

Ministère des Enseignements Secondaire et Supérieur
(MESS)

Secrétariat Général

Université Polytechnique de Bobo-Dioulasso (U.P.B.)

Ecole Supérieure d'Informatique (E.S.I)



Cycle des Ingénieurs de Travaux Informatiques (C.I.T.I)
Option : Analyse et Programmation (A.P)

Rapport de fin de cycle

THEME : « Gestion des congés et absences professionnels à la CARFO »

Période du 1^{er} octobre au 31 décembre 2013

Auteurs : Arnaud Boureima LOUE
Pascal OUEDRAOGO

Maitre de stage

M. Aboudou TRAORE
Informaticien à la CARFO

Superviseur

Dr Telesphore TIENDREBEOGO
Enseignant à l'UPB

Année Académique : 2012-2013

DEDICACE

A ALLAH Tout Puissant pour m'avoir permis d'atteindre ce niveau.

A toute ma famille et particulièrement à mes parents, à mes frères qui m'ont apporté leur soutien inconditionnel.

A mes amis et à tous ceux qui ont contribué d'une façon ou d'une autre à l'aboutissement de ce travail.

Enfin à tous ceux qui concourent au développement de l'informatique au niveau mondial et particulièrement au Burkina Faso.

Arnaud Bouréma LOUE

A Dieu le Miséricordieux pour m'avoir permis de voir ce monde.

A toute ma famille particulièrement à ma maman et aux frères qui ont été présents à mes côtés.

Enfin à tous ceux qui ont contribué à l'aboutissement de ce projet.

Pascal OUEDRAOGO

REMERCIEMENTS

Au terme de ce stage, nous tenons vivement à remercier un certain nombre d'acteur qui ont concouru à l'aboutissement de notre projet.

Il s'agit entre autre :

- du Directeur Général de la **CARFO B. Olivier SAWADOGO** pour avoir bien voulu nous accepter dans sa société ;
- de monsieur **Noel BADO**, directeur du DIS ;
- de notre maître de stage monsieur **TRAORE Aboudou** pour son guide et ses conseils ;
- de notre superviseur **Dr Telesphore TIENDREBEOGO** pour son assistance et ses conseils ;
- à tout le personnel de la **CARFO** auprès duquel nous avons trouvé un climat très social et ambiant qui nous a permis de disposer des informations indispensables pour l'aboutissement de notre projet.

Nos remerciements vont aussi à l'endroit :

- de l'administration de l'Université Polytechnique de Bobo-Dioulasso en particulier celle de l'Ecole Supérieure d'Informatique (ESI) ;
- du corps enseignant de l'ESI pour avoir assuré notre formation ;
- de tous ceux et toutes celles qui ont contribué d'une manière ou d'une autre à la réalisation de ce stage et à son bon déroulement.

SIGLES, ABREVIATIONS ET ACRONYMES

Sigles et abréviations	Significations
API	Application Programming Interface
CARFO	Caisse Autonome de Retraite des FONctionnaires
EDI	Environnement de Développement Intégré
ESI	Ecole Supérieure d'Informatique
DIS	Direction de l'Informatisation et des Statistiques
DRH	Directeur des Ressources Humaines
HTML	Hyper-Text Markup Language
JDO	Java Data Objects
JSF	Java Server Faces
JSP	Java Server Pages
SAP	Service d'Administration du Personnel
SGBD	Système de Gestion de Base de Données
SIGA	Système Intégré de Gestion Automatisée
SMS	Short Message System
SQL	Structured Query Language
UML	Unified Modeling Language
UP	Unified Process
UPS	Uninterruptable Power System
XHTML	eXtensible Hyper-Text Markup Language

PREAMBULE

L'Université Polytechnique de Bobo-Dioulasso (UPB), jadis Centre Universitaire de Bobo-Dioulasso (CUPB), a été créée le 23 mai 1997 par décret n°97-54/PRES/PM/MESSRS. Installée à une quinzaine de kilomètres de l'ouest de Bobo-Dioulasso, elle est composée de six (06) établissements et trois écoles doctorales: LERNSE, LARESBA, GERME.

- Ecole Supérieure d'Informatique (ESI) ;
- Institut du Développement Rural (IDR) ;
- Institut Universitaire de Technologie (IUT) ;
- Institut des Sciences de la Santé (INSSA) ;
- Unité de Formation et de Recherche Sciences et Technologie (UFR/ST) ;
- Unité de Formation et de Recherche Sciences Juridiques Politiques Economiques et de Gestion (UFR/SJPEG).

L'Ecole Supérieure d'Informatique où nous avons suivi notre formation, a une organisation pédagogique qui s'articule autour des trois axes suivants :

- la formation au Cycle des Ingénieurs de Travaux Informatiques (CITI) options Analyse et Programmation (AP) et Réseau et Maintenance Informatiques (REMI) ;
- la formation au Cycle des Ingénieurs de Conception en Informatique (CICI) ;

La formation au Cycle des Ingénieurs de Travaux Informatiques option Analyse et Programmation a pour objectif de former des cadres moyens opérationnels et évolutifs qui sont aptes à :

- participer efficacement à la conception, à la réalisation et à la maintenance d'applications informatiques ;
- assurer la formation des utilisateurs ;
- gérer des centres informatiques.

Durant leur cursus universitaire, les étudiants en ingénierat de travaux informatiques, parvenus en troisième année, doivent effectuer un stage obligatoire au cours duquel ils réalisent leur projet de fin de cycle.

L'objectif de ce stage est non seulement de mettre en exergue les connaissances acquises à l'école mais aussi de favoriser l'insertion professionnelle de ces futurs diplômés.

TABLE DES MATIERES

DEDICACE	i
REMERCIEMENTS	ii
SIGLES, ABREVIATIONS ET ACRONYMES.....	iii
PREAMBULE	iv
LISTE DES FIGURES.....	x
LISTE DES TABLEAUX.....	xi
INTRODUCTION GENERALE	1
CHAPITRE 1 : INCEPTION.....	2
INTRODUCTION	3
I.1 Présentation de la structure d'accueil.....	4
I.1.1 Historique.....	4
I.1.2 Missions et attributions	5
I.1.3 Fonctionnement.....	6
I.1.4 Organisation et stratégie.....	6
I.1.5 Paysage des applications existantes et infrastructure informatique	7
I.1.5.1 L'environnement matériel informatique	7
I.1.5.2 L'environnement logiciel	8
I.1.5.3 L'environnement réseau.....	8
I.2 Problématique et résultats attendus.....	10
I.2.1 Problématique	10
I.2.2 Résultats attendus.....	11
I.3 Langage de modélisation et démarche d'analyse.....	12
I.3.1 Le langage de modélisation.....	12
I.3.1.1 Présentation d'UML.....	12
I.3.1.2 Justification du choix du langage de modélisation.....	12
I.3.2 La méthode d'analyse et de conception	13
I.3.3.1 Définition	13
I.3.3.2 Présentation de UP [1].....	13
I.3.3.3 Justification du choix du processus unifié.....	15
I.3.3 Les acteurs du projet	15
I.3.3.1 Le comité de pilotage.....	15
I.3.3.2 Le groupe de projet	15
I.3.3.3 Le groupe des utilisateurs.....	16
I.3.4 Planning prévisionnel.....	16

I.3.5	Explication des écarts.....	17
I.4	Etude de l'existant.....	17
I.4.1	Modélisation de l'existant	17
I.4.1.1	Le diagramme de cas d'utilisation du système existant	18
I.4.1.2	Description textuelle de cas d'utilisation	19
I.4.2	Diagnostic de l'existant.....	19
	CONCLUSION.....	20
	CHAPITRE II : ELABORATION.....	21
	INTRODUCTION	22
II.1	Méthode de calcul des coûts de développement	23
II.2	Le Système de Gestion de Bases de Données	24
II.2.1	Etude comparative de quelques SGBD [4]	25
II.2.2	Choix du SGBD	26
II.3	Description des scenarii	26
II.3.1	Premier scenario.....	26
II.3.1.1	Description générale.....	26
II.3.1.2	Présentation de l'architecture réseau.....	26
II.3.1.3	Besoins logiciels du premier scenario.....	26
II.3.1.4	Besoins matériels du premier scenario.....	27
II.3.1.5	Coût de développement de l'application.....	28
II.3.1.6	Coût de formation des utilisateurs.....	28
II.3.1.7	Coût total du premier scénario	28
II.3.2	Deuxième scénario.....	30
II.3.2.1	Description générale.....	30
II.3.2.2	Architecture réseau.....	30
II.3.2.3	Besoins logiciels du deuxième scenario.....	31
II.3.2.4	Besoins matériels du deuxième scenario.....	32
II.3.2.5	Coût du développement.....	32
II.3.2.6	Coût de formation des utilisateurs.....	33
II.3.2.7	Coût total du deuxième scénario	33
II.3.3	Troisième scénario	33
II.3.3.1	Description générale.....	33
II.3.3.2	Architecture réseau du troisième scénario	34
II.3.3.3	Besoins logiciels du troisième scénario	35
II.3.3.4	Besoins matériels du troisième scenario	36
II.3.3.5	Coût de développement.....	37

II.3.3.6	Coût de formation des utilisateurs.....	37
II.3.3.7	Coût total du troisième scénario.....	37
II.3.4	Critique des scénarii.....	38
II.3.4.1	Premier scénario.....	38
II.3.4.2	Deuxième scénario.....	38
II.3.4.3	Troisième scénario.....	39
II.3.4.4	Etude comparative des scénarii.....	39
II.3.4.5	Choix de scénario.....	40
II.3.5	Modélisation du futur système.....	40
II.3.5.1	Le diagramme de cas d'utilisation.....	40
II.3.5.2	Quelques diagrammes de séquence.....	45
II.3.5.3	Quelques diagrammes d'activités.....	51
II.3.5.4	Le diagramme de classe.....	53
II.3.5.4.1	Présentation du diagramme de classe.....	53
II.3.5.4.2	Quelques règles de gestion.....	57
	CONCLUSION.....	57
	CHAPITRE III : CONSTRUCTION ET TRANSITION.....	58
	INTRODUCTION.....	59
III.1	CONSTRUCTION.....	59
III.1.1	Description des fonctionnalités de l'application.....	59
III.1.2	Langages de programmation.....	61
III.1.2.1	Etude comparative de langages de programmation web.....	61
III.1.2.2	Choix du langage de programmation.....	64
III.1.2.3	Framework JEE.....	64
III.1.2.4	Autres langages.....	65
III.1.3	L'environnement de développement intégré.....	65
III.1.4	Présentation de quelques écrans de GCAP.....	65
III.1.5	Politique de sécurité.....	68
III.1.5.1	Notion de sécurité.....	68
III.1.5.2	Sécurité applicative.....	68
III.1.5.3	La politique de sécurité contre les virus.....	68
III.1.5.4	Politique de sécurité pour la gestion des mots de passe et des accès non autorisés.....	68
III.1.5.5	Politique de sécurité contre les catastrophes.....	69
III.1.5.6	Mise en place des sauvegardes et des restaurations [15].....	69
III.2	TRANSITION.....	70
III.2.1	Tests et déploiement.....	70

III.2.2 Formation des utilisateurs	70
CONCLUSION	71
CONCLUSION GENERALE	72
REFERENCES BIBLIOGRAPHIQUES ET WEBOGRAPHIQUES	I
ANNEXES	III

LISTE DES FIGURES

Figure 1 : Parc informatique de la CARFO.....	7
Figure 2 : Le réseau de la CARFO connecté avec RESINA.....	9
Figure 3 : Enchaînement d'activités préconisé par UP [2].....	14
Figure 4: Diagramme de GANTT du planning prévisionnel.....	16
Figure 5 : Diagramme de cas d'utilisation de l'existant.....	18
Figure 6 : Architecture réseau du deuxième scénario.....	30
Figure 7 : Architecture réseau du troisième scénario.....	34
Figure 8 : Diagramme de cas d'utilisation du futur système.....	41
Figure 9: Diagramme de séquence du cas d'utilisation : « Faire demande ».....	46
Figure 10: Diagramme de séquence du cas d'utilisation « S'authentifier».....	47
Figure 11 : Diagramme de séquence du cas d'utilisation : « Traiter demande ».....	48
Figure 12: Diagramme de séquence du cas d'utilisation « Faire demande ».....	49
Figure 13: Diagramme de séquence du cas d'utilisation « Gérer allocation de congé ».....	50
Figure 14: Diagramme d'activité du cas d'utilisation « S'authentifier ».....	51
Figure 15 : Diagramme d'activité du cas d'utilisation « Faire demande congé ».....	52
Figure 16: Diagramme d'activité du cas d'utilisation « Traiter demande ».....	53
Figure 17 : Diagramme de classe.....	56
Figure 18: Page de connexion.....	65
Figure 19 : Page d'accueil du DRH.....	66
Figure 20 : Ecran de demande de congé annuel.....	66
Figure 21 : Ecran de notification envoyée à un agent.....	67
Figure 22 : Ecran de traitement de demande de congé annuel par un directeur de direction.....	67

LISTE DES TABLEAUX

Tableau 1: <i>Description textuelle du cas d'utilisation « Faire demande scenario nominal »</i>	19
Tableau 2 : <i>Tableau récapitulatif du diagnostic du système existant</i>	20
Tableau 3: <i>Tableau comparatif de quelques SGBD</i>	25
Tableau 4: <i>Besoins logiciels du premier scenario</i>	27
Tableau 5: <i>Besoins matériels du premier scenario</i>	27
Tableau 6 : <i>Coût de formation des utilisateurs du premier scenario</i>	28
Tableau 7: <i>Estimation du coût total de l'application pour le premier scenario</i>	29
Tableau 8: <i>Besoins logiciels du deuxième scénario</i>	31
Tableau 9: <i>Besoins matériels du deuxième scenario</i>	32
Tableau 10: <i>Coût de formation des utilisateurs</i>	33
Tableau 11: <i>Estimation du coût total de l'application pour le deuxième scenario</i>	33
Tableau 12: <i>Besoins logiciels du troisième scénario</i>	35
Tableau 13: <i>Besoins matériels du troisième scenario</i>	36
Tableau 14: <i>Coût de formation des utilisateurs pour le troisième scenario</i>	37
Tableau 15: <i>Coût total du troisième scenario</i>	37
Tableau 16: <i>Etude comparative des scénarii</i>	40
Tableau 17: <i>Quelques règles de gestion</i>	57
Tableau 18: <i>Etude comparative de langages de programmation web [7]</i>	62
Tableau 19: <i>Framework JEE utilisés</i>	64

INTRODUCTION GENERALE

Dans le cadre du stage de fin du Cycle des Ingénieurs de Travaux Informatiques que nous avons été accueillis du **01 octobre 2013 au 31 décembre 2013** à la **CARFO** où il nous a été soumis le thème : « *Gestion des congés et absences professionnels à la CARFO* »

Cette gestion connaît de nombreuses difficultés dues au nombre important de tâches et à leur gestion manuelle.

Notre travail s'articulera autour de trois (03) chapitres : le premier chapitre situe le stage c'est à-dire la présentation de la structure d'accueil, traite de la problématique du thème, des résultats attendus avant de présenter la méthode d'analyse et de conception. Ensuite, il présentera le système d'information existant, puis à le critiquer afin d'en dégager les forces et les faiblesses. Le deuxième chapitre a pour objectif de déterminer les scénarii possibles pour le système à mettre en place. Le dernier abordera l'étude détaillée du futur système.

CHAPITRE 1 : INCEPTION

INTRODUCTION

Pour mener un projet informatique dans une structure il faut nécessairement bien connaître ladite structure, avoir une bonne maîtrise du thème lié au projet et définir une méthode de travail.

L'inception, première phase de la démarche d'analyse UP (Unified Process) permet de cerner les contours du problème et de juger l'opportunité du projet. Dans ce chapitre, il sera question de faire une présentation de la structure d'accueil, de montrer la problématique liée à la gestion des congés et absences professionnels et d'explicitier les résultats attendus. En outre, notre démarche d'analyse et le planning prévisionnel seront présentés. Enfin, nous ferons une étude de l'existant.

