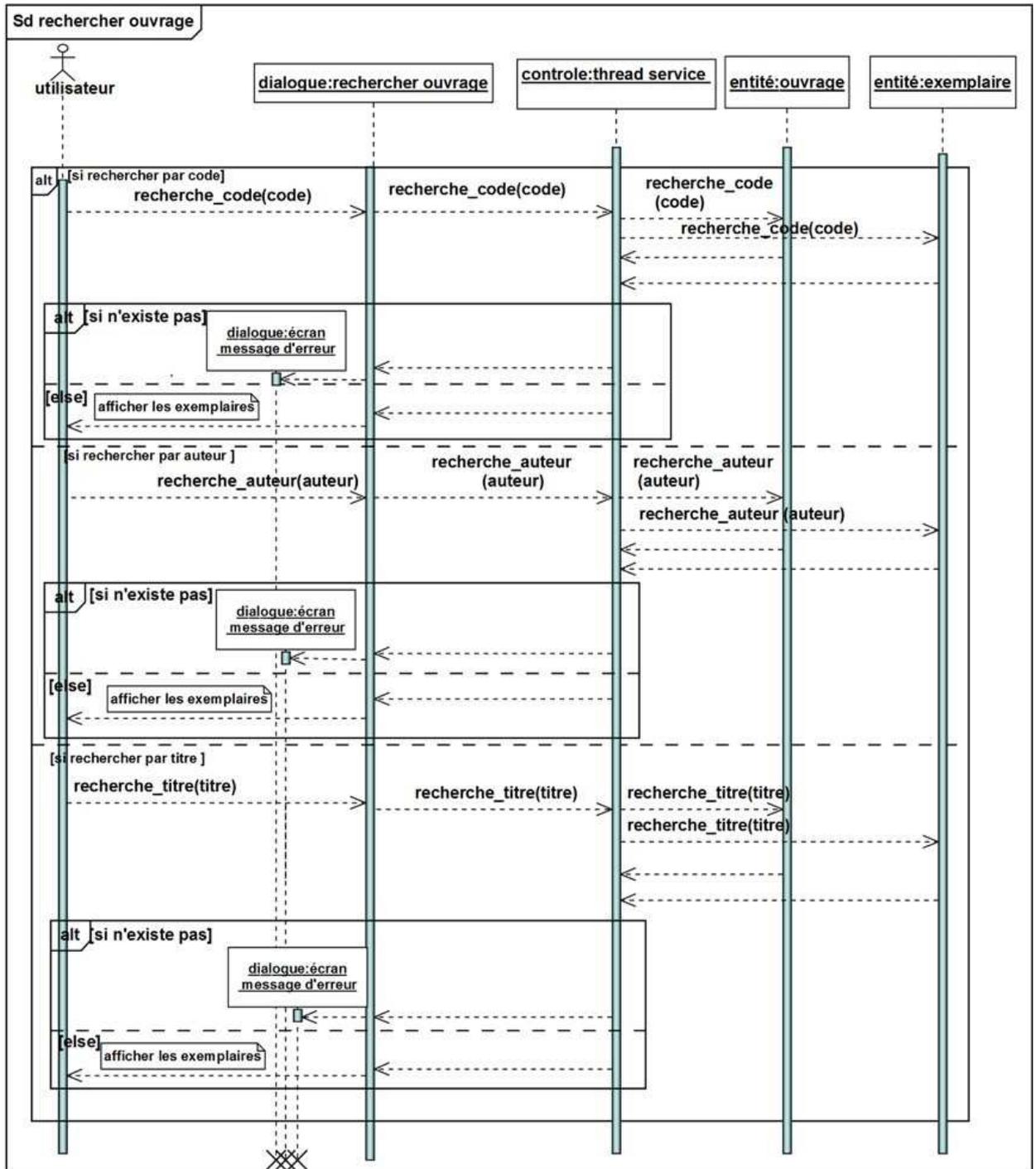


Figure IV.13 : Diagramme de séquence d'interaction du cas rechercher ouvrage.

