

Remerciements

*Nous tenons à exprimer notre profonde gratitude à
notre promoteur M. R. Soualah pour son aide et son
orientation pour la réalisation de ce projet.*

*Un grand merci à tous ceux qui ont contribué au
bon déroulement de ce modeste travail.*

Dédicaces

A mes chers parents

A mes deux chers frères

A ma chère petite sœur

A tous mes amis

Fatima

A la mémoire de mon père

A ma chère mère

A mes deux grandes sœurs et leurs petites familles

A mon frère et sa petite famille

A ma petite sœur et son mari

A tous mes amis

Meriem

Table des matières

Table des matières

Chapitre I : Etat de l'art

Introduction générale :	1
I L'organisme d'accueil :	3
I.1 Présentation de l'organisme d'accueil :	3
I.2 Organigramme de FGEI :	3
I.3 Le traitement de courrier au sein de FGEI :	5
I.3.1 Etude de l'existant :	5
I.3.2 Position du problème :	8
II Gestion électronique de courrier (GEC):	8
II.1 Définition :	8
II.2 Avantages de la GEC:	8
II.3 La mise en place d'un système GEC:	9
II.4. La dématérialisation :	9
II.4.1 Numérisation :	9
II.4.2 Le formatage :	14
II.4.3 Le traitement :	15
II.4.4 L'indexation :	17
II.4.5 L'enregistrement :	17
II.4.6 Le classement :	17
II.4.7 La recherche :	18
II.4.8 La consultation :	18
II.4.9 L'archivage :	18
II.4.10 La diffusion :	18
II.5 Processus métier :	19
II.5.1 Définition :	19
II.5.2 Gestion de processus métier (BPM) :	19
II.5.3 Avantages :	19

Table des matières

II.5.4 Modélisation des processus métier :	20
II.5.5 langages de modélisation :	20
II.5.6 Le workflow :	21
II.5.7 Moteur de workflow :	21
III L'open source :	22
III.1 Logiciel libre :	22
III.2 Du logiciel libre est née l'open source :	22
III.3 Etude comparative des logiciels GEC open source :	23
IV. Conclusion :	26

Chapitre II : Analyse et conception :

I. Introduction :	27
II. Méthodes d'analyse et de conception :	27
II.1 Les méthodes cartésiennes ou fonctionnelles :	27
II.2 Les méthodes systémiques :	27
II.3 Les méthodes objet :	28
II.4 Choix retenu :	28
II.5 Présentation générale de la méthode OMT:	29
III. Identification des acteurs :	30
IV. Les besoins utilisateurs :	31
V. Détermination des diagrammes des cas d'utilisation :	32
V.1 Diagramme de cas d'utilisation général du profil consultant :	32
V.2 Description textuelles des diagrammes des cas d'utilisation pour consultant :	33
V.3 Diagramme de cas d'utilisation général du profil utilisateur :	36
V.3.1 Description textuelle des diagrammes de cas d'utilisation pour utilisateur	37
V.4 Diagramme de cas d'utilisation général du profil administrateur	40
V.5.1 Description textuelles des cas d'utilisation pour le profil « administrateur »	41

Table des matières

VI. Les diagrammes de séquence :	43
VI.1 Diagramme de séquence « authentification »:	43
VI.2 Diagramme de séquence « ajout» :	44
VI.3 Diagramme de séquence « suppression» :	45
VI.4 Diagramme de séquence « modification»	46
VI.5 Diagramme de séquence « changer mot de passe »	47
VI.6 Diagramme de séquence « rechercher un courrier »	48
VII. Les diagrammes d'activités :	49
VII.1 Diagramme d'activité « s'authentifier »	49
VII.2 Diagramme d'activité d'ajout :	50
VII.3 Diagramme d'activité de modification :	51
VII.4 Diagramme d'activité de suppression.....	52
VII.5 Diagramme d'activité « rechercher un courrier »	53
VIII. Les diagrammes de collaboration :	53
VIII.1 Diagramme de collaboration d'authentification :	54
VIII.2 Diagramme de collaboration d'ajout :	55
VIII.3 Diagramme de collaboration de modification :	56
VIII.4 Diagramme de collaboration de suppression :	57
VIII.5 Diagramme de collaboration « rechercher un courrier » :	58
IX. Conception de la base de données :	58
IX.1. Diagramme de classe générale pour la base de données :	59
IX.2. Modèle logique de données :	60
XI. Conclusion :	62

Chapitre III : Réalisation

I. Introduction :	63
II. Etude du logiciel choisi « OpenCourrier » :	63
II.1 Installation :	63

Table des matières

II.2	Dépoilement :	65
II.3	Création de la base de données :	65
II.4	Connexion de l'utilisateur :	65
II.5	schéma des fichiers de l'application :	65
III.	Travail réalisé :	67
III.1	Modélisation des processus métier :	67
III.2	L'implémentation :	70
III.3	Les interfaces de notre systèmes	92
III.	Conclusion :	100

Listes des figures

Chapitre I : Etat de l'art

Figure I.1 : organigramme de FGEL	4
Figure I.2 : tableau synthèse de l'existant.....	5
Figure I.3 : interface Firewire.....	11
Figure I.4 : interface USB 2.0.....	12
Figure I.5 : interface SCSI.....	12
Figure I.6 : port parallèle.....	12
Figure I.7 : scanner à plat.....	13
Figure I.8 scanner à défilement.....	13
Figure I.9 : appareils photo numérique.....	14

Chapitre II : analyse et conception

Figure II.1 : Diagramme de cas d'utilisation général du profil consultant.....	32
Figure II.2 : Diagramme de cas d'utilisation général du profil utilisateur.....	36
Figure II.3 : Diagramme de cas d'utilisation général du profil administrateur.....	40
Figure II.4 : Diagramme de séquence « authentification ».....	43
Figure II.5 : Diagramme de séquence « ajout».....	44
Figure II.6 : Diagramme de séquence « suppression»:.....	45
Figure II.7 : Diagramme de séquence « modification»:.....	46
Figure II.8 : Diagramme de séquence « changer mot de passe » :.....	47
Figure II.9 : Diagramme de séquence « rechercher un courrier »:.....	48
Figure II.10 : Diagramme d'activité « s'authentifier ».....	49
Figure II.11 : Diagramme d'activité d'ajout :	50
Figure II.12 : Diagramme d'activité de modification :.....	51

Figure II.13 : Diagramme d'activité de suppression:.....	52
Figure II.14 : Diagramme d'activité « rechercher un courrier »:.....	53
Figure II.15 : Diagramme de collaboration d'ajout:.....	54
Figure II.16 : Diagramme de collaboration d'ajout :.....	55
Figure II.17 : Diagramme de collaboration de modification:.....	56
Figure II.18 : Diagramme de collaboration de suppression:.....	57
Figure II.19 : Diagramme de collaboration « rechercher un courrier »:.....	58
Figure II.20 : Diagramme de classe générale:.....	59

Chapitre III : réalisation

Figure III.1 : Diagramme Bpmn.....	68
Figure III.2 : exécution du diagramme BPMN.....	69
Figure III.3 : Interface « tableau de bord ».....	93
Figure III.4 : Interface : « courrier arrivé ».....	94
Figure III.5 : Interface « tableau de bord du courrier ».....	95
Figure III.6 : Interface « formulaire de recherche d'un courrier».....	96
Figure III.7 : Interface « formulaire d'ajout d'un courrier.....	97
Figure III.8 : interface « circulation ».....	98
Figure III.9 : Interface « formulaire d'ajout d'une circulation ».....	99

Introduction

générale

Toute organisation administrative acquiert et produit tout au long de son activité un grand nombre de courriers. La gestion et la conservation de ces courriers sont devenues deux activités essentielles et, lorsqu'elles sont bien menées, elles présentent un véritable atout pour améliorer ses activités.

L'avènement des nouvelles technologies de l'information et de la communication et le développement rapide du numérique ont remis en cause la fonction et la finalité de ces courriers et imposent une réflexion et une mise en place d'outils permettant de centraliser la gestion et la diffusion de ces derniers et d'y accéder d'une manière sécurisée depuis n'importe quel poste au sein de l'organisation.

Pour que ces courriers puissent être gérés par un système de gestion électronique de courriers « GEC », ils doivent être disponibles sous forme numérique ce qui engendre un processus de dématérialisation « voir page 9 ».

L'utilisateur a également de plus en plus besoin de suivre l'information et de la tracer afin de normaliser les processus et les procédures pour garantir une qualité de travail optimale et efficace. Cette évolution nécessite la mise en place d'une organisation pour un traitement du courrier rationnel et efficient. C'est pourquoi des solutions de travail collaboratif ont été conçues notamment le concept de gestion du processus métier « BPM ».

Parmi les raisons qui expliquent le passage du papier au numérique pour la majorité des organisations, nous citerons :

- Gain de temps et de réactivité grâce à la consultation électronique de tous les courriers instantanément contrairement à la consultation manuelle.
- Gain de place grâce à la diminution des archives papier.
- Meilleure qualité de travail, car il n'y a plus de risque de transmettre des courriers erronés ou non mis à jour.

C'est dans ce cadre que s'inscrit notre projet de fin d'étude ayant comme objectif d'implémenter une solution de gestion électronique de courrier pour la faculté de génie électrique et informatique.

Après une étude détaillée des besoins exprimés par le personnel responsable de gestion du courrier et grâce à des informations collectées nous avons élaboré un système permettant de répondre à l'ensemble de ses attentes.

Dans ce contexte, le travail réalisé comporte trois chapitres principaux à savoir :

- Chapitre 1 : donne une présentation générale du projet : l'organisme d'accueil et le système existant en citant ses défaillances suivi d'un exposé sur l'état de l'art du domaine de gestion de courrier.
- Chapitre 2 : présente l'analyse des différents besoins fonctionnels auxquelles doit satisfaire l'application ainsi que la conception détaillée du système.
- Chapitre 3 : décrit les tâches accomplies en titre de réalisation.

Chapitre I :

Etat de l'art

I L'organisme d'accueil :

I.1 Présentation de l'organisme d'accueil :

La Faculté de Génie Electrique et Informatique (FGEI) faisait partie du centre universitaire qui était implanté à Oued Aissi depuis 1978 jusqu'au 1984. En 1985, le centre universitaire était érigé en instituts nationaux d'enseignement supérieur(INES). De 1999 à nos jours, les instituts sont devenus des facultés dont FGEI qui a son siège depuis 2001 à Hasnaoua II (Bastos) elle comporte quatre départements qui sont :

- Département Informatique
- Département Electronique
- Département Electrotechnique
- Département Automatique

La faculté de génie électrique et informatique s'est orientée vers la généralisation de la mise en place du LMD.

Elle assure six semestres pour l'obtention du diplôme de fin d'étude pour la licence, et quatre autres semestres pour l'obtention du diplôme de fin d'étude pour le master (soit master 1 ou bien master 2) et six semestres pour le diplôme du doctorat.

La faculté comporte cinq Laboratoires de recherche qui contribuent significativement à la promotion et la valorisation de la recherche scientifique (4 projets CNPRU, 07 projets PNR), et à l'encadrement de Magistérants et Doctorants engagés dans le cadre des formations post-graduées habilitées (06 formations doctorales et 01 formation de Magister).

- Laboratoire des Technologies Avancés en Génie Electrique (LATAGE).
- Laboratoire de Génie Electrique (LGE)
- Laboratoire d'Analyse, et Modélisation des Phénomènes Aléatoires(LAMPA).
- Laboratoire de Recherche en Informatique (LARI).
- Laboratoire de Conception et Conduite des Systèmes de Production (L2CSP).

I.2 Organigramme de FGEI :

Figure I.1 : organigramme de FGEI

I.3 Le traitement de courrier au sein de FGEI :

I.3.1 Etude de l'existant :

Qui ? Quoi ? Où ? Quand ? Comment ? Pourquoi ? et Combien ? Synthétisent l'ensemble des résultats obtenus lors de notre stage pratique au sein de FGEI de manière à dégager des objectifs et des besoins pour la mise en place du système de gestion électronique du courrier.

	Questions posées	Réponses
Qui ?	1- Qui produit les courriers sortants ? 2- Qui réceptionne les courriers entrants ? 3- Qui est en mesure de les dépouiller ? 4- Qui consulte le courrier ? 5- Qui modifie ou supprime les courriers ? 6- Qui est en mesure de les conserver ?	1- Pour le courrier départ L'ensemble du personnel administratif peut être considéré comme producteur de courriers 2- Secrétariat du doyen 3- Secrétariat du doyen 4- Chacun est en mesure de consulter son courrier approprié 5- La secrétaire du doyen 6- Chacun est en mesure de conserver un certain nombre de courrier selon l'organisation interne
Quoi ?	1- Quel est le moyen de réceptionner le courrier ? 2- Quels sont les types du	1- un agent est responsable d'apporter les courriers au secrétariat 2- Les courriers produits sont

Etat de l'art et étude de l'existant

	<p>courrier entrant/sortant ?</p> <p>3- Quels sont les courriers prioritaires ?</p> <p>4- Quels est le format de ces courriers ?</p> <p>5- Quels sont les informations de ces courriers ?</p> <p>6- Quels types de courriers sont à conserver ?</p> <p>7- Quel est votre type de classement ?</p>	<p>de type papier</p> <p>3- les courriers signalés prioritaire</p> <p>4- lettre</p> <p>5- expéditeur, date, référence, destinataire, objet, signature</p> <p>6- les courriers considérés comme finis</p> <p>7- le classement se fait par ordre chronologique suivi du classement numérique, les courriers des enseignants sont classés par ordre alphabétique</p>
Où ?	<p>1- A quel endroit sont stockés les documents papier ?</p>	<p>1- le courrier est réparti sur l'ensemble de la structure, il est en général situé au plus proche des personnes qui en ont besoin</p> <p>2-</p>
Quand ?	<p>1- Quand est-ce que vous faites l'archive</p>	<p>1- Quand le courrier est considéré comme fini</p>
Comment ?	<p>1 - Comment se fait le tri des courriers ?</p>	<p>1- Au premier lieu : - lettres urgentes, fax, télégramme</p>

Etat de l'art et étude de l'existant

	<p>2- Comment vous repartez les courriers dans les différents services ?</p> <p>3- Comment se fait l'enregistrement des courriers ?</p> <p>4- Comment se fait l'indexation de ces courriers ?</p> <p>5- Comment se fait la recherche ?</p> <p>6- Comment se fait l'archivage de ces courriers ?</p>	<p>En second lieu :</p> <p>Lettres courantes, imprimés divers</p> <p>2-transmission des courriers se fait par les navettes internes</p> <p>3-l'enregistrement se fait dans un registre papier.</p> <p>4-L'indexation se fait en mentionnant les informations essentiels sur le courrier</p> <p>5-La recherche se fait par l'ordre de classement</p> <p>6-Le courrier considéré comme fini se classe dans des chemises puis dans boites à archive</p>
Pourquoi ?	<p>1- pourquoi voulez-vous automatiser la gestion des courriers?</p>	<ul style="list-style-type: none"> • Perte de temps, • l'augmentation continue des volumes de papier depuis des dizaines d'années avec les problèmes de gestion et de stockage des courriers

Combien ?	1- A combien estimez-vous le volume de ces courriers ? 2- Combien de personne sont concernées ?	1- 200 courriers par an. 2- l'ensemble de la structure est concernés
------------------	--	---

Figure I.2 : tableau synthèse de l'existant

I.3.2 Position du problème :

- Difficultés pour gérer leurs courriers car l'espace devient de plus en plus encombré par la quantité importante de papier engendrée par ces courriers. Cette contrainte conduit à la perte des courriers.

- Toute mise à jour survenue sur l'un des courriers enregistrés n'est détectée par les autres services concernés que par voie humaine, en conséquence elle ne se fait pas en temps réel et risque de transmettre des courriers erronés.

- Pour collecter des informations nécessaires sur les courriers déjà traités nous constatons une lenteur causée par la recherche manuelle malgré le bon classement.

- Notre mission consiste alors à concevoir et à mettre en place un système qui devra permettre d'uniformiser les pratiques pour le personnel et de ne plus se confronter aux problèmes actuels.

II Gestion électronique de courrier (GEC):

II.1 Définition :

La GEC représente un ensemble d'outils et de techniques qui permettent de dématérialiser, organiser, gérer, stocker et distribuer le courrier physique (document papier), sous forme électronique [1].

II.2 Avantages de la GEC:

- **Gain de temps :** un système GEC annule les recherches manuelles en intégrant des moteurs de recherche interne, ce qui facilite la consultation des courriers numérisés grâce à une indexation claire de ces courriers. Il permet aussi de diminuer les

déplacements pour faire circuler le courrier papier, en le diffusant entre les services répartis via un réseau.

- **Gain d'espace :** Numériser les documents sous format électronique, permet de gagner un espace non négligeable au sein des locaux.
- **Gain de productivité et travail collaboratif :** La GEC facilite le travail collaboratif car une fois mis en réseau, les membres de l'organisation pourront partager et consulter les courriers instantanément.
- **Gain économique :** Une mise en place d'un système GEC permet de réduire la masse de papier allouée à la gestion administrative, ainsi que le nombre des impressions, et des cartouches d'encre ou des toners.

II.3 La mise en place d'un système GEC:

Il existe des solutions différentes de gestion du courrier, pour mettre en place un système GEC adéquat, il faut :

- Identifier les outils et les fonctionnalités techniques nécessaires pour mettre en œuvre un processus de gestion électronique des courriers entrants et sortants.
- Evaluer ses besoins et définir sa solution de gestion électronique des courriers entrants et sortants.
- Evaluer les choix techniques pour la conception et l'exploitation d'une solution gestion électronique des courriers entrants et sortants.
- Mesurer le niveau de performance et la qualité d'un système de gestion électronique des courriers entrants et sortants.
- Evaluer les services à associer.

II.4. La dématérialisation :

Il est important de rappeler que la dématérialisation est l'une des facettes de la Gestion Electronique de Courriers (GEC), l'autre étant la gestion des processus. Nous allons citer un certain nombre d'étapes communes à tous les projets de dématérialisation :

II.4.1 Numérisation :

La numérisation des documents consiste à transformer, à l'aide d'un numériseur (scanner, appareil photo numérique, ...), un document papier, une page de texte par exemple, en une simple image numérisée.

Cette image brute est ensuite traitée afin d'en extraire le contenu sous une forme exploitable, en particulier pour l'indexation. On fait appel pour cela aux outils de reconnaissance de reconnaissance de caractères [2].

- **Scanner :**

Le scanner est un périphérique électronique qui permet de transformer un document papier ou une partie de document en une image numérique. Le document est soumis au balayage d'un rayon lumineux ; un capteur transforme la lumière reçue en un signal électrique qui est transféré à l'ordinateur, pour y être ensuite sauvegardé, traité ou analysé [3] , La numérisation d'un document est rendue possible grâce aux capteurs ultra sensibles et à la lumière rediffusée par les documents.

- **Caractéristique d'un scanner :**

Les scanners sont dotés de technologies et caractéristiques de pointe dans le but de faciliter la gestion documentaire quotidienne.

- **Reconnaissance optique des caractères :**

Les scanners de GEC sont en général équipés de logiciels de reconnaissance optiques. Ils traduisent en fichiers texte les images scannées contenant des données écrites. Ils allègent ainsi leur poids en les convertissant au format Word ou PDF.

- **Format de document numérisé :**

Selon leur taille, les scanners sont capables d'accueillir différentes taille de documents, généralement A4 (21 x 29,7 cm), plus rarement A3 (29,7 x 42 cm).

- **La résolution :**

Exprimée en points par pouce (notés ppp ou dpi pour dot per inch), la résolution définit la finesse de la numérisation.

- **Les trois modes :**

Les scanners actuels utilisés en GEC sont capables de numériser dans les trois modes suivants : noir et blanc, niveau de gris, couleur.

- **Le mode noir et blanc :**

Le mode noir et blanc, s'impose quand le document est destiné à un logiciel de reconnaissance de caractères (OCR). Dans ce mode, le scanner va décider individuellement pour chaque point s'il est blanc ou noir. Ainsi, lorsqu'il analysera l'image résultante, le logiciel d'OCR va rencontrer des lettres aux contours bien nets, son travail sera facilité et le résultat (le texte reconnu) sera de meilleure qualité. Il est le plus économique sur le plan de l'espace de stockage.

- Le mode niveau de gris :

Le mode niveau de gris est moins économique, mais permet un bon rendu des pages numérisés. Son rendu esthétique lui permet même de numériser assez fidèlement des images grâce à ses 256 niveaux de gris.

- Le mode couleur :

Le mode couleur repose sur le même principe que le niveau de gris, mais en superposant cette fois trois numérisations en trois couleurs différentes : rouge, vert, bleu. Il est très gourmand en mémoire de stockage, mais c'est de loin le mode qui donne les meilleurs résultats en termes de rendu des détails, de lisibilité et d'esthétisme.

Pour la numérisation de dossiers documentaires composés de documents textuels et d'images – par exemple un article de revue – ce sont les modes niveau de gris ou couleur qui sont préconisés.

- La précision /la linéarité:

C'est la capacité du scanner à restituer des lignes parallèles sans déformation, ni perte de distinction. Cette caractéristique est intrinsèque au scanner, gérée automatiquement, et n'est pas paramétrable par l'utilisateur

- La volumétrie quotidienne du scanner :

La performance d'un scanner s'exprime en ppm, ce qui correspond au nombre de pages numérisées à la minute.

Dans le cadre des projets de dématérialisation des courriers administratifs, il n'est en général pas exigé une volumétrie extrêmement élevée. Une valeur comprise entre 500 et 1000 pages/jours est ainsi souvent suffisante.

- **Interface** : il s'agit de la connectique du scanner. Les principales interfaces sont les suivantes :

Firewire : Il s'agit de l'interface de prédilection, car son débit est particulièrement adapté à ce type de périphériques ;



Figure I.3 : interface Firewire

USB 2.0 : Présent sur la totalité des ordinateurs récents, il s'agit d'une interface standard, conseillée si l'ordinateur ne possède pas de connexion firewire ;



Figure I.4 : interface USB 2.0

SCSI : Interface de prédilection pour le scanner à la fin des années 90, le standard SCSI est actuellement délaissé au profit du Firewire et de l'USB 2.0 ;



Figure I.5 : interface SCSI

Port parallèle : Lent par nature, ce type de connectique est de moins en moins utilisée et à éviter si l'ordinateur possède une des connectiques précédentes.



Figure I.6 : port parallèle

- **Types de scanner :**

Scanner à plat : C'est le scanner classique. Il est destiné à numériser des documents opaques. Couplé à un dos muni d'un éclairage. Il convient à de nombreuses utilisations, est facile à utiliser, et largement disponible.



Figure I.7 : scanner à plat

Scanner à défilement : ce type de scanner est principalement utilisé dans le monde bureautique. Le document est entraîné, comme pour un fax, c'est lui qui se déplace et non les capteurs ou un miroir.



Figure I.8 : scanner à défilement

Appareil photo numérique : Les appareils photo numériques combinent un scanner avec les optiques de l'appareil afin de former un outil pouvant produire des images de haute qualité. Bien qu'ils soient plus lents et plus difficiles à utiliser qu'un scanner à plat, les appareils numériques s'adaptent parfaitement à tous types de documents et d'objets.



Figure I.9 : appareil photo numérique

Autres scanners : Il existe d'autres types de scanners, tel que le scanner de photogravure, le scanner à tambour, le scanner à main...etc, mais les plus utilisés dans la GEC sont les modes que nous venons de présenter.

II.4.2 Le formatage :

Une fois la numérisation physique terminée, certains pilotes de scanner proposent plusieurs formats de fichiers pour l'enregistrement de l'image. En général, les formats sont BMP, TIF, PDF et JPEG.

- **Format fichier image :**

Consiste en fait en une photographie numérique du document : le codage informatique retranscrit l'image du document, avec le style de caractères, la mise en page et les illustrations. A ce stade, et avant un quelconque traitement de reconnaissance de caractères, il n'y a aucune possibilité d'intervention ou d'interrogation directement sur le texte numérisé. Il existe en fait deux modes de représentation numérique d'image, ou catégories de formats de fichiers images : le vectoriel et le bitmap.

Le bitmap s'applique en particulier à toutes les images numérisées par un scanner. Il convient non seulement aux images photographiques, mais s'applique également à la représentation de pages de texte. Dans ce cas, bien évidemment, il s'agit d'un document consultable – sans recherche en texte intégral - mais non modifiable – autrement dit non compréhensible par la machine, et non d'un réel document en mode texte – ou mode caractère. Pour passer à un mode texte, il faut utiliser la technologie dite de « reconnaissance de caractères ». Il existe deux principaux formats de fichier pour le bitmap : le TIFF et le JPEG.