I.1 Présentation de la structure d'accueil

La **CARFO** est un établissement de prévoyance social du Burkina Faso. Elle a pour vocation la gestion des retraites des fonctionnaires de l'Etat et des militaires. La **CARFO** est incontestablement intégrée dans le quotidien des travailleurs de l'administration publique.

Une Direction de l'Informatisation et des Statistiques (DIS) a été créée au sein de la **CARFO** afin d'améliorer les rendements de celle-ci en utilisant les Nouvelles Technologies de l'Information et de la Communication (NTIC). En effet, vu l'importance de ses activités et le nombre croissant de ses pensionnés, l'objectif de la création de la DIS est d'assurer promptement la satisfaction de la clientèle de l'établissement. La DIS vise une meilleure exploitation des services offerts par les NTIC en automatisant les tâches des différents services de la **CARFO**.

I.1.1 Historique

Avant les indépendances il n'existait pas de régime de retraite, ni de caisse de retraite. La gestion des pensions de retraite était confiée à une section de la direction du budget. Cette direction avait pour mission d'assurer le traitement des dossiers de pension des fonctionnaires affiliés à deux organismes de retraite différents. Le premier qui est la Caisse Locale de Retraite (CLR) avec son siège à Dakar, gérait le régime de retraite des anciens cadres locaux de l'ex Afrique Occidentale Française (A.O.F). Le second, appelé Caisse de Retraite de la France d'Outre-Mer (CRFOM) basé à Paris était chargé de gérer le régime de retraite des fonctionnaires des anciens cadres généraux et supérieurs des ex Territoires d'Outre-Mer (TOM).

Avec l'avènement des indépendances, ces caisses furent dissoutes en 1959 pour laisser la liberté à chacun des nouveaux Etats indépendants l'initiative de la mise en place de son propre système de retraite et d'en assurer la gestion. Notre pays, le Burkina Faso, à l'instar des autres ex-colonies a procédé à la mise en place progressive d'un dispositif juridique et organisationnel permettant la prise en charge de la retraite des agents de l'Etat. Il a créé un service des pensions dont l'organisation et le fonctionnement ont été définis par l'arrêté n°271/MF/B/DIR/F8 du 13/10/61.

Suite au manque d'expérience et de problème de gestion, l'Etat a jugé nécessaire de créer une structure autonome pour gérer les pensions, d'où la naissance de la **CARFO** en 1986.

La forme juridique d'un établissement public à caractère industriel et commercial dotée de la personnalité juridique et de l'autonomie financière, la **CARFO** a été transformée en un établissement public de prévoyance sociale (EPPS) par décret n°2008-

155/PRES/PM/MFPRE/MEF du 03 avril 2008. Elle est régie par la loi **n°016-2006/AN du 16 mai 2006** portant création de la catégorie d'Etablissements publics de prévoyance sociale et le décret **n°2008-156/PRES/PM/MFPRE/MF du 03 avril 2008** approuvant ses statuts particuliers.

La **CARFO** n'a débuté ses activités de façon autonome qu'en 1989. De plus amples informations sur l'historique de la **CARFO** sont disponibles sur son site officiel www.carfo.org.

I.1.2 Missions et attributions

La **CARFO** a pour missions et attributions la gestion :

- du régime de retraite des fonctionnaires, militaires et magistrats institué par la loi n° **47/94/ADP du 29 novembre 1994** portant régime général de retraite des fonctionnaires, militaires et magistrats étendu aux agents contractuels de la fonction publique recrutés à partir du 1er janvier 1999 à travers la loi n° **006-2001/AN du 17 mai 2001** ;
- du régime de prévention et de réparation des risques professionnels institué par la loi n° **022/2006/AN du 16 novembre 2006** portant régime de prévention et de réparation des risques professionnels applicable aux agents de la fonction publique, aux militaires et aux magistrats ;
- de tout autre régime qui viendrait à être créé par la loi. Se fondant sur la mission qui lui est assignée, la **CARFO** sert actuellement les prestations suivantes :
 - la pension de retraite ;
 - la pension d'invalidité ;
 - la pension de survivants (veuves/veufs et orphelins) ;
 - les remboursements des retenues pour pension.

La **CARFO** gère désormais les risques professionnels et s'attèle à fournir les prestations suivantes au titre des accidents du travail et des maladies professionnelles :

- l'assistance médicale, chirurgicale, pharmaceutique hospitalière et accessoires ;
- la fourniture, l'entretien, le renouvellement et la réparation des appareils de prothèse ;
- le paiement des rentes d'incapacité ;
- le paiement des rentes de survivants ;
- la rééducation professionnelle, la réadaptation fonctionnelle ;
- la prise en charge du transport et des frais de séjour pour soins.

Outre les activités liées à son objet social, la **CARFO** assure par ailleurs la liquidation et le paiement des capitaux décès pour le compte de l'État. Cette prestation n'est donc pas à la

charge de la **CARFO** mais de l'État qui en assure le financement. La pension est payée actuellement trimestriellement aux guichets de la **CARFO** à Ouagadougou ou auprès des services déconcentrés du Trésor Public (trésoreries et perceptions) dans les autres localités. Les échéances de paiement sont les suivantes : mars, juin, septembre et décembre. Il convient de noter également que les pensions peuvent être versées mensuellement à la demande du pensionné.

I.1.3 Fonctionnement

La **CARFO** est placée sous la tutelle de trois (03) ministères :

- le ministère chargé de la fonction publique pour la tutelle technique ;
- le ministère chargé de l'économie et des finances pour la tutelle financière ;
- le ministère chargé du commerce pour la tutelle de gestion.

La **CARFO** est administrée par un conseil d'administration de seize (16) membres répartis à part égale entre l'État et les organisations professionnelles des travailleurs. Les travailleurs retraités sont représentés au conseil d'administration de la **CARFO** en qualité de membres observateurs sans voix délibérative à raison de deux (2) membres.

I.1.4 Organisation et stratégie

La **CARFO** a subi de nombreuses réformes notamment sa transformation en Etablissement Public de Prévoyance Sociale (**EPPS**), qui vise à faire d'elle une véritable institution de sécurité sociale autonome et déconcentrée, et en matière de protection sociale des agents de l'Administration Publique une vitrine des pays de la sous régions. C'est dans cette optique que la **CARFO** s'est dotée en juillet 2008 d'un document d'orientation dénommé stratégie de renforcement du Système de Gestion de Retraite des Agents de l'Etat (**SYGRAE**). Le **SYGRAE** est un ensemble d'axes d'intervention et d'orientation des actions de la **CARFO** à l'horizon 2018. Il propose de relever trois défis majeurs qui sont :

- assurer sa mutation vers les normes de la **CIPRES** ;
- renforcer la confiance et l'espoir des usagers;
- réussir l'autonomie de gestion du système de sécurité sociale des agents de l'Administration Publique.

Pour relever ces défis, des objectifs stratégiques ont été définis pour la décennie à venir qui sont notamment :

- le renforcement des capacités de l'institution à faire face à ces obligations ;
- l'adaptation des prestations des services aux besoins de la clientèle ;
- le renforcement de la viabilité financière du régime de retraite.

I.1.5 Paysage des applications existantes et infrastructure informatique

I.1.5.1 L'environnement matériel informatique

Comme moyen matériel, la CARFO regorge en son sein :

- d'une gamme d'ordinateurs de bureau de marque HP et Lenovo;
- des PC de marque HP pour les agents de la DIS ;
- un serveur de base de données **Windows Serveur 2003 R2** ;
- un serveur de base de données **Oracle 9i**
- des modems de connexion ADSL, des routeurs et des switch ;
- des imprimantes de marques HP;
- des télécopieurs de marque HP.

La figure 1 présente la répartition du matériel informatique de la *CARFO*. [0]

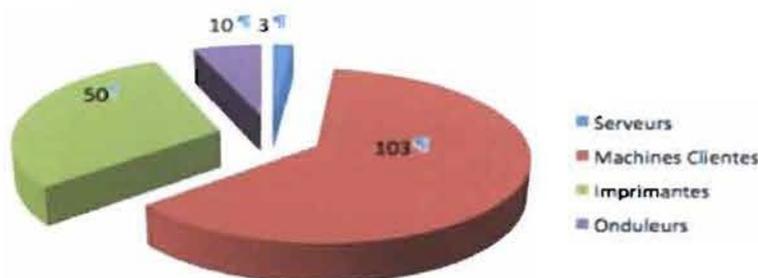


Figure 1 : Parc informatique de la CARFO

I.1.5.2 L'environnement logiciel

Le système informatique de la **CARFO** repose sur **SIGA** (Système Intégré de Gestion Automatisée) qui est une application développée en interne pour la gestion automatisée des différentes tâches de l'institution. SIGA présente les caractéristiques suivantes :

- il a été développé sous « **Developer 2000 d'Oracle** »;
- c'est une application bâtie sur une architecture client/serveur à deux couches ou 2-tiers ;

Nous avons aussi recensés sur le site de la **CARFO** :

- **Oracle 9i** comme système de gestion de base de données ;
- le logiciel de gestion de la comptabilité **CIPRES** (Confédération Inter Africaine de Prévoyance Sociale) ;
- les systèmes d'exploitation Windows XP et Windows 7 installés sur les postes ;
- les outils Microsoft office 2007 et 2010 installés sur les ordinateurs de bureau ;
- un anti-virus Kaspersky Lab 2006 sur chaque ordinateur.

I.1.5.3 L'environnement réseau

Le réseau local de la **CARFO** est un sous réseau du réseau qui relie les administrations publiques du Burkina Faso nommé **RESINA**. Ce réseau présente les caractéristiques suivantes:

- c'est un réseau WAN (Wide Area Network) ayant une topologie en anneau FDDI (Fiber Distributed Data Interface) ;
- c'est une interconnexion par fibre optique de trente(30) bâtiments administratifs ;
- il abrite les départements ministériels et quelques institutions à l'instar de la **CARFO** ;
- c'est une interconnexion de douze(12) villes par liaison spécialisée ;
- **RESINA** offre les services de transport et d'administration de l'information.

Il faut noter que des réseaux locaux sans fil (WI-FI) existent également dans chaque direction.

La figure 2 présente l'architecture réseau de la **CARFO**.

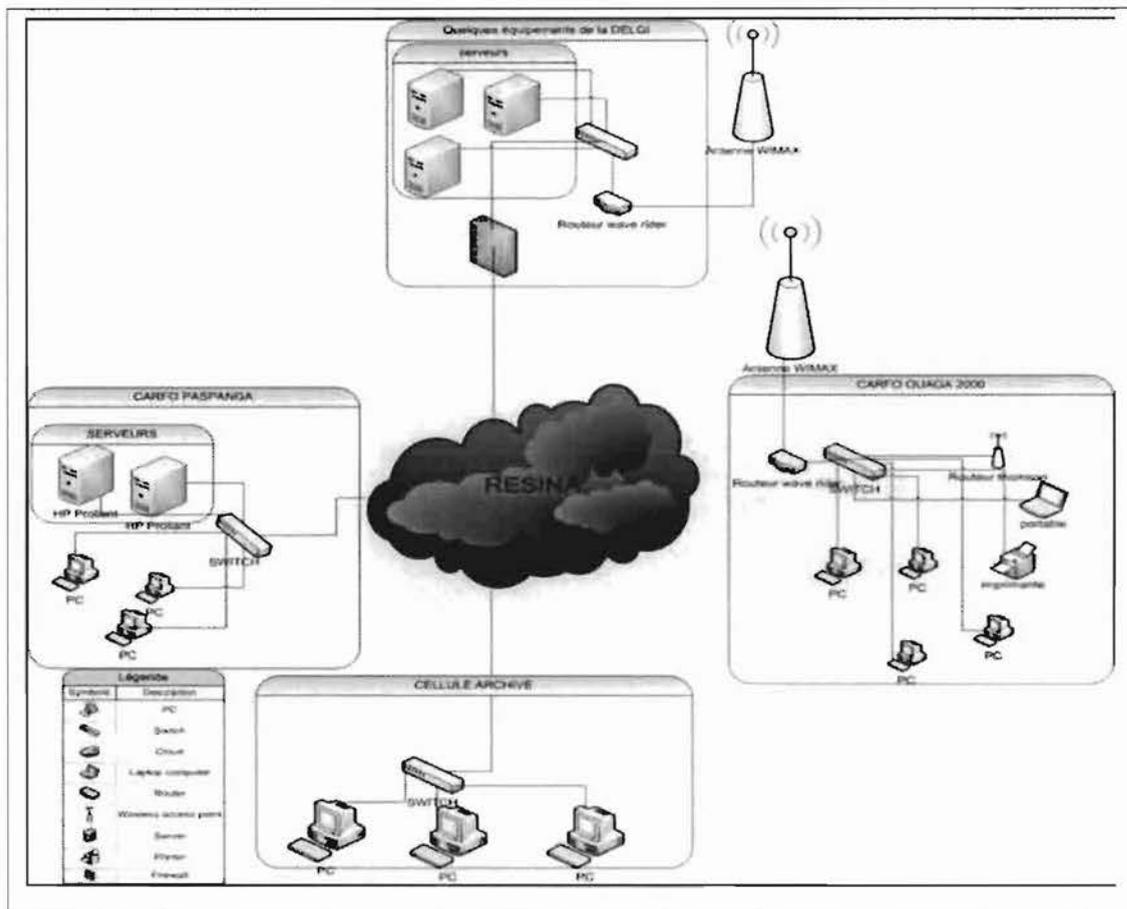


Figure 2 : Le réseau de la CARFO connecté avec RESINA

I.2 Problématique et résultats attendus

I.2.1 Problématique

Une des attributions de la direction des ressources humaines est la gestion du mouvement du personnel de la *CARFO*.

La dite direction est tenue de connaître l'état de présence ou d'absence de chaque agent de la *CARFO*.

Pour ce faire toutes les demandes de congés et de permissions d'absences passent par elle.

Elle les traite en collaboration avec les différents directeurs de services, chefs de services et le DG.

Cependant, malgré la grandeur de cette boîte qu'ait la *CARFO* (avec plus de 140 agents) et malgré la taille des données manipulées, la gestion des congés et des permissions d'absences demeure essentiellement manuelle (80% manuelle et utilisation sommaire de Microsoft Excel).

En plus, il s'est avéré que les logiciels de gestions de congés et d'absences existant sur le marché ne correspondent pas véritablement aux attentes des utilisateurs dans l'environnement de la *CARFO*.

Les insuffisances que présentent ces logiciels sont entre autres :

- le caractère monoposte de certains logiciels ;
- la nécessité de faire des paramétrages toute chose qui s'avère souvent très fastidieuse ;
- les difficultés d'appropriation par le personnel de l'entreprise ;
- la non prise en compte de certains besoins de l'entreprise ;
- le coût élevé de la licence.

C'est donc pour apporter une solution à ces insuffisances mais aussi pour accompagner le DIS dans sa politique d'informatisation de toutes les activités de la *CARFO*, qu'il nous a été soumis le présent thème : « Gestion des congés et absences professionnels »

Des solutions de logiciels de gestion de congés et d'absences, il en existe sur le marché.

Nous en avons étudié deux :

- **PHP_CONGES** : Gestion des congés
 - version : 1.5.1
 - langue(s) de l'interface : français, anglais, autre ;
 - type : application desktop ;
 - systèmes supportés : UNIX-like, Windows, MacOS X ;
 - licence : GPL ;
 - coût : gratuit.

○ **NEREO** (disponible en ligne)

- Version : 2014
- Langue(s) de l'interface : français ;
- Type : application web ;
- Licence : propriétaire ;
- Coût : entre 1 et 3€ par mois et par employés selon le type de l'entreprise.

De l'étude que l'étude que nous avons réalisé, il ressort que ces logiciels ne satisfont pas aux attentes des utilisateurs. Cependant quelles sont les attentes des utilisateurs ?

I.2.2 Résultats attendus

Les résultats attendus découlent de la problématique.

Il s'agira pour le groupe de projet d'apporter une solution informatique pour la gestion des congés et des absences au sein de la *CARFO*. Et cette solution n'est pas des moindres, elle se veut pallier les insuffisances constatés chez les logiciels existant sur le marché.