2.13-ajouter pénalité

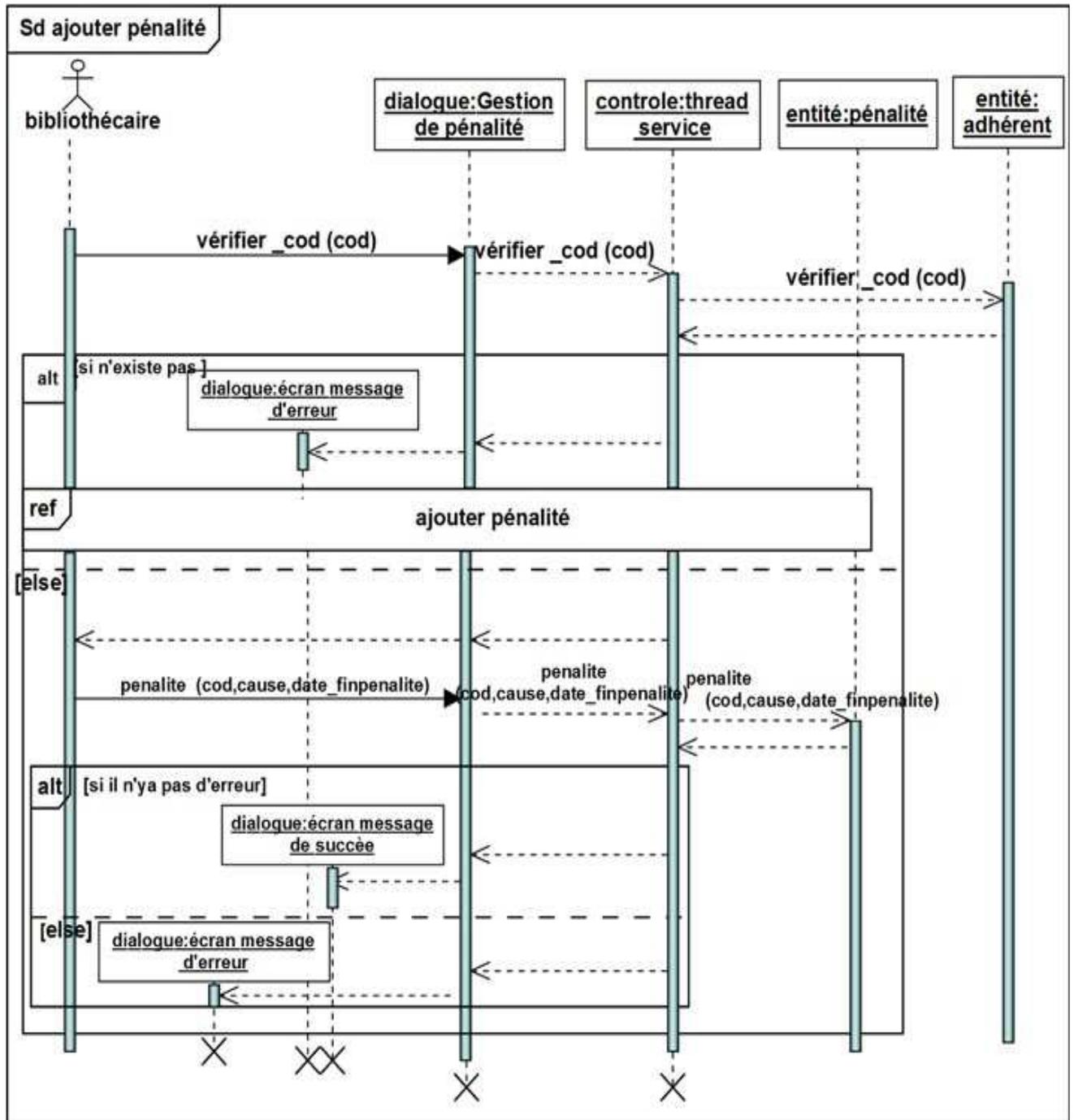


Figure IV.14 : Diagramme de séquence d'interaction du cas ajouté pénalité.