Le **vectoriel** propose un champ de représentation infiniment plus limité, mais beaucoup plus flexible. Il convient notamment aux dessins au trait, aux graphiques et présentation, ainsi qu'au domaine graphique technique. Mais il n'est pas adapté à la représentation d'images avec un rendu photographique, Il existe deux principaux formats de fichier pour le vectoriel : le PostScript, le SVG18.

- **Le portable document format PDF :**

Le format PDF est un format de fichier qui préserve les polices, les images, les objets graphiques et la mise en forme de tout document source, quelles que soient l'application et la plate-forme utilisées pour le créer. Les fichiers PDF peuvent être créés avec des options personnalisées, tant aux niveaux de la compression des images et des textes, de la qualité d'impression du fichier, que du verrouillage (interdiction d'impression, de modification...).

Le PDF est dit « multi-format » puisqu'il peut aujourd'hui intégrer une couche de texte, des images, mais également des vidéos, du son et des formulaires.

- Il peut être enrichi de pièces jointes et de commentaires.
- Il organise le document en pages physiques au format papier A4, ce qui permet notamment d'imprimer directement le document.
- Il est constitué non seulement du corps du document mais aussi d'options de visualisation et de navigation, comme les signets par exemple, qui permettent de disposer de l'arborescence du document et ainsi d'accéder à ce dernier de manière hiérarchique et non linéaire.
- Il autorise une recherche en texte intégral, fonctionnalité inexistante pour le TIFF ou le JPEG.
- Il permet une forme de navigation via des liens hypertextes et des boutons de navigation.
- Il est possible de pré programmer un niveau de zoom pour faciliter la lecture à l'écran.

II.4.3 Le traitement :

- **La compression :**

Les fichiers numérisés peuvent parfois être très détaillée, et par conséquent la taille du fichier peut être très grande. Cela peut être un problème à la fois en termes d'espace disque et en termes de temps nécessaire pour télécharger ces fichiers. Par conséquent, la meilleure solution est souvent de compresser le fichier, de sorte à avoir les mêmes informations dans un format plus pratique.

Pour ce faire, il existe différents outils optimaux qui peuvent compresser les fichiers numérisés Selon les fins d'utilisation.

- **Reconnaissance de caractères :**

Le texte qui se trouve sur le document scanné est considéré comme de simples pixels noirs disposés sur un fond blanc. Il n'existe donc pas en tant que texte à proprement parlé. Impossible alors de l'utiliser.

Une reconnaissance de caractères est donc le processus qui va traiter la photo de texte pour transformer les pixels qui la compose en caractères alphanumériques. Le résultat de cette reconnaissance de caractères peut être un PDF, un document Word ou encore un tableau Excel, par exemple.

Il existe deux technologies de reconnaissance de caractères :

- **OCR (Optical Character Recognition) :**

Matériel ou logiciel de conversion d'un document image (codé en mode image) en un document texte (codé en mode caractère). Les documents traités sont de diverses catégories d'écriture (dactylographiée, numérique, manuscrite, bâton et cursive) ; ils peuvent être multi-écritures et hétérogènes (imprimés, dégradés, bruités). Plusieurs techniques de reconnaissance sont utilisées par les moteurs de reconnaissance optique de caractères, avec ou sans apprentissage.

Le principe technologique

Le principe d'une technologie OCR est de lire le document pour détecter les formes, puis les comparer à des bibliothèques de formes pour en faire correspondre un caractère. Si des erreurs surviennent lors de la reconnaissance d'un caractère, l'OCR compare alors le mot entier au contenu de son dictionnaire intégré pour en déduire l'équivalence la plus proche et ainsi corriger le caractère mal lu. Ainsi le texte pourra alors être segmenté selon l'information recherché.

- **ICR (Intelligent Character Recognition) :**

Est une technique avancée de reconnaissance optique de caractères, l'ICR ajoute cependant une technique de reconnaissance de caractères manuscrits et intègre un mécanisme d'apprentissage. Un moteur d'ICR est capable d'apprendre de nouveaux caractères et d'améliorer ses performances pour ses prochaines tâches de reconnaissances.

Autrement dit, si un caractère manuscrit qui représente un caractère "A" est identifié difficilement, il sera possible d'apprendre au moteur ICR qu'il s'agit d'un "A". Lorsque de nouveau une matrice représentant potentiellement un "A" se présentera, le système utilisera sa base de caractères enrichie par apprentissage pour en déduire que la matrice correspond au

caractère "A". L'ICR est également associé à des règles permettant au moteur de prendre des décisions en cas de doute. Les techniques d'ICR sont utilisées dans des solutions de lecture automatique des documents (LAD).

II.4.4 L'indexation :

Consiste à déterminer les termes ou expressions qui serviront de clés d'accès (métadonnées¹, mots-clés) pour retrouver le document dans le système de classement. Cela consiste en premier lieu à attacher des métadonnées (informations liées au document dont les objectifs principaux sont de le décrire et de le localiser en vue de faciliter et d'améliorer son repérage, sa gestion, son usage ou sa préservation) aux documents. Il y a deux types d'indexation.

- **L'indexation manuelle (par type):**

Elle offre une description formelle du document en utilisant ses métadonnées (type, auteur, titre, source, date, etc.) dont le vocabulaire est standardisé afin de permettre l'utilisation de ces métadonnées par le plus grand nombre d'outils de recherche.

- **L'indexation automatique (par mots-clés) :**

Elle vise plutôt le contenu du document pour faciliter les opérations de recherche. Il s'agit de répertorier les termes qui apparaissent le plus souvent ou de sélectionner les termes qui se rapportent au document dans une liste de mots prédéfinie (liste de mots liés par des relations de hiérarchie ou d'équivalence)

II.4.5 L'enregistrement :

L'enregistrement d'un document peut répondre à deux besoins complémentaires : la mémorisation du document pour des utilisations ultérieures (consultation, diffusion) ou le déclenchement de tâches.

Les supports de stockage doivent être adaptés le mieux possible avec le volume des documents. Ils doivent aussi, en fonction de la fréquence de consultation et de l'importance des données, offrir un temps d'accès fiable.

II.4.6 Le classement :

Cette opération consiste à ranger les documents dans un espace informatique accessible aux utilisateurs. Elle est facilitée par la mise en place d'un plan de classement

¹ C'est une donnée servant à définir ou décrire une autre donnée

commun à l'ensemble de l'organisation. On s'attachera alors à définir ce dernier de manière « naturelle », c'est-à-dire compréhensible par l'ensemble des utilisateurs.

II.4.7 La recherche :

La recherche dépend directement des possibilités d'indexation des courriers offertes par le système de GEC

La recherche pourra être aidée grâce à l'ensemble des mots-clés conseillés pour l'indexation des courriers lors d'enregistrement.

II.4.8 La consultation :

La consultation des documents peut s'effectuer de plusieurs manières :

- En se déplaçant au sein du plan de classement : on veillera à ce que l'ergonomie de la GEC facilite cette navigation ;
- En utilisant un moteur de recherche qui permet de retrouver les contenus gérés, au moyen de recherches structurées en utilisant les métadonnées ou de recherche en texte intégral.

II.4.9 L'archivage :

Consiste à enregistrer un document sous certaines conditions dans le système qui permettra d'en assurer la préservation pendant un certain laps de temps avec toute la sécurité requise. L'archivage se distingue donc du stockage par la notion de sécurité (la modification des documents est interdite, la destruction est également interdite sauf sous contrôle strict, toute action effectuée sur le document doit être tracée) afin que l'objet conserve sa valeur (notamment légale).

II.4.10 La diffusion :

Elle consiste à mettre en ligne les courriers numériques sur le réseau Internet ou sur un intranet via des serveurs web. Ces courriers intégrés dans l'application sont ainsi accessibles quasi immédiatement, depuis n'importe quel poste connecté au réseau de diffusion, et ce simultanément par plusieurs utilisateurs.

II.5 Processus métier :

II.5.1 Définition :

Un processus métier est un enchaînement d'actions réalisées par différents acteurs collaborant pour délivrer un résultat tangible et une valeur ajoutée métier pour l'entreprise [4].

Un processus métier est transverse, il s'appuie en général sur plusieurs structures et applications d'une organisation, voire de plusieurs organisations.

Caractéristiques d'un processus métier :

- Durée (moyenne) : un jour, plusieurs années.
- Fréquence d'exécution : 1000 exécutions par jour ou une exécution par an.
- Nombre d'utilisateurs : (par type d'utilisateur).
- Ressource utilisée (applications, référentiels ...).

II.5.2 Gestion de processus métier (BPM) :

Le BPM (Business Process Management) est l'ensemble des techniques, méthodes et outils permettant de modéliser, exécuter, contrôler et analyser les processus opérationnels ; en utilisant un modèle de données unique et partagé, en suivant des règles métiers préétablies et en collaborant avec d'autres systèmes de l'organisation en s'appuyant sur des acteurs qui peuvent être : des êtres humains, organisations, des applications, documents et autres sources d'information[5].

II.5.3 Avantages :

Une stratégie de gestion des processus métiers permet :

- D'organiser la masse de tâches que l'entreprise réalise chaque jour
- De diffuser la bonne information, au bon moment et à la bonne personne
- D'apporter un meilleur contrôle et pilotage de l'activité par le suivi et la traçabilité de l'exécution des processus métiers
- De mettre en valeur les dysfonctionnements de l'entreprise, d'en déterminer les causes et d'y remédier efficacement.

II.5.4 Modélisation des processus métier :

La modélisation est une phase primordiale dans la gestion des processus métier, dans laquelle, les concepteurs définissent, d'une manière abstraite ou détaillée, les processus métier ou redéfinissent un processus existant dans le but de l'améliorer.

Pour cela, des modèles et des langages sont utilisés afin de permettre de décrire les éléments de base d'un processus. En effet, cette phase est composée de trois étapes :

- **La première étape** permet de définir un processus de haut niveau, indépendamment des aspects techniques. Le concepteur y décrit seulement la structure, les ressources nécessaires et les interfaces du processus grâce à des langages graphiques comme UML et BPMN.

- **La deuxième étape** de la phase de modélisation est la configuration des processus abstraits où les détails fonctionnels du processus sont spécifiés. Le concepteur décrit de nombreuses informations techniques telles que : le format des messages échangés, les protocoles de transport utilisés, les services et les applications invoquées dans le processus. Dans cette étape, des langages d'exécution sont utilisés comme XPDL et BPEL.

- **La dernière étape** de cette phase est l'évaluation du processus exécutable en utilisant les techniques de simulation (BPS : Business Process Simulation) ou de la vérification formelle (comme les réseaux de Pétri) en vue de s'assurer du bon fonctionnement du processus avant son exécution. Dans cette étape, le processus peut être optimisé en utilisant les résultats de simulation pour définir des éventuelles améliorations. Le processus est ensuite redéfini et simulé ou vérifié de nouveau.

II.5.5 langages de modélisation :

Les composantes de la modélisation qui sont 'échange' et 'représentation' ont conduit à définir des langages communs, ou notations communes, qui deviendront des standards.

Les principaux langages sont :

- **UML (Unified Methode Language) :**

UML est devenu une référence lorsqu'il a été retenu par un groupement public, l'Object Management Group (OMG) en 1997, il est actuellement à la version 2.5 Ce langage de modélisation est le plus largement utilisé pour décrire les systèmes d'information.

- **BPMN (Business Process Model and Notation) :**

Il s'agit d'un standard pour la modélisation de processus qui définit une notation graphique commune à tous les outils de modélisation. Le BPMN permet de représenter un processus métier avec une notation graphique complète (éléments graphiques et diagrammes).

Il découple les informations métier des informations techniques et fournit une correspondance vers des langages d'exécution. Il s'agit d'une notation assez proche d'UML appliquée aux processus. Aujourd'hui, La spécification BPMN est devenue stable et elle est adaptée aux besoins du marché. Plusieurs éditeurs l'ont adoptée et introduite dans leurs outils.

- **YAWL (Yet Another Workflow Language) [8]:**

Ce langage est destiné à représenter les processus métier en étendant la syntaxe des réseaux de Pétri pour qu'elle supporte tous les patrons de flux de contrôle. En réalité, le YAWL est langage à la fois graphique et d'exécution, en effet, un moteur d'exécution a été implémenté pour ce langage à l'inverse des autres langages de définition de haut niveau, tel qu'UML, qui sont dédiés à la modélisation graphique des processus et qui doivent être traduits vers un langage d'exécution pour que le processus soit implémenté

- **XPDL [8]:**

Le langage XPDL (XML Process Definition Language) est proposé par Workflow Management Coalition (WFMC) qui regroupe plusieurs fournisseurs de système de workflow. XPDL est construit à partir du langage XML Extensible Markup Language, il permet de spécifier un processus métier en définissant les activités, les transitions, les partenaires et les interactions entre eux. XPDL est adopté par la plupart des moteurs de workflow.

II.5.6 Le workflow :

Le workflow qu'on peut traduire littéralement par « flux de travail » désigne deux réalités. Tout d'abord, le workflow désigne tout ensemble de tâches effectuées de manière séquentielle ou simultanément par un groupe de personne, afin d'atteindre un résultat donné ». Et d'autre part, ce concept désigne l'automatisation complète ou partielle des processus de travail à l'aide de logiciels informatiques « les applications de workflow » selon un ensemble de règle procédurale bien défini.

De façon plus générale, nous pouvons dire que le workflow est « l'ensemble des moyens mis en œuvre pour automatiser et gérer entièrement les processus d'une organisation »

II.5.7 Moteur de workflow :

Le moteur de workflow s'exécute sur la partie serveur, il permet d'exécuter des instances de workflow, Le serveur stocke et gère les bases de données contenant les modèles

de workflow, les carnets d'adresses des utilisateurs et des groupes. Le serveur gère aussi le moteur de workflow. Ce moteur est responsable de l'activation du workflow, du suivi et des notifications.

III L'open source :

A l'origine du concept, un logiciel « open source » signifie logiciel à code source ouvert. « Open source » s'est largement imposée dans le monde professionnel mais également dans le milieu universitaire. Depuis ses débuts, le champ de l'open source s'est profondément modifié, ce qui a conduit certains auteurs à lever des ambiguïtés sur sa définition

III.1 Logiciel libre :

Selon la FSF (Free Software Foundation), le logiciel libre est un logiciel fourni avec l'autorisation pour quiconque de l'utiliser, de le copier, et de le distribuer, soit sous une forme conforme à l'original, soit avec des modifications, ou encore gratuitement ou contre un certain montant [6].

La définition de logiciel libre a connu plusieurs révisions au fil du temps pour clarifier certaines subtilités. Mais de manière concrète, pour qu'un logiciel soit qualifié de libre, il doit respecter 4 critères de libertés :

- **La liberté d'exécuter** le programme comme vous le souhaitez, et à toute fin
- **La liberté d'étudier** le fonctionnement du programme, et le modifier comme vous le souhaitez pour votre ordinateur
- **La liberté de redistribuer** les copies que vous avez reçues
- **La liberté de distribuer** des copies de vos versions modifiées pour donner à toute la communauté une chance de profiter de vos modifications

Les libertés 2 et 4 nécessitent un accès au **code source**. Tout logiciel qui n'offre pas aux utilisateurs ces 4 libertés est qualifié de logiciel propriétaire, selon le mouvement du logiciel libre.

III.2 Du logiciel libre est née l'open source :

L'association entre logiciel libre et logiciel gratuit a constitué un obstacle à l'adoption commerciale, alors qu'un logiciel libre devrait être également disponible pour l'usage, le développement et la distribution de nature commerciale. En 1998, les partisans du logiciel

libre se sont donc réunis pour apporter les idées et les avantages du logiciel libre à l'industrie du logiciel commercial. Le mouvement du logiciel libre a donc été rebaptisé devenant ainsi l'open source, en mettant en avant les avantages du partage du code et la collaboration pour les entreprises.

L'open source repose donc sur les principes du logiciel libre, mais les fondateurs du nouveau mouvement ont défini dix conditions préalables que doit respecter un projet pour être qualifié d'open source :

- La redistribution doit être libre ;
- Le programme doit être distribué avec le code source, sinon il doit y avoir un moyen très médiatisé pour l'obtenir sans frais ;
- La licence doit autoriser les modifications et les œuvres dérivées, et doit leur permettre d'être distribuées sous les mêmes termes que la licence du logiciel original ;
- Pour maintenir l'intégrité du code source de l'auteur, la licence peut exiger que les œuvres dérivées portent un nom ou un numéro de version différent de ceux du logiciel original ;
- La licence ne doit discriminer aucune personne ou groupe de personnes ;
- La licence ne doit pas défendre d'utiliser le programme dans un domaine d'activité spécifique
- Les droits attachés au programme doivent s'appliquer à tous ceux à qui il est redistribué, sans obligation pour ces parties d'obtenir une licence supplémentaire ;
- La licence ne doit pas être spécifique à un produit ;
- La licence ne doit pas imposer des restrictions sur d'autres logiciels distribués avec le logiciel sous licence. Par exemple, la licence ne doit pas exiger que tous les autres programmes distribués sur le même support doivent être des logiciels open source ;
- La licence doit être technologiquement neutre.

III.3 Etude comparative des logiciels GEC open source :

Afin de choisir un système de gestion électronique du courrier, il fallait mener une étude comparative des systèmes GEC existants, pour cette fin nous avons cherché les systèmes les plus performants et connus et nous avons établi la grille d'évaluation suivante, qui met l'accent sur les critères les plus parlants :

Etat de l'art et étude de l'existant

Critères	Maarch Courrier	Cute Flow	CheckDocSql	Opencourrier
Définition	Maarch Courrier est une solution verticalisée de gestion des courriers, construite au-dessus du framework Maarch de gestion documentaire (GED) et d'Archivage Électronique (SAE).	CuteFlow est un logiciel de workflow simple. Il permet la circulation et la validation de documents entre utilisateurs tout en gérant les permissions de chacun.	checkDocSql est une solution qui permet de : -la gestion de documents de la correspondance entrante et sortante; -la gestion de la consultation des documentes et dossiers enregistrés.	openCourrier est un logiciel de gestion des courriers entrants et sortants dans une organisation
Points forts	-Paramétrable et adaptable à tous types d'organisations	Logiciel de travail collaboratif dans un cadre intra/internet, le système est basé sur du PHP/MySQL est donc facile à personnaliser.	- Robustesse -Etendue fonctionnelle	-paramétrable -Facile à configurer
Points faibles	Complexe à configurer	-il a quelques bugs	Complexe à configurer	La forme du Courrier se limite à deux types : PDF et image

Etat de l'art et étude de l'existant

fonctionnalités	-Numérisation rapide des documents Ajout, modification, consultation, suppression et recherche des courriers - notifications Gestion des utilisateurs	-Création de listes d'utilisateurs -Création de modèles de documents -Définition de la circulation du document	-gestion différenciée des droits des utilisateurs -Droits d'administration, d'encodage et de consultation différenciés selon différents profils d'utilisateurs -ajout, consultation modification, suppression des documents, dossier, organisation -recherche multicritères d'un dossier ou d'un document	-gestion des droits utilisateurs -gestion des profils utilisateur Gestion des services Ajout, modification, recherche et consultation des courriers
contraintes techniques	-XAMPP -PostgreSQL - SlickSVN	-PHP -MySQL -Serveur web pour PHP	-MySQL -PHP et le PHP module pour LDAP -EasyPHP -mysql et Apache existent -ModeliXe -Java jdk sdk -Tomcat	-PHP -MySQL -Serveur web pour PHP

Etat de l'art et étude de l'existant

			-Cocoon Jdbc (Mysql- connector)	
Documentation	Guide de configuration et d'utilisation Sur le site source de maarch	-Documentation incorporée au code source sur la configuration et l'utilisation	-Documentation incorporée au code source sur la configuration et l'utilisation	-Documentation incorporée au code source sur la configuration et l'utilisation
Ergonomie	Bonne qualité ergonomique	-Quelques bugs à l'affichage	-Pas de surcharge inutile -Contenu lisible	-Pas de surcharge inutile -Contenu lisible

IV. Conclusion :

Dans ce chapitre nous avons présenté théoriquement notre sujet pour mieux comprendre le système implémenté. Le chapitre suivant sera consacré à l'étude des besoins utilisateurs auxquels doit répondre notre application ainsi que sa conception.

Chapitre II :
Analyse
& conception

Analyse et conception :

I. Introduction :

Dans ce présent chapitre nous allons donner une brève introduction aux méthodes d'analyse et de conception ainsi que le choix retenu comme démarche adoptée. Nous étudierons ensuite, les besoins fonctionnels de notre système. La spécification formelle des besoins sera présentée par des diagrammes de cas d'utilisation. Enfin, nous allons entamer une partie cruciale du développement logiciel qui constitue un pont entre la spécification et la réalisation et qui comporte la conception de l'application en utilisant la modélisation UML.

II. Méthodes d'analyse et de conception :

Une méthode d'analyse et de conception permet la construction d'applications fonctionnellement conformes aux attentes des utilisateurs

Malgré la diversité des méthodes d'analyse et de conception, il est possible de les classer en quatre catégories :

- Méthodes cartésiennes
- Méthodes systémiques
- Méthodes objets

II.1 Les méthodes cartésiennes ou fonctionnelles :

Avec cette méthode, le système à étudier est abordé par les fonctions qu'il doit assurer plutôt que par les données qu'il doit gérer. Le processus de conception est vu comme un développement linéaire [7]. Il y a décomposition systématique du domaine étudié en sous-domaines, eux-mêmes décomposés en sous-domaines jusqu'à un niveau considéré élémentaire. SADT (Structured-Analysis-Design-Technique) en est un exemple.

II.2 Les méthodes systémiques :

Les méthodes systémiques s'appuient sur la compréhension du système d'information en tant que système s'insérant dans l'ensemble des autres systèmes de l'organisation. Le système d'information est couplé à la fois au système opérationnel de l'organisation dont il est une représentation et au système de décision dont il est le support. On met l'accent sur l'aspect

global, la mise en évidence d'éléments et de relations entre ces éléments. Comme exemple de méthode systémique nous pouvons citer MERISE, AXIAL ...

II.3 Les méthodes objet :

La modélisation objet consiste à modéliser un ensemble d'éléments du monde réel en un ensemble d'entités informatiques. Ces entités informatiques sont appelées objet .

Entre 1970 et 1990, de nombreux analystes ont mis au point des approches orientées objets, si bien qu'en 1994 il existait plus de 50 méthodes objet. Toutefois seules 3 méthodes ont véritablement émergées [7]:

- La méthode OMT de Rumbaugh
- La méthode BOOCH'93 de Booch
- La méthode OOSE de Jacobson

II.4 Choix retenu :

Le choix de la méthode se fera en prenant en compte des critères essentiels du système à concevoir. Pour des raisons d'efficacité, de rapidité et d'analyse complète nous opterons en effet pour la méthode OMT qui est une méthode objet utilisant le langage de modélisation UML

Notre choix de la méthode orientée objet s'explique du fait des avantages qu'elle offre.

- Cette approche a été inventée pour faciliter l'évolution d'applications complexes. Elle apporte l'indépendance entre les programmes, les données et les procédures parce que les programmes peuvent partager les mêmes objets sans avoir à se connaître comme le mécanisme d'import/export.
- De plus, l'approche objet permet une conception qui facilite la maintenance des applications (l'encapsulation des données et des traitements). Cela est dû au fait qu'il est possible par exemple de modifier une méthode sans toucher à son interface ou de créer une sous-classe héritée de celle qui nous intéresse.
- L'adoption d'une approche objet pour la conception s'appuie sur une méthode ou un langage efficace pour modéliser le système d'information. La qualité d'une conception est intimement liée à la méthode utilisée pour sa conduite

II.5 Présentation générale de la méthode OMT:

OMT permet de modéliser un système selon trois points de vue complémentaires, chacun capturant des aspects essentiels du système, tous requis pour une description complète:

- Le modèle objet est le point de vue des données.
- Le modèle dynamique est le point de vue du contrôle et des comportements.
- Le modèle fonctionnel est le point de vue des transformations apportées aux données.

- **Le modèle objet: (Le "Quoi?") :**

Le modèle objet saisit la structure statique du système, en montrant les classes d'objets, leurs attributs, leurs opérations et les relations entre les classes.

C'est le modèle le plus important. Il met l'accent sur le fait que le système est construit autour d'objets plutôt que des fonctionnalités.

- **Le modèle dynamique: (Le "Quand?") :**

Le modèle dynamique décrit les interactions entre les objets

- diagrammes de transition d'états
- contrôle (conditions de déclenchement d'une opération)

- **Le modèle fonctionnel: (Le "Comment?")**

Le modèle fonctionnel décrit les transformations appliquées aux données

- diagrammes de flots de données
- processus.