La solution à apporter sera une solution locale, adaptée parfaitement aux besoins des utilisateurs, et une solution moindre coût.

Cette solution doit permettre entre autre :

- de créer des comptes pour chaque agent ;
- de faire des demandes de congés et d'absences en ligne ;
- de calculer les allocations de congés ;
- de proposer des outils d'aide aux managers pour la prise de décision concernant le traitement des demandes ;
- de notifier les agents par message sur leur téléphone et sur leur espace messagerie ;
- d'avoir des statistiques sur les absences ;
- de concilier simplicité et performance (permettre plusieurs connexions simultanées) ;
- d'être de moindre coût ;
- de garantir la sécurité aussi bien au niveau base de données qu'au niveau applicatif ;

I.3 Langage de modélisation et démarche d'analyse

L'analyse et la conception d'un projet informatique nécessitent l'utilisation d'un langage de modélisation et d'une méthode ou démarche de développement appropriée. Ces outils sont des outils d'aide à la conception qui permettent la description du système logiciel, une meilleure compréhension de celui-ci par les acteurs et surtout une participation et une collaboration active entre informaticiens et acteurs à travers le recueil d'information. L'ensemble des règles de modélisation peut être appuyé par un processus ou méthode d'analyse dont le rôle est d'assurer l'harmonie au sein d'un ensemble d'éléments coopératifs, et d'expliquer comment il convient de se servir du langage de modélisation.

I.3.1 Le langage de modélisation

Un langage de modélisation est un langage artificiel qui peut être utilisé pour exprimer de l'information ou de la connaissance ou des systèmes dans une structure qui est définie par un ensemble cohérent de règles. Ces règles sont utilisées pour l'interprétation de la signification des composants dans la structure.

Le choix du langage par le groupe de projet s'est porté sur UML.

I.3.1.1 Présentation d'UML

UML est un langage de modélisation fondé sur les concepts orientés objets. Il a été conçu pour la modélisation de tous les phénomènes de l'activité de l'entreprise indépendamment des techniques d'implémentation mise en œuvre par la suite. En plus des éléments constitutifs de tout langage, UML a choisi la notation graphique fondée sur des diagrammes. Il est né de la fusion de trois (03) méthodes de référence : OMT, BOOCH, OOSE. Il n'est ni une méthode, ni un processus mais un langage de modélisation. UML n'impose pas une démarche particulière pour l'analyse d'un système mais préconise d'adopter une démarche ayant les caractéristiques suivantes :

- itérative et incrémentale ;
- centrée sur l'architecture logicielle ;
- guidée par le besoin des utilisateurs du système ;

I.3.1.2 Justification du choix du langage de modélisation

Le groupe de projet a choisi UML comme langage de modélisation à cause de nombreux avantages que présente ce langage.

En effet UML présente l'avantage d'être le standard de la modélisation objet universellement reconnu.

Il est un langage visuel. Sa notation graphique permet d'exprimer visuellement des solutions objets facilitant ainsi la comparaison et l'évaluation de celles-ci. C'est un langage formel et normalisé doté d'un gain de précision et d'un gage de stabilité. Il est aussi un support de communication performant car il cadre l'analyse tout en facilitant la compréhension des représentations abstraites complexes.

En outre, UML est indépendant des langages d'implémentation et des processus de développement, il offre une certaine souplesse aux concepteurs en n'imposant pas de démarche particulière.

Enfin, son concept objet scié parfaitement avec le type d'application que le groupe de projet devra mettre en œuvre.

I.3.2 La méthode d'analyse et de conception

I.3.3.1 Définition

Dans les méthodes d'analyse et de conception, on distingue l'analyse de la conception.

La phase d'analyse permet de lister les résultats attendus, en terme de fonctionnalités, de performance, de robustesse, maintenance, de sécurité, d'extensibilité, etc.

La phase de conception, permet de décrire de manière non ambiguë, le plus souvent en utilisant un langage de modélisation, le fonctionnement futur du système, afin d'en faciliter la réalisation.

Un processus ou méthode d'analyse et de conception définit une séquence d'étapes, en partie ordonnée, qui concourent à l'obtention d'un système logiciel ou à l'évolution d'un système existant. L'objet d'un processus de développement est de produire des logiciels de qualité qui répondent aux besoins de leurs utilisateurs dans des temps et des coûts prévisibles.

Le Processus Unifié ou Unified Process (UP) en anglais est la méthode d'analyse que le groupe de projet a utilisé pour conduire ce projet.

I.3.2.2 Présentation de UP [1]

Le Processus Unifié est une méthode de développement logiciel ayant les caractéristiques suivantes :

- **itératif et incrémental** : une itération désigne la succession des étapes de l'enchaînement d'activités, tandis qu'un incrément correspond à une avancée dans les différents stades de développement ;

- **piloté par les cas d'utilisation d'UML** : les cas d'utilisation illustrent, détectent puis décrivent les besoins fonctionnels et leur ensemble constitue le modèle de cas d'utilisation qui dicte les fonctionnalités complètes du système ;
- **centré sur l'architecture** : l'architecture d'un système logiciel peut être décrite comme les différentes vues du système qui doit être construit ;
- **centré sur la diminution des risques**.

UP est un ensemble de principes génériques pouvant être adaptés en fonction des spécificités des projets. Un tableau comparatif des implémentations célèbres d'UP est présenté en annexe (Annexe 1).

Le cycle de développement du Processus Unifié organise les tâches et les itérations en quatre phases.

- **inception** : elle consiste à faire une spécification des besoins et aussi une sorte d'étude de faisabilité où on effectue les recherches nécessaires pour décider si on poursuit ou non le projet ;
- **élaboration** : à ce niveau, on développe de façon incrémentale l'architecture du noyau, les risques et la plupart des besoins sont identifiés ;
- **construction** : cette phase consiste à la construction des sous-ensembles exécutables et stables du produit final ;
- **transition** : dans cette phase on procède au déploiement du système sur des sites opérationnels.

Chaque phase est constituée d'un ensemble d'activités.

La figure 3 présente l'enchaînement des activités qu'UP préconise dans l'exécution des différentes phases.

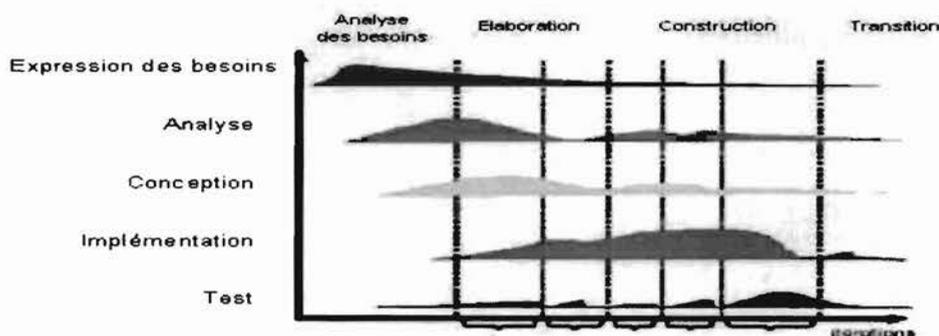


Figure 3 : Enchaînement d'activités préconisé par UP [2]

I.3.3.3 Justification du choix du processus unifié

Vu la taille de la structure d'accueil, et pour bien mener ce projet, la démarche d'analyse que nous devons choisir, doit être adaptée. Nous avons choisi UP car à travers ses caractéristiques, nous pourrions cerner efficacement les besoins des utilisateurs des différents services, accélérer tout en gardant la cohérence du processus de développement du système, limiter les coûts de réalisation. En plus, UP est un processus pouvant être adapté à une large classe de système logiciels, à différents domaines d'application, à différents types d'entreprises, à différents niveaux de compétences et à différentes tailles de l'entreprise. Ainsi, il nous permettra de proposer un système qui répond aux besoins des utilisateurs. Egalement, l'adaptabilité de ce processus fait de lui, le processus idéal pour notre projet.

I.3.3 Les acteurs du projet

Ce sont toutes les personnes qui interviennent dans l'exécution de ce projet. Ils sont regroupés en groupe de pilotage, groupe de projet et en groupe des utilisateurs.

I.3.3.1 Le comité de pilotage

Le comité de pilotage est un groupe d'encadreurs chargés de veiller au bon fonctionnement du projet. Il a pour rôle de guider le groupe de travail, de valider les choix méthodologiques et les orientations générales, de définir les moyens à mettre en place pour la réalisation du projet, de coordonner les activités et de donner la validation finale au projet.

Il est constitué de :

- *M. Aboudou TRAORE*, ingénieur informaticien à la *CARFO*, notre maître de stage;
- *Dr Telesphore TIENDREBEOGO*, enseignant à l'UPB, notre superviseur ;
- *M. Noel BADO*, directeur de la DIS ;
- *M. Nazaire OUATTARA*, directeur des ressources humaines.

I.3.3.2 Le groupe de projet

Le groupe de projet est chargé de l'étude, de la conception et de la réalisation du projet avec l'aide du groupe d'utilisateurs et sous la supervision du comité de pilotage.

Il est composé de :

- *Arnaud Boureima LOUE*, élève ingénieur en Analyse et Programmation, 3^{ème} année, ESI ;
- *Pascal OUEDRAOGO*, élève ingénieur en Analyse et Programmation, 3^{ème} année, ESI ;

I.3.3 Le groupe des utilisateurs

Le groupe des utilisateurs a un rôle consultatif. Il est chargé de fournir toutes les informations nécessaires à la bonne conduite du projet. Il intervient également dans la validation des dossiers d'études et des prototypes produits par le groupe de projet.

Il se compose de tous les utilisateurs potentiels du système à mettre en œuvre. Il s'agit de de tous les agents de la **CARFO** notamment les managers.

I.3.4 Planning prévisionnel

Pour réaliser un projet, il est nécessaire d'établir un planning prévisionnel en accord avec tous les acteurs du projet et de veiller à son respect.

Ce planning prévisionnel est le tableau de bord de chaque acteur du projet. La réussite du projet en ce qui concerne les exigences du délai de réalisation y va du respect de ce planning.

Le chef de projet se doit donc de veiller au respect du planning prévisionnel.

En accord avec les autres acteurs du projet, le groupe de projet a adopté le planning suivant représenté par un diagramme de GANTT.

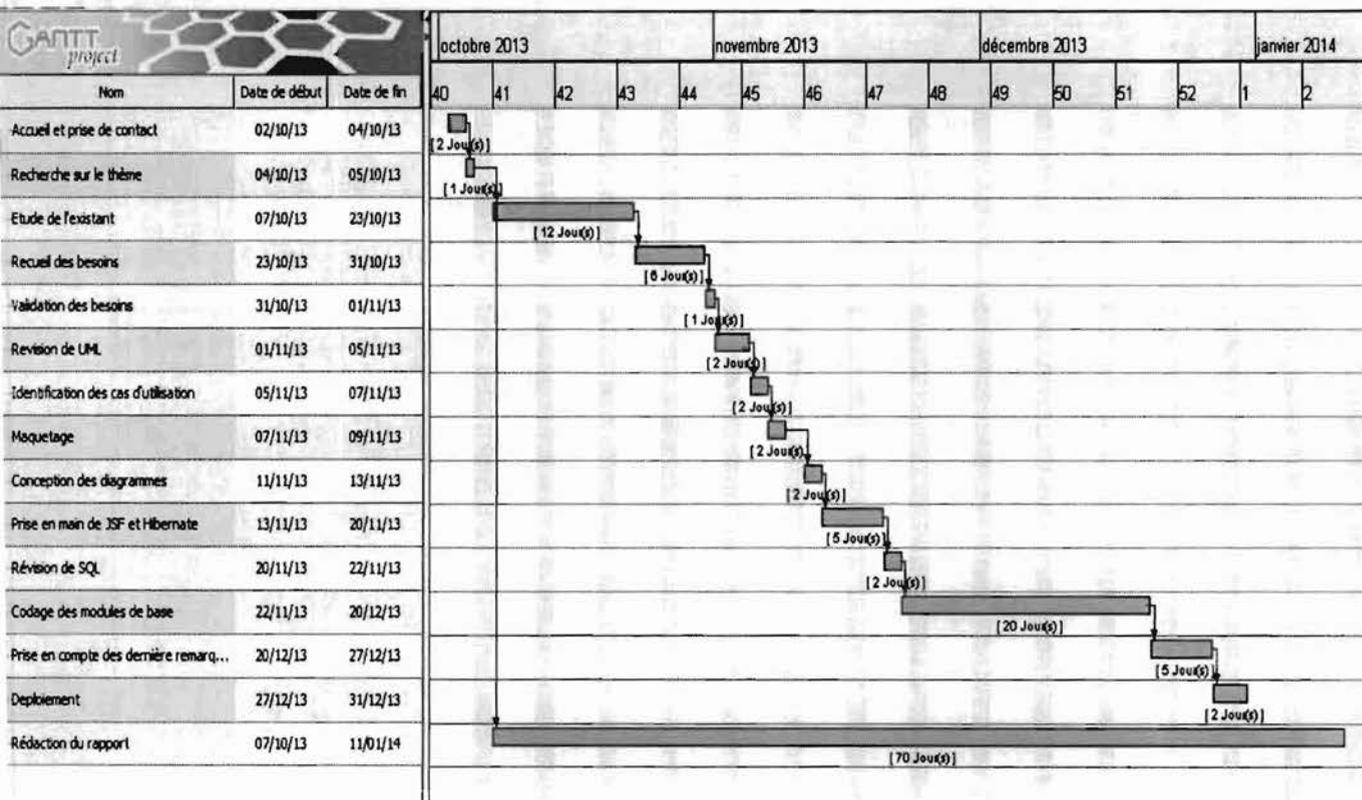


Figure 4: Diagramme de GANTT du planning prévisionnel

Cependant ce planning prévisionnel n'a pas pu être respecté pour certaines raisons.

I.3.5 Explication des écarts

Les raisons qui expliquent le non-respect du planning réel par les acteurs du projet sont multiples :

- après trois semaines, le thème définit premièrement a été changé par le présent ;
- le stage a été interrompu du 25 novembre au 8 décembre pour des raisons académiques ;

I.4 Etude de l'existant

L'étude de l'existant a pour objectif de permettre au groupe de projet de comprendre le fonctionnement du système actuel, d'en dégager les forces et faiblesses, de recenser le souhait des utilisateurs, de connaître les contraintes à considérer dans la conception du futur système. Cette étude nous facilitera ainsi dans la proposition des solutions pouvant satisfaire les besoins des utilisateurs.

I.4.1 Modélisation de l'existant

Les différentes interviews menées par le groupe de projet ont permis d'une part de délimiter le projet, d'autre part de connaître les activités des différents services de la *CARFO*, les interactions entre ces services, les informations mises en jeu dans le système à mettre en œuvre.

Le groupe de projet a jugé nécessaire de modéliser l'existant à travers un diagramme de cas d'utilisation.