2.14-supprimer pénalité

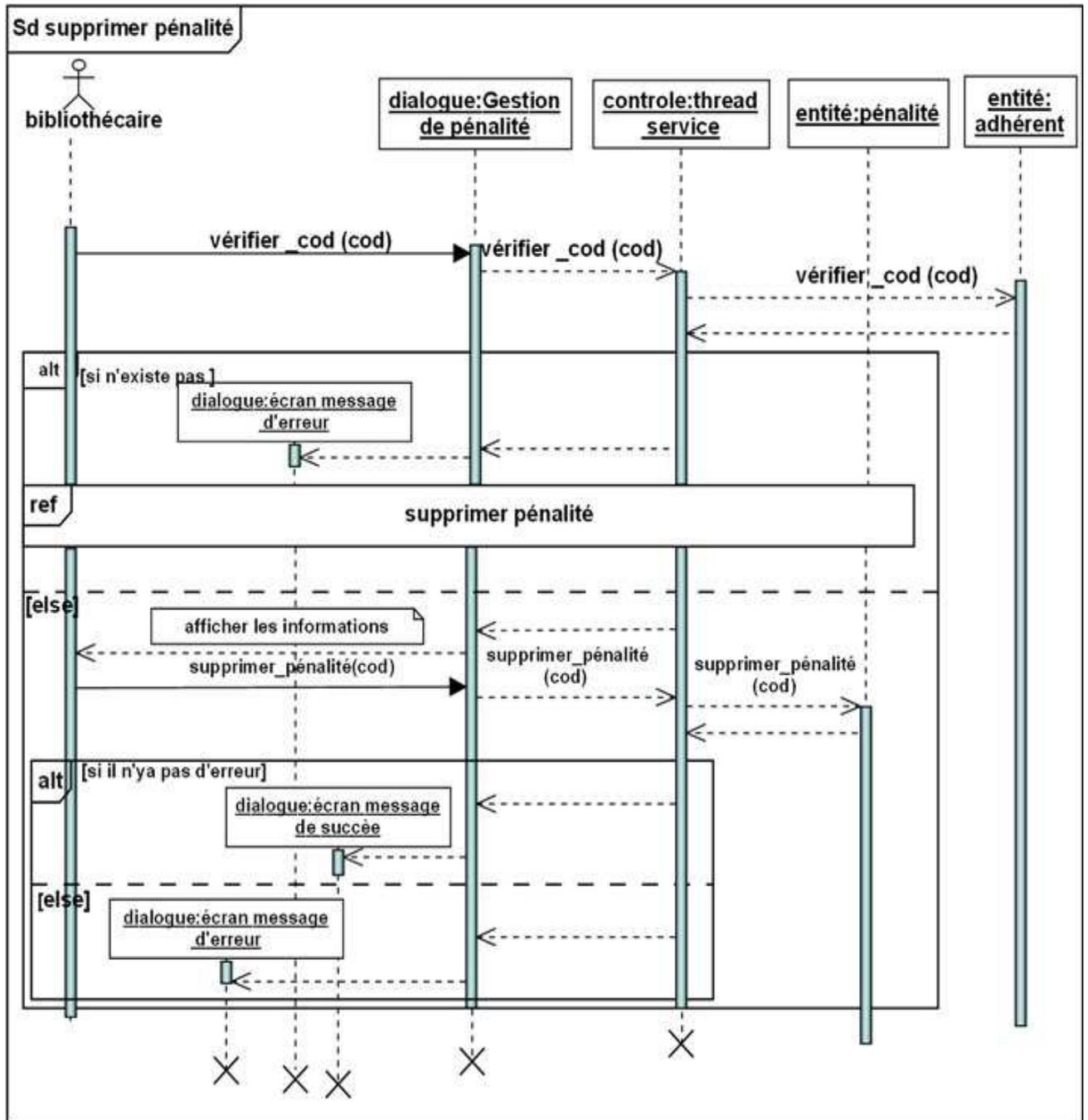


Figure IV.15 : Diagramme de séquence d'interaction du cas supprimé pénalité.