Ces trois modèles sont complémentaires et inter-reliés.

Le modèle objet est fondamental et préalable aux deux autres.

➤ **La démarche d'analyse et de conception d'OMT.**

Comme la plupart des méthodes orientées objet, OMT propose une démarche en trois phases.

- **L'étape d'analyse**

Elle permet d'analyser l'existant et les contraintes de réalisation, c'est une base d'accord entre client et développeur. L'analyste doit communiquer avec son client pour qu'il définisse ses besoins, Cette étape est une description de ce que le système doit faire.

- **L'étape de conception**

Phase qui suit l'étape d'analyse des besoins, Elle permet de présenter l'architecture d'ensemble du système, Elle a comme objectifs de répondre à la question « comment faire le système ? » et de décomposer de façon modulaire le système à mettre en place.

C'est le niveau logique, il définit les structures de données et les algorithmes d'implémentation des classes.

- **L'implémentation du Système.**

Cette phase consiste à la mise en œuvre des programmes, elle varie en fonction des langages et des systèmes de gestion de bases de données choisis conformément aux spécifications définies dans les phases précédentes. A la sortie il sera produit un modèle physique (modules implémentés mais non testés, documentation de programmation expliquant le code).

III. Identification des acteurs :

Notre Système fournit une interaction avec plusieurs types d'acteurs, ils sont définis comme étant des utilisateurs.

Chaque utilisateur est associé à un profil, les profils sont hiérarchiques et listés de façon croissante, c'est-à-dire que les permissions d'un profil sont valables pour les suivants profils, Nous identifions dans le cadre de ce projet trois profils :

- Le profil consultant : rôle des employés qui ont comme charge de consulter les courriers.
- Le profil utilisateur : est toute personne ayant en plus des droits de consultation, le droit d'enregistrer de nouveaux courriers arrivés/départs et de créer une circulation pour ces derniers.
- Le profil administrateur : l'administrateur est l'acteur chargé de paramétrage du système.

Pour cela nous allons spécifier ci-dessous les besoins que doit fournir le système pour chaque profil.

IV. Les besoins utilisateurs :

Le profil consultant : le système permet au consultant simple de :

- S'identifier pour accéder à son espace privé.
- Changer le mot de passe.
- Visualiser les courriers arrivés/départ de son service.
- Rechercher un courrier déjà enregistré.
- Visualiser l'archive.
- Editer la liste de tous les courriers de son service.
- Se déconnecter.

Le profil utilisateur : le système permet à l'utilisateur de :

- numériser un courrier.
- Ajouter et modifier un courrier arrivé/départ.
- Archiver les courriers.
- Ajouter et modifier une circulation pour un courrier.

Le profil administrateur : le système permet à l'administrateur de :

- Gérer les profils (ajouter, modifier, supprimer un profil).
- Gérer les utilisateurs (ajouter, modifier, supprimer un utilisateur).
- Gérer les droits utilisateurs (ajouter, modifier, supprimer un droit-utilisateur).
- Gérer les services (ajouter, modifier, supprimer un service).
- Gérer les types-correspondants (ajouter, modifier, supprimer un type-correspondant)

V. Détermination des diagrammes des cas d'utilisation :

V.1 Diagramme de cas d'utilisation général du profil consultant :

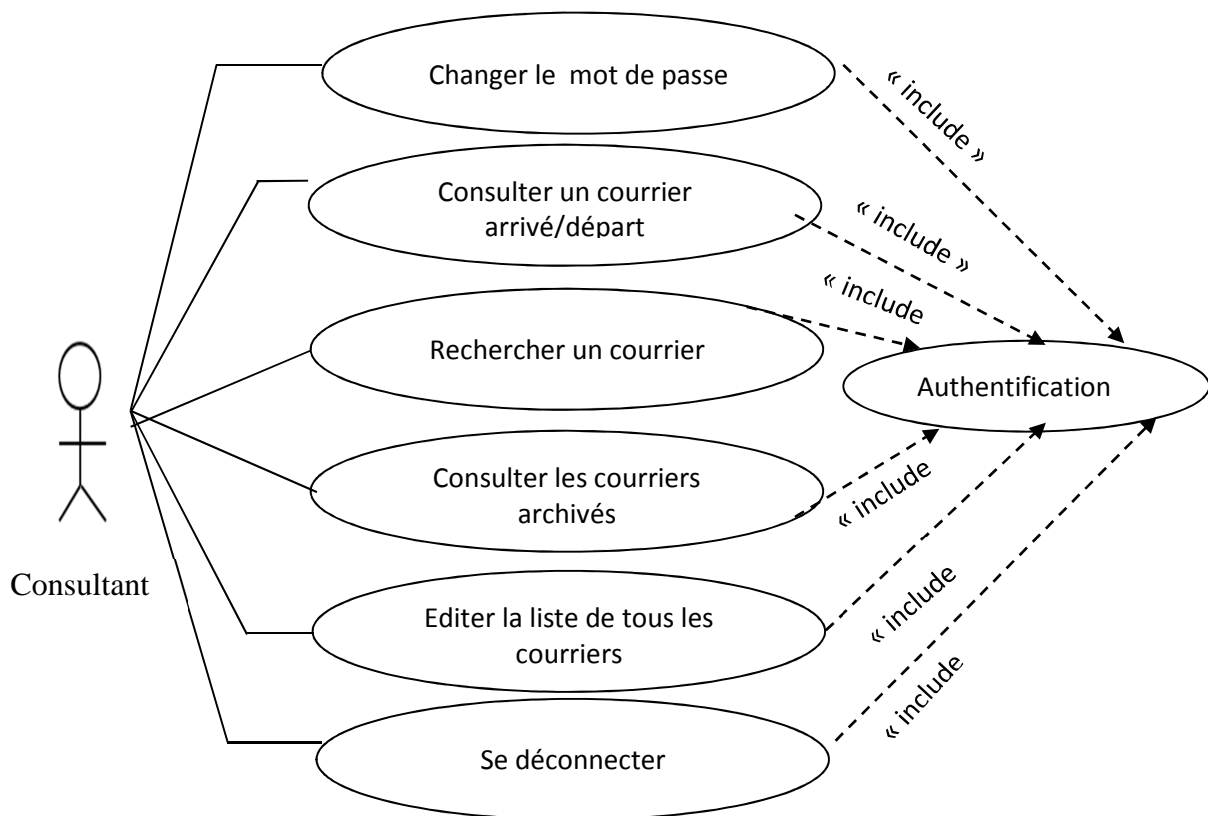


Figure II.1 : Diagramme de cas d'utilisation général du profil consultant

V.2 Description textuelles des diagrammes des cas d'utilisation pour consultant :

➤ **Cas 1** : s'authentifier.

- **Acteurs concernés** : consultant, utilisateur, administrateur
- **Pré-conditions** : L'utilisateur accède à l'application
- **Scénarios nominaux** :
 - 1- L'utilisateur saisit le login et le mot de passe
 - 2- Le système affiche l'espace privé d'utilisateur
- **Scénarios d'exception** :
 - L'authentification n'a pas réussi, le système affiche un message d'erreur.
- **Scénarios alternatifs** :
 - L'utilisateur décide de quitter l'application
- **Post-conditions** :
 - aucun

➤ **Cas 2** : changer le mot de passe.

- **Acteurs concernés** : consultant, utilisateur, administrateur
 - **Pré-conditions** : L'utilisateur doit être authentifié
- **Scénarios nominaux** :
 - 1- L'utilisateur clique sur le lien « changer le mot de passe »
 - 2- Le système affiche le formulaire
 - 3- l'utilisateur remplit le formulaire et le valide
 - 4- Le système enregistre la modification
- **Scénarios d'exception** :
 - La modification n'a pas réussi, le système affiche un message d'erreur.
- **Scénarios alternatifs** :
 - L'utilisateur décide de quitter la modification et clique sur «retour »
- **Post-conditions** :
 - La modification du mot de passe est enregistrée dans la base de données

➤ **Cas 3** : consulter courrier arrivé/départ

- **Acteurs concernés** : consultant, utilisateur, administrateur
- **Pré-conditions** : L'utilisateur doit être authentifié
- **Scénarios nominaux** :
 - 1- L'utilisateur clique sur le lien « courrier arrivé/départ »
 - 2- Le système affiche la liste des courriers enregistrés
 - 3- l'utilisateur clique sur un courrier pour le visualiser
 - 4- Le système affiche le courrier
- **Scénarios d'exception** :
 - aucun.
- **Scénarios alternatifs** :
 - L'utilisateur décide de quitter la consultation et clique sur «retour »
- **Post-conditions** :
 - aucun

➤ **Cas 4** : rechercher un courrier.

- **Acteurs concernés** : consultant, utilisateur, administrateur
- **Pré-conditions** : L'utilisateur doit être authentifié
- **Scénarios nominaux** :
 - 5- L'utilisateur clique sur le lien « recherche »
 - 6- Le système affiche le formulaire
 - 7- l'utilisateur saisit une ou plusieurs informations concernant le courrier et valide
 - 8- Le système affiche le courrier correspond à cette recherche.
- **Scénarios d'exception** :
 - 4- Aucun courrier ne correspond à cette recherche.
- **Scénarios alternatifs** :
 - 3- a - L'utilisateur décide de quitter la recherche et clique sur «retour »
- **Post-conditions** :
 - aucun

➤ **Cas 5**: consulter courriers archivés

- **Acteurs concernés :** consultant, utilisateur, administrateur
- **Pré-conditions :** L'utilisateur doit être authentifié
- **Scénarios nominaux :**
 - 1- L'utilisateur clique sur le lien « archive »
 - 2- Le système affiche la liste des courriers archivés
 - 3- l'utilisateur clique sur un courrier pour le visualiser
 - 4- Le système affiche le courrier
- **Scénarios d'exception :**
 - aucun.
- **Scénarios alternatifs :**
 - L'utilisateur décide de quitter la consultation et clique sur « retour »
- **Post-condition :**
 - aucun

➤ **Cas 6:** éditer la liste de tous les courriers

- **Acteurs concernés :** consultant, utilisateur, administrateur
- **Pré-conditions :** L'utilisateur doit être authentifié
- **Scénarios nominaux :**
 - 1- L'utilisateur clique sur le lien « édition »
 - 2- Le système affiche la liste des courriers
- **Scénarios d'exception :**
 - aucun.
- **Scénarios alternatifs :**
 - L'utilisateur décide de quitter la consultation et clique sur « retour »
- **Post-conditions :**
 - aucun

V.3 Diagramme de cas d'utilisation général du profil utilisateur :

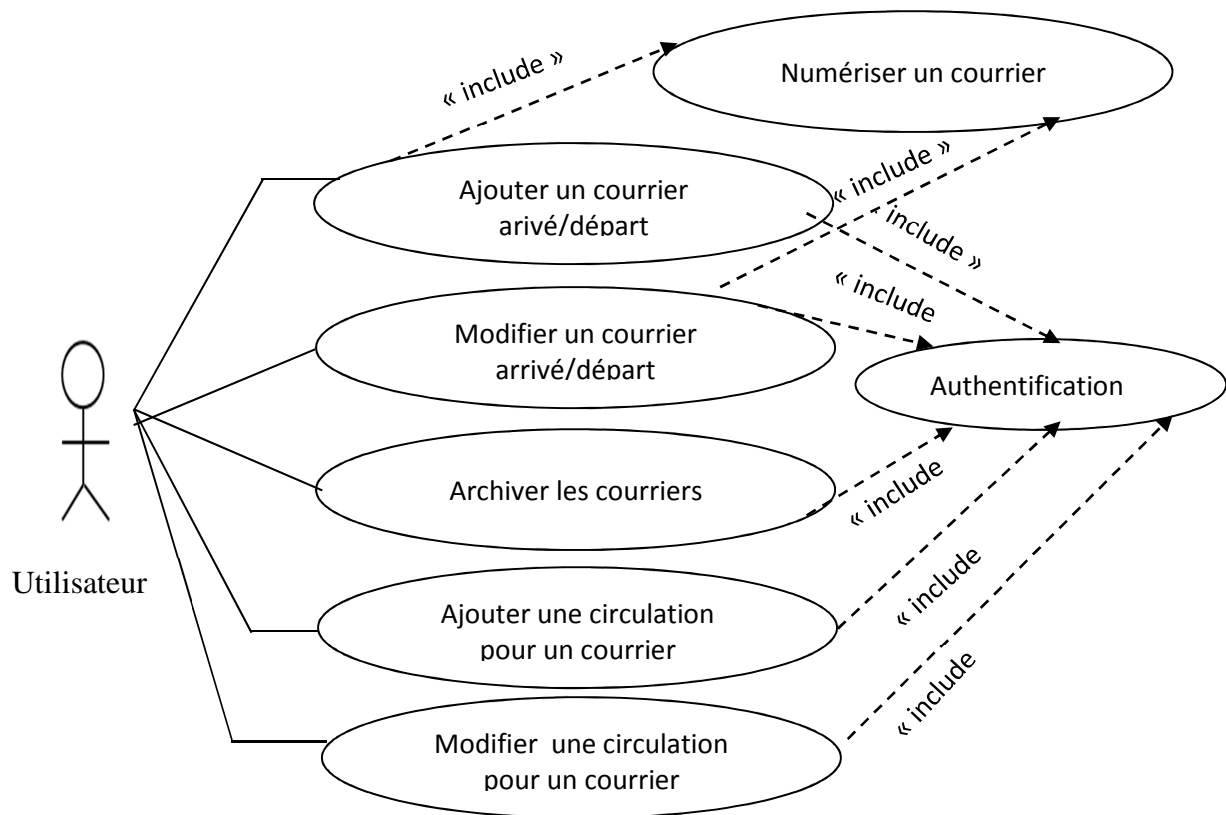


Figure II.2 : Diagramme de cas d'utilisation général du profil utilisateur

V.3.1 Description textuelle des diagrammes de cas d'utilisation pour utilisateur

➤ **Cas 1** : ajouter un courrier.

- **Acteurs concernés** : utilisateur, administrateur
- **Pré-condition** :
 - L'utilisateur doit être authentifié
 - Le courrier doit être numérisé.
- **Scénarios nominaux** :
 - 1- L'utilisateur clique sur le lien « ajouter »
 - 2- Le système affiche le formulaire d'ajout
 - 3- l'utilisateur remplit et valide
 - 4- Le système enregistre les données
- **Scénarios d'exception** :
 - L'ajout n'a pas réussi, le système affiche un message d'erreur.
- **Scénarios alternatifs** :
 - L'utilisateur décide de quitter l'ajout
- **Post-condition** :
 - Le courrier est enregistré dans la base de données

➤ **Cas 2** : modifier un courrier.

- **Acteurs concernés** : utilisateur, administrateur
- **Pré-conditions** : L'utilisateur doit être authentifié
- **Scénarios nominaux** :
 - 1- L'utilisateur choisit un courrier clique sur le lien « modifier ».
 - 2- Le système affiche le formulaire de modification.
 - 3- l'utilisateur remplit le formulaire et le valide
 - 4- Le système enregistre la modification.
- **Scénarios d'exception** :
 - La modification n'a pas réussi, le système affiche un message d'erreur.
- **Scénarios alternatifs** :
 - L'utilisateur décide de quitter la modification et clique sur «retour »
- **Post-conditions** :
 - La modification du courrier est enregistrée dans la base de données

➤ **Cas 3** : archiver les courriers.

- **Acteurs concernés** : utilisateur, administrateur
- **Pré-condition** : L'utilisateur doit être authentifié
- **Scénarios nominaux** :
 - 1- L'utilisateur clique sur le lien « archivage »
 - 2- Le système affiche un formulaire
 - 3- l'utilisateur remplit le formulaire et le valide
 - 4- Le système enregistre les données
- **Scénarios d'exception** :
 - L'archivage n'a pas réussi, le système affiche un message d'erreur.
- **Scénarios alternatifs** :
 - L'utilisateur décide de quitter l'archivage et clique sur le lien « retour »
- **Post-condition** :
 - L'archivage est enregistré dans la base de données

➤ **Cas 4** : ajouter une circulation pour un courrier.

- **Acteurs concernés** : utilisateur, administrateur
- **Pré-condition** :
 - L'utilisateur doit être authentifié
- **Scénarios nominaux** :
 - 1- L'utilisateur clique sur le lien « ajouter »
 - 2- Le système affiche le formulaire d'ajout
 - 3- l'utilisateur remplit et valide
 - 4- Le système enregistre les données
- **Scénarios d'exception** :
 - L'ajout n'a pas réussi, le système affiche un message d'erreur.
- **Scénarios alternatifs** :
 - L'utilisateur décide de quitter l'ajout
- **Post-condition** :
 - La circulation du courrier est enregistrée dans la base de données
 -

➤ **Cas 5** : modifier une circulation d'un courrier.

- **Acteurs concernés** : utilisateur, administrateur
- **Pré-conditions** : L'utilisateur doit être authentifié

- **Scénarios nominaux :**

- 1- L'utilisateur choisit une circulation et clique sur le lien « modifier ».
- 2- Le système affiche le formulaire de modification.
- 3- l'utilisateur remplit le formulaire et le valide
- 4- Le système enregistre la modification.

- **Scénarios d'exception :**

- La modification n'a pas réussi, le système affiche un message d'erreur.

- **Scénarios alternatifs :**

- L'utilisateur décide de quitter la modification et clique sur «retour »

- **Post-conditions :**

- La modification de la circulation du courrier est enregistrée dans la base de données

V.4 Diagramme de cas d'utilisation général du profil administrateur

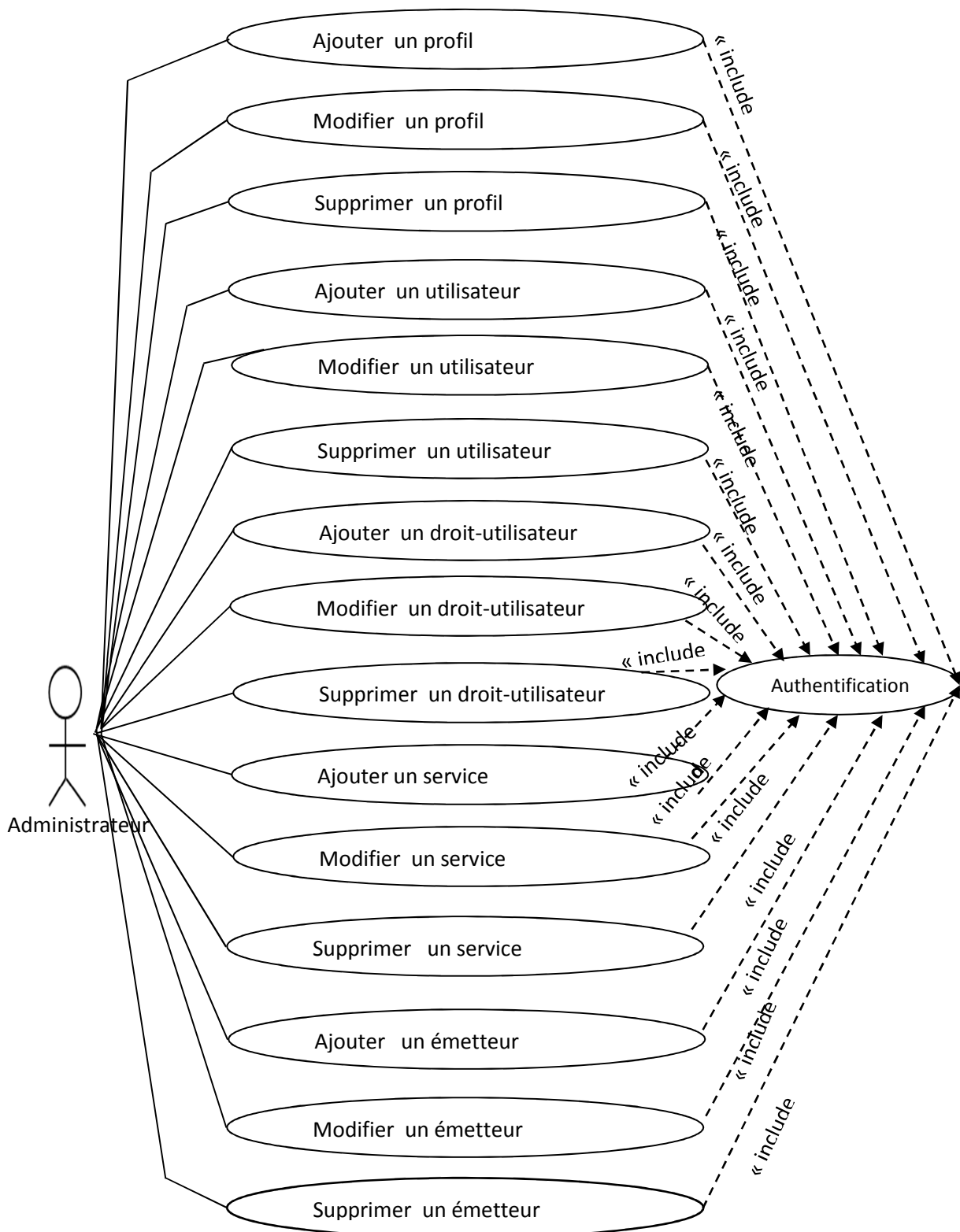


Figure II.3 : Diagramme de cas d'utilisation général du profil administrateur

V.5.1 Description textuelles des cas d'utilisation pour le profil « administrateur »

- **Cas 1** : ajouter un profil /utilisateur /droit-utilisateur /service /émetteur.
- **Acteurs concernés** : administrateur
 - **Pré-condition** :
 - L'utilisateur doit être authentifié
 - **Scénarios nominaux** :
 - 1- L'administrateur clique sur le lien « ajouter »
 - 2- Le système affiche le formulaire d'ajout
 - 3- l'administrateur remplit et valide
 - 4- Le système enregistre les données
 - **Scénarios d'exception** :
 - L'ajout n'a pas réussi, le système affiche un message d'erreur.
 - **Scénarios alternatifs** :
 - L'administrateur décide de quitter l'ajout
 - **Post-condition** :
 - Les données sont enregistrées dans la base de données
- **Cas 2** : modifier un profil /utilisateur /droit-utilisateur /service /émetteur.
- **Acteurs concernés** : administrateur
 - **Pré-conditions** : L'administrateur doit être authentifié
 - **Scénarios nominaux** :
 - 1- L'administrateur choisit un courrier clique sur le lien « modifier ».
 - 2- Le système affiche le formulaire de modification.
 - 3- l'administrateur remplit le formulaire et le valide
 - 4- Le système enregistre la modification.
 - **Scénarios d'exception** :
 - La modification n'a pas réussi, le système affiche un message d'erreur.
 - **Scénarios alternatifs** :
 - L'administrateur décide de quitter la modification et clique sur «retour »
 - **Post-conditions** :
 - La modification est enregistrée dans la base de données

➤ **Cas 3** : supprimer un profil /utilisateur /droit-utilisateur /service /émetteur.

- **Acteurs concernés** : administrateur
- **Pré-condition** : L'administrateur doit être authentifié
- **Scénarios nominaux** :
 - 1- L'administrateur choisit un élément et clique sur « supprimer »
 - 2- Le système supprime l'élément
- **Scénarios alternatifs** :
 - L'administrateur décide de quitter la suppression et clique sur « retour »
- **Post-condition** :
 - La suppression des éléments dans la base de données est effectuée.