I.4.1.1 Le diagramme de cas d'utilisation du système existant

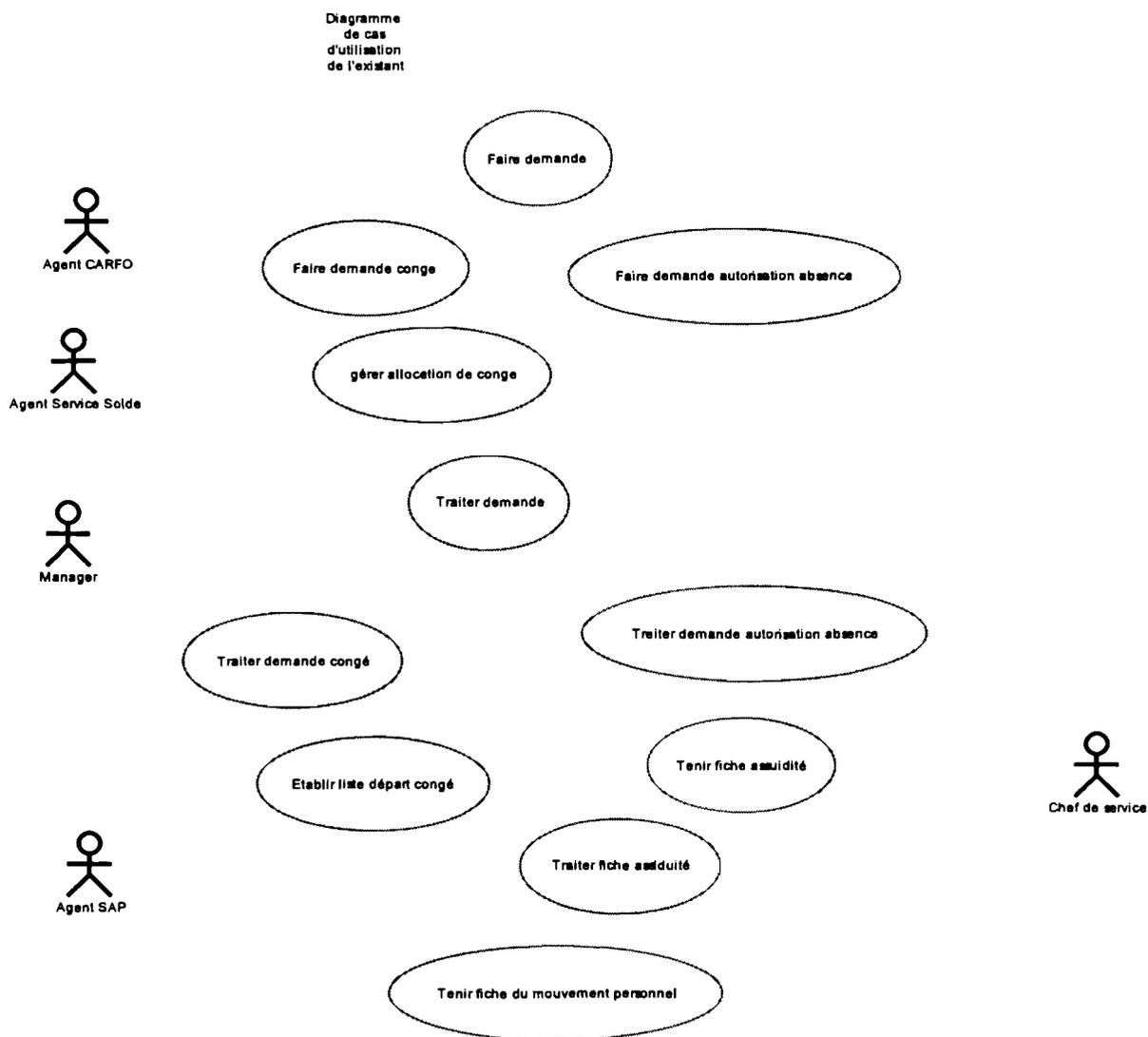


Figure 5 : Diagramme de cas d'utilisation de l'existant

I.4.1.2 Description textuelle de cas d'utilisation

La description textuelle de cas d'utilisation permet de décrire plus explicitement l'ensemble des activités qui composent le cas d'utilisation.

Tableau 1: *Description textuelle du cas d'utilisation « Faire demande scenario nominal »*

CU3 : Faire demande	Scenario nominal	Folio 1/1
Résumé : Ce cas permet à un agent de faire une demande d'autorisation d'absence		Version 1.0
		19/11/2013
Préconditions : Fiche de demande disponible.		
Acteurs : Agent CARFO		
<Début>		
1. Retirer fiche		
2. Remplir la fiche		
3. Joindre d'éventuelles pièces justificatives		
4. Déposer la demande auprès du chef de service		
<Fin>		

I.4.2 Diagnostic de l'existant

Le diagnostic de l'existant consiste à donner une appréciation sur le fonctionnement du système en place. Il a pour objectif de faire ressortir les forces et faiblesses dudit système.

Ainsi, le diagnostic fait par le groupe de projet est reporté dans le tableau 2.

Tableau 2 : Tableau récapitulatif du diagnostic du système existant

Forces	Faiblesses
<ul style="list-style-type: none"> ✓ La volonté et la disponibilité du personnel ✓ La maîtrise de Microsoft Office Excel par certains acteurs ✓ L'existence de matériel informatique dans de nombreux services (ordinateurs, imprimantes) 	<ul style="list-style-type: none"> ✓ Utilisation de Microsoft Office Excel ne permet pas de manipuler certaines informations ✓ Lenteur dans le traitement des demandes d'absences ✓ Absences d'informations aidant à la prise de décision pour le traitement des demandes ✓ Absences de statistique sur les absences ✓ Calcul manuel des allocations pouvant entraîner des erreurs ✓ Absence d'automatisation de certaines tâches ✓ Mauvais suivi de l'assiduité des agents

CONCLUSION

Dans ce chapitre il a été question de présenter la structure d'accueil, de montrer la problématique liée au thème et pour ensuite définir notre approche et planning d'exécution du projet. Nous avons également fait une étude de l'existant.

Comme méthode d'analyse pour mener le projet nous avons choisi UP et UML comme langage de modélisation.

Tout ceci va être une base sur laquelle nous pourrons nous appuyer pour proposer des solutions adaptées répondant aux attentes des utilisateurs et du groupe de pilotage.

CHAPITRE II : ELABORATION

INTRODUCTION

Ce chapitre est dédié à l'étude des scénarii autrement dit des solutions possibles pour la mise en place du futur système.

Cette étude consistera à faire une estimation des coûts matériels et logiciels, des risques et des exigences des différents scénarii en vue de permettre à l'ensemble des acteurs et des décideurs de cerner les avantages et les inconvénients de chaque scénario. A l'issue de cette étude un choix consensuel entre le groupe de projet, le groupe de pilotage et le groupe des utilisateurs sera fait.

II.1 Méthode de calcul des coûts de développement

On distingue plusieurs méthodes permettant d'estimer le coût développement d'un logiciel parmi lesquelles nous avons le modèle COCOMO (CONstructive COSt MOdel) [3]. Cette méthode existe en trois versions : simple, intermédiaire et détaillée.

Nous utiliserons le modèle COCOMO simple qui est le mieux documenté, il donne des estimations des coûts en s'appuyant sur la taille (estimée) du logiciel et sur le type de logiciel ou projet à réaliser. Il existe trois (03) types de projets que sont :

- les projets de mode organique : ces projets sont réalisés par une équipe de taille relativement petite travaillant dans un environnement familier et dans un domaine d'application connu de l'équipe ;
- les projets de mode semi-détaché : ce sont des types de projets qui ne sont pas trop complexes. L'équipe de développement se connaît un peu, et les technologies peuvent être mal connues, mais pas d'une grande difficulté d'appréhension ;
- les projets de mode embarqué : le système à développer est une partie d'un système complexe et les modifications de spécifications destinées à contourner des problèmes logiciels sont en général impossibles.

Les formules permettant de calculer le coût, ou plus exactement l'effort requis pour le développement du logiciel en fonction du type de projet sont les suivantes :

- Mode organique : $HM = 2,4 (KLSL)^{1,05}$;
- Mode semi-détaché : $HM = 3 (KLSL)^{1,12}$;
- Mode embarqué : $HM = 3,6 (KLSL)^{1,20}$.
 - HM (signifie Homme-Mois) : représente l'effort requis pour le développement de l'application ;
 - KLSL (Kilo-Lignes-Sources du logiciel) : correspond à 1/1000 du nombre de lignes de code du logiciel.

Le modèle COCOMO simple permet également d'estimer le temps nécessaire au développement d'un projet (TDEV). Les équations pour les différents types de projets sont les suivantes :

- Mode organique : $TDEV = 2,5 (HM)^{0,38}$;
- Mode semi-détaché : $TDEV = 2,5 (HM)^{0,35}$;
- Mode embarqué : $TDEV = 2,5 (HM)^{0,32}$.

Le nombre de personnes requis pour réaliser le projet dans cet intervalle de temps est donc :

➤ $N = HM / TDEV.$

Le coût total de réalisation est donné par :

➤ $Coût = HM * ValeurHM.$

Où ValeurHM représente le salaire moyen d'un informaticien dans le pays des développeurs.

Pour le calcul des coûts de réalisation du système de gestion des congés et absences professionnels, nous utiliserons la méthode COCOMO simple en mode semi-détaché, notre projet étant classé dans la catégorie des projets de type semi-détaché.

II.2 Le Système de Gestion de Bases de Données

Un Système de Gestion de Bases de Données en abrégé SGBD, est un ensemble de logiciels permettant la gestion efficace des données. Il permet ainsi le stockage persistant des données, l'accès aux données, le partage des données, la sécurité des données, la restauration des données.

II.2.1 Etude comparative de quelques SGBD [4]

Dans le tableau 3 présente un comparatif de quelques SGBD.

Tableau 3: Tableau comparatif de quelques SGBD

SGBD	Avantages	Inconvénients
PostgreSQL 9.3.1 	<ul style="list-style-type: none"> ✓ Il est open source et gratuit ✓ Il est fiable et relativement performant tout en restant simple d'utilisation et d'administration ✓ il supporte la majorité du standard SQL-92 ✓ Il très riche fonctionnellement : héritage de tables et multitude de modules ✓ Il offre une sauvegarde efficace à chaud ✓ Il propose un puissant langage procédural : le PL/pgSQL 	<ul style="list-style-type: none"> ✓ Une modification du fichier de sécurité pg_hba.conf nécessite un redémarrage pour être prise en compte ✓ Ses sauvegardes sont peu évoluées ✓ Il ne supporte que les bases de moyenne importance ✓ Il n'a pas d'ordonnanceur intégré ✓ Il n'a pas de fonctions d'agrégat OLAP
Oracle 9i 	<ul style="list-style-type: none"> ✓ Il est très riche fonctionnellement ✓ Il intègre une fonction d'audit évoluée ✓ Il jouit d'une haute disponibilité ✓ Il a un puissant langage procédural : le PL/SQL 	<ul style="list-style-type: none"> ✓ Son prix est très élevé ✓ Son administration est complexe ✓ Il est très gourmand en ressources ✓ Son optimiseur est très faible ✓ Il n'offre pas de type auto-incrément déclaratif, et ses séquences ne peuvent être dédiées à une table spécifique
MySQL 5.7.17 	<ul style="list-style-type: none"> ✓ Il est orienté service web ✓ Il est gratuit et Open Source ✓ Il est simple d'installation et d'utilisation ✓ Il est très intégré dans l'environnement Apache/PHP 	<ul style="list-style-type: none"> ✓ Il ne convient pas pour les grosses bases de données ✓ Il a peu de richesse fonctionnelle ✓ Il n'offre pas de sauvegarde à chaud

II.2.2 Choix du SGBD

Suite à cette étude comparative de SGBD, le groupe de projet en accord avec le groupe de pilotage a choisi **MySQL 5.7.17** pour mener ce projet.

Le choix de MySQL pour rester dans la logique du schéma directeur du service informatique de la **CARFO**, en plus, MySQL est une référence des SGBD du monde libre surtout pour le déploiement des applications web.

II.3 Description des scénarii

L'objectif de l'étude des scénarii est de déterminer l'architecture logique du futur système, les coûts enregistrés par équipements (matériel et logiciel) à acquérir ainsi que le coût total de l'application.

II.3.1 Premier scénario

Ce premier scénario consiste à mettre en place une application à deux couches (application 2-tiers).

II.3.1.1 Description générale

L'application devra être intégrée à **SIGA**.

SIGA est l'application de base utilisée à la **CARFO**, c'est une application 2-tiers développée avec **Developer 2000 d'Oracle**. Il permet la gestion automatisée des pensions des retraités, des risques professionnels etc.

Ce scénario implique que pour utiliser l'application, les utilisateurs doivent être à la **CARFO** car cette solution n'offre pas d'accès distant à l'application.

II.3.1.2 Présentation de l'architecture réseau

Comme souligné plus haut, l'application sera intégrée à **SIGA** qui lui ait déployé sur le réseau local existant de la **CARFO**. (cf. *figure 2 page 9*).

II.3.1.3 Besoins logiciels du premier scénario

Le tableau 4 présente la liste ainsi que les coûts associés des matériels à acquérir pour ce premier scénario.

Tableau 4: Besoins logiciels du premier scenario

Désignation	Coût (FCFA)/Disponibilité	
	Coût	Disponibilité
<i>Environnement de développement (Oracle)</i>	0	Déjà existant
<i>SGBD (Oracle 9i)</i>	0	Déjà existant
<i>Antivirus (Kaspersky Lab 2006)</i>	0	Déjà existant
<i>Licence Developer 2000</i>	0	Déjà existant

II.3.1.4 Besoins matériels du premier scenario

Le tableau 5 présente la liste des matériels à acquérir ainsi que les coûts associés pour ce premier scenario.

Tableau 5: Besoins matériels du premier scenario

Désignation	Caractéristiques	Nombre	Coût unitaire	Coût total
<i>Serveur de bases de données</i>	Serveur HP ProLiant DL580 Gen8/ Intel® Xeon® E7-4800 v2 / 3 To/52, 8 kg/ DDR3 RDIMM	1	Existant	0
<i>Ordinateurs</i>	PC de marque HP	suffisant	Existant	0
<i>Imprimantes</i>	HP laser jet P2015	suffisant	Existant	0
<i>Onduleur</i>	APC Back-UPS RS 800VA	suffisant	Existant	0
<i>Switch</i>	D-LINK DES-1008D SWITCH 8 PORTS RJ-45 D-Link 10/100 Mbps	suffisant	Existant	0
<i>Coût total des matériels à acquérir</i>				0

II.3.1.5 Coût de développement de l'application

En se référant aux applications de gestion de congés et d'absences déjà sur le marché, le nombre de lignes de code pour l'application à mettre en place est estimé à **5500** lignes.

En application les formules de COCOMO, l'estimation du coût de développement de l'application est le suivant :

- $HM = 3 * (5500/1000)^{1,12} = 20,25$ Homme-mois
- $TDEV = 2.5 * (20,25)^{0,35} = 7,16$ mois
- $N = 20,25/7,16 = 2,83$ hommes soit environ **3 hommes**
- Coût Total = $20,25 * 200000$ FCFA = **4 050 000 FCFA**

II.3.1.6 Coût de formation des utilisateurs

Pour permettre aux utilisateurs de pouvoir utiliser efficacement l'application, une formation sera faite à leur égard. Les détails de l'évaluation des coûts de la dite formation sont consignés dans le tableau 6.

Tableau 6 : Coût de formation des utilisateurs du premier scénario

Coût/session (FCFA)	Nombre de session	Nombres de personnes par session	Coût total (FCFA)
250000	2	50	500 000

II.3.1.7 Coût total du premier scénario

Suite aux différentes estimations faites, le coût total du premier scénario est donné dans le tableau 7.

Tableau 7: Estimation du coût total de l'application pour le premier scenario

Désignation	Coût (FCFA)
Coût du matériel à acquérir	0
Coût de développement	4 050 000
Coût des logiciels à acquérir	0
Coût de la formation des utilisateurs	500 000
Coût total du premier scenario	4 550 000

II.3.2 Deuxième scénario

Ce scénario consiste à mettre en place une application 3-tiers qui ne sera pas accessible via l'internet.

II.3.2.1 Description générale

L'application sera déployée sur le réseau local existant de la **CARFO**. L'application sera installée sur un serveur d'application et la base de données sur un serveur de bases de données.

Chaque l'utilisateur, depuis son poste, accèdera à l'application par son navigateur. L'application ne sera pas rendu accessible à distance via le réseau Internet.

II.3.2.2 Architecture réseau

L'application sera déployée sur le réseau local existant de la **CARFO**.