2.15-afficher liste des adhérents pénalisés

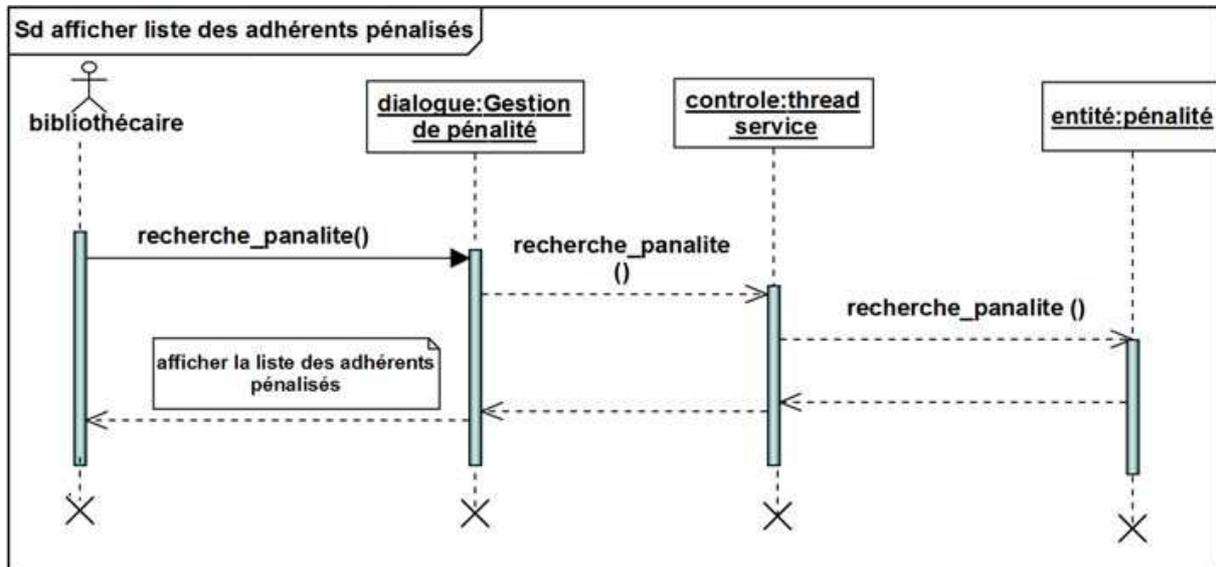


Figure IV.16 : Diagramme de séquence d’interaction du cas afficher liste des adhérents pénalisés.

3-Diagramme de classes de conception

L’objectif de cette étape est de produire le diagramme de classes qui servira pour l’implémentation, ce diagramme » permet de représenter la description et l’aspect statique du système en intégrant dans chaque classe.