VI. Les diagrammes de séquence :

VI.1 Diagramme de séquence « authentification »:

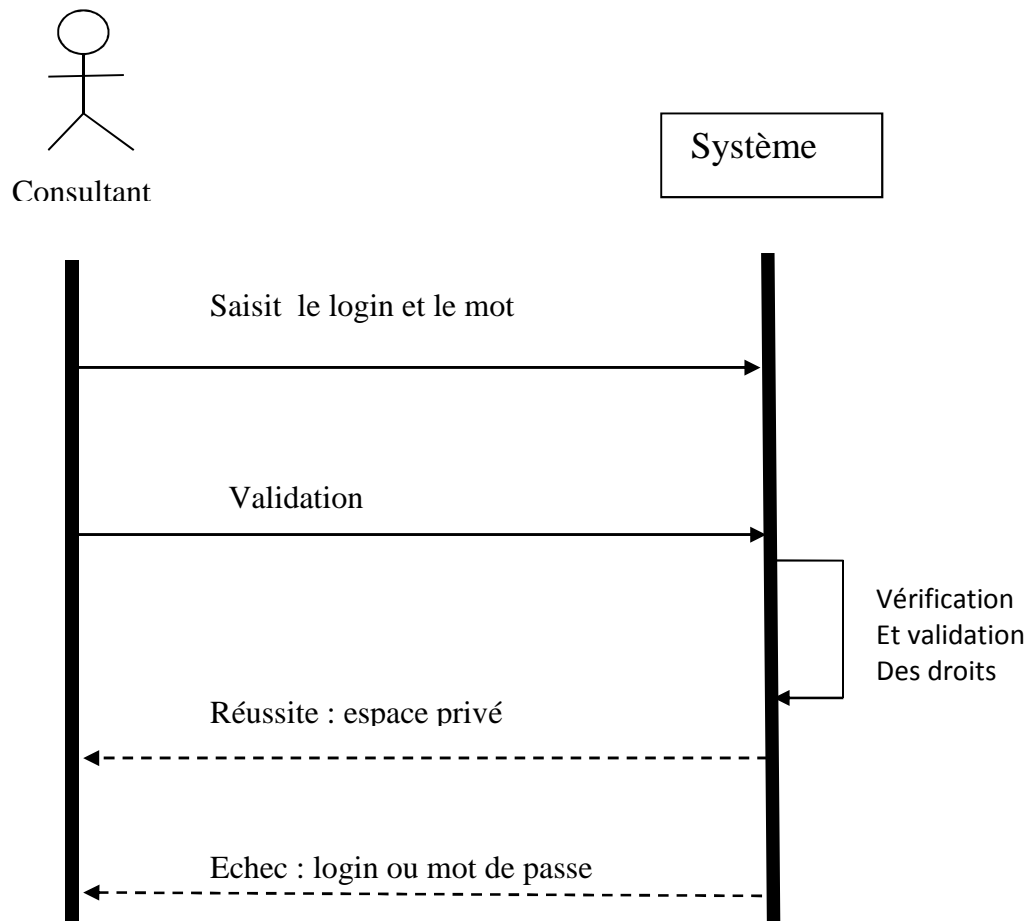


Figure II.4 : Diagramme de séquence « authentification »

- L'utilisateur doit se connecter en utilisant son login + mot de passe.
- Si l'authentification est valide, le système lui affiche son espace privé
Sinon il affiche un message d'erreur et renvoie la page d'authentification
- Tout au long d'une session l'utilisateur n'a la possibilité d'accéder qu'aux services dont il est autorisé.

VI.2 Diagramme de séquence « ajout » :

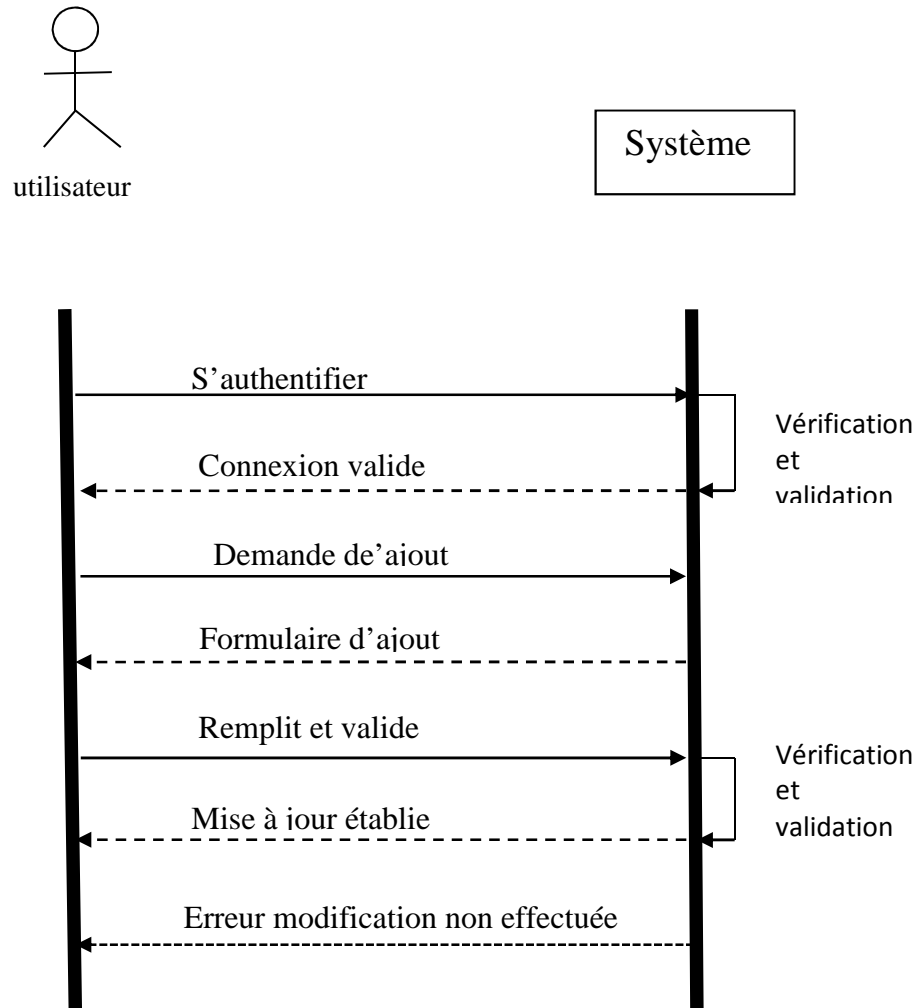


Figure II.5 : Diagramme de séquence « ajout »

- l'utilisateur s'authentifie
- Il clique sur le lien « ajouter »
- Le système lui affiche le formulaire
- L'utilisateur remplit le formulaire et valide
- Le système enregistre les informations s'il n'y a pas d'erreur sinon affiche un message d'erreur

VI.3 Diagramme de séquence « suppression » :

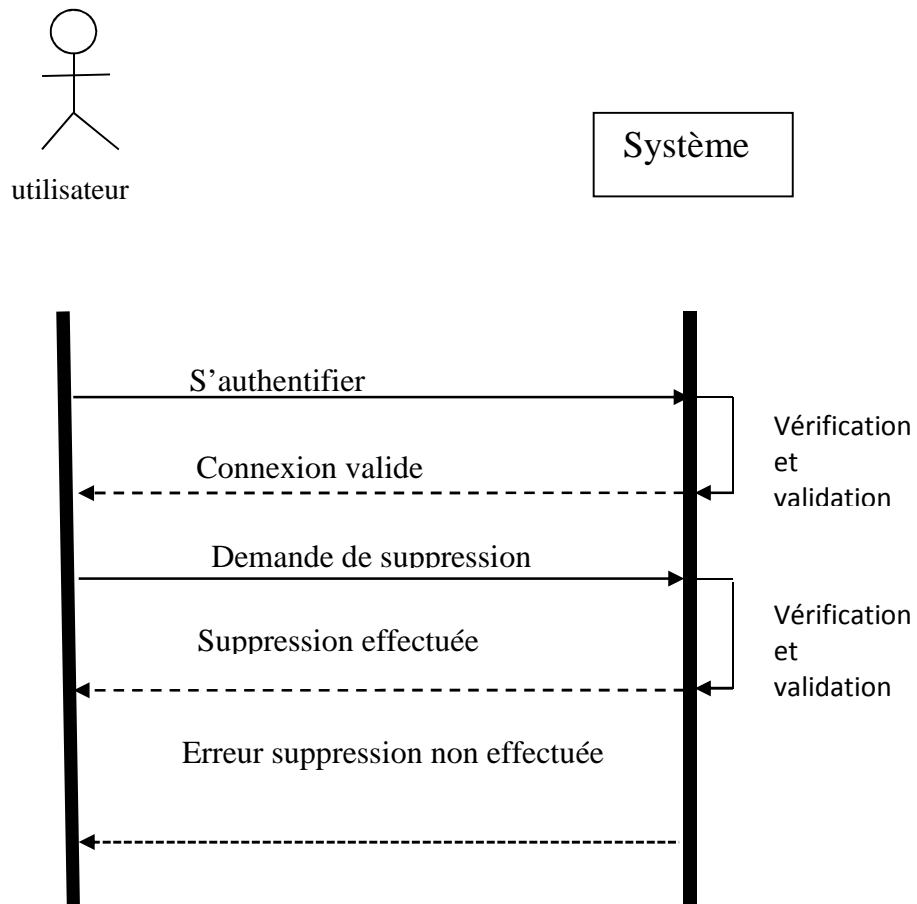


Figure II.6 : Diagramme de séquence « suppression »

- l'utilisateur s'authentifie
- Il clique sur le lien « supprimer »
- Le système demande une confirmation si oui il supprime les informations.

VI.4 Diagramme de séquence « modification »

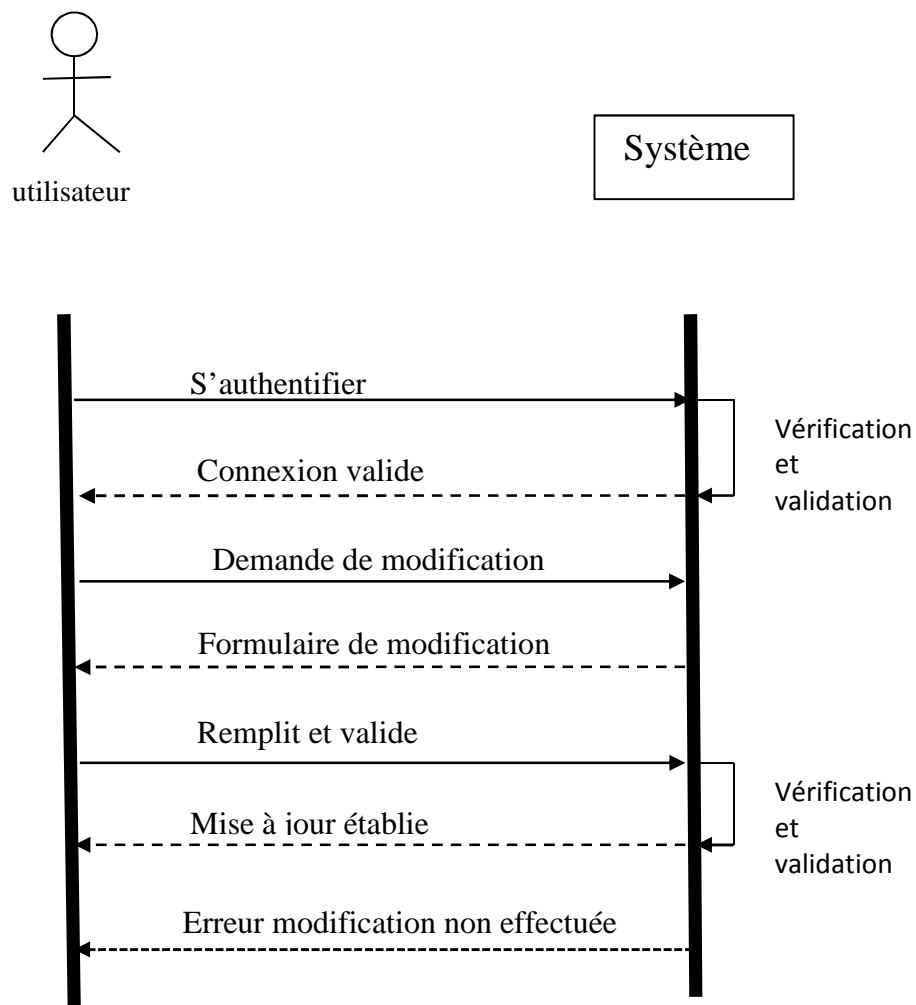


Figure II.7 : Diagramme de séquence « modification »

- l'utilisateur s'authentifie
- Il clique sur le lien « ajouter »
- Le système lui affiche le formulaire
- L'utilisateur remplit le formulaire et valide
- Le système enregistre les informations s'il n'y a pas d'erreur sinon affiche un message d'erreur

VI.5 Diagramme de séquence « changer mot de passe »

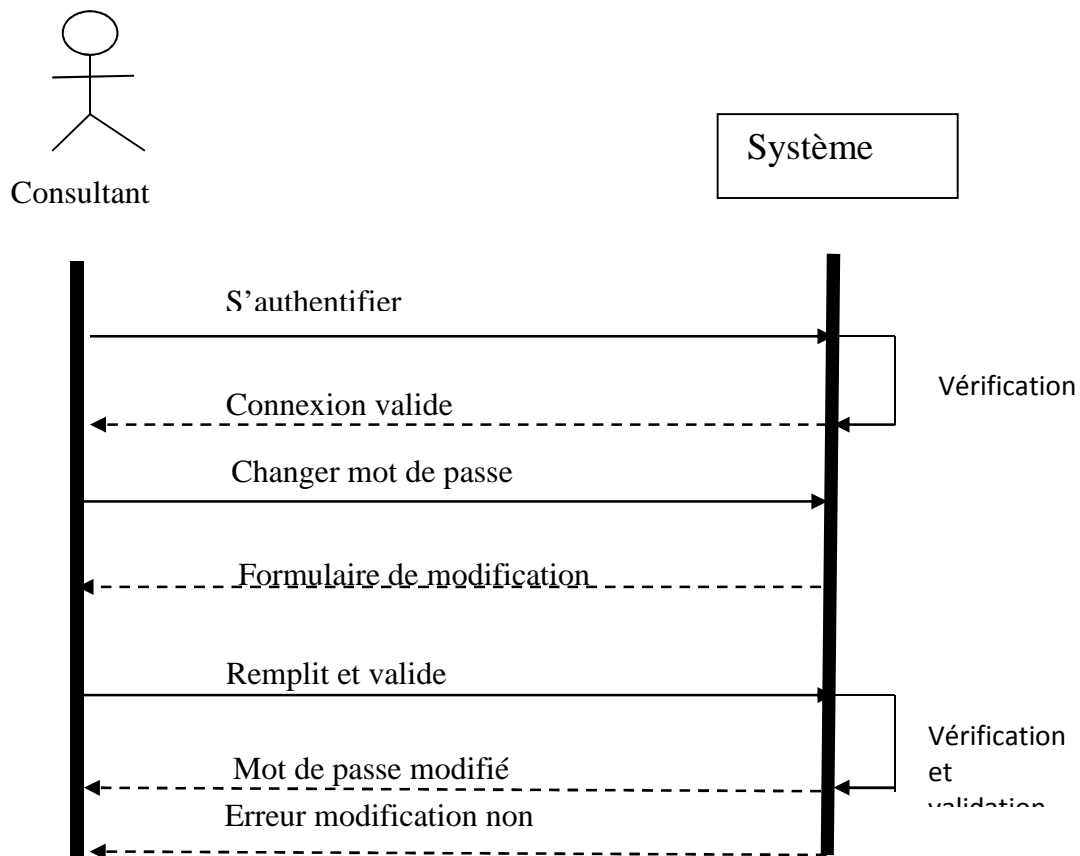


Figure II.8 : Diagramme de séquence « changer mot de passe »

- l'utilisateur s'authentifie
- Il clique sur le lien « changer le mot de passe »
- Le système lui affiche le formulaire
- L'utilisateur remplit le formulaire et valide
- Le système mis à jour les modifications si 'il n'y a pas d'erreur sinon affiche un message d'erreur

VI.6 Diagramme de séquence « rechercher un courrier »

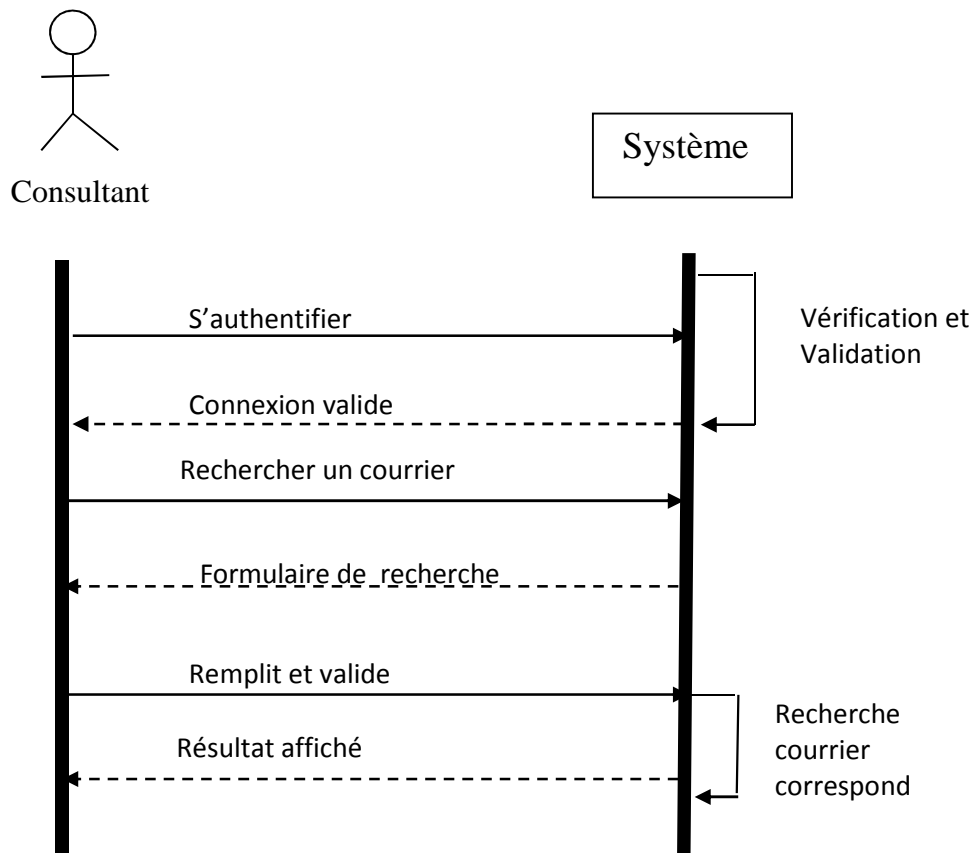


Figure II.9 : Diagramme de séquence « rechercher un courrier »

- l'utilisateur s'authentifie
- Il clique sur le lien « recherche »
- Le système lui affiche le formulaire
- L'utilisateur remplit le formulaire et valide
- Le système affiche le courrier correspond aux informations saisies.

VII. Les diagrammes d'activités :

VII.1 Diagramme d'activité « s'authentifier »

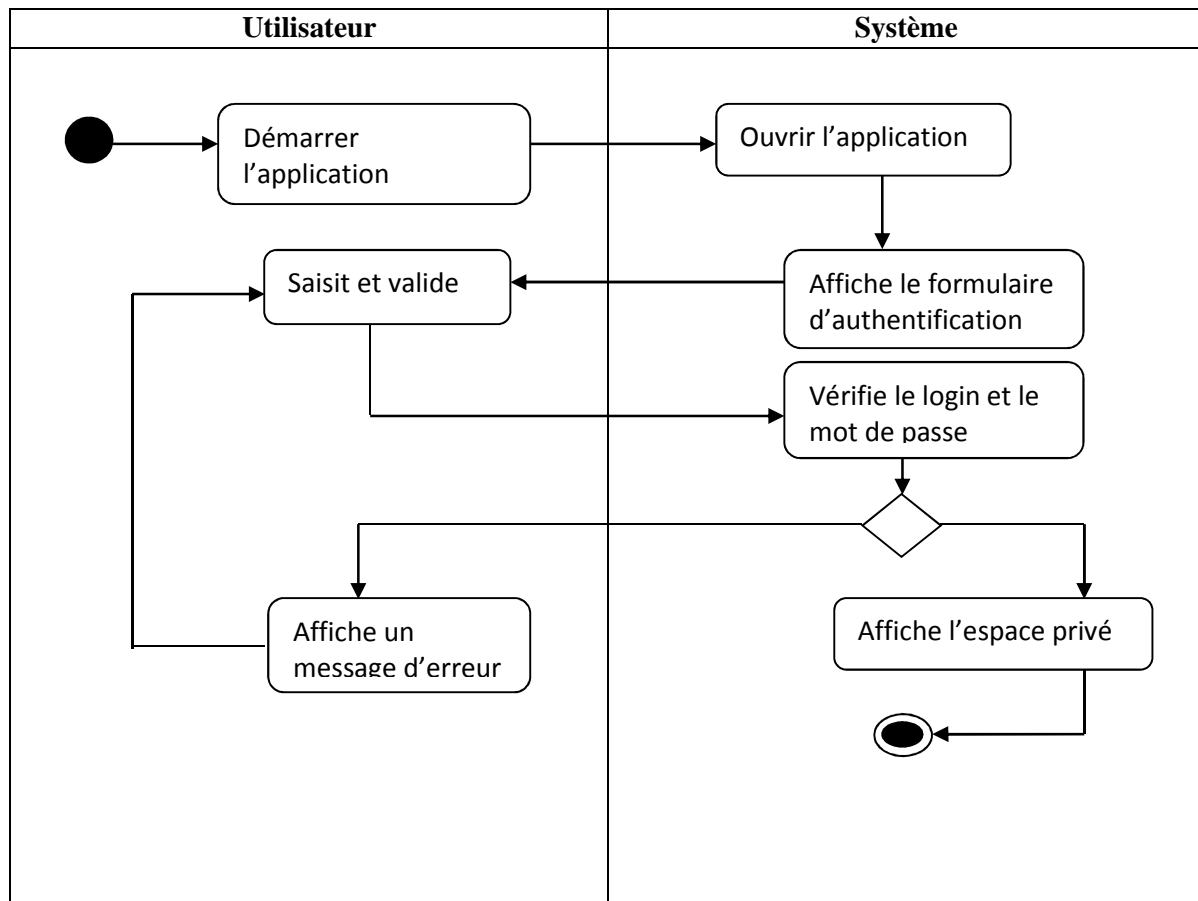


Figure II.10 : Diagramme d'activité « s'authentifier »

VII.2 Diagramme d'activité d'ajout :

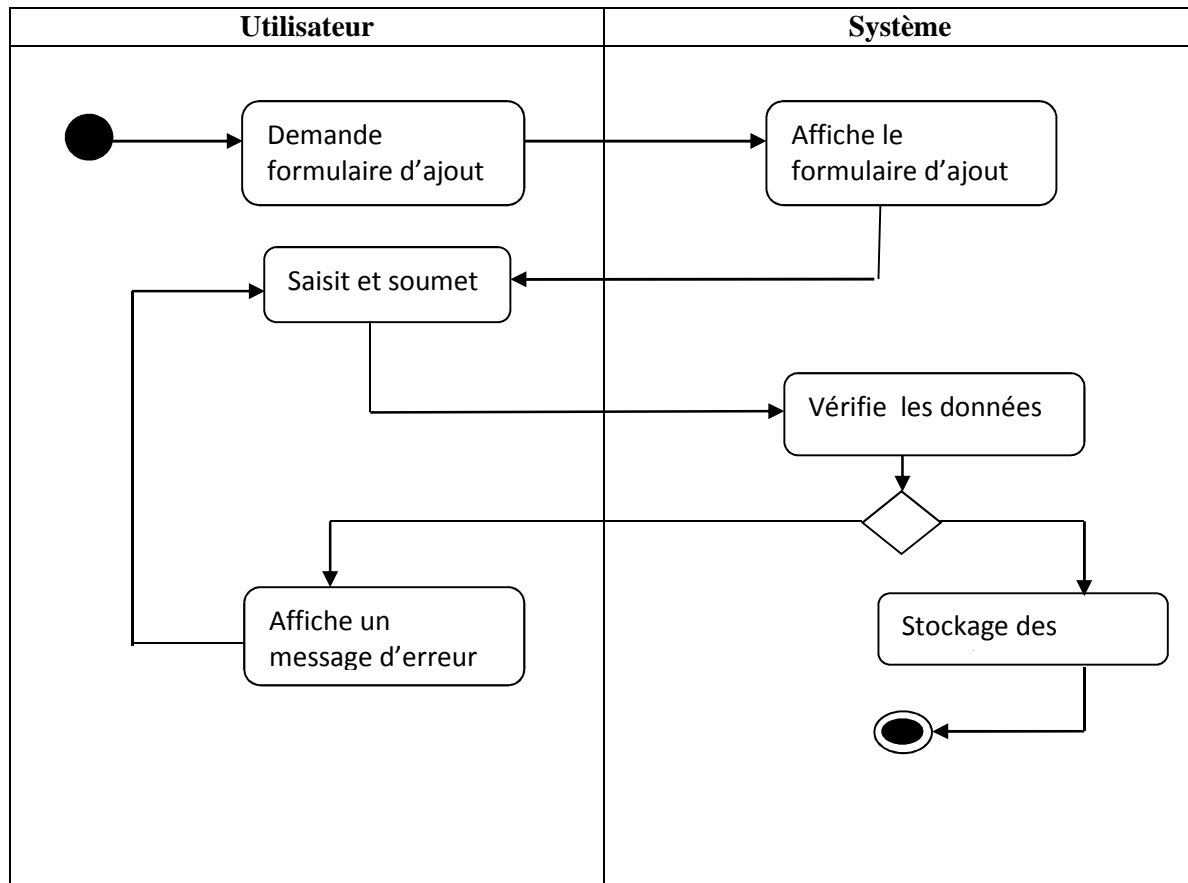


Figure II.11 : Diagramme d'activité d'ajout :