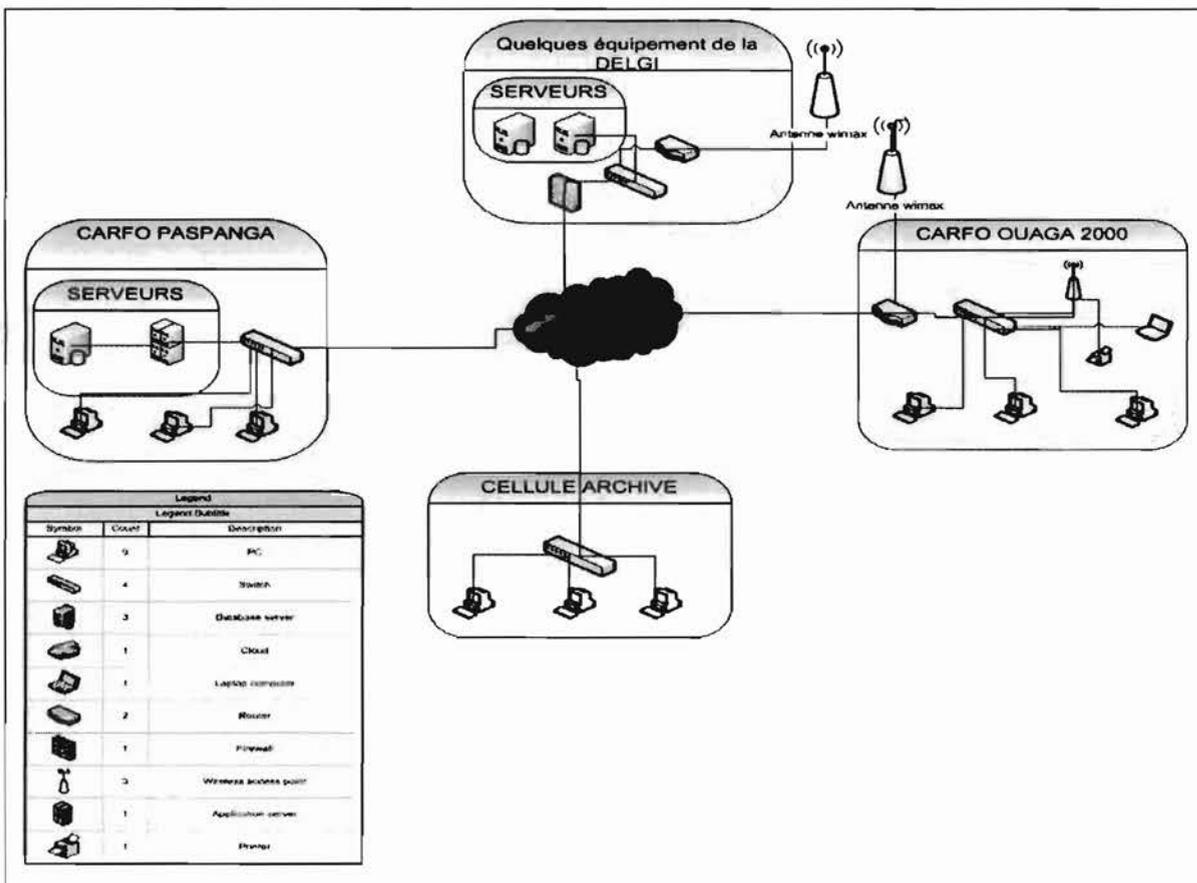


Figure 6 : Architecture réseau du deuxième scénario

II.3.2.3 Besoins logiciels du deuxième scénario

Tableau 8: *Besoins logiciels du deuxième scénario*

Désignation	Coût (FCFA)/Disponibilité	
	Coût	Disponibilité
<i>Environnement de développement (Netbeans 7.3.1)</i>	0	Déjà existant
<i>SGBD (Myql 5.7.17)</i>	0	Déjà existant
<i>Serveur d'application (Tomcat 7.0.42)</i>	0	Déjà existant
<i>Anti-virus(Kaspersky Kaspersky Lab 2006)</i>	0	Déjà existant
<i>JEE 5</i>	0	Déjà existant
<i>Coût total des logiciels à acquérir</i>		0

II.3.2.4 Besoins matériels du deuxième scenario

Tableau 9: *Besoins matériels du deuxième scenario*

Désignation	Caractéristiques	Nombre	Coût unitaire	Coût total
<i>Serveur de bases de données</i>	Serveur HP ProLiant DL580 Gen8/ Intel® Xeon® E7-4800 v2 / 3 To/52, 8 kg/ DDR3 RDIMM	1	0 (existant)	0
<i>Serveur d'application [5]</i>	Format tour, Xeon® Quad Core, E3-1230, 32 Go, 365 Watts, 500 Go / RAID 1	1		749 931
<i>Ordinateurs</i>	PC de marque HP	suffisant	0 (existant)	0
<i>Imprimantes</i>	HP laser jet P2015	suffisant	0 (existant)	0
<i>UPS</i>	APC Back-UPS RS 800VA	suffisant	0 (existant)	0
<i>Switch</i>	D-LINK DES-1008D SWITCH 8 PORTS RJ-45 D-Link 10/100 Mbps	suffisant	0 (existant)	0
Coût total des matériels à acquérir				749 931

II.3.2.5 Coût du développement

Le nombre de lignes de code pour ce scenario est estimé à 6500 lignes.

En application des formules de COCOMO l'estimation du coût de développement de l'application est le suivant :

$$\textcircled{O} \text{ HM} = 3(6500/1000)^{1,12} = 24,41 \text{ Homme-mois}$$

$$\textcircled{O} \text{ TDEV} = 2.5(24,41)^{0,35} = 7,65 \text{ mois}$$

$$\textcircled{O} \text{ N} = 24,41/7,65 = 3,19 \text{ hommes soit environ 3 hommes}$$

$$\text{Coût Total} = 24,42 * 200000 \text{ FCFA} = 4\,882\,000 \text{ FCFA}$$

II.3.2.6 Coût de formation des utilisateurs

Tableau 10: Coût de formation des utilisateurs

Coût horaire (FCFA)	Nombre d'heures par utilisateur	Nombres d'utilisateurs	Coût total (FCFA)
5000	15	100	500 000

II.3.2.7 Coût total du deuxième scénario

Tableau 11: Estimation du coût total de l'application pour le deuxième scénario

Désignation	Coût (FCFA)
Coût du matériel à acquérir	749 931
Coût de développement	4 882 000
Coût des logiciels à acquérir	0
Coût de la formation des utilisateurs	500 000
Coût total du deuxième scénario	6 131 931

II.3.3 Troisième scénario

Ce troisième et dernier scénario consiste à mettre en place une application 3-tiers qui sera lui accessible à distance via le réseau Internet.

II.3.3.1 Description générale

L'application à développer sera de type 3-tiers et va tourner sur un serveur d'application.

Chaque utilisateur pourra accéder en local à l'application, mais aussi à distance via le réseau Internet.

II.3.3.2 Architecture réseau du troisième scénario

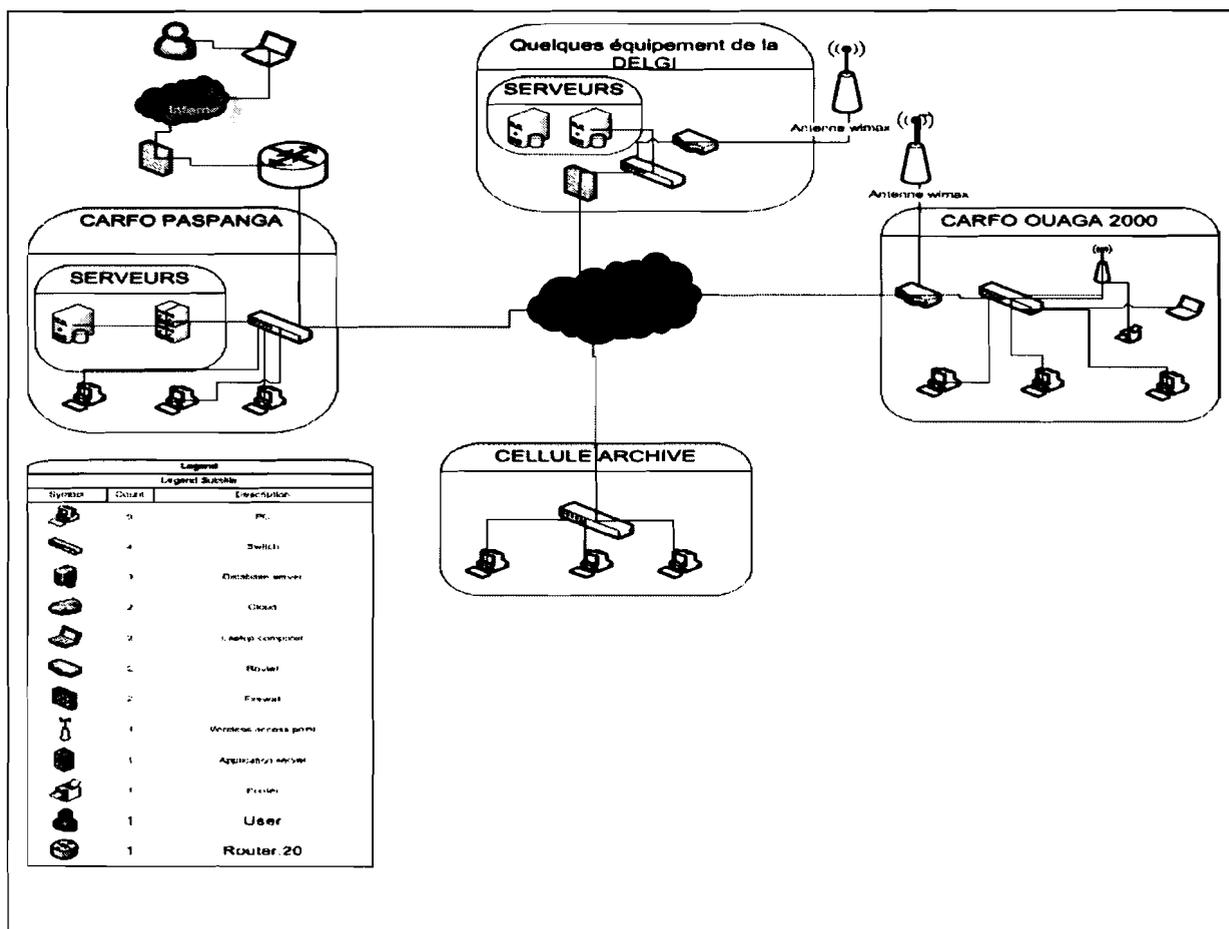


Figure 7 : Architecture réseau du troisième scénario

II.3.3.3 Besoins logiciels du troisième scénario

Tableau 12: *Besoins logiciels du troisième scénario*

Désignation	Coût (FCFA)/Disponibilité	
	Coût	Disponibilité
<i>Environnement de développement (NetBeans 7.3.1)</i>	0	Déjà existant
<i>SGBD (MySQL 5.7.17)</i>	0	Déjà existant
<i>Serveur d'application (Apache tomcat 7.0.42)</i>	0	Déjà existant
<i>Anti-virus(Kaspersky Kaspersky Lab 2006)</i>	0	Déjà existant
<i>Adresse internet</i>	50 000	Non existant
<i>Coût total des logiciels à acquérir</i>		50 000

II.3.3.4 Besoins matériels du troisième scénario

Tableau 13: Besoins matériels du troisième scénario

Désignation	Caractéristiques	Nombre	Coût unitaire	Coût total
<i>Serveur de bases de données</i>	Serveur HP ProLiant DL580 Gen8/ Intel® Xeon® E7-4800 v2 / 3 To/52, 8 kg/ DDR3 RDIMM	1	0 (existant)	0
<i>Serveur d'application</i>	Format tour, Xeon® Quad Core, E3-1230, 32 Go, 365 Watts, 500 Go / RAID 1	1		749 931
<i>Ordinateurs</i>	PC de marque HP	suffisant	0 (existant)	0
<i>Imprimantes</i>	HP laser jet P2015	suffisant	0 (existant)	0
<i>UPS</i>	APC Back-UPS RS 800VA	suffisant	0 (existant)	0
<i>Switch</i>	D-LINK DES-1008D SWITCH 8 PORTS RJ-45 D-Link 10/100 Mbps	suffisant	0 (existant)	0
<i>Routeur [6]</i>	Routeur ADSL / Série : Cisco systems 2900/ Port LAN : 2*RJ-45/ Standard LAN : Gigabit Ethernet (10/100/1000 Mbits/s)/ Assignation d'adresse IP : Support DHCP Dynamique, Statique/ Technologie : ADSL, ADSL 2, ADSL 2+	1	35 750	500 000
<i>Firewall</i>	Cisco RV082 V3			168909
Coût total des matériels à acquérir				1418840

II.3.3.5 Coût de développement

Le nombre de lignes de code de l'application dans ce scénario est estimée à 6900 (400 lignes de plus comparativement au deuxième scénario).

Les 400 lignes de codes supplémentaires sont utilisées pour renforcer la sécurité applicative liée à la menace que représente l'accessibilité via l'Internet.

En application des formules de COCOMO, l'estimation du coût de développement de l'application est le suivant :

- $HM = 3(6900/1000)^{1,12} = 26,1$ Homme-mois
- $TDEV = 2.5(8,7)^{0,35} = 7,83$ mois
- $N = 26,1/7,83 = 3,33$ hommes soit environ 3 hommes
- Coût Total = $26,1 * 200000$ FCFA = 5 220 000 FCFA

II.3.3.6 Coût de formation des utilisateurs

Tableau 14: Coût de formation des utilisateurs pour le troisième scénario

Coût horaire (FCFA)	Nombre d'heures par utilisateur	Nombres d'utilisateurs	Coût total (FCFA)
5000	15	100	500 000

II.3.3.7 Coût total du troisième scénario

Suite aux différentes estimations faites, le coût total du premier scénario est donné dans le tableau 15.

Tableau 15: Coût total du troisième scénario

Désignation	Coût (FCFA)
<i>Coût du matériel à acquérir</i>	1 418 840
<i>Coût de développement</i>	5 220 000
<i>Coût des logiciels à acquérir</i>	50 000
<i>Coût de la formation des utilisateurs</i>	500 000
<i>Coût total du troisième scénario</i>	7 188 840

II.3.4 Critique des scénarii

II.3.4.1 Premier scenario

○ Avantages

- système entièrement isolé de l'extérieur ce qui représente un gain de sécurité ;
- mise en œuvre plus ou moins facile ;
- mode d'utilisation de l'application familier aux utilisateurs ;
- scenario moins couteux.

○ Inconvénients

- difficultés de maintenance de l'application, elle doit se faire sur chaque poste client ;
- impossibilité de connexion à distance à l'application ;
- application trop lourd en mémoire centrale pour les ordinateurs clients.

II.3.4.2 Deuxième scenario

○ Avantages

- système entièrement isolé de l'extérieur ce qui représente un gain de sécurité ;
- pas de surcharge des ordinateurs clients qui utilisent seulement leur navigateur web pour accéder à l'application ;
- rapidité de traitement des requêtes ;
- facilité de maintenance car la maintenance se fait sur une seule machine sur laquelle est installée l'application, le serveur d'application ;
- facilité de développement grâce à la séparation par couche (vue, contrôleur et modèle) ;
- facilité d'évolution de l'application grâce à la séparation par couche ;
- convivialité de l'interface utilisateur ;

○ Inconvénients

- impossibilité de connexion à distance à l'application ;
- mode d'utilisation pas familier aux utilisateurs ;
- présence d'un maillon faible (le serveur d'applications) qui en cas de panne entraîne une indisponibilité totale de l'application ;
- coût plus ou moins élevé.

II.3.4.3 Troisième scénario

Le scénario 3 présente aussi bien des avantages que des inconvénients.

O Avantages

- absence de contraintes géographiques de l'utilisateur ;
- accessibilité en tout temps ;
- pas de surcharge des ordinateurs clients qui utilisent seulement leur navigateur web pour accéder à l'application ;
- rapidité de traitement des requêtes ;
- facilité de maintenance car la maintenance se fait sur une seule machine sur laquelle est installée l'application, le serveur d'applications ;
- facilité de développement grâce à la séparation par couche (vue, contrôleur et modèle) ;
- facilité d'évolution de l'application grâce à la séparation par couche ;
- convivialité de l'interface utilisateur ;

O Inconvénients

- insécurité liée à la présence sur l'Internet ;
- mode d'utilisation pas familier aux utilisateurs ;
- présence d'un maillon faible (le serveur d'application) qui en cas de panne entraîne une indisponibilité totale de l'application ;
- coût plus ou moins élevé.

II.3.4.4 Etude comparative des scénarii

Le but de l'étude comparative des scénarii est de comparer suivant certains critères les différents scénarii afin d'en choisir un.