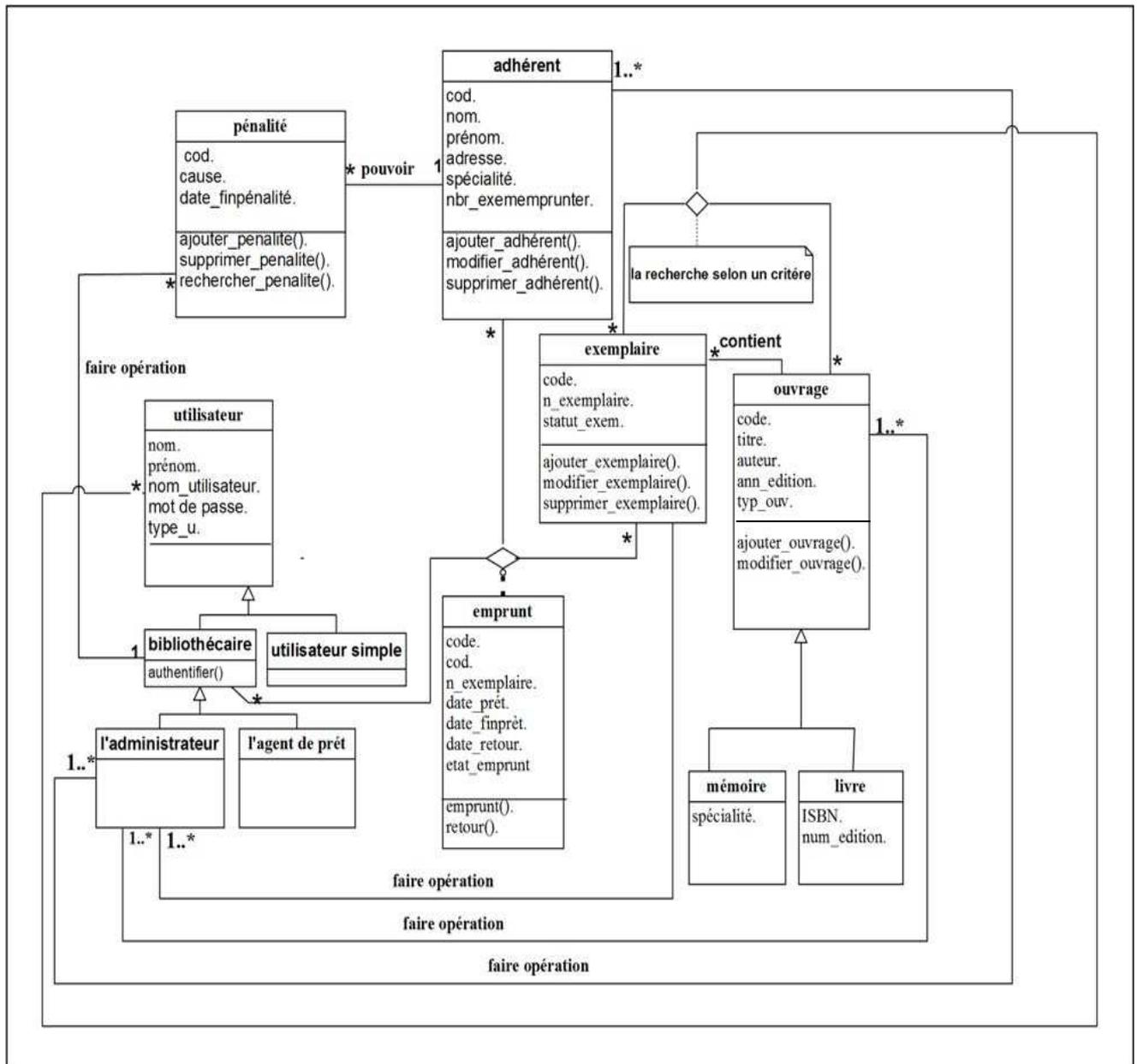


Figure IV.17 : Diagramme de class conception.

4 -Le modèle logique de données

Après l'application des règles de transformations et de passage du diagramme de classe vers le modèle logique de données, nous avons dégagé les différentes tables relatives au diagramme de classe, elles sont comme suit:

Utilisateur (nom, prénom, login, password, typ_u).

Ouvrage (code, ISBN, titre, auteur, num_edition, ann_edition, spécialité, typ_ouv).

Exemplaire (code, statut_exem, n_exemplaire).

Adhèrent (cod, nom, prénom, spécialité, adresse, nbr_exemprunter).

Emprunt (code, cod, n_exemplaire, date_prêt, date_fin_prêt, date_retour, etat_emp).

Pénalité (cod, cause, date_fin pénalité).

Remarque

Normalement la base **ouvrage** peut être décomposé en trois base qui sont :

Ouvrage (contient tous les informations communes entre les livres et les mémoires), **livre** (contient l'ISBN et le numéro d'édition) et **mémoire** (contient l'information spécialité). Cela permet de gagner l'espace disque, mais comme nous n'avons pas eu assez de temps en a utilisé une seule base (**ouvrage**) qui contient toutes les informations des livres et des mémoires, des champs dans la base vont donc rester vide.

5- Diagramme de déploiement

Les diagrammes de déploiement permettent de visualiser le système matérielle en même temps que le système logiciel et d'afficher les spécifications de l'interconnexion matérielle et logicielle. Le diagramme de déploiement présente dans la figure suivant montre la répartition physique des PCs clients connectés a un PC serveur, dont lequel nous souhaitons implémenter notre base de données.

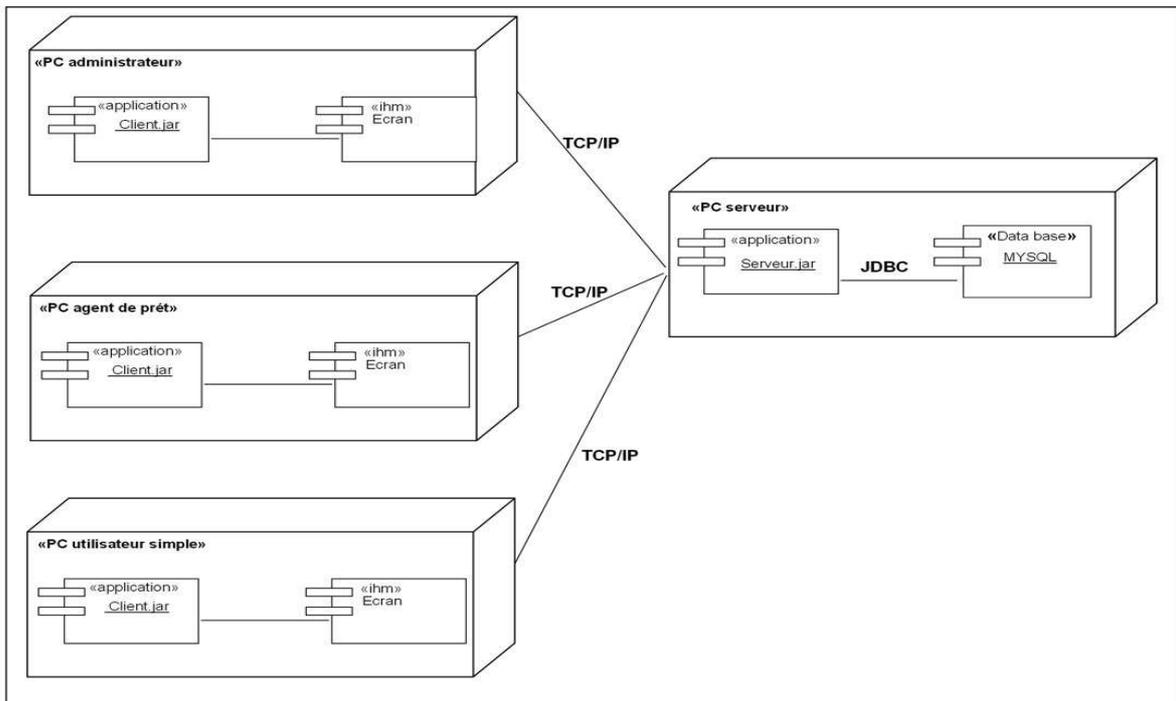


Figure IV.18 : Diagramme de déploiement.