VII.3 Diagramme d'activité de modification :

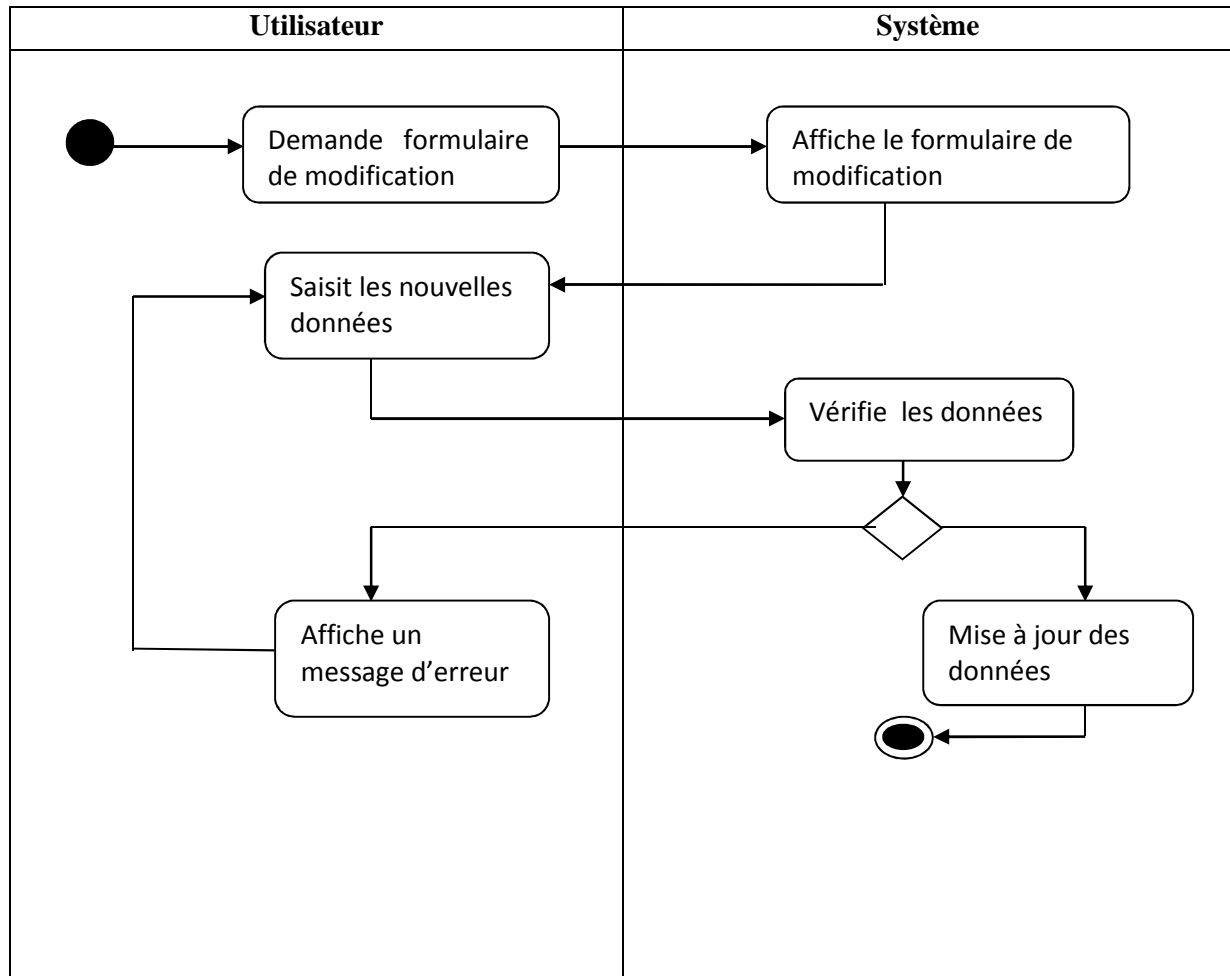


Figure II.12 : Diagramme d'activité de modification

VII.4 Diagramme d'activité de suppression

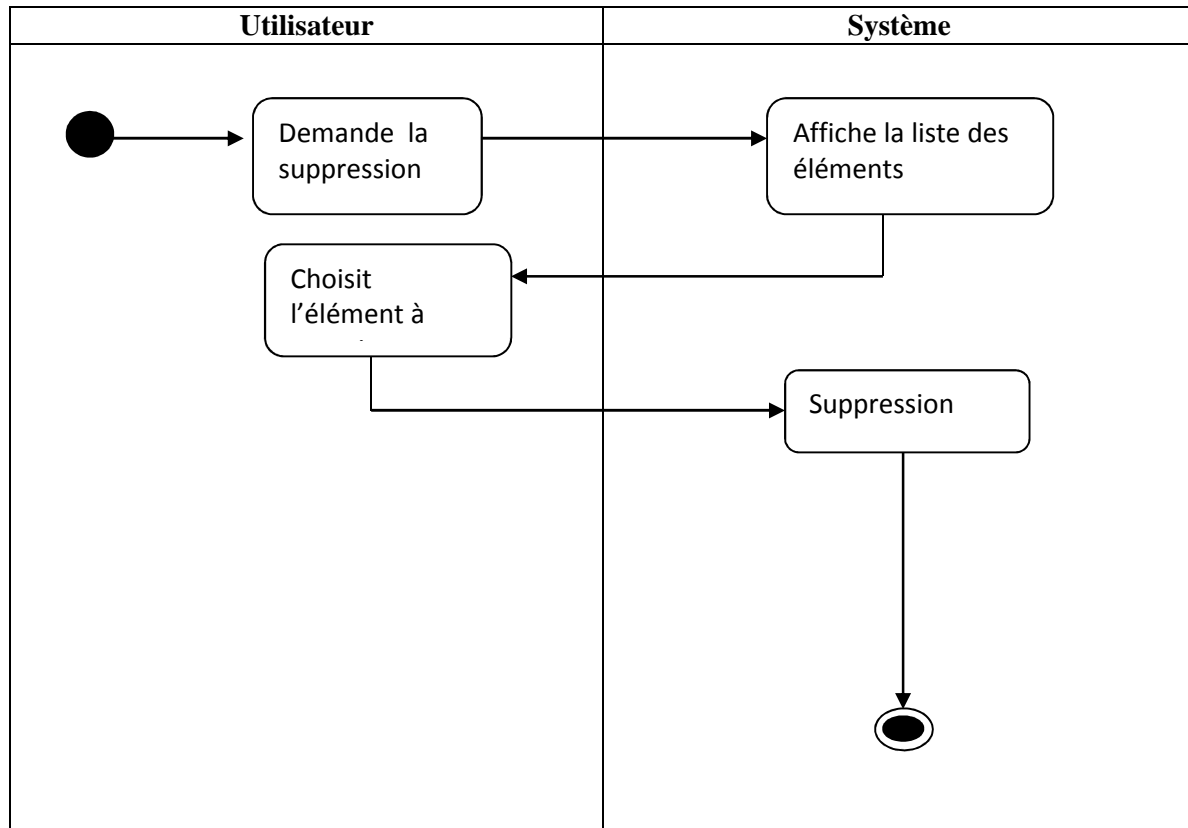


Figure II.13 : Diagramme d'activité de suppression

VII.5 Diagramme d'activité « rechercher un courrier »

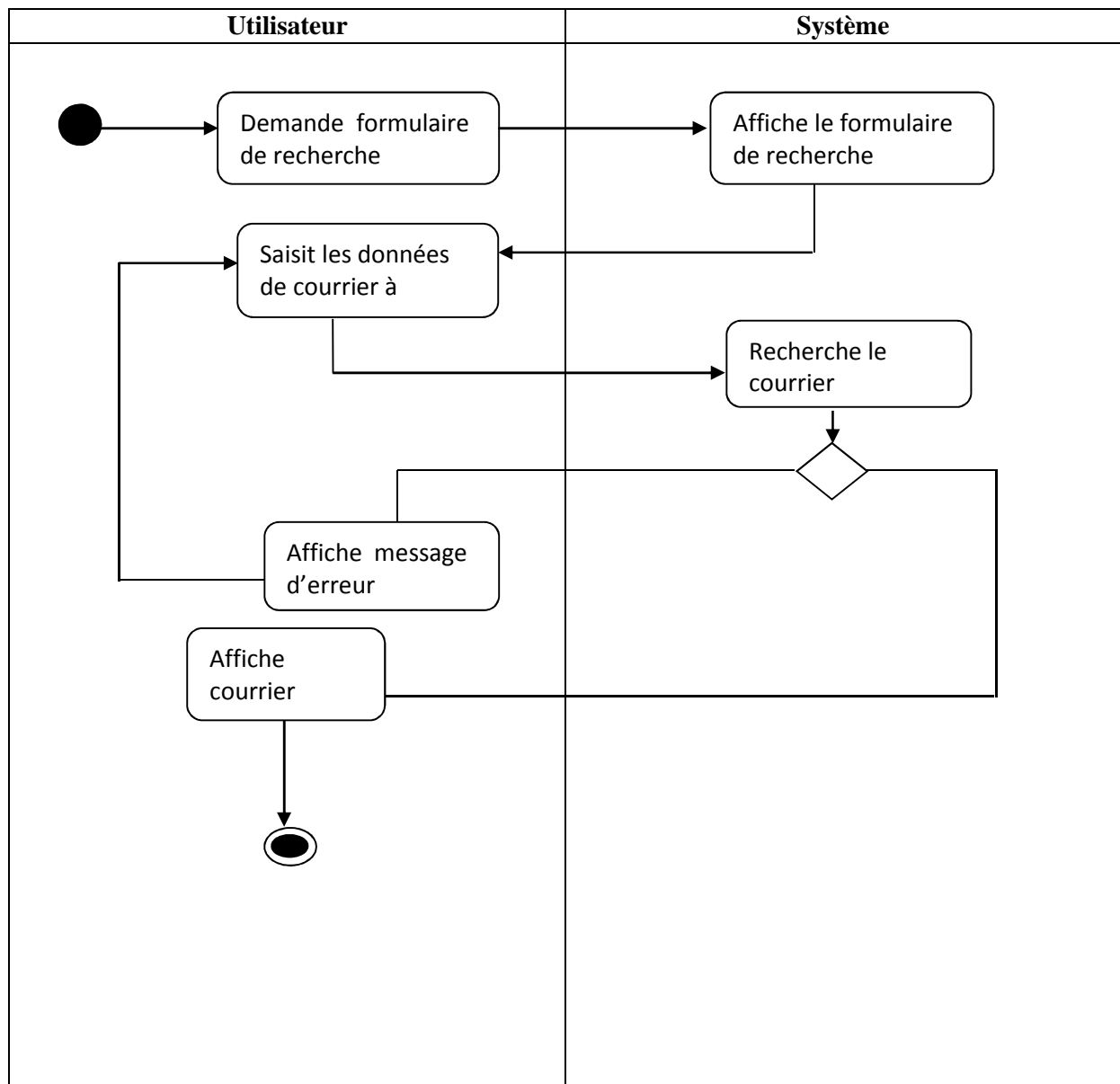


Figure II.14 : Diagramme d'activité « rechercher un courrier »

VIII. Les diagrammes de collaboration :

VIII.1 Diagramme de collaboration d'authentification :

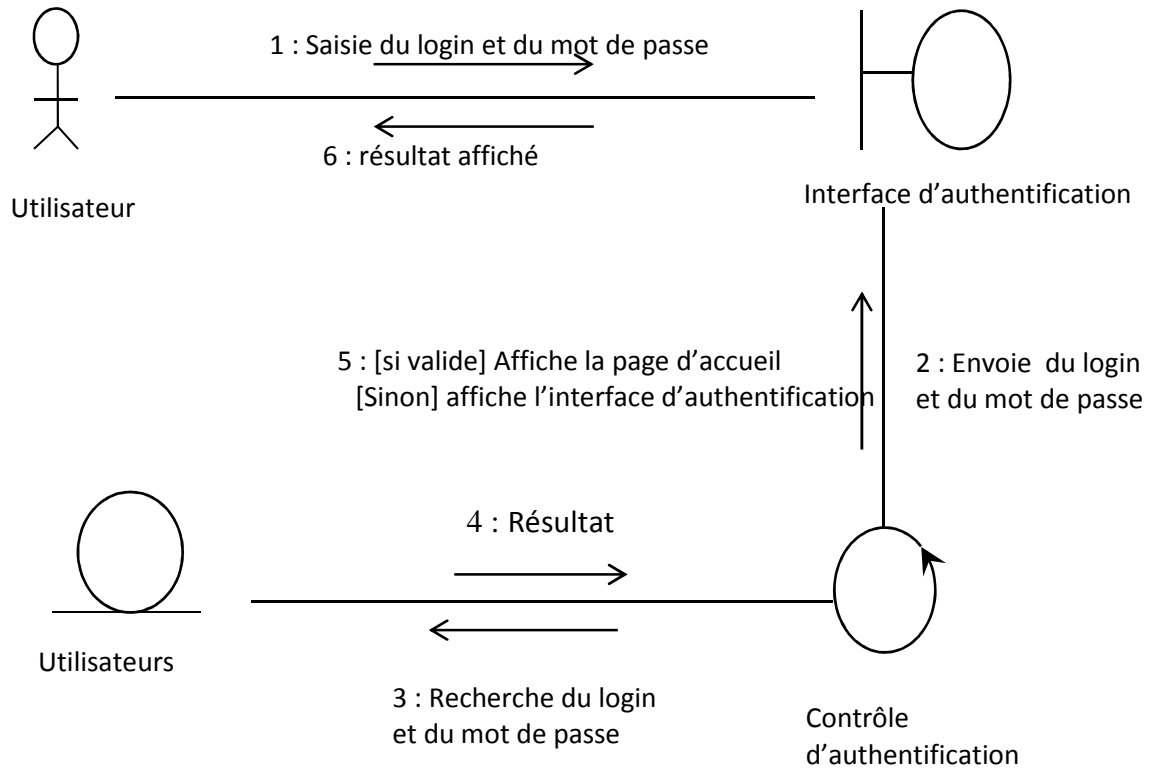


Figure II.15 : Diagramme de collaboration d'authentification

- l'utilisateur saisit son login et son mot de passe sur l'interface d'authentification.
- Si l'authentification est valide, le système lui affiche son espace privé
Sinon il affiche un message d'erreur et renvoie la page d'authentification

VIII.2 Diagramme de collaboration d'ajout :

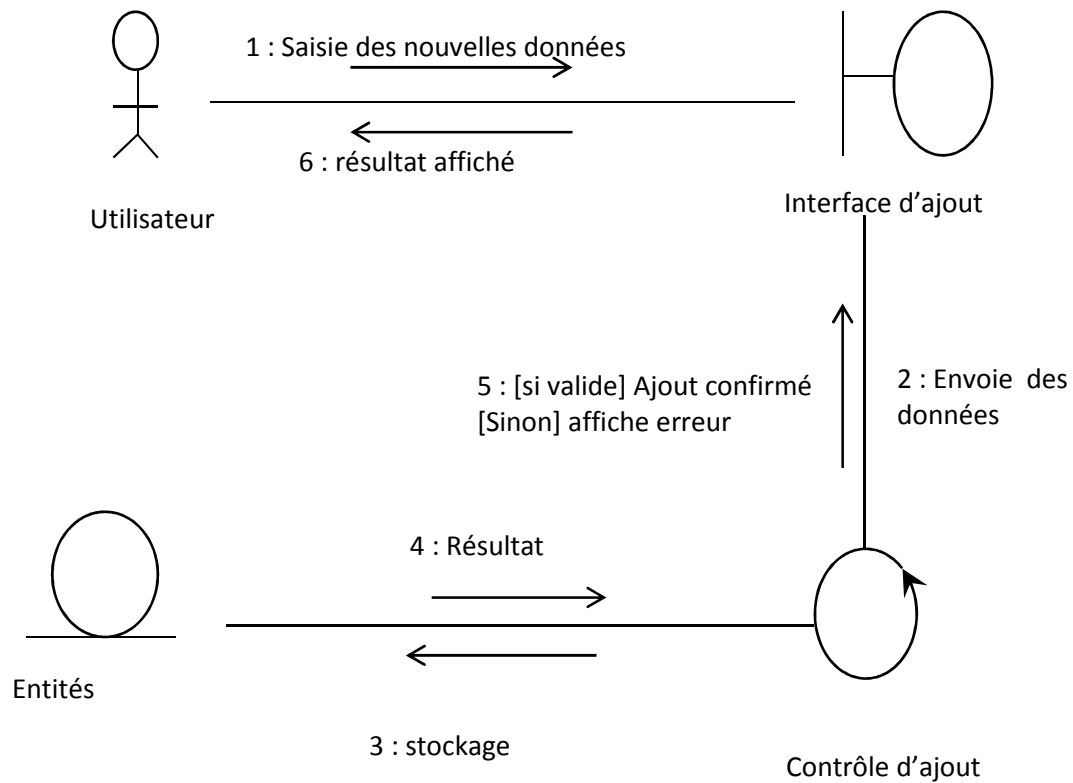


Figure II.16 : Diagramme de collaboration d'ajout

- l'utilisateur s'authentifie et clique sur le lien « ajouter »
- Le système lui affiche le formulaire
- L'utilisateur remplit le formulaire et valide
- Le système enregistre les informations s'il n'y a pas d'erreur sinon affiche un message d'erreur

VIII.3 Diagramme de collaboration de modification :

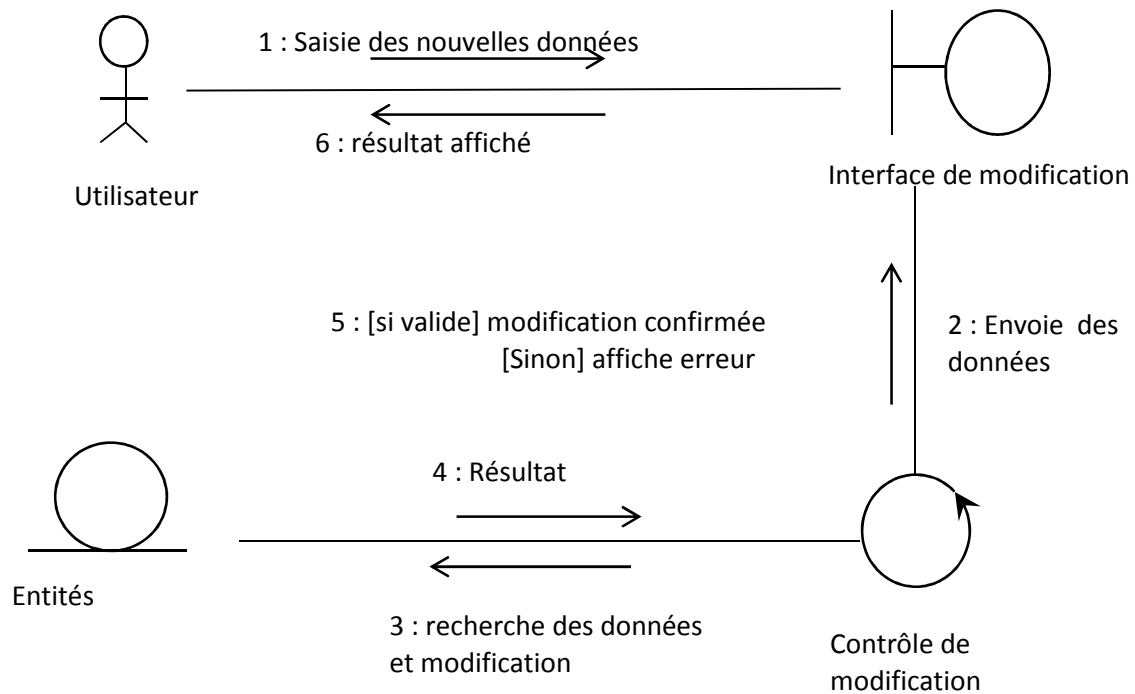


Figure II.17 : Diagramme de collaboration de modification

- après avoir être authentifié l'utilisateur saisit de nouvelles données sur le formulaire de modification et valide.
- Le système enregistre les informations s'il n'y a pas d'erreur sinon affiche un message d'erreur

VIII.4 Diagramme de collaboration de suppression :

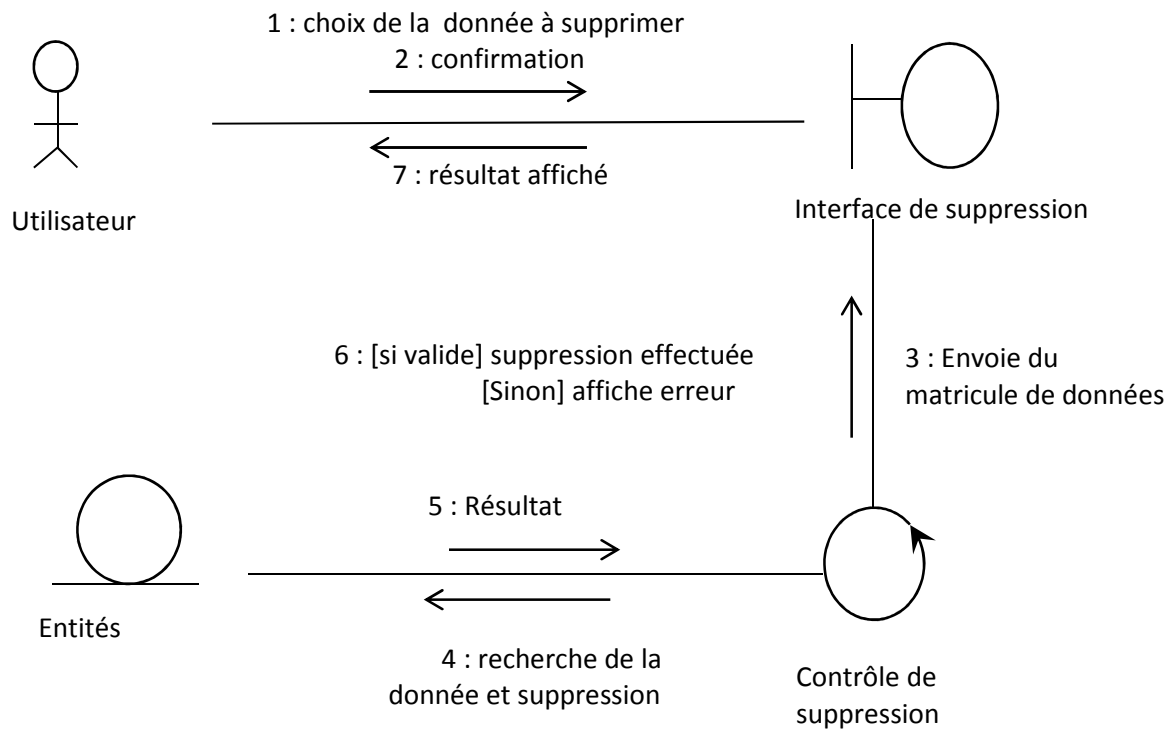


Figure II.18 : Diagramme de collaboration de suppression

- l'utilisateur choisit l'élément à supprimer
- Il clique sur « supprimer »
- Le système demande une confirmation si oui il supprime les informations

VIII.5 Diagramme de collaboration « rechercher un courrier » :

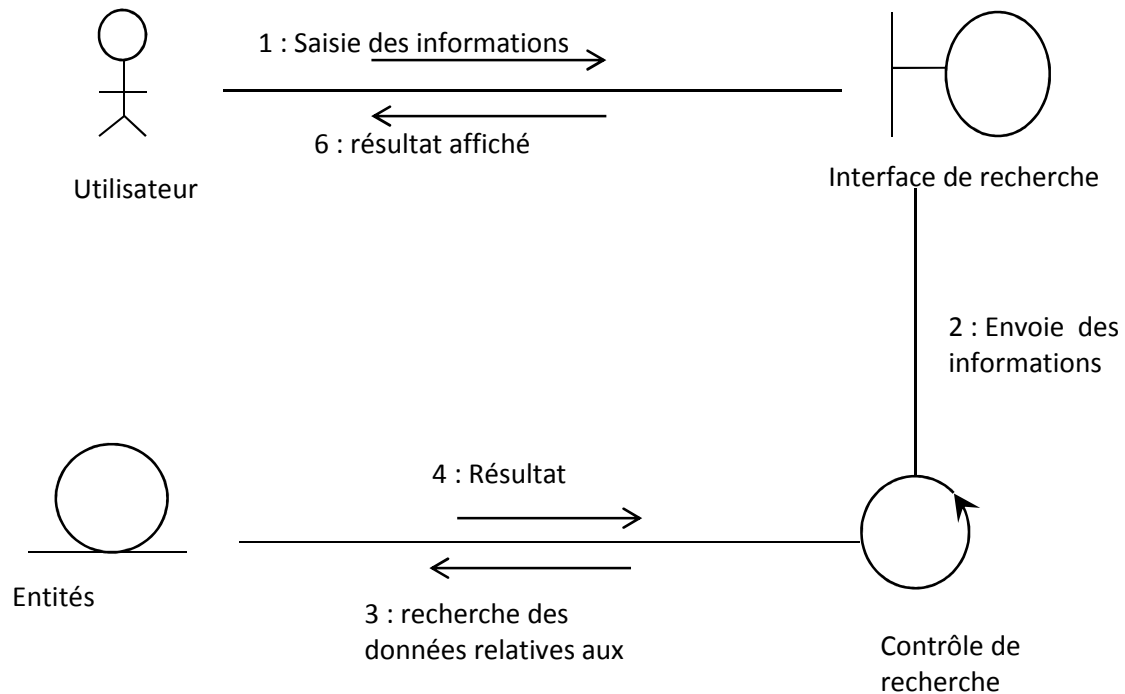


Figure II.19 Diagramme de collaboration « rechercher un courrier »

- l'utilisateur saisit les informations correspondants au courrier à rechercher
- Le système affiche le courrier correspond aux informations saisies.