Ce choix doit être consensuel entre le groupe de projet, le groupe de pilotage et le groupe des utilisateurs.

Le tableau 16 synthétise l'étude comparative des trois scénarii.

Tableau 16: Etude comparative des scénarii

Critères	Premier scenario	Deuxième scenario	Troisième scenario
<i>Maintenance</i>	Difficile	Facile	Facile
<i>Sécurité</i>	Elevée	Faible	Faible
<i>Coût</i>	Faible	Elevé	Elevé
<i>Intérêt formateur</i>	Faible	Elevé	Elevé
<i>Evolution</i>	Difficile	Facile	Facile
<i>Ergonomie</i>	Moins conviviale	Conviviale	Conviviale
<i>Accessibilité</i>	Limité	Limité	Forte accessibilité

II.3.4.5 Choix de scénario

Au regard de l'étude comparative de ces trois scénarii et des besoins et objectifs de l'entreprise, le troisième scenario a été choisi par le groupe de projet en consensus avec le groupe de pilotage.

Le critère d'accessibilité a été le critère le plus déterminant dans le choix de ce scénario.

II.3.5 Modélisation du futur système

II.3.5.1 Le diagramme de cas d'utilisation

Le diagramme de cas d'utilisation permet de recueillir l'ensemble des besoins fonctionnels du futur système à mettre en place. Il montre aussi les interactions possibles entre les acteurs avec le futur système.

La figure 5 présente le diagramme de cas d'utilisation du futur système.

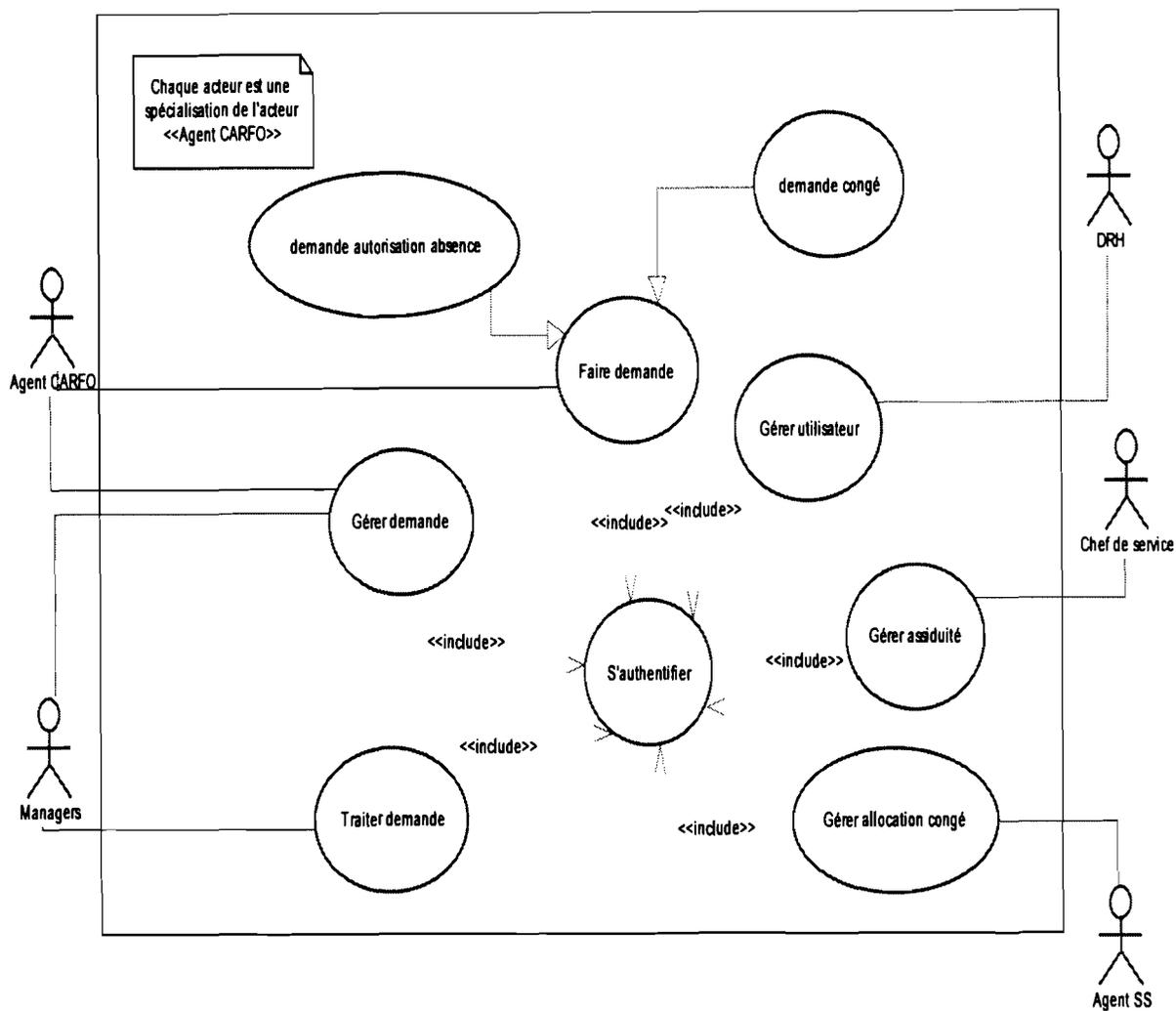


Figure 8 : Diagramme de cas d'utilisation du futur système

Description textuelle de cas d'utilisation

La description textuelle des cas d'utilisation permet de bien expliciter les cas d'utilisation en montrant notamment les activités qu'ils comportent.

CU1 : Faire demande	ScENARIO nominal	Folio 1/1
Résumé : Ce cas permet à un agent de faire une demande d'autorisation d'absence	Version 1.0	
	19/11/2013	
Préconditions : L'agent veut faire une demande d'autorisation d'absence		
Acteurs : Agent CARFO		
<Début>		
1. Inclusion du cas « S'authentifier »		
2. Remplir le formulaire de la demande		
4. Envoyer le formulaire		
<Fin>		

CU3 : S'authentifier	ScENARIO nominal	Folio 1/3
Résumé : Ce cas permet aux utilisateurs de s'authentifier avec des mots de passe avant d'accéder aux fonctionnalités de l'application afin de garantir la sécurité des données	Version 1.0	
	19/11/2013	
Préconditions : L'application est lancée		
Acteurs : Tous les utilisateurs		
<Début>		
1. Le système demande à l'utilisateur de saisir son nom d'utilisateur et son mot de passe		
2. L'utilisateur saisit les informations demandées		
3. Le système vérifie la véracité des informations entrées par l'utilisateur (A1) (E1)		
4. Le système ouvre l'espace de travail de l'application correspondant au profil de l'utilisateur connecté		
<Fin>		

CU3 : S'authentifier	Scenario alternatif	Folio 2/3
Résumé : Ce cas permet aux utilisateurs de s'authentifier avec des mots de passe avant d'accéder aux fonctionnalités de l'application afin de garantir la sécurité des données		Version 1.0
		19/11/2013
Préconditions : L'application est lancée		
Acteurs : Tous les utilisateurs		
<Début>		
A1. Le mot de passe ou le nom d'utilisateur entré n'est pas correct en moins de trois essais		
A1.1 : le système informe l'utilisateur que la connexion a échoué		
A1.2 : on repart à l'activité (03) du scenario nominal		
<Fin>		

CU3 : S'authentifier	Scenario d'erreur	Folio 3/3
Résumé : Ce cas permet aux utilisateurs de s'authentifier avec des mots de passe avant d'accéder aux fonctionnalités de l'application afin de garantir la sécurité des données		Version 1.0
		19/11/2013
Préconditions : L'application est lancée		
Acteurs : Tous les utilisateurs		
<Début>		
E1. Le mot de passe ou le nom d'utilisateur entré n'est pas correct pour plus de trois essais		
E1.1 : Le système informe l'utilisateur que son compte a été bloqué		
E1.2 : le système propose à l'utilisateur de contacter l'administrateur		
E1.3 : Fin de scenario		
<Fin>		

CU4 : Gérer assiduité	Scenario nominal	Folio 1/1
Résumé : Ce cas permet à un directeur de service de contrôler l'assiduité des agents qui sont à sa décharge	Version 1.0	
	19/11/2013	
Préconditions : Fin de semaine		
Acteurs : Directeur de service		
<p><Début></p> <ol style="list-style-type: none"> 1. Inclusion du cas « S'authentifier » 2. Enregistrer les absences constatées des agents 3. Modifier des absences 4. Supprimer des absences <p><Fin></p>		

CU5 : Gérer utilisateur	Scenario nominal	Folio 1/1
Résumé : Ce cas permet au DRH de faire des tâches liées à la gestion des Utilisateurs	Version 1.0	
	19/11/2013	
Préconditions : Un besoin		
Acteurs : DRH		
<p><Début></p> <ol style="list-style-type: none"> 1. Inclusion du cas « S'authentifier » 2. Créer un utilisateur (nom, login, profil,...) 3. Modifier un utilisateur (login, profil,...) 4. Supprimer un utilisateur <p><Fin></p>		

CU5 : Traiter demande	Scenario nominal	Folio 1/1
Résumé : Ce cas permet à un manager de traiter une demande	Version 1.0	
	19/11/2013	
Préconditions : Une demande a été envoyée		
Acteurs : Directeur de service		
<Début>		
1. Inclusion du cas « S'authentifier »		
2. Consulter demande		
3. Consulter l'outil d'aide à la décision		
4. Porter avis à la demande		
<Fin>		

II.3.5.2 Quelques diagrammes de séquence

Un diagramme de séquence permet de décrire l'enchaînement des activités au sein d'un cas d'utilisation.

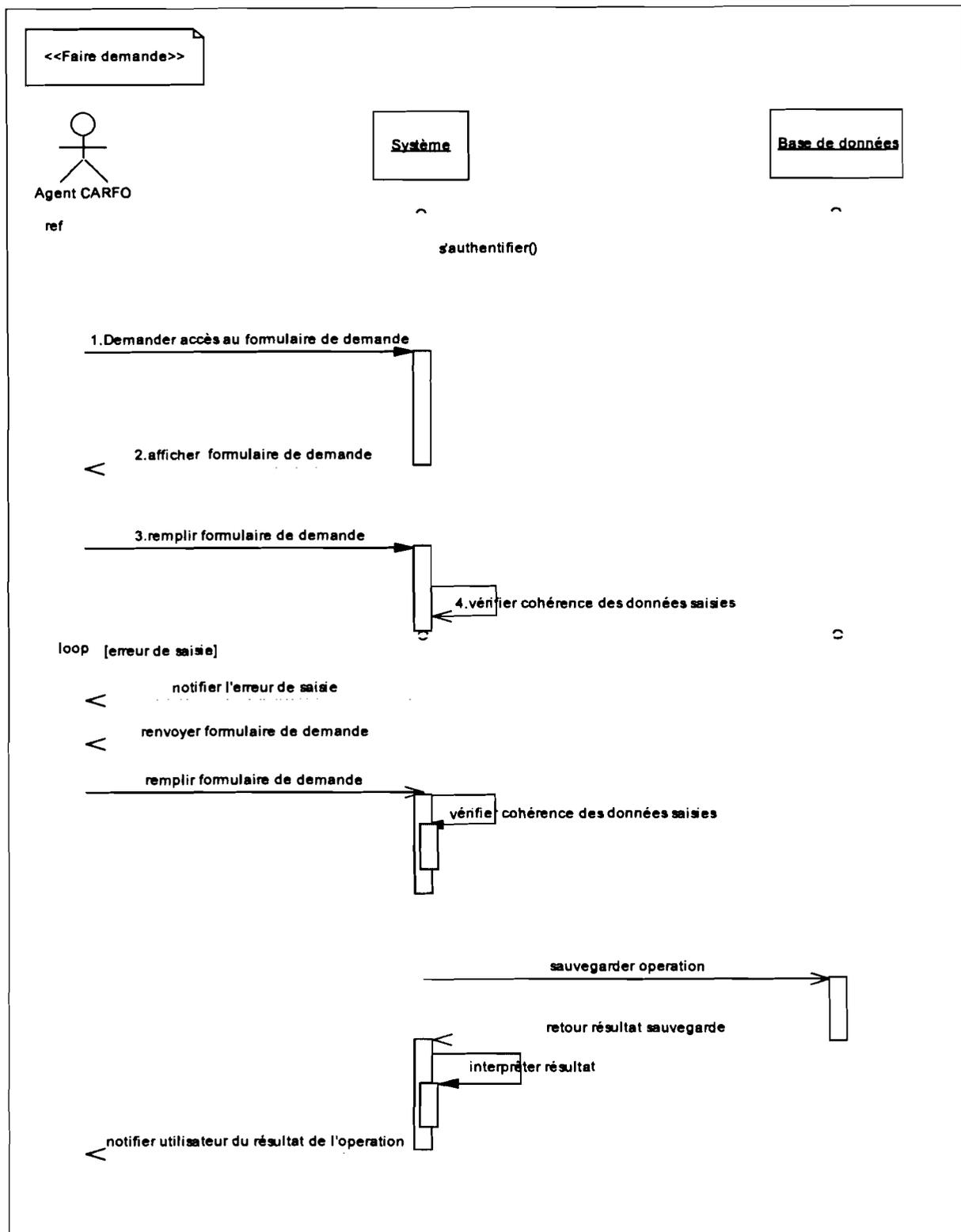


Figure 9: Diagramme de séquence du cas d'utilisation : « Faire demande »

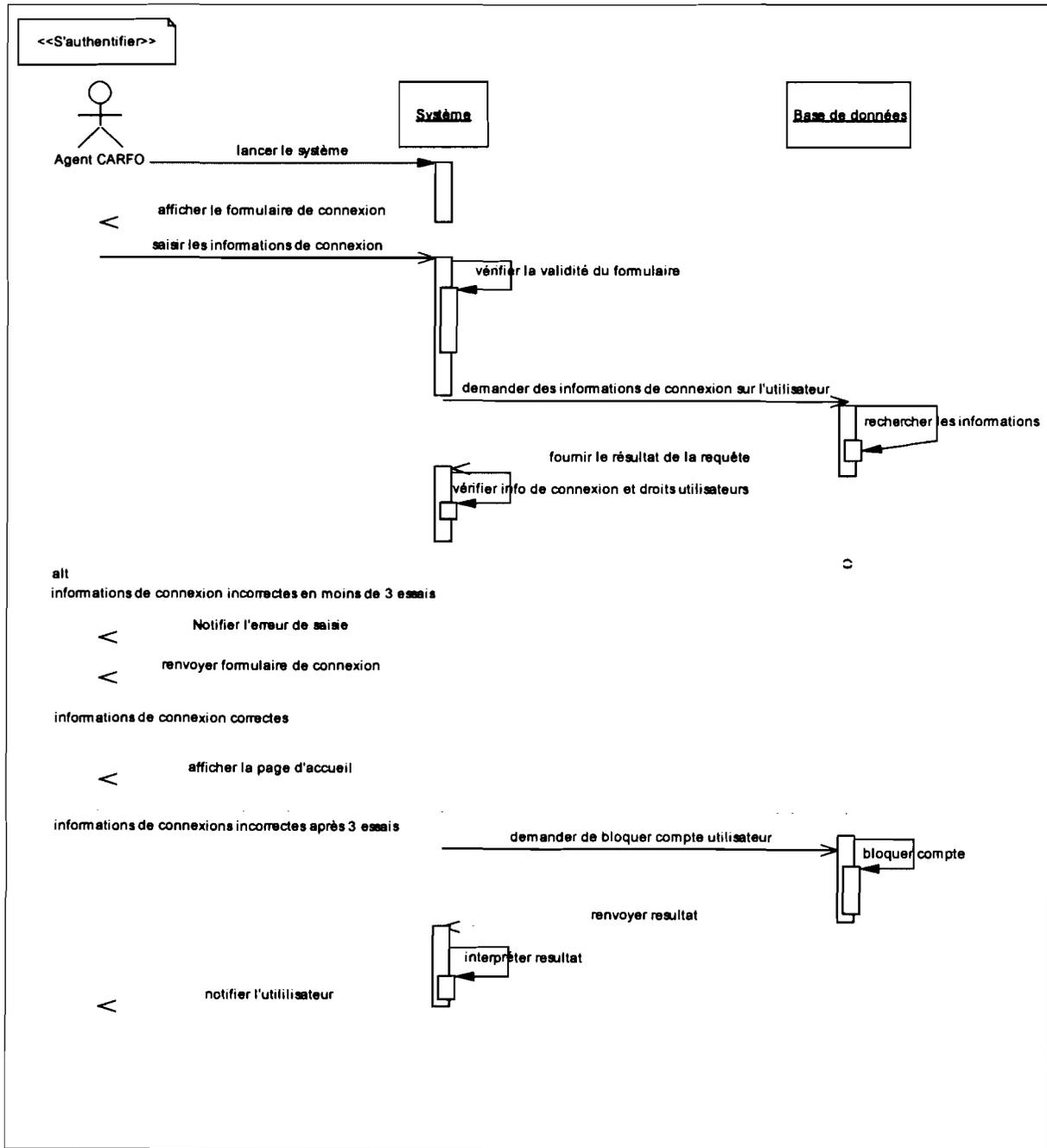


Figure 10: Diagramme de séquence du cas d'utilisation « S'authentifier »