Conclusion

Dans ce chapitre nous avons dégagé l'architecture statique de notre système et trouvé les tables qui nous allons utiliser. dans le chapitre suivant nous allons aborder la réalisation du système.

Chapitre V

Réalisation

Introduction

1- Environnement de travail.

2- Quelques interfaces de l'application.

Conclusion

Chapitre V: REALISATION

Introduction

Dans ce chapitre nous allons présenter les outils utilisés pour implémenter notre application et présenter quelques interfaces homme –machine.

1-Environnement de travail

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

1.1-Java

Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton de Sun Microsystems. Le langage Java a été introduit par la société SUN en 1995. Il possède de nombreuses caractéristiques ont cite :

- JAVA fonctionne comme une machine virtuelle (Indépendant de toute plate forme).
- JAVA est simple (pas de pointeur).
- JAVA autorise le multitâche (multithreading).
- JAVA peut être utilisé sur INTERNET.
- JAVA est gratuit.
- C'est un langage compilé : avant d'être exécuté, il doit être traduit dans le langage de la Machine sur laquelle il doit fonctionner.
- JAVA contient une très riche bibliothèque de classes(Packages) qui permettent de :
 - Créer des interfaces graphiques.
 - Utiliser les données multimédia.
 - Communiquer à travers les réseaux... etc.

1.2-NetBeans

NetBeans est à l'origine un EDI (environnement de développement intégré) Java. NetBeans fut développé à l'origine par une équipe d'étudiants à Prague, racheté ensuite par Sun Microsystems. Quelque part en 2002, Sun a décidé de rendre NetBeans open-source sa conception est complètement modulaire. Ce qui fait de NetBeans une boîte à outils facilement améliorable ou modifiable.

La licence de NetBeans permet de l'utiliser gratuitement à des fins commerciales ou non. Elle permet de développer tous types d'applications, qu'ils peuvent être gratuits ou payants.

NetBeans comprend un explorateur de bases de données qui supporte toutes les bases relationnelles pour lesquelles un connecteur JDBC existe (selon les versions des gestionnaires de bases de données) on peut citer : MySQL, Oracle ...etc.

1.3 -SGBDR(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.

1.4 - PHPMYAdmin

PHPMYAdmin est une application graphique qui peut gérer un serveur MySQL. Il permet aux l'utilisateur de :

- parcourir, créer, copier, ajouter, supprimer, renommer et modifier bases de données, tables, champs et index.
- Maintenir serveur, bases de données et tables en proposant une configuration serveur.

1.5-l'architecture 3 –tiers

Pour réaliser notre application nous avons utilisé l'architecture client/serveur 3-tiers elle est partagée entre les clients et le serveur tel que :

Le Client

Les clients sont les ordinateurs et les logiciels qui envoient des demandes au serveur par le biais des connexions réseaux.

Serveur

Le serveur est le logiciel et l'ordinateur qui l'héberge dont le rôle est de répondre de manière automatique aux demandes envoyées par les clients. Le programme serveur va créer un thread de service par chaque client connecté. Pour gérer les applications entre les clients et le serveur on utilise les sockets telle que :

Les threads

Dans le modèle des processus chaque processus est indépendant .chaque processus possède son propre espace d'adressage et ne peut pas manipuler les données des autres processus.il est intéressant dans autre modèle d'avoir plusieurs petits processus dans un même processus qui manipulent les données. Ces petits processus sont appelés **threads** (en français processus indépendants). Java permet d'exécuter plusieurs programmes indépendants les uns des autres. Ceci permet une exécution parallèle de différentes tâches de façon autonome, et permet aussi de partager le même espace d'adressage et manipuler les mêmes données

Les sockets

Les socquettes permettent de gérer les communications entre deux ordinateurs en utilisant le protocole TCP/IP.

Un socket est identifiée par :

- une adresse IP : c'est une adresse numérique identifier une carte réseaux d'une machine.

- un port : attribué automatiquement ou choisie par le programme, il identifier l'application.