IX. Conception de la base de données :

IX.1. Diagramme de classe générale pour la base de données :

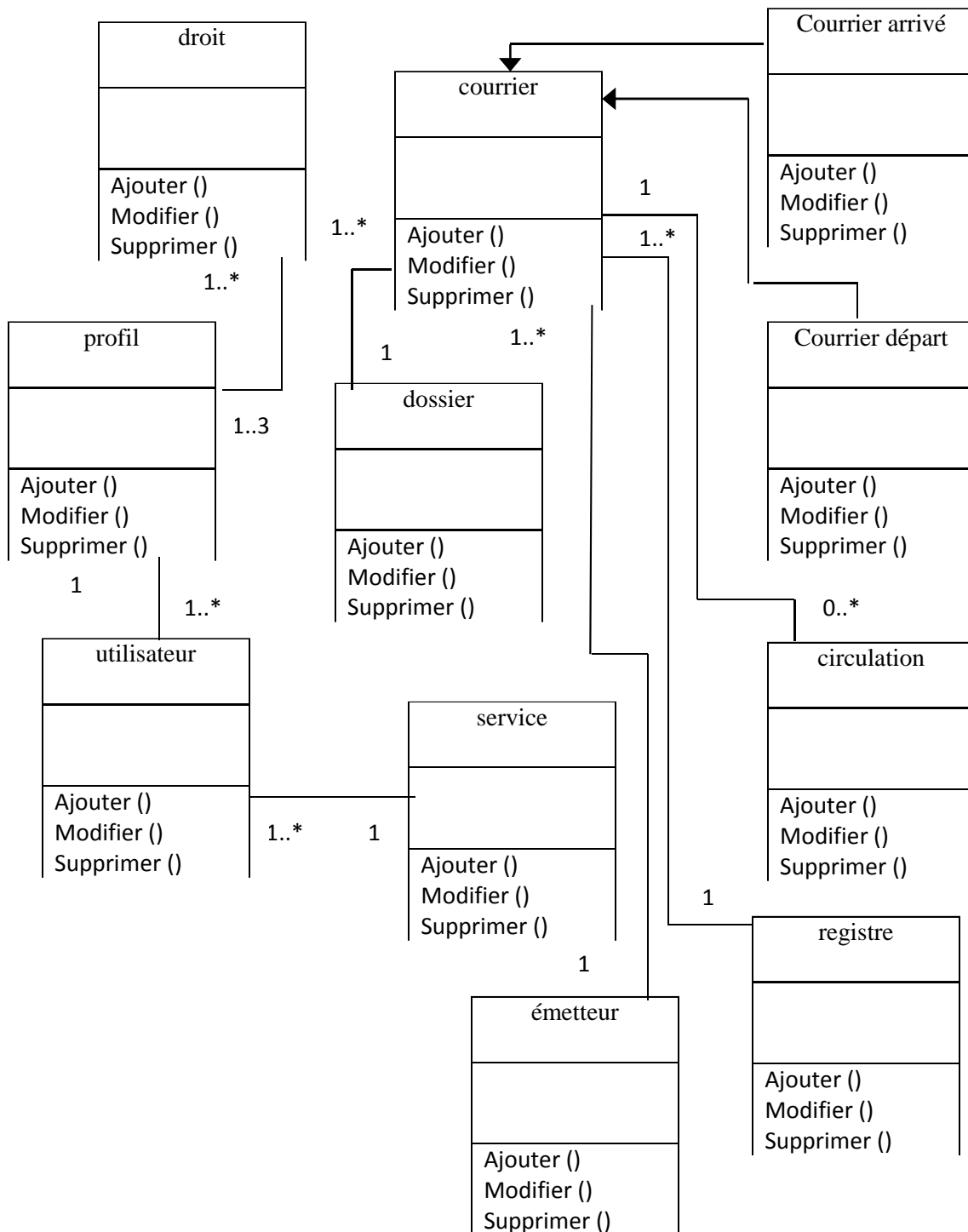


Figure II.20 : Diagramme de classe générale

IX.2. Modèle logique de données :

D'après le modèle conceptuel de données on peut directement obtenir le modèle logique de données contenant les tables suivantes :

- 1- Courrier (Idcourrier, Naturecourrier, Typecourrier, Datearrive, Datecourrier, Emetteur, Objet, Copiea, Traitement).
- 2- Dossier (Iddossier, Idcourrier* ,Datedossier, Typedossier).
- 3- Registre (Idregistre, Iddossier*, Idcourrier*).
- 4- Circulation (Idcirculation, Idourrier*, Traitement, Datebuttoir, Dateexecution, delaireponse, detailreponse).
- 5- Notification (Idnotification,).
- 6- Service (Idservice, Libelle, Serviceattache).
- 7- Emetteur (Idemetteur, Nom, Prenom, tel).
- 8- Utilisateur (Idutilisateur, Nom, Email, Login, Motpasse, idProfil*, idService*).
- 9- Profil (Idprofil, typeprofil).
- 10- Droit (Iddroit, idprofil*).

Remarque :

(*): Clés étrangère.

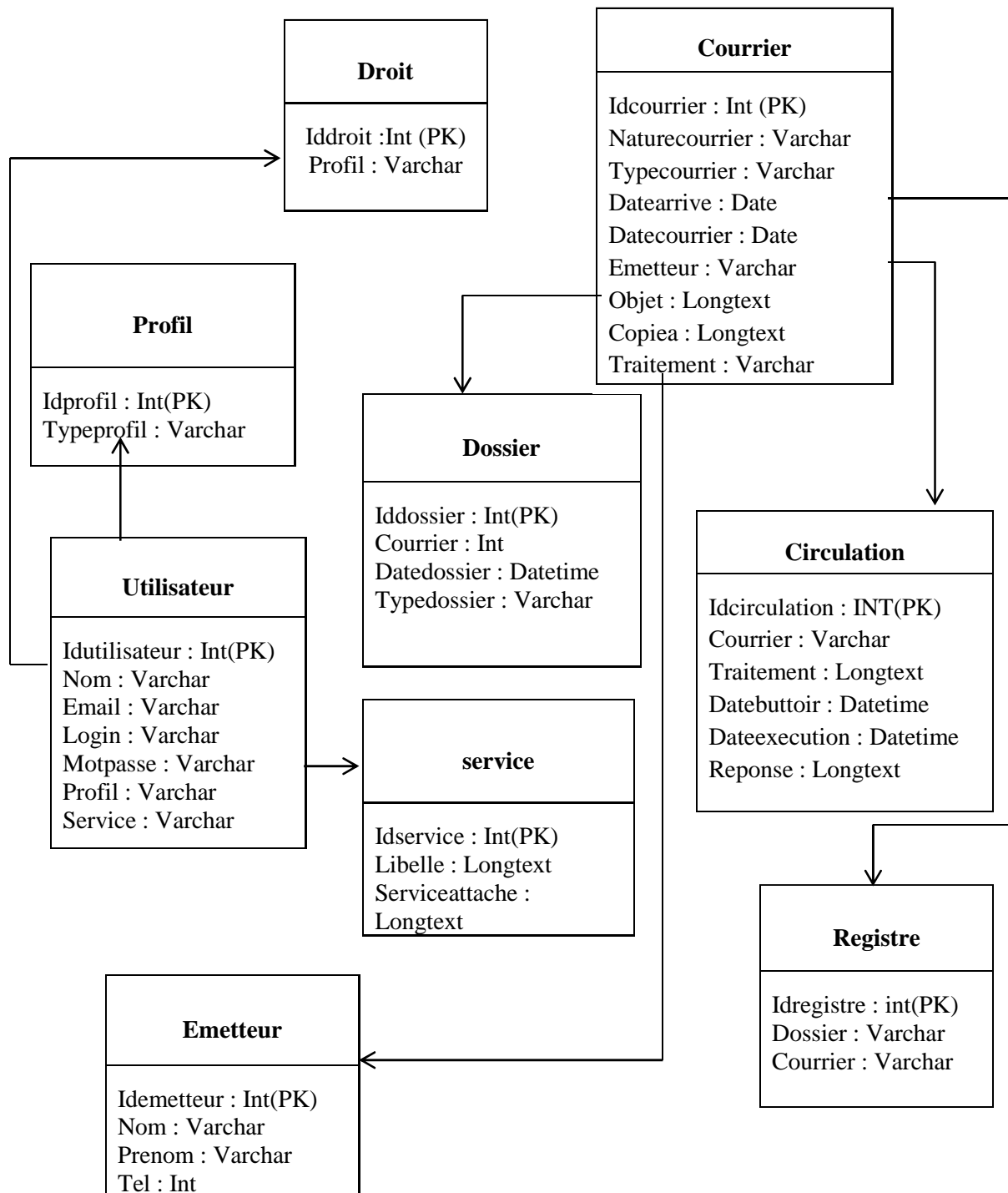


Figure II.21 : Modèle logique de données

XI. Conclusion :

Dans ce chapitre, nous venons de présenter une analyse globale de l'application tout en spécifiant les besoins utilisateurs que notre travail doit satisfaire et respecter ainsi que la phase conception qui vient pour permettre la détermination des différents objets contribuant à assurer les fonctionnalités souhaitées. Cette phase est une préparation à la phase du codage garantissant une organisation claire et précise.

Nous essayons dans le chapitre réalisation de montrer les fonctionnalités réalisées suite à l'implémentation qui a été faite.

Chapitre III :

réalisation

I. Introduction :

Après avoir achevé l'étape de conception de l'application, nous entamons dans ce chapitre la phase de réalisation. Nous allons présenter, en premier lieu, le logiciel open-source que nous avons choisi parmi les logiciels présentés dans le premier chapitre (voir page23). Ensuite, nous allons donner un aperçu sur le travail accompli à travers des captures d'écran et des codes réalisés.

II. Etude du logiciel choisi « OpenCourrier » :

Suite à la réalisation des tests et à l'évaluation présenté dans le premier chapitre, nous avons opté pour le logiciel OpenCourrier. Ce logiciel couvre les fonctionnalités suivantes :

- Ajouter, consulter archiver des courriers ;
- Trouvez, à l'aide du moteur de recherche, un courrier précis.
- Paramétrer l'application :
 - Gestion des utilisateurs et leurs droits d'utilisation ;
 - Gestion des différents services,
 - Gestion des organisations qui émettent le courrier,

II.1 Installation :

Afin de pouvoir installer ce logiciel, il faut avoir au préalable :

- Un serveur web ;
- PHP ;
- Un serveur de base de données MySQL ;

Pour ce fait nous avons choisi la plate-forme de développement Web Wampserver et l'IDE Netbeans pour l'implémentation.

➤ Wampserver :

Il s'agit d'une plateforme de développement Web, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP.

C'est un environnement comprenant deux serveurs (un serveur web Apache et un serveur de bases de données MySQL), un interpréteur de script (PHP), ainsi qu'une

administration SQL phpMyAdmin. Il permet donc d'installer en une seule fois tout le nécessaire au développement local du PHP.

➤ **L'interface Phpmyadmin :**

C'est est une interface gratuite réalisée en langage PHP pour le SGBD MySQL afin de faciliter la gestion des bases de données MySQL sur un serveur. Il s'agit de l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur PHP.

➤ **Le serveur apache :**

C'est le serveur Web le plus utilisé au monde actuellement, car il englobe plusieurs avantages dont :

- C'est un Serveur gratuit.
- Un niveau de performance élevé pour des besoins matériels modestes.
- Modulaire.
- Configurable.
- Robuste.
- Très portable (Linux, Windows ...)

➤ **Le serveur Mysql :**

MySQL est un système de gestion de bases de données relationnelles (SGBDR) libre, permettant De manipuler des instructions adressées à la base de données Sous forme de requêtes SQL

➤ **Le langage de requête Sql (Standard Query Language) :**

Sql est un langage permettant d'interroger les bases de données de manière simple. Il est doté d'une syntaxe particulière que l'on doit respecter pour que la communication avec la base se passe au mieux.

A part le fait d'envoyer directement les requêtes SQL telles quelles au SGBD, le but ultime de l'utilisation de SQL sera aussi d'inclure ces requêtes SQL dans un programme écrit dans un autre langage. Ceci permet de coupler le SGBD à un langage informatique, donc à un programme. Tel est le cas de MySQL avec PHP.

➤ **PHP (Hypertext Preprocessor):**

C'est un langage de script qui est principalement utilisé pour être exécuté par un serveur HTTP, mais il peut fonctionner comme n'importe quel langage interprété en utilisant les scripts et son interpréteur sur un ordinateur. PHP permet de développer des scripts suivant le modèle procédural et/ou le modèle objet.

➤ **Netbeans :**

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

II.2 Déploiement :

Pour déployer l'application nous avons décompressé le fichier téléchargé dans le répertoire de serveur web apache.

II.3 Création de la base de données :

Pour initialiser la base de données nous avons importé ces deux fichiers :

- ../data/mysql/init.sql
- ../data/mysql/init_metier.sql

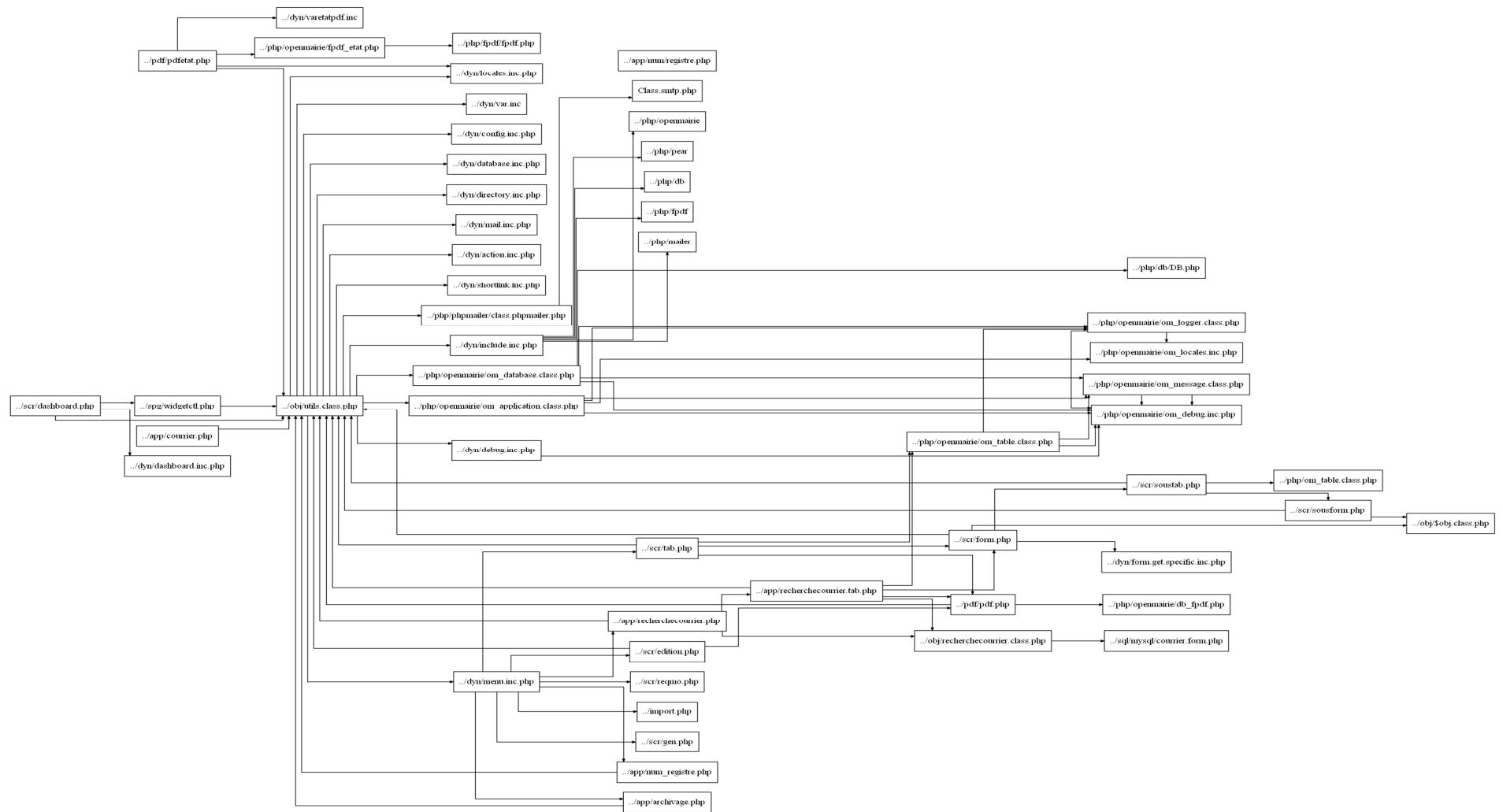
II.4 Connexion de l'utilisateur :

Pour la première authentification, l'utilisateur doit saisir les coordonnées suivantes :

- Login : admin
- Mot de passe : admin

II.5 schéma des fichiers de l'application :

Réalisation



III. Travail réalisé :

III.1 Modélisation des processus métier :

BPMN est un langage graphique standardisé et basé sur une structure XML. Il permet de dessiner des diagrammes représentant le déroulement et l'enchaînement de tâches. Un diagramme BPMN peut être modélisé via un outil de modélisation BPMN permettant l'export dans un format standard en XML.

BPMN est constitué d'un ensemble d'éléments de modélisation. Ces éléments sont constitués d'un symbole (objet graphique sur un diagramme) et d'une liste d'attributs (la plupart du temps invisible sur un diagramme).

De manière simple, les tâches représentent une action effectuée par un humain ou un service informatique. Ces actions sont liées entre elles par des flèches de connexion, nommées flux de séquence, qui représentent le cours du processus, son « flux ». Les portes représentent des points de jonction où le flux se divise en chemins parallèles ou conditionnels. Enfin des « événements » représentent toute chose se produisant et influençant le déroulement du flux (début, pause, fin, arrêt, erreur, attente de signal etc.). Tous ces éléments sont ensuite placés dans des bassins représentant les limites du processus. Ce bassin est divisé en couloirs représentant les acteurs du dispositif.

Le diagramme BPMN suivant représente l'ensemble des tâches effectuées lors de traitement du courrier « statistique service national » par le personnel de la faculté Génie électrique et informatique.



Figure III.1 : diagramme BPMN

A l'exécution :

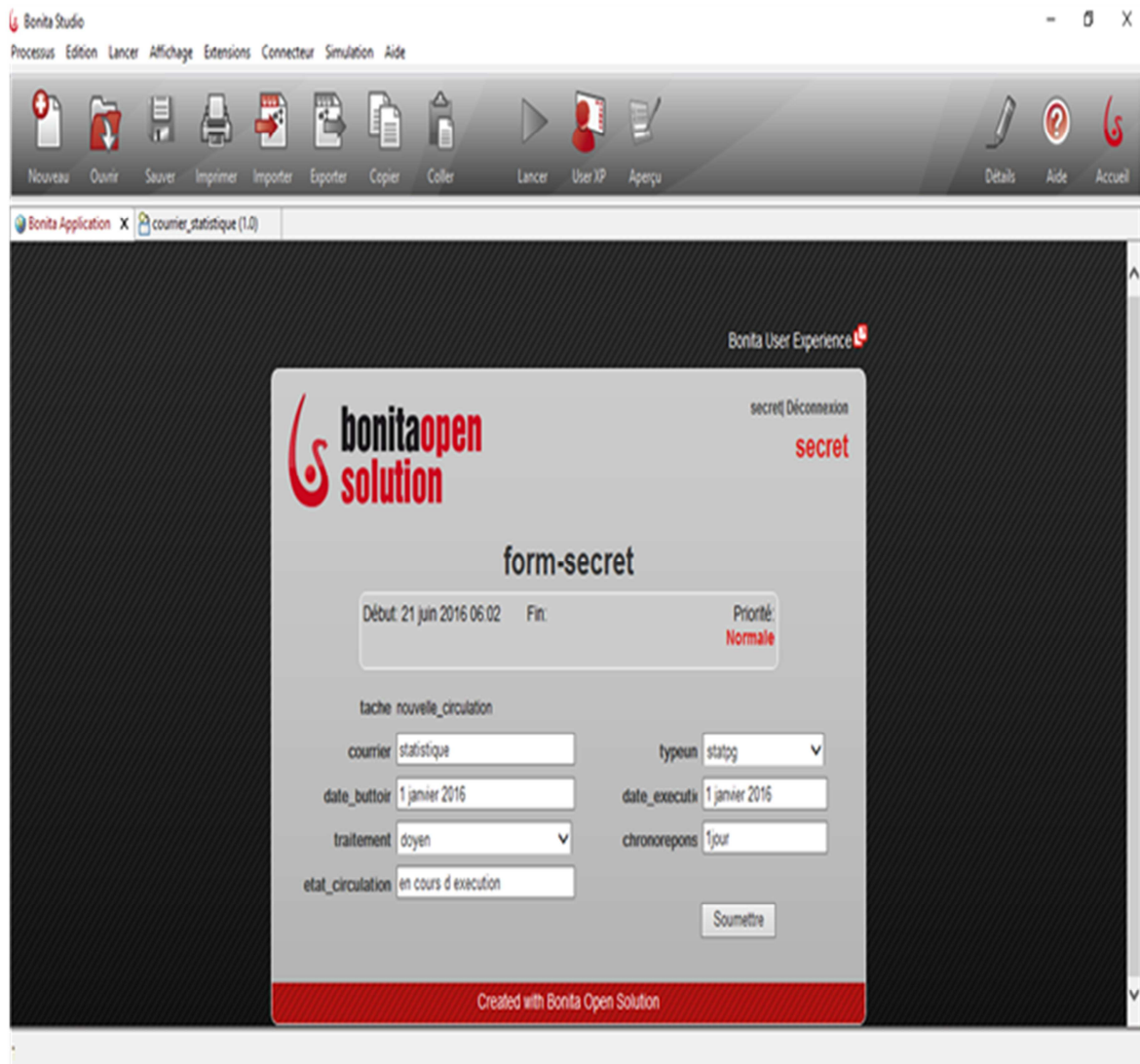


Figure III .2 : exécution du diagramme BPMN

III.2 L'implémentation :