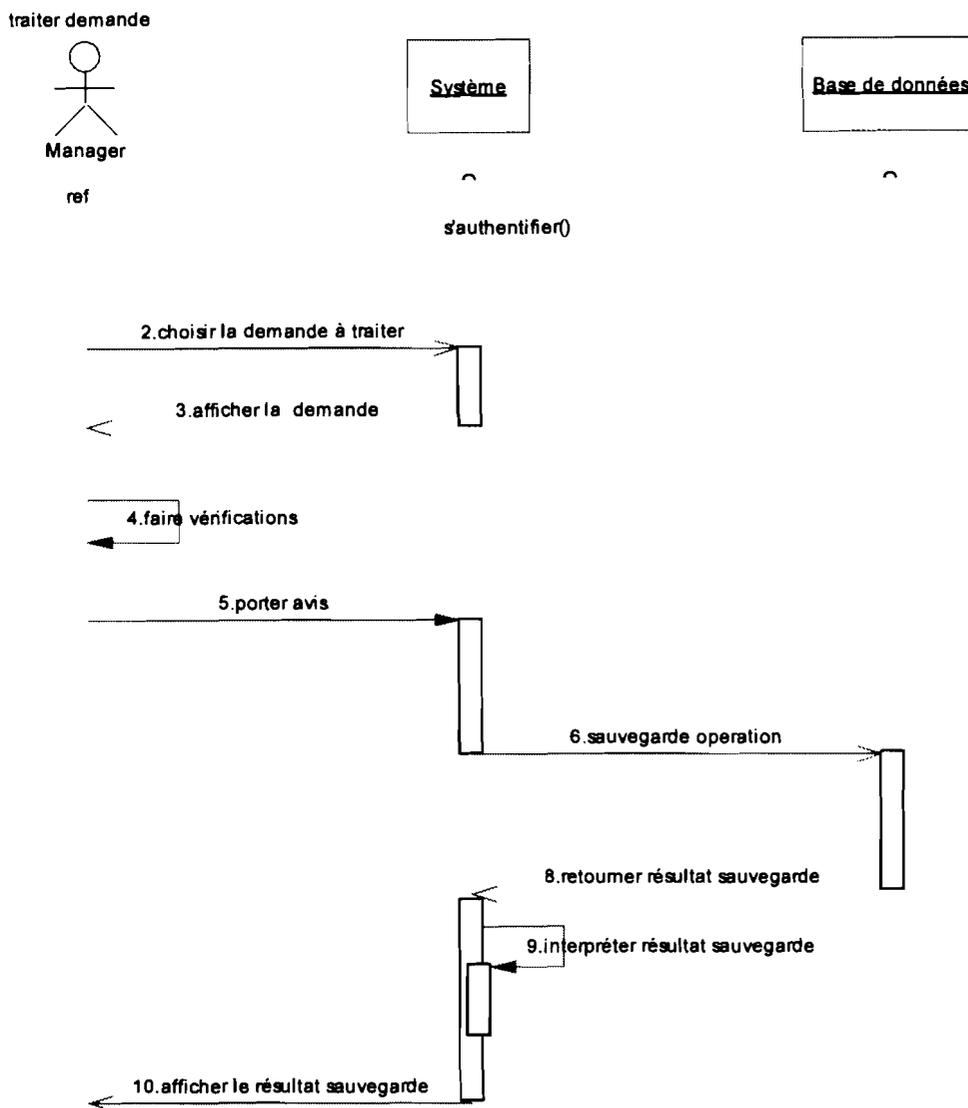


Figure 11 : Diagramme de séquence du cas d'utilisation : « Traiter demande »

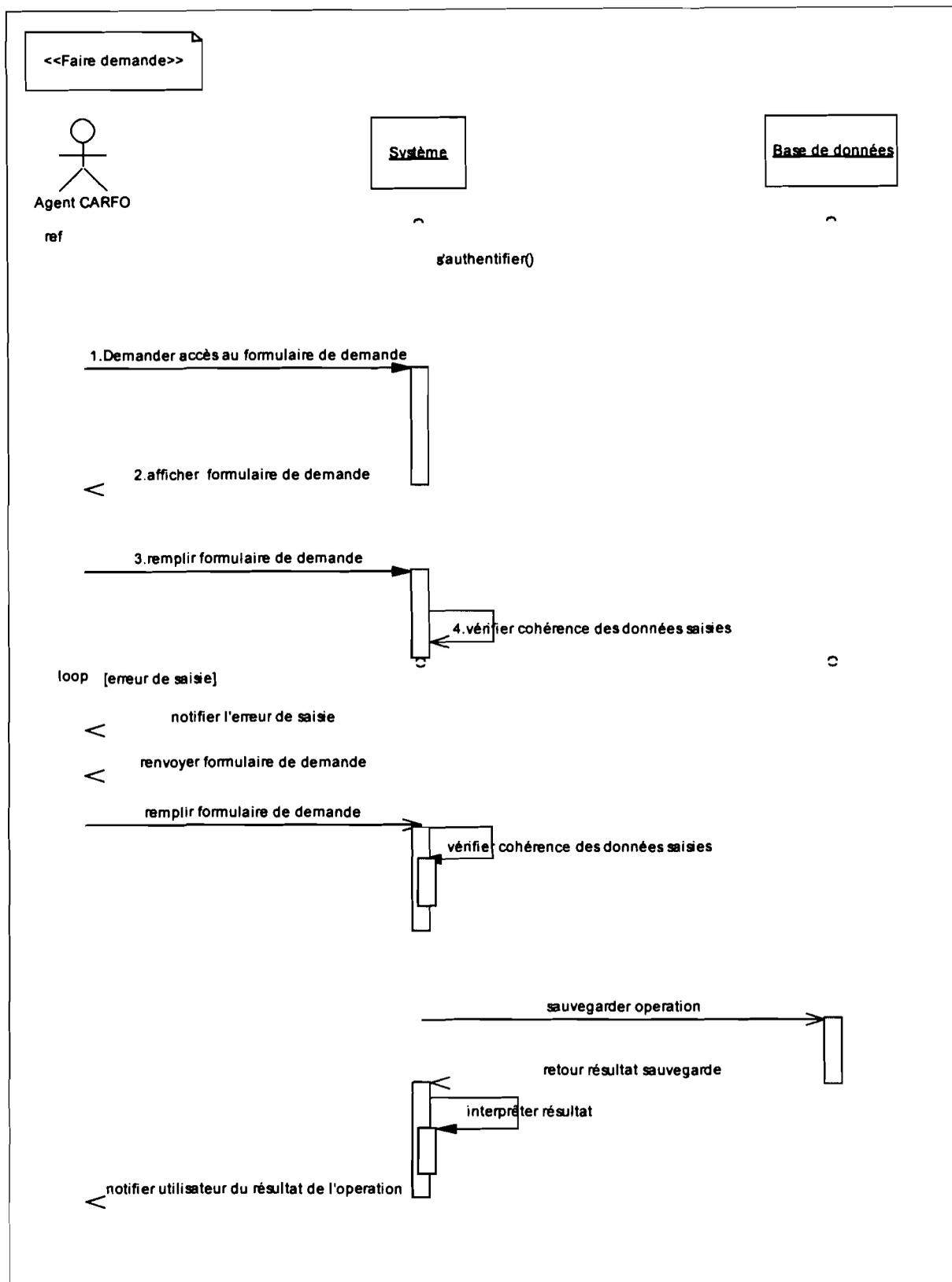


Figure 12: Diagramme de séquence du cas d'utilisation « Faire demande »

gerer allocation de congé

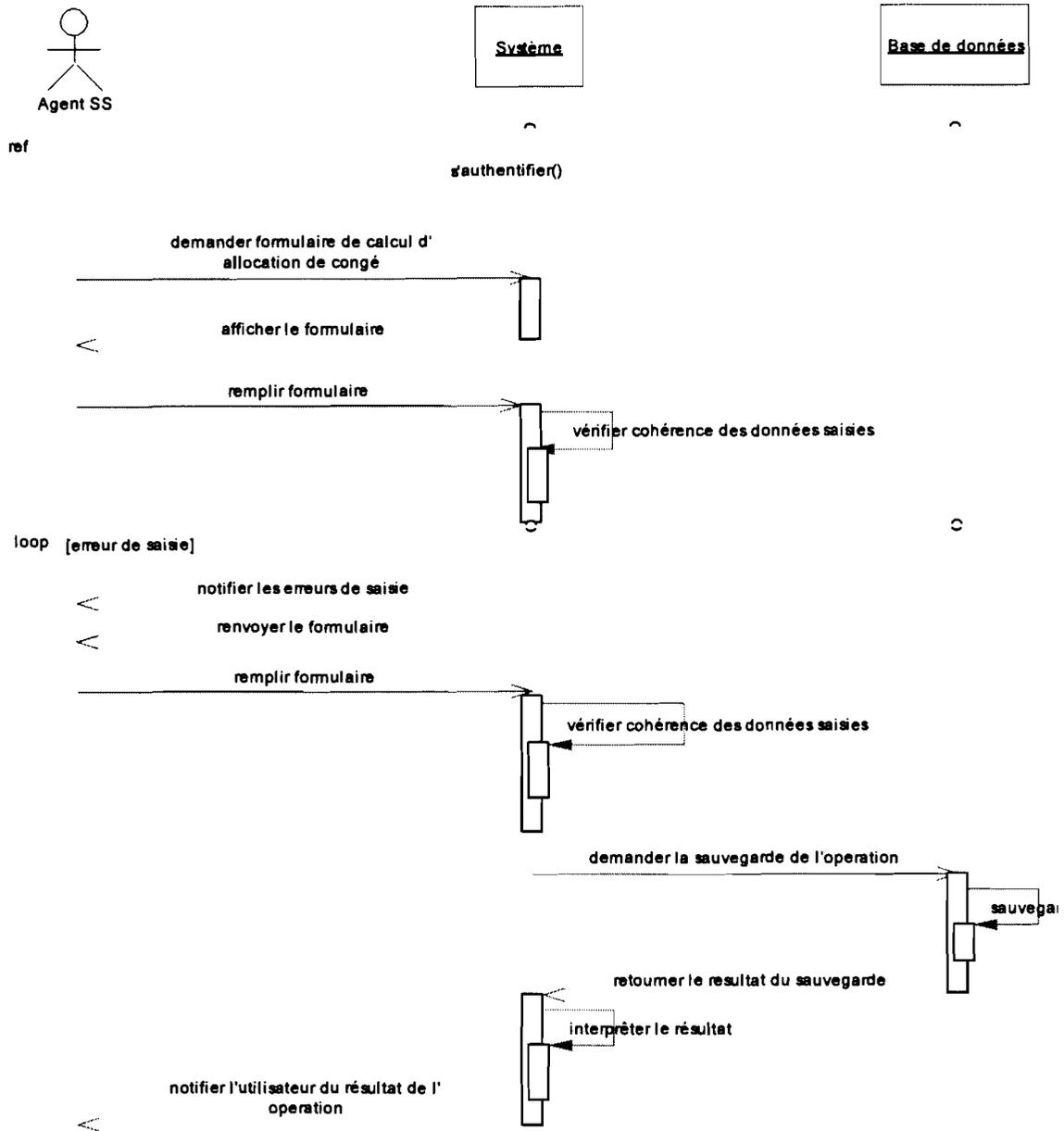


Figure 13: Diagramme de séquence du cas d'utilisation « Gérer allocation de congé »

II.3.5.3 Quelques diagrammes d'activités

Un diagramme d'activité permet de décrire les flux de contrôle dans un cas d'utilisation ou entre plusieurs cas d'utilisation.

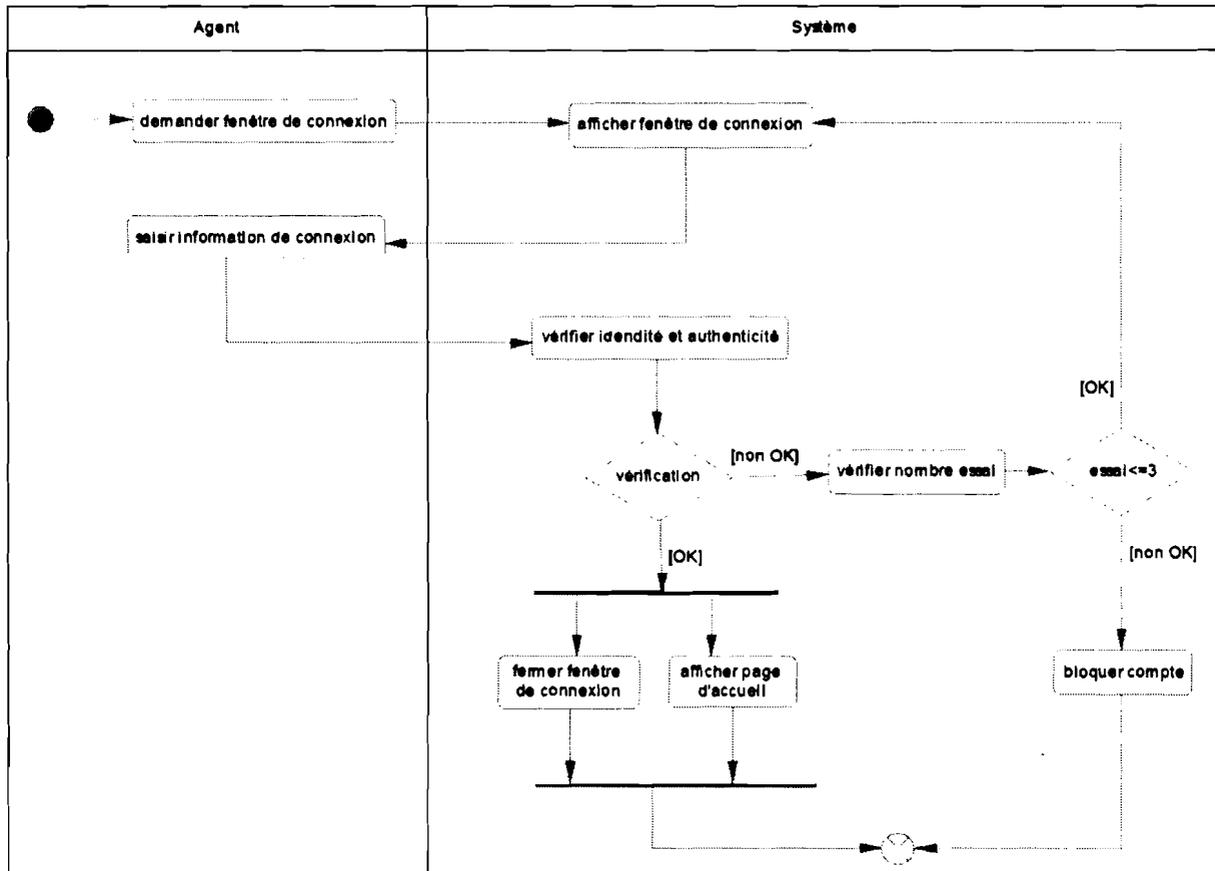


Figure 14: Diagramme d'activité du cas d'utilisation « S'authentifier »