Adresse de socket = adresse IP+port.

2- Quelque interface de l'application

Interface d'accueil : Cette interface d'authentification permettant aux différents utilisateurs d'accéder au système.



Figure V.1 : L'interface d'accueil.

L'interface de menu principal : Permet aux bibliothécaires de choisir les différentes tâches qu'ils veulent effectuer (gestion d'ouvrage, gestion des adhérents, gestion des emprunts.....) après authentification.



Figure V.2: L'interface de menu principale.

L'interface de gestion d'ouvrage: Cette interface permet à l'administrateur d'entrer le code de l'ouvrage puis de choisir les opérations qu'il veut effectuer sur un ouvrage ou un exemplaire d'un ouvrage (ajouter ouvrage, modifier ouvrage, ajouter exemplaire.....).

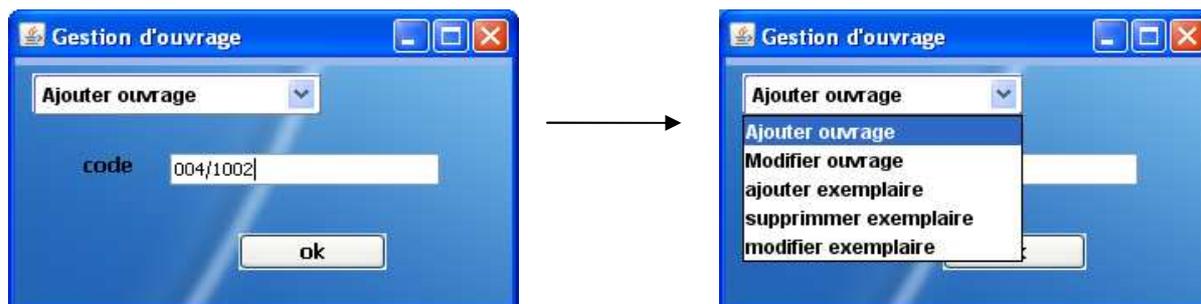


Figure V.3 :L'interface gestion d'ouvrage et d'exemplaire.

L'interface des ouvrages : Permet à l'administrateur d'effectuer les mises à jour des ouvrages (l'ajout et la modification).



Interface d'ouvrage

insérer tout les informations demander SVP:

Type de document Livres

Code 004/1200

ISBN

Titre UML2 pour la pratique

Auteur Jhon pierce

Num édition

Année édition 2002

Specialite

Retour arrière

enregistrer annuler

Figure V.4 : L'interface d'ouvrage.

L'interface des exemplaires : Permet à l'administrateur d'effectuer les mises à jour des exemplaires d'un ouvrage (l'ajout, la suppression et la modification).



Interface d'exemplaire

code 004/2048

etat_exem disponible

n_exemplaire 1 à 6

ok

Figure V.5 : L'interface d'exemplaire.

L'interface de gestion des adhérents : Permet à l'administrateur d'effectuer les mises à jour des adhérents (l'ajout, la suppression et la modification).



Figure V.6 : L'interface de gestion d'adhérent.

L'interface de gestion des emprunts : Permet aux bibliothécaires de faire l'emprunt ou le retour d'un exemplaire d'un ouvrage.



Figure V.7 : L'interface de gestion d'emprunt.

L'interface de gestion de recherche : Permet aux différents utilisateurs de faire la recherche des exemplaires selon un critère spécifié (par code, par titre, par auteur).