Le fichier « ../gen/obj/circulation.class.php »

```
<?php
require_once ("../php/openmairie/dbformdyn.class.php");

class circulation_gen extends dbForm
{ var $table="circulation";
  var $clePrimaire="circulation";
  var $typeCle="N";
  var $retourformulaire;

  function setvalF($val) {
    //affectation valeur formulaire $this-
    >valF['circulation'] = $val['circulation'];

    $this->valF['courrier'] = $val['courrier'];
    $this->valF['traitement'] =
    $val['traitement'];

    if($val['datebuttoir']!=""){
      $this->valF['datebuttoir'] = $this->dateDB($val['datebuttoir']);
    }
    if($val['dateexecution']!=""){
      $this->valF['dateexecution'] = $this->dateDB($val['dateexecution']);
    }

    $this->valF['etat'] = $val['etat']; $this-
    >valF['observation'] = $val['observation']; $this-
    >valF['delaireponse'] = $val['delaireponse'];
    $this->valF['detailreponse'] =
    $val['detailreponse'];
  }

  //=====
  //cle primaire automatique [automatic primary key]
  //=====
  =

  function setId(&$db) {
    //numero automatique
    $this->valF[$this->table] = $db->nextId(DB_PREFIXE.$this->table);
  }

  function setValFAjout($val) {
    //numero automatique -> pas de controle ajout cle primaire
  }

  function verifierAjout() {
```

```
//numero automatique -> pas de verification de cle primaire
}

//=====
// verifier avant validation [verify]
//=====
= /**
 * Methode
 verifier */

function verifier($val = array(), &$db = NULL, $DEBUG = false) {
    // On appelle la methode de la classe
    parent parent::verifier($val, $db,
    $DEBUG);
    // On verifie si le champ n'est pas vide
    /*
        if ($this->valF['courrier'] ==
            "") { $this->correct = false;
                $this->addToMessage(_("Le champ")." "._("courrier")."
                _("est obligatoire"));
            }

        */
    }

    //=====
    == // Formulaire [form]
    //=====
    ==

function setType(&$form,$maj) {
    //type
    if ($maj==0){ //ajout $form-
        >setType('circulation','hidden');// cle automatique

        if($this->retourformulaire=='courrier')
            $form-
            >setType('courrier','hiddenstatic');
        else
            $form-
            >setType('courrier','select'); $form-
            >setType('traitement','text');

        if($this->retourformulaire=='') $form-
            >setType('datebuttoir','date');
        else
            $form->setType('datebuttoir','date2');

        if($this->retourformulaire=='') $form-
            >setType('dateexecution','date');
        else
            $form->setType('dateexecution','date2');
```

```

    $form->setType('etat','text'); $form-
    >setType('observation','textarea');
    $form->setType('delaireponse','text');
    $form-
    >setType('detailreponse','textarea');
}// fin ajout

if ($maj==1){ //modifier $form-
    >setType('circulation','hiddenstatic');
    if($this->retourformulaire=='courrier')
        $form->setType('courrier','hiddenstatic');
    else
        $form->setType('courrier','select');

    $form->setType('traitement','text');
    if($this->retourformulaire=='')
        $form->setType('datebuttoir','date');
    else
        $form->setType('datebuttoir','date2');
    if($this->retourformulaire=='')
        $form->setType('dateexecution','date');
    else
        $form->setType('dateexecution','date2');
    $form->setType('etat','text'); $form-
    >setType('observation','textarea'); $form-
    >setType('delaireponse','text'); $form-
    >setType('detailreponse','textarea');
}// fin modifier

if ($maj==2){ //supprimer $form-
    >setType('circulation','hiddenstatic'); $form-
    >setType('courrier','hiddenstatic'); $form-
    >setType('traitement','hiddenstatic'); $form-
    >setType('datebuttoir','hiddenstatic'); $form-
    >setType('dateexecution','hiddenstatic'); $form-
    >setType('etat','hiddenstatic'); $form-
    >setType('observation','hiddenstatic'); $form-
    >setType('delaireponse','hiddenstatic'); $form-
    >setType('detailreponse','hiddenstatic');
}//fin supprimer
}

function setOnChange(&$form,$maj) {
//javascript controle client $form-
    >setOnChange('circulation','VerifNum(this)'); $form-
    >setOnChange('courrier','VerifNum(this)'); $form-
    >setOnChange('datebuttoir','fdate(this)'); $form-
    >setOnChange('dateexecution','fdate(this)');
}

function setTaille(&$form,$maj) {
//taille des champs affiches (text) $form-
    >setTaille('circulation',8); $form-
    >setTaille('courrier',8); $form-

```

```

>setTaille('traitement',40); $form-
>setTaille('datebuttoir',10); $form-
>setTaille('dateexecution',10); $form-
>setTaille('etat',40); $form-
>setTaille('observation',80); $form-
>setTaille('delaireponse',40); $form-
>setTaille('detailreponse',80);
}

function setMax(&$form,$maj) {
//Longueur max en saisie (text) $form-
>setMax('circulation',8); $form-
>setMax('courrier',8);

    $form->setMax('traitement',40);
    $form->setMax('datebuttoir',10);
    $form-
    >setMax('dateexecution',10);
    $form->setMax('etat',40); $form-
    >setMax('observation',6); $form-
    >setMax('delaireponse',40);
    $form-
    >setMax('detailreponse',6);
}

function setLib(&$form,$maj) {
//Libelle des champs $form-
>setLib('circulation',_('circulation')); $form-
>setLib('courrier',_('courrier')); $form-
>setLib('traitement',_('traitement')); $form-
>setLib('datebuttoir',_('datebuttoir')); $form-
>setLib('dateexecution',_('dateexecution'));
    $form->setLib('etat',_('etat')); $form-
>setLib('observation',_('observation')); $form-
>setLib('delaireponse',_('delaireponse'));
    $form-
>setLib('detailreponse',_('detailreponse'));
}

function setSelect(&$form, $maj,&$db,$debug) {
    if(file_exists ("../sql/".$db->phptype."/". $this->table.".form.inc"))
        include ("../sql/".$db->phptype."/". $this->table.".form.inc");
    if($maj<2){
        // courrier
        $contenu=array(
        );
        $res = $db->query($sql_courrier);
        // Logger
        $this->addToLog("setSelect()[gen/obj]: db-
>query(\"".$sql_courrier."\"); ", VERBOSE_MODE);
        if (database::isError($res)) die($res-
            >getMessage().$sql_courrier);
        else{
            if ($debug == 1)

```

```

        echo " la requete ".$sql_courrier." est
executée<br>"; $contenu[0][0]='';
$contenu[1][0]=_('choisir')."&nbsp;"._('courrier');
$k=1;
        while ($row=& $res-
            >fetchRow()){
            $contenu[0][$k]=$row[0];
            $contenu[1][$k]=$row[1];
            $k++;
        }
        $form-
        >setSelect('courrier',$contenu); }// fin
        error db
    }// fin
maj }// fin
select

//=====
= // sous Formulaire [subform]
//=====
=

function setValsousformulaire(&$form,$maj,$validation,$idxformulaire,
$retourformulaire,$typeformulaire,&$db,$DEBUG=null){
    $this->retourformulaire =
    $retourformulaire; if($validation==0) {
        if($retourformulaire =='courrier') $form-
            >setVal('courrier', $idxformulaire);
    }// fin validation
}// fin setValsousformulaire

//=====
= // cle secondaire [secondary key]
//=====
=

}// fin classe

```

Le fichier « ../obj/circulation.class.php »

```
<?php
```

```
require_once ("../gen/obj/circulation.class.php");
```

```
class circulation extends circulation_gen {
```

```

    function circulation($id,&$db,$debug) {
        $this-
        >constructeur($id,$db,$debug);
    }// fin constructeur

```

```

function verifier($val, &$db, $DEBUG) {
    parent::verifier($val, $db,
        $DEBUG);
    /*
    if ($this->valF['traitement']==""){
        $this->correct=false;
        $this->addToMessage(_("service")." ".$_("execution")." ".$_("
courrier")." ".$_("obligatoire"));
    }

function setType(&$form, $maj) {
    //
    parent::setType($form,$maj);
    // compatibilite ascendante version 1.07+
    // variable $scanreponse_tache dans var.inc
    //
    $form->setType('courrier', 'hiddenstatic');
    $form->setType('objetcourrier', 'hiddenstatic');
    $form->setType('registre', 'hiddenstatic');
    $form->
    >setType('servicetraitement', 'hiddenstatic');
    //
    $form->setType('circulation', 'hiddenstatic');
    //
    $form->setType('archive_fichier', 'hidden');

    if($maj<2)//ajout et modif
    {
        $form->setType('etat', 'select');

        $form->setType('traitement', 'hiddenstatic');

        $form->setType('observation',
'textarea');
        if(isset($this->idxformulaire))
            $form->setType('delaireponse', 'comboD2');
        else
            $form->setType('delaireponse', 'comboD');
        $form->setType('datebuttoir', 'date2');
        $form->setType('dateexecution', 'date2');
        $form->setType('detailreponse', 'textarea');
    }
    else if($maj == 2)//suppression
    {
        $form->setType('etat', 'hiddenstatic');
        $form->setType('traitement', 'hiddenstatic');
        $form->setType('observation', 'hiddenstatic');
        $form->setType('delaireponse', 'hiddenstatic');
        $form->setType('datebuttoir', 'hiddenstatic');
        $form->setType('dateexecution', 'hiddenstatic');
        $form->setType('detailreponse', 'hiddenstatic');
    }
}

```

```

//
if ($maj == 0) {
    $form->setType('circulation', 'hidden');
    $form->setType('traitement', 'select');
}
}

function setGroupe(&$form, $maj) {
    // $form->setGroupe('delaireponse', 'D'); //
    $form->setGroupe('detailreponse', 'F');
}

function setRegroupe(&$form, $maj) {
    //
    $form->setRegroupe('courrier', 'D', _('courrier'), "collapsible");
    $form->setRegroupe('registre', 'G', '');
    $form->setRegroupe('objetcourrier', 'G', '');
    $form->setRegroupe('servicetraitement', 'F', '');
    //
    if ($maj == 1) {
        $form->setRegroupe('datebuttoir', 'D', _('circulation'), "collapsible");
    }

    else {
        $form->setRegroupe('circulation', 'D', _('circulation'),
            "collapsible");

        $form->setRegroupe('datebuttoir', 'G', '');
    }

    $form->setRegroupe('dateexecution', 'G', '');
    $form->setRegroupe('etat', 'G', '');
    $form->setRegroupe('traitement', 'F', '')

    $form->setRegroupe('observation', 'D', _('details'), "collapsible");
    $form->setRegroupe('delaireponse', 'G', '');
    $form->setRegroupe('detailreponse', 'F', '');
}

function setOnChange(&$form, $maj) {
    // javascript controle client
    $form->setOnChange('circulation', 'VerifNum(this)');
    $form->setOnChange('courrier', 'VerifNum(this)');
    $form->setOnChange('datebuttoir', 'fdate(this)');
    $form->setOnChange('dateexecution', 'fdate(this)');
}

function setSelect(&$form, $maj, &$db, $DEBUG)
{
    //

```



```

if (isset($GLOBALS["f"])) {
    $f = $GLOBALS["f"];
} else {
    // Ce cas ne doit jamais arrivé
    die("Erreur");
}
if(file_exists ("../sql/".$db->phptype."/".$this->table.".form.inc"))
    include ("../sql/".$db->phptype."/".$this->table.".form.inc");

if($maj<2) //ajout et mise a jour
{

    if(isset($this->idxformulaire))
        $courrier=$this->idxformulaire;
    else

        //$courrier=0;

        if ($f->getParameter('service_circulation') == '2')
        {
            $sql_souserv = $sql_servexec.$courrier;

        }
        else
            //if ($f->getParameter('service_circulation') == '1')
            {
                $sql_souserv = $sql_tousserv;
            }
            //else echo _("Erreur de configuration (voir /dyn/var.inc)");

    $res = $db->query($sql_souserv);

    if (database::isError($res))

        die($res->getMessage().$sql_souserv);
    else
    {

        if ($DEBUG == 1)
            echo " la requete ".$sql_souserv." est ex&ecute;cut&ecute;e<
br>";

        $contenu[0][0]="";
        $contenu[1][0]=_("votre_choix"
        );

        $k = 1;
        while($row=& $res->fetchRow())
        {

```

```

        if
            ($row[0]=="*
            ") { $k++;
        } else {
            // dans contenu[0][k] --> Le num du service = la valeur
            qu' on va entrer dans la base
            // dans contenu[0][k] --> Le numero du service et
            son libelle = ce qui apparait
            $contenu[0][$k]=$row[0];
            $contenu[1][$k]=$row[0].'
            '.$row[1]; $k++;
        }
    }
    $form->setSelect("traitement",$contenu);
}
//solde ->
oui/non
$contenu=array()
;
//$contenu[0][0]='';
$contenu[0][0]='en_cours
';
$contenu[0][1]='terminee
';
//$contenu[1][0]='Choisissez
';
$contenu[1][0]=_("en_cours")
;
$contenu[1][1]=_("terminee")
;

$form->setSelect("etat",$contenu);
//chronorep -> int (8)
$contenu[0][0]="courrier";// table
$contenu[0][1]="registre"; // zone
origine $contenu[1][0]="objetcourrier";
$contenu[1][1]="detailreponse";
$form
->setSelect("delaireponse",$contenu);

}
}

// *****
// argument supplementaire $db
// *****

function setValsousformulaire(&$form, $maj, $validation, $idxformulaire,

$retourformulaire, $typeformulaire, &$db) {
    // specifique : initialisation des services traitement de
    taches if ($validation == 0) {

```

```

if (($maj == 0) || ($maj==1)) { $form-
    >setVal($retourformulaire, $idxformulaire);
    // $form->setVal('courrier', $idxformulaire);
    //
    $temp = $db->query("SELECT * from
    service,courrier WHERE service.service =
    courrier.traitement
    AND courrier.courrier = ".$idxformulaire);

    if (database::isError($temp))
        { die($temp-
        >getMessage());
        }
    else{
        $row = $temp->fetchRow(DB_FETCHMODE_ASSOC); $form-
        >setVal("servicetraitement", "(".$row["service"].") ").
        $row['servicelib']);
        $form->setVal("objetcourrier",
        $row['objetcourrier']); $form->setVal("registre",
        $row['registre']);
    }
}
}
$this->idxformulaire = $idxformulaire;
}

//Function de mise à jour du champ nbtache de la table
courrier function updateNbCirculation($idCourrier, $db) {
    $sqlCount="SELECT count(circulation) FROM circulation WHERE
    courrier= $idCourrier";

    $nbcirculation=$db-
    >getOne($sqlCount); if
    (database::isError($nbcirculation))
        die($nbcirculation->getMessage());
    else {
        $sqlUpdate="UPDATE courrier SET nbcirculation=$nbcirculation
    WHERE courrier=$idCourrier";
        $res= $db->autoExecute("courrier",array('nbcirculation'
    => $nbcirculation),DB_AUTOQUERY_UPDATE,"courrier=".$idCourrier);
        if (database::isError($res))
            die($res-
            >getMessage());
    }
}

function triggerSupprimerApres($id, &$db = NULL, $val = array(), $DEBUG = false)
{
    $this->updateNbCirculation($val['courrier'], $db);
}

//Lier le courrier depart chronoreponse au courrier au quel la tache est liee
function triggerModifierApres($id, &$db = NULL, $val = array(), $DEBUG = false)
{
    //

```

```

if (isset($GLOBALS["f"])) {

    $f = $GLOBALS["f"];
} else {
    // Ce cas ne doit jamais
    arrivé die("Erreur");
}

if($this->valF['detailreponse']!="") {
    // Recuperation du courrier lie a la tache
    $sql_depart="SELECT * FROM courrier WHERE registre='".$this->valF[' delaireponse']."' AND typecourrier='Depart'";
    $res_depart=$db->query($sql_depart); $f->isDatabaseError($res_depart);
    if($res_depart->numrows()==1) {
        require_once
        "../obj/courrier_lie.class.php";
        $courriers_lies=new courrier_lie($f);
        $sql_courrier="SELECT * FROM courrier WHERE courrier='".$this->valF['
courrier']";

        $res_courrier=$db->query($sql_courrier); $f->isDatabaseError($res_courrier); $row_depart=$res_depart->fetchrow(DB_FETCHMODE_ASSOC);
        $row_courrier=$res_courrier->fetchrow(DB_FETCHMODE_ASSOC);
        // Test si Le courrier depart fait parti d'une affaire
        // et compte Le nombre de courriers Lies
        if($row_depart['liaison']!=0) {
            $sql_count_link_depart="SELECT count(*) FROM courrier
WHERE liaison='".$row_depart['liaison']";
            $nb_linked_depart=$db->getOne($sql_count_link_depart); $f->isDatabaseError($nb_linked_depart);
        } else {
            $nb_linked_depart=
            0;
        }
        // Test si Le courrier lie a la tache fait parti d'une affaire
        // et compte Le nombre de courriers Lies
        if($row_courrier['liaison']!=0) {
            $sql_count_link_courrier="SELECT count(*) FROM courrier
WHERE
liaison='".$row_courrier['liaison']"; $nb_linked_courrier=$db->getOne($sql_count_link_courrier); $f->isDatabaseError($nb_linked_courrier);
        } else {
            $nb_linked_courrier=
            0;
        }
        // Test si Les deux courriers font partis d'une affaire
        if($nb_linked_depart!= $nb_linked_courrier OR ($nb_linked_courrier==
0 AND $nb_linked_depart==0)) {

```

```

        if($nb_linked_depart!=0 AND $nb_linked_courrier!=0) {
            echo "<script> lienCirculation('".$nb_linked_depart."','".$nb_linked_courrier."','".$row_courrier['liaison']."','".$row_depart['liaison']."');</script>";

        } elseif($nb_linked_depart!=0 AND
            $nb_linked_courrier==0) { $courriers_lies-
                >addLink($row_courrier['courrier'],
$row_depart['liaison']);
        } elseif($nb_linked_depart==0 AND $nb_linked_courrier!=0) {

            $courriers_lies-
>addLink($row_depart['courrier'], $row_courrier['liaison']);
        } else { $code_liaison=$db-
            >nextId("code_liaison");
            $courriers_lies->addLink($row_depart['courrier'],
$code_liaison);
            $courriers_lies->addLink($row_courrier['courrier'],
$code_liaison);
        }
    }
}
}
}

// Envoi de mail lors de l'ajout
function triggerajouterapres($id, &$db = NULL, $val = array(), $DEBUG = false) {
    // Récupération de l'instance de la classe
    utils if (isset($GLOBALS["f"])) {
        $f = $GLOBALS["f"];
    } else {
        // Ce cas ne doit jamais arrivé
        die("Erreur");
    }
    // Si l'option notification par mail est
    activée if ($f->config['notification_email'])
    {
        //
        $recipient = array();
        //
        $sql = " SELECT email ";
        $sql .= " FROM om_utilisateur ";
        $sql .= " WHERE email NOT LIKE '' AND service =
        '".$val["traitement"]."'
";

        $res = $db->query($sql);
        if (database::isError($res))
        { die($res-
            >getMessage());
        } else {
            while ($row=& $res->fetchRow()) { if(preg_match("#^[a-z0-9._-
            ]+@[a-z0-9._-]{2,}\.[a-z]{2,4}$#i",
$row[0])) {

```

```

        $recipient[]=$row[0];
    }
}
//
if (count($recipient)>0) {
    $title = str_replace("[ville]", $f->collectivite['ville'],
$f-> config['notification_email_title']);
    $message = str_replace("[ville]", $f->collectivite['ville'],
$f-> config['notification_email_corps']);
    $message = str_replace("[id_task]", $this->valF['circulation'],
$message);
    // Test de l'envoi de mail dans un try/catch pour récupérer les
éventuelles

erreurs de la librairie
try {
    $ret = $f->sendMail($title, $message, $recipient);
} catch (phpmailerException
    $e) { $ret = false;
    $this->addLog("sendMail(): ". $e->errorMessage(),
DEBUG_MODE); } catch (Exception $e) {
    $ret = false;
    $this->addLog("sendMail(): ". $e->getMessage(), DEBUG_MODE);
}
// Gestion du message d'information à
l'utilisateur if ($ret === true) {
    $this->addToMessage(_("Envoi d'un email de notification au
service."));
} else {
    $this->addToMessage(_("Aucun email de notification au
service. Contactez votre administrateur."));
}
} else {
    $this->addToMessage(_("Aucun email de notification au service
car aucune adresse renseignée."));
}
}
//Lier le courrier depart chronoreponse au courrier au quel la tache est
liee
if($this->valF['detailreponse']!="") {
    if($this->valF['detailreponse']!="")
    {
        $sql_depart="SELECT * FROM courrier WHERE registre='".$this->valF[' detailreponse']."' AND typecourrier='Depart'";
        $res_depart=$db->query($sql_depart); $f->isDatabaseError($res_depart);
        if($res_depart->numrows()==1) {
            require_once
            "../obj/courrier_lie.class.php";
            $courriers_lies=new courrier_lie($f);

```

```

        $sql_courrier="SELECT * FROM courrier WHERE courrier=".$this->
        valF['courrier'];

        $res_courrier=$db->query($sql_courrier); $f-
        >isDatabaseError($res_courrier); $row_depart=$res_depart-
        >fetchrow(DB_FETCHMODE_ASSOC);
        $row_courrier=$res_courrier-
        >fetchrow(DB_FETCHMODE_ASSOC);
        if($row_depart['liaison']!=0) {
            $sql_count_link_depart="SELECT count(*) FROM courrier
WHERE liaison=".$row_depart['liaison'];
            $nb_linked_depart=$db-
            >getOne($sql_count_link_depart); $f-
            >isDatabaseError($nb_linked_depart);
        } else {
            $nb_linked_depart=
            0;
        }
        if($row_courrier['liaison']!=0) {
            $sql_count_link_courrier="SELECT count(*) FROM
courrier
WHERE liaison=".$row_courrier['liaison']; $nb_linked_courrier=$db-
            >getOne($sql_count_link_courrier); $f-
            >isDatabaseError($nb_linked_courrier);

        } else {
            $nb_linked_courrier=
            0;
        }
        if($nb_linked_depart!=$nb_linked_courrier
OR ( $nb_linked_courrier==0 AND $nb_linked_depart==0)) {
            if($nb_linked_depart!=0 AND $nb_linked_courrier!=0) {
                echo "<script>
lienCirculation(".$nb_linked_depart.",",",
$nb_linked_courrier.",",", $row_courrier['liaison'],",", $row_depart['liaison'],");</
script>";

            } elseif($nb_linked_depart!=0 AND
$nb_linked_courrier==0) { $courriers_lies-
            >addLink($row_courrier['courrier'],
$row_depart['liaison']);
            } elseif($nb_linked_depart==0 AND
$nb_linked_courrier!=0) { $courriers_lies-
            >addLink($row_depart['courrier'],
$row_courrier['liaison']);
            } else { $code_liaison=$db-
            >nextId("code_liaison");
            $courriers_lies->addLink($row_depart['courrier'],
$code_liaison);
            $courriers_lies->addLink($row_courrier['courrier'],
$code_liaison);
        }
    }
}

```

```

    }
}
$this->updateNbCirculation($val['courrier'], $db);
}
} // fin class

```

?>

Le fichier « ../sql/mysql/ajoutercirculation.inc »

<?php

```

include('../gen/sql/mysql/circulation.inc.php');

//
$ent = _("courrier");

if(isset($idx)){ if($idx != '']')
    if (trim($idx!=''))
        $ent = $ent."><font id='idz1'>&nbsp;".$idx."&nbsp;</font>";
    }
    if(isset($idz) ){ if (trim($idz!=''))
        $ent = $ent."&nbsp;<font
        id='idz1'>&nbsp;".strtoupper($idz)."&nbsp;</font>";
    }

//
$edition = "";

//
$sousformulaire=array('circulation','dossier');

$table=DB_PREFIXE."courrier
inner join ".DB_PREFIXE."service on courrier.traitement=service.service";

$champAffiche=array(
    "courrier.courrier
    as id",
    "registre as
    '"._("registre")."',
    "nature as
    '"._("nature")."',
    "concat(substring(datearrivee,9,2),'/',substring(datearrivee,6,2
), '/', substring(datearrivee,1,4)) as '"._("Date d'arrivee ou
signature").'",
    "concat(substring(datecourrier,9,2),'/',substring(datecourrier,6,2
), '/', substring(datecourrier,1,4)) as '"._("datecourrier")."',
    "emetteurnom as '"._("emetteur_destinataire")."',
    "servicelib as '"._("traitement")."',
    "substring(objetcourrier, 1,20) as
    '"._("objetcourrier")."',
    /*"nbtache as '"._("Tache(s)")."'*/
);

```



```

$champRecherche=array( "courrier", "registre", "nature", "emetteurnom",
                        "emetteurprenom", "emetteur", "emetteurad1", "emetteurad2",
                        "emetteurcp", "emetteurville", "objetcourrier", "courriertype", "pj",
                        "referent"
                        );

// tri
$tri= " order by courrier.courrier desc";
// suivant service
if($_SESSION['service
']== '*')
    $selection="";
else
    $selection=" where traitement='".$_SESSION['service']."'";

if ($retourformulaire==
    'emetteur_destinataire') $selection=" where
    courrier.emetteur = '".$_IDX.'"';

$href[0]['lien
']= "#";
$href[2]['lien
']= "#";
$href[1]=
array(
    "lien" => "../scr/form.php?obj=circulation&idx=",
    "id" =>
"&tricol=".$_tricol."&origin=".$_obj."&premier=".$_premier."&
recherche=".$_recherche1."&selectioncol=".$_selectioncol,
    "lib" => "<span class=\"om-icon om-icon-16 om-icon-fix add-16\"
title=\"\"._(\"Ajouter_circulation\")\">\"._(\"Ajouter\")\"</span>",
);
$href[5] = array(
    "lien" => "../app/courrier.php?idx=",
    "id" =>
"&tricol=".$_tricol."&origin=".$_obj."&premier=".$_premier."&
recherche=".$_recherche1."&selectioncol=".$_selectioncol,
    "lib" => "<span class=\"om-icon om-icon-16 om-icon-fix tb-16\"
title=\"\"._(\"Tableau de bord du courrier\")\">\"._(\"TB\")\"</span>",
);

?>