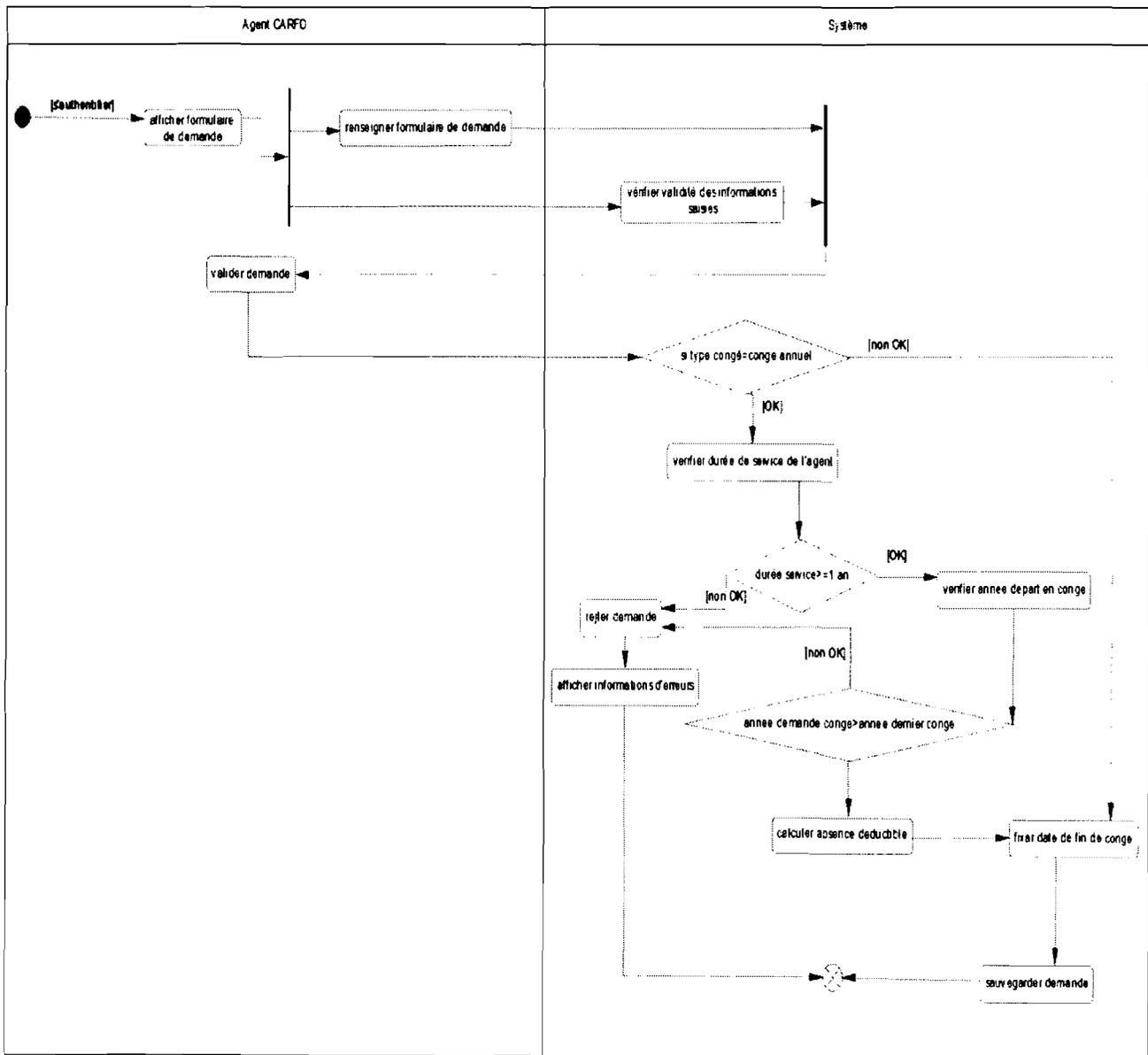


Figure 15 : Diagramme d'activité du cas d'utilisation « Faire demande congé »

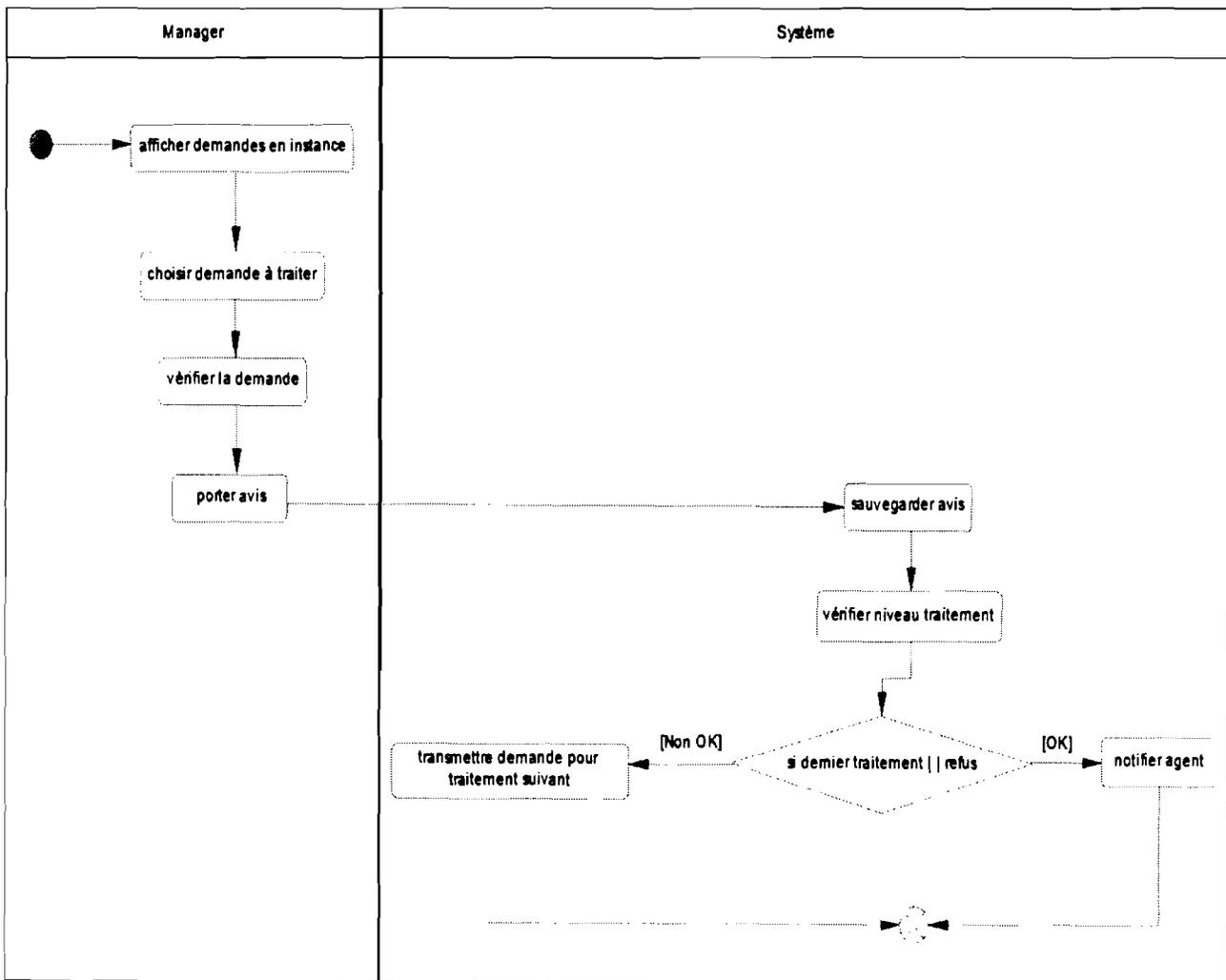


Figure 16: Diagramme d'activité du cas d'utilisation « Traiter demande »

II.3.5.4 Le diagramme de classe

Un diagramme de classe permet de décrire l'ensemble des objets du système, c'est l'un des diagrammes les plus importants d'UML.

II.3.5.4.1 Présentation du diagramme de classe

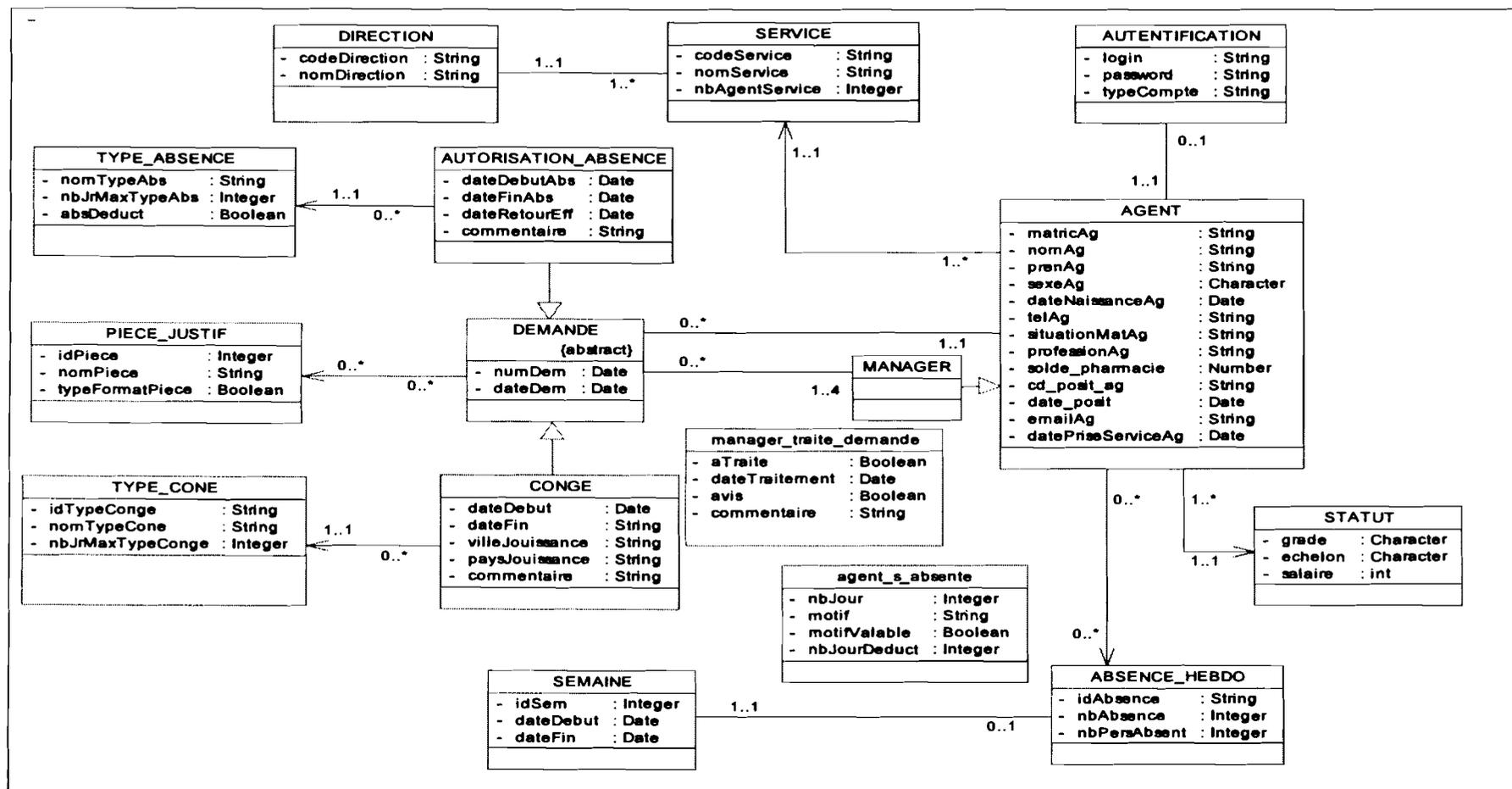


Figure 17 : Diagramme de classe

II.3.5.4.2 Quelques règles de gestion

Afin de respecter la logique métier du futur système, nous avons établi des règles de gestion (RG). Certaines découlent du fonctionnement du système actuel et d'autres ont été introduites par le groupe de projet dans le but de corriger les insuffisances du système existant (*cf. tableau 17*).

Tableau 17: *Quelques règles de gestion*

Numéro	Description
RG 1	Une demande est traitée par au moins une personne et au plus 4 personnes
RG 2	Un agent possède 0 ou 1 compte
RG 3	Une demande nécessite 0 ou plusieurs pièces justificatives
RG 4	Un congé correspond à un type de congé et un seul
RG 5	Un agent envoie 0 ou plusieurs messages
RG 6	Un agent reçoit un ou plusieurs messages
RG 7	Un service appartient à une et une seule direction
RG 8	Une direction possède 1 à plusieurs services
RG 9	Un agent peut s'absenter 0 à plusieurs fois dans la semaine
RG 10	Un agent a un et un seul statut
RG 11	Plusieurs agents peuvent avoir le même statut
RG 12	Un agent peut faire 0 ou plusieurs demandes
RG 13	Une demande appartient à un et un seul agent

CONCLUSION

Il a été question essentiellement dans ce chapitre de proposer des scénarii et de faire le choix du scénario qui répond au mieux aux besoins des utilisateurs.

Le choix du scénario s'est porté sur le scénario 3 qui permettra au futur système d'être accessible via Internet. De plus, une modélisation du système futur a été faite en vue de permettre de mieux appréhender la phase de construction.

CHAPITRE III : CONSTRUCTION ET TRANSITION

INTRODUCTION

Après avoir fait le choix de notre scénario et fait sa modélisation, la mise en œuvre du système est indispensable car elle va permettre au groupe des utilisateurs et celui de pilotage d'avoir une idée concrète du système futur.

Nous aborderons dans cette partie une présentation succincte des fonctionnalités du système future et de quelques maquettes de l'application. Ensuite, nous décrivons la politique de sécurité à mettre en œuvre pour assurer le bon fonctionnement du système.

III.1 CONSTRUCTION

III.1.1 Description des fonctionnalités de l'application

L'application porte le nom de : «**GCAP**» (Gestion des Congés et Absences Professionnels) Conformément aux besoins des utilisateurs (traduit dans le diagramme de cas d'utilisation) l'application présente les fonctionnalités suivantes :

- Pour tous les utilisateurs :
 - demande de congés et d'autorisations d'absences en ligne (congé administratif, congés maladies) ;
 - notification automatique par SMS en réponse à une demande ;
 - modification de demandes ;
 - utilisation de la messagerie ;
 - annulation de demandes ;
 - consultation de l'état de compteur (nombre de jour pris et restants) ;
 - consultation du nombre de jour auquel il a droit (affichage automatique lors de la demande de congé) ;
 - consultation de l'historique des congés et absences.

O Les chefs de service

- validation ou refus des demandes de congés ou d'absences (avec commentaires optionnelles) ;
- notification automatique en cas de demande à traiter ;
- listing de l'historique des absences et congés des agents du service ;
- consultation des états de compteurs (Nombre de jours pris et restants) pour chaque agent du service ;
- consultation des statistiques des congés et absences des agents du service ;
- enregistrement des absences hebdomadaires constatées et qui n'ont pas fait cas de demande préalable;

O Les directeurs de direction

- validation ou refus des demandes de congés ou d'absences (avec commentaires optionnels) ;
- listing de l'historique des absences et congés des agents par service et de tous les services de la direction ;
- notification automatique en cas de demande à traiter ;
- consultation des états de compteurs (Nombre de jours pris et restants) pour chaque agent de chaque service de la direction ;
- consultation des statistiques des congés et absences des agents par service et de tous les services de la direction ;

O Pour le service de l'administration du personnel

- consultation de la liste de départ en congé des agents par service, par direction, de tous les services de toutes les directions ;
- consultation de la liste des agents autorisés à s'absenter (par service, par direction,...).

O Pour le DRH

- validation ou refus des demandes de congés ou d'absences (avec commentaires optionnels) ;
- consultation de la liste de départ en congé des agents par service, par direction, de tous les services de toutes les