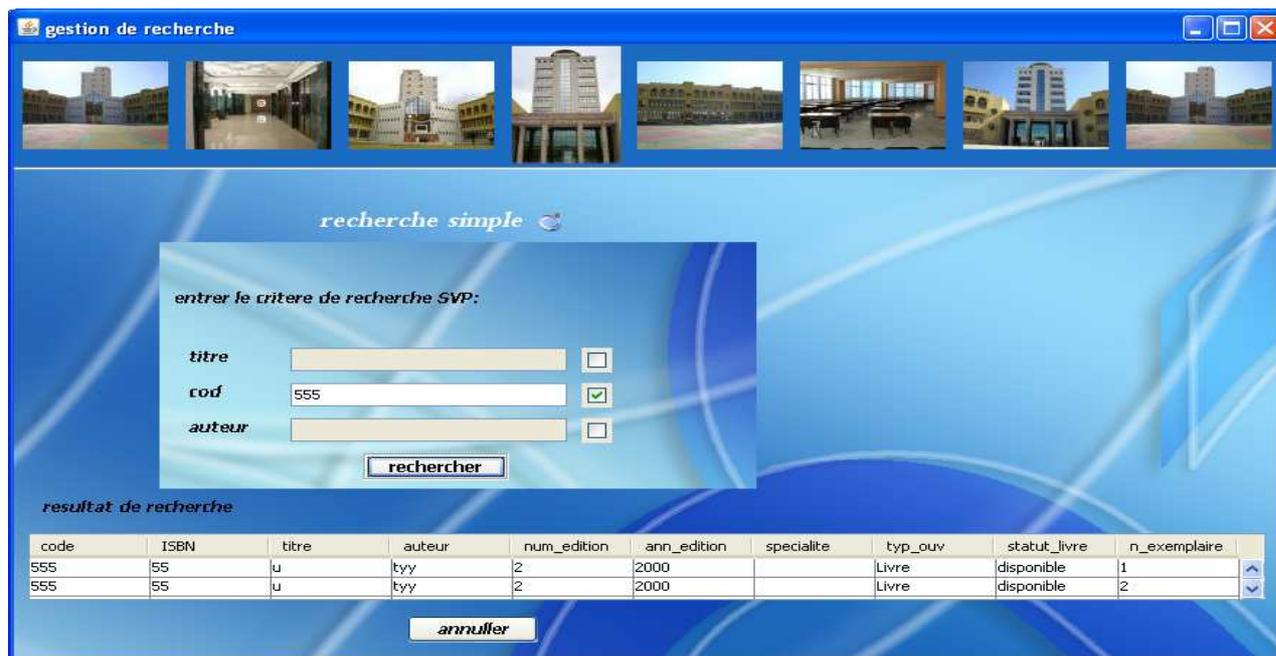


Figure V.8 : L'interface de gestion de recherche.

L'interface de gestion de pénalité : Permet aux bibliothécaires de faire la mis à jour d'une pénalité (l'ajout, la suppression) et l'affichage de la liste des adhérents pénalisés.



Figure V.9 : L'interface de gestion de pénalité.

Conclusion

Dans ce chapitre, nous avons décrit brièvement les interfaces de notre application et l'environnement utilisé de développment. Notre application est convivial et facile a utilisé, et permet de faire toutes les opérations demandé.

Conclusion générale

Dans ce mémoire nous avons développé une application client/serveur pour la gestion des emprunts dans la bibliothèque du centre universitaire de Mila, Notre projet était une bonne occasion pour sortir du cadre théorique et appliquer les connaissances acquises lors des études universitaires dans un environnement réel de travail qui nous a permis de s'initier dans le domaine professionnel et d'apprendre plusieurs attitudes telles que le travail en groupe dans une administration pour extraire les besoins des acteurs du système à mettre en œuvre ainsi que les règles de communication au sein d'une hiérarchie administrative stricte. Le profit essentiel été d'apprendre comment utiliser une architecture client/serveur et la faire intégrer dans notre application tout en respectant les contraintes temporelles.

L'objectif principal de notre projet été de permettre aux utilisateurs de l'application de faire toutes les taches de la gestion des emprunts des ouvrages dans la bibliothèque avec une interface claire et facile. Comme nous n'avons pas eu assez de temps, on n'a pas put implémenter toutes les opérations qui peuvent être effectués sur les ouvrages et les adhérents.

Finalement nous espérons que ce modeste mémoire aura une utilité dans le cadre de la formation des future étudiants du centre universitaire de Mila. Il pourrait bien sur être amélioré et complété en y ajoutant d'autres modules comme le calcul des statistiques et la gestion du personnel de la bibliothèque.

Bibliographie

- « Conception et réalisation d'une application web pour la gestion des stocks cas d'étude magasin de la faculté des sciences exactes de l'université de Bejaia », Mémoire de fin d'étude, Université de Bejaia, Juin 2010.
- « Conception et développement d'une application de gestion d'une bibliothèque » Mémoire de fin d'étude, Université de Tunis, 2008/2009.
- « Gestion Electronique des Documents », Mémoire de fin d'étude, université de Constantine ,2008.
- « UML2 », édition 2007-2008 Laurent Audibert, ([http : //www-lipn.univ-paris13.fr/Audibert/pages/enseignement/cours.htm](http://www-lipn.univ-paris13.fr/Audibert/pages/enseignement/cours.htm)).
- « Conception et réalisation d'une application web pour la gestion du prêt entre bibliothèques » Mémoire de fin d'étude, école nationale supérieur d'informatique, 2010.
- « UML2 pour la pratique (étude de cas et exercices corrigés)», 5 éditions, Eyrolles.