```

le fichier « ../sql/mysql/circulation.inc.php »

```
<?php
```

```
include('../gen/sql/mysql/circulation.inc.php');
```

```

if(isset($idx)){ if($idx != '']')
    if (trim($idx!=''))
        $ent = $ent."<font id='idz1'>&nbsp;".$idx."&nbsp;</font>";
    }
    if(isset($idz) ){ if (trim($idz!=''))
        $ent = $ent."&nbsp;<font
        id='idz1'>&nbsp;".strtoupper($idz)."&nbsp;</font>";
    }
}

$href[0]['lien']= "#";
//$href[1]['Lien
']= "#";
//$href[2]['Lien
']= "#";
?>

```

le fichier « ../sql/mysql/circulation.form.php »

```

<?php

include('../gen/sql/mysql/circulation.form.inc.php');

        $champs = array(
            "courrier", "circulation", "datebuttoir", "dateexecution", "etat",
            "circulation.traitement","observation", "delaireponse", "detailreponse"
        );

$sql_servexec = "SELECT DISTINCT service,
                    servicelib FROM service
                    INNER JOIN courrier
                    ON service.servicerattachement =
                    courrier.traitement AND courrier.courrier =
                    ";

$sql_tousserv = "SELECT DISTINCT service, servicelib FROM service";

?>

```

le fichier « ../gen/sql/mysql/circulation.form.php »

```

<?php

$DEBUG=0;
$ico="";
$ent="option->circulation";
$tableSelect=DB_PREFIXE."circul
ation";
$champs=array("circulation","courrier","traitement","datebuttoir","dateexecution","et
at"," observation","delaireponse","detailreponse");
//champs select
$sql_courrier="select * from ".DB_PREFIXE."courrier";
?>

```

le fichier « ../sql/mysql/circulation_widget.php »

```
<?php
/**
 * //Ce script permet de définir les paramètres d'affichage du listing des
 * //circulation dans le contexte d'un widget sur le tableau de bord.
 */

\} On surcharge le script des paramètres permettant d'afficher le
\} listing standard des courriers.
include('../sql/mysql/circulation.inc');

//
$serie = 10;
$href = array();

?
```

le fichier « ../sql/mysql/circulation.pdf.php »

```
<?php

$DEBUG=0;
$orientation='L';// orientation P-> portrait L-
>paysage $format='A4';// format A3 A4 A5
$police='arial';
$margeleft=10;// marge
gauche $margetop=5;//
marge haut
$margeright=5;// marge
droite

$border=1; // 1 -> bordure 0 -> pas de
bordure $C1=0;// couleur texte R
$C2=0;// couleur texte V
$C3=0;// couleur texte B
$size=10; //taille POLICE
$height=4; // hauteur ligne
tableau $align='L';

$fond=1;// 0- > FOND transparent 1 ->
fond $C1fond1=234;// couleur fond R 241
$C2fond1=240;// couleur fond V 241
$C3fond1=245;// couleur fond B 241
$C1fond2=255;// couleur fond R
$C2fond2=255;// couleur fond V
$C3fond2=255;// couleur fond B
$libtitre='Liste circulation'; //
libelle titre

$flagsessionliste=0;// 1 - > affichage session liste ou 0 -> pas
d'affichage $bordertitre=0; // 1 -> bordure 0 -> pas de bordure
```

```

$aligntitre='L'; // L,C,R
$heighttitre=10;// hauteur ligne
titre
$grastitre='B';//$gras='B' -> BOLD OU
$gras='' $fondtitre=0; //0- > FOND
transparent 1 -> fond
$C1titrefond=181;// couleur fond R
$C2titrefond=182;// couleur fond V
$C3titrefond=188;// couleur fond B
$C1titre=75;// couleur texte R
$C2titre=79;// couleur
texte V $C3titre=81;//
couleur      texte      B
$sizetitre=15;
$flag_entete=1;//entete colonne : 0 -> non affichage , 1 ->
affichage $fondentete=1;// 0- > FOND transparent 1 -> fond
$heightentete=10;//hauteur ligne entete colonne
$C1fondentete=210;// couleur
fond R $C2fondentete=216;//
couleur fond V
$C3fondentete=249;// couleur
fond B $C1entetetxt=0;//
couleur texte R
$C2entetetxt=0;// couleur
texte V $C3entetetxt=0;//
couleur texte B
$C1border=159;// couleur
texte R $C2border=160;//
couleur texte V
$C3border=167;// couleur
texte B
$l0=24; // largeur colone -> champs 0 -
circulation $be0='L';// border entete
colone
$b0='L';// border cellule colone
$ae0='C'; // align cellule
entete colone $a0='L';

$l1=24; // largeur colone -> champs 1 -
courrier $be1='L';// border entete
colone
$b1='L';// border cellule colone
$ae1='C'; // align cellule
entete colone $a1='L';

$l2=120; // largeur colone -> champs 2 -
traitement $be2='L';// border entete colone
$b2='L';// border cellule colone
$ae2='C'; // align cellule
entete colone $a2='L';

$l3=30; // largeur colone -> champs 3 -
datebuttoir $be3='L';// border entete
colone
$b3='L';// border cellule colone
$ae3='C'; // align cellule
entete colone $a3='L';

```

```
$l4=82; // largeur colone -> champs4 -
dateexecution $be4='LR';// border entete
colone
$b4='LR';// border cellule
colone $ae4='C'; // align
cellule entete colone $a4='L';
$widthtableau=280;
$bt=1;// border 1ere et derniere ligne du tableau par page->0 ou 1
$sql="select circulation, courrier, traitement,
concat(substring(datebuttoir,9,2),'/',
substring(datebuttoir,6,2),'/',substring(datebuttoir,1,4)) as datebuttoir,concat(
substring(dateexecution,9,2),'/',substring(dateexecution,6,2),'/',substring(dateex
ecution, 1,4)) as dateexecution from ".DB_PREFIXE."circulation";
?>
```

le fichier « ../sql/mysql/courrier.pdf.php »

<?php

```
$DEBUG=0;
$orientation='L';// orientation P-> portrait L-
> paysage $format='A3';// format A3 A4 A5
$police='arial';
$margeleft=5;// marge
gauche $margetop=10;//
marge haut
$margeright=5;// marge
droite
$border=1; // 1 -> bordure 0 -> pas de bordure

$C1=0;// couleur
texte R $C2=0;//
couleur texte V
$C3=0;// couleur
texte B

$size=10; //taille POLICE
$height=4.6; // hauteur ligne
tableau $align='L';
$fond=1;// 0- > FOND transparent 1 -> fond

$C1fond1=234;// couleur
fond R 241 $C2fond1=240;//
couleur fond V 241
$C3fond1=245;// couleur
fond B 241
```

```

$C1fond2=255;// couleur
fond R $C2fond2=255;//
couleur fond V
$C3fond2=255;// couleur
fond B

$libtitre='Liste courrier'; // libelle titre
$flagsessionliste=0;// 1 -> affichage session liste ou 0 -> pas
d'affichage $bordertitre=0; // 1 -> bordure 0 -> pas de bordure
$aligntitre='L'; // L,C,R
$heighttitre=10;// hauteur ligne
titre

$grastitre='B';//$gras='B' -> BOLD OU
$gras='' $fondtitre=0; //0- > FOND
transparent 1 -> fond
$C1titrefond=181;// couleur fond R
$C2titrefond=182;// couleur fond V
$C3titrefond=188;// couleur fond B

$C1titre=75;// couleur
texte R $C2titre=79;//
couleur texte V
$C3titre=81;// couleur
texte B

$sizetitre=15;
$flag_entete=1;//entete colonne : 0 -> non affichage , 1 ->
affichage $fondentete=1;// 0- > FOND transparent 1 -> fond
$heightentete=10;//hauteur ligne entete colonne
$C1fondentete=210;// couleur
fond R $C2fondentete=216;//
couleur fond V
$C3fondentete=249;// couleur
fond B

$C1entetetext=0;// couleur
texte R $C2entetetext=0;//
couleur texte V
$C3entetetext=0;// couleur
texte B

$C1border=159;// couleur
texte R $C2border=160;//
couleur texte V
$C3border=167;// couleur
texte B

$l0=24; // largeur colone -> champs 0 -
registre $be0='L';// border entete
colone
$b0='L';// border cellule colone
$ae0='C'; // align cellule
entete colone $a0='L';

```

```
$l1=24; // largeur colone -> champs 1
- nature $be1='L';// border entete
colone
$b1='L';// border cellule colone
$ae1='C'; // align cellule
entete colone $a1='L';

$l2=30; // largeur colone -> champs2 -
typecourrier $be2='LR';// border entete
colone
$b2='LR';// border cellule
colone $ae2='C'; // align
cellule entete colone $a2='L';
//
$l3=30; // largeur colone -> champs2 -
datearrivee $be3='LR';// border entete
colone
$b3='LR';// border cellule
colone $ae3='C'; // align
cellule entete colone $a3='L';

$l4=30; // largeur colone -> champs2 -
datecourrier $be4='LR';// border entete
colone
$b4='LR';// border cellule
colone $ae4='C'; // align
cellule entete colone $a4='L';

$l5=30; // largeur colone -> champs2 -
emetteur $be5='LR';// border entete
colone
$b5='LR';// border cellule
colone $ae5='C'; // align
cellule entete colone $a5='L';

$l6=80; // largeur colone -> champs2 -
objetcourrier $be6='LR';// border entete
colone
$b6='LR';// border cellule
colone $ae6='C'; // align
cellule entete colone $a6='L';

$l7=80; // largeur colone -> champs2 - copiea $be7='LR';// border entete colone

$b7='LR';// border cellule
colone $ae7='C'; // align
cellule entete colone $a7='L';

$l8=80; // largeur colone -> champs2 -
traitement $be8='LR';// border entete
colone
```

```
$b8='LR';// border cellule  
colone $ae8='C'; // align  
cellule entete colone $a8='L';
```

```
$widthtableau=408;  
$bt=1;// border 1ere et derniere ligne du tableau par page->0 ou 1  
$sql="select registre, nature, typecourrier, datearrivee, datecourrier,  
emetteur, objetcourrier, copiea, traitement from ".DB_PREFIXE."courrier";  
?>
```

III.3 Les interfaces de notre systèmes

Réalisation

localhost / localhost / ope X openMaine :: openCourr X Boîte de réception - merie X

localhost/opencourrier_3.3.1/scr/dashboard.php

utilis | tizi ouzou | (*) tous services | Mot de passe | Deconnexion

Tableau de bord

► Courrier
► circulation
► Traitement
► Export

Tableau De Bord

Votre session est maintenant ouverte.

Bienvenue utilis dans votre tableau de bord paramétrable en cliquant ici

notification

| Notification | Courrier | Service | Date | Vu |
|---------------------------------|----------|---------|------|----|
| Aucun enregistrement. | | | | |
| Acceder au lien | | | | |

Courrier Arrivee

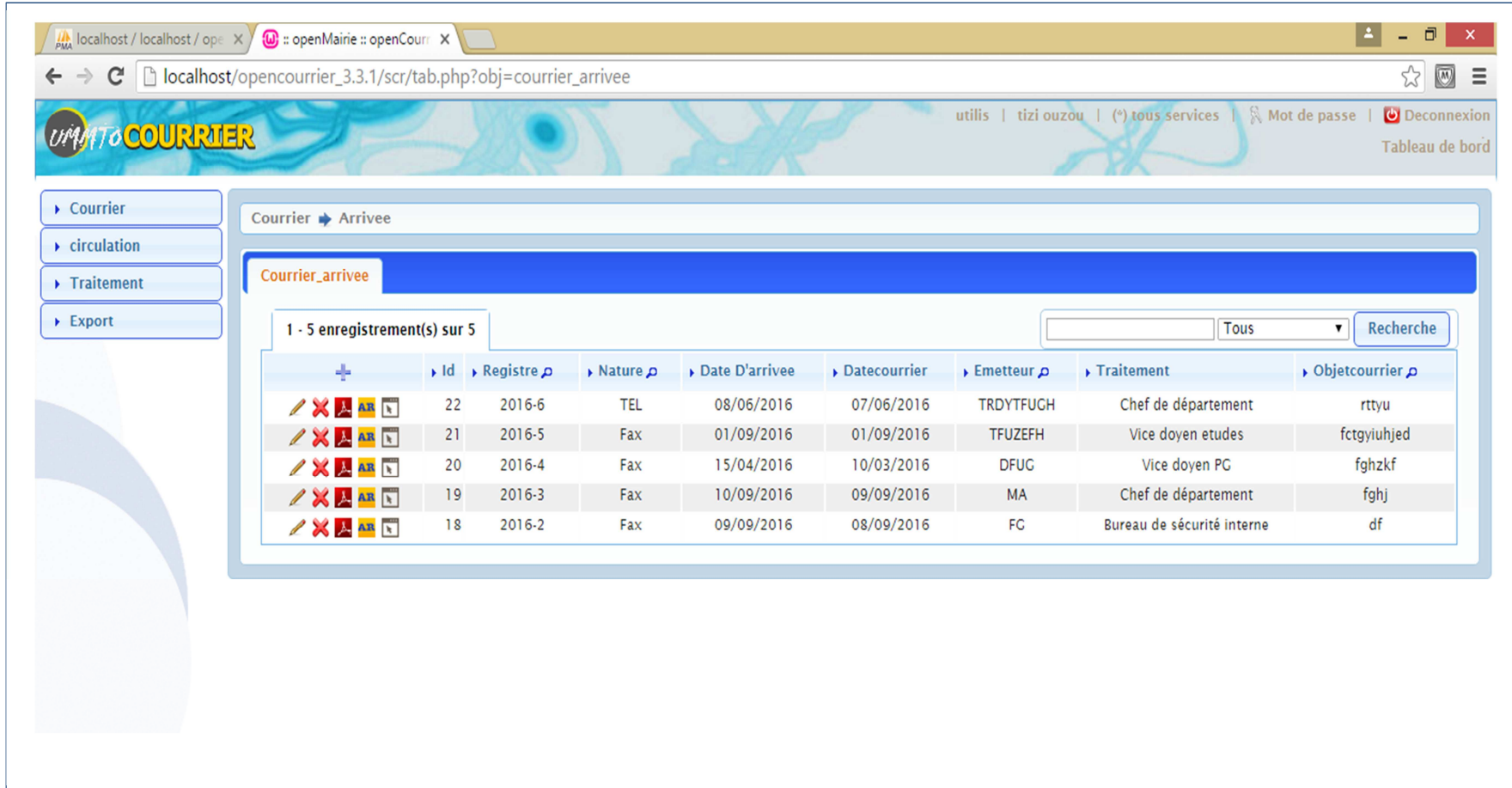
| Registre | Date D'arrivee | Datecourrier | Emetteur | Objetcourrier | Traitement |
|---------------------------------|----------------|--------------|-----------|---------------|----------------------------|
| 2016-6 | 08/06/2016 | 07/06/2016 | TRDYTFUGH | rttyu... | Chef de département |
| 2016-5 | 01/09/2016 | 01/09/2016 | TFUZEFH | fctgyihjed... | Vice doyen etudes |
| 2016-4 | 15/04/2016 | 10/03/2016 | DFUG | fghzkf... | Vice doyen PG |
| 2016-3 | 10/09/2016 | 09/09/2016 | MA | fghj... | Chef de département |
| 2016-2 | 09/09/2016 | 08/09/2016 | FG | df... | Bureau de sécurité interne |
| Acceder au lien | | | | | |

circulation

| Circulation | Courrier | Traitement | Datebuttoir | Dateexecution | Etat | Delaireponse |
|---------------------------------|----------|---|-------------|---------------|----------|--------------|
| 1 | 10 | | 21/09/2016 | 14/09/2016 | en_cours | 4jours |
| 6 | | Adjoint du
0 personnel et
enseignant | 00/00/0000 | 00/00/0000 | en_cours | |
| 5 | | Adjoint chef-dep
0 chargé scolarité de
graduation | 14/09/2016 | 14/09/2016 | en_cours | 3jours |
| Acceder au lien | | | | | | |

Figure III.3 : Interface « tableau de bord »

Réalisation



localhost / localhost / ope X :: openMairie :: openCourr X

localhost/opencourrier_3.3.1/scr/tab.php?obj=courrier_arrivee

utilis | tizi ouzou | (*) tous services | Mot de passe | Deconnexion
Tableau de bord

Courrier

↳ Courrier

↳ circulation

↳ Traitement

↳ Export

Courrier ➔ Arrivee

Courrier_arrivee

1 - 5 enregistrement(s) sur 5

Tous Recherche





















| | Id | Registre | Nature | Date D'arrivee | Datecourrier | Emetteur | Traitement | Objetcourrier |
|---|----|----------|--------|----------------|--------------|-----------|----------------------------|---------------|
|     | 22 | 2016-6 | TEL | 08/06/2016 | 07/06/2016 | TRDYTFUGH | Chef de département | rttyu |
|     | 21 | 2016-5 | Fax | 01/09/2016 | 01/09/2016 | TFUZEFH | Vice doyen etudes | fctgyiuhjed |
|     | 20 | 2016-4 | Fax | 15/04/2016 | 10/03/2016 | DFUG | Vice doyen PG | fghzxf |
|     | 19 | 2016-3 | Fax | 10/09/2016 | 09/09/2016 | MA | Chef de département | fghj |
|     | 18 | 2016-2 | Fax | 09/09/2016 | 08/09/2016 | FC | Bureau de sécurité interne | df |

Figure III. 4 : Interface « courrier arrivé »

Réalisation

localhost / localhost / ope X :: openMairie :: openCourr X :: openMairie :: openCourr X

localhost/opencourrier_3.3.1/app/courrier.php?idx=20&tricol=&origin=courrier_arrivee&premier=0&recherche=&selectioncol=

utilis | tizi ouzou | (*) tous services | Mot de passe | Deconnexion

Tableau de bord

Courrier | Tableau De Bord

Cette page rassemble les informations relatives a un courrier.

Courrier

Courrier no 2016-4 du 10/03/2016 arrivee le 15/04/2016
Emetteur : DFUG
Objet : fghzxf
Traitement : Vice doyen PG Vice doyen PG

circulation

| Circulation | Traitement | Date buttoir | Date execution | delaireponse | Observation |
|-------------|---|--------------|----------------|--------------|-------------|
| 5 | Adjoint chef-dep chargé scolarité de graduation | 14/09/2016 | 14/09/2016 | 3jours | RAS |

dossier

| Dossier | Fichier | Date du dossier | Type de dossier | Observation |
|---------|------------|-----------------|-----------------|-------------|
| 8 | 0/a_20.pdf | 27/04/2016 | arrive | |

Courriers lies

| Courrier | Type de registre | N de registre | Date | Emetteur |
|----------|------------------|---------------|------|----------|
|----------|------------------|---------------|------|----------|

Figure III.5 : Interface « tableau de bord du courrier »

Réalisation

localhost / localhost / ope X :: openMairie :: openCourr X Boîte de réception - merr X

localhost/opencourrier_3.3.1/app/recherchecourrier.php

utilis | tizi ouzou | (*) tous services | Mot de passe | Deconnexion
Tableau de bord

Courrier Recherche

courrier

typecourrier
Tous

registre

Date d'arrivée ou signature
Debut: Debut:

Fin: Fin:

objetcourrier

correspondant

nom prenom

adresse Complement adresse

CP ville

traitement

traitement
votre choix

diffusion referent

Utilisation de % pour zones de saisie : %BAS finit par 'BAS' %BAS commence par 'BAS' %BAS contient 'BAS'

Recherche

Figure III.6: Interface « formulaire de recherche d'un courrier »

Réalisation

The screenshot shows a web browser window with the URL `localhost/opencourrier_3.3.1/scr/form.php?obj=courrier_arrivee&tricol=`. The page features a header with the 'openMairie' logo and navigation links like 'utilis', 'tizi ouzou', '(* tous services', 'Mot de passe', and 'Deconnexion'. A sidebar on the left contains a menu with 'Courrier', 'circulation', 'Traitement', and 'Export'. The main content area is titled 'Courrier Arrivee' and includes tabs for 'Courrier_arrivee' and 'Dossier'. The 'Courrier_arrivee' tab is active, displaying a form with several sections: 'Scan_pdf' (with a message 'Pas de scan disponible sur le serveur'), 'courrier' (with a dropdown for 'typecourrier arrivee __ nature'), 'horodatage' (with date and time fields), 'emetteur' (with fields for 'code', 'civilite', 'nom', 'prenom', 'adresse', 'creation', 'CP', 'ville', and 'type_correspondant'), and 'Details' (with fields for 'objet', 'Bible', and 'courriertype').

Figure III.7 : Interface « formulaire d'ajout d'un courrier »

Réalisation

localhost / localhost / ope X openMaine :: openCourr X

localhost/opencourrier_3.3.1/scr/tab.php?obj=circulation

utilis | tizi ouzou | (*) tous services | Mot de passe | Deconnexion

Tableau de bord

UMMTO COURRIER

Option Courrier

Circulation

1 - 3 enregistrement(s) sur 3

Tous Recherche

| | Circulation | Courrier | Traitement | Datebuttoir | Dateexecution | Etat | Delaireponse |
|--|-------------|----------|---|-------------|---------------|----------|--------------|
| | 1 | 10 | | 21/09/2016 | 14/09/2016 | en_cours | 4jours |
| | 6 | 5 | Adjoint du personnel et enseignant | 00/00/0000 | 00/00/0000 | en_cours | |
| | 5 | 20 | Adjoint chef-dep chargé scolarité de graduation | 14/09/2016 | 14/09/2016 | en_cours | 3jours |

Figure III.8 : Interface « circulation »

The screenshot shows a web browser window with the URL `localhost/opencourrier_3.3.1/scr/form.php?obj=circulation&idx=22&tricol=&origin=ajoutercirculation&premier=0&recherche=&selectioncol=`. The page has a header with a logo and a breadcrumb trail: `Option > Courrier > 22 > 22`. A left sidebar contains navigation links: `Courrier`, `circulation`, `Traitement`, and `Export`. The main content area is titled `Circulation` and contains two sections: `courrier` and `details`. The `courrier` section includes fields for `datebuttoir`, `dateexecution`, `etat` (a dropdown menu currently showing `en_cours`), and `traitement`. The `details` section includes a large text area for `observation`, a `delaireponse` field with a slider, and a `detailreponse` text area. At the bottom, there is a button `Modifier l'enregistrement de la table : 'Circulation'` and a `Retour` link.

Figure III.9: Interface « formulaire d'ajout d'une circulation »

III. Conclusion :

Dans ce chapitre nous avons, au premier lieu, présenté le logiciel OpenCourrier puis les différents outils et langages que nous avons utilisé pour implémenter notre système. Par la suite, nous avons présenté l'implémentation et les interfaces de notre système.

Conclusion

générale

Conclusion général

L'objectif de notre travail était de développer un système de gestion du courrier pour la Faculté de Génie Electrique et Informatique

Pour mener à terme notre application, nous avons donné un aperçu général sur les concepts concernant la gestion électronique du courrier à savoir la dématérialisation, la gestion des processus métier et les logiciels de GEC en open source.

Ensuite, on a présenté un chapitre sur analyse et conception de notre application, puis un chapitre sur la réalisation de l'application.

L'application à laquelle nous avons abouti, répond à plusieurs de ces fonctionnalités à savoir :

- gestion des courriers entrant/sortants.
- archivage des courriers.
- circulation d'un courrier au sein de l'entreprise (le suivi).

Durant les trois mois de stage à la faculté Génie Electrique et Informatique où nous avons mené une étude d'analyse et de conception sur la gestion du courrier, et d'après l'étude qui précède, il apparaît que l'informatisation de la gestion du courrier serait d'un apport considérable pour FGEL.

Bibliographie

Bibliographie

- [1] : www.la-gec.com
- [2] : www.ged.prestataires.com/conseils/dematerialisation-de-documents
- [3] : www.wikipedia.org/wiki/Scanner
- [4] : www.theses.insa-lyon.fr/publication/2010ISAL0016/these.pdf
- [5]:www.memoireonline.com/10/10/3966/m_Guide-pour-implementer-un-outil-BPM-une-opportunite-pour-reconfigurer-ses-processus-metiers5.html
- [6] : www.logiciellibre.net/ossfswhat.php
- [7] :www.memoireonline.com/12/08/1760/m_Plateformes-de-services-integres-pour--mobiles2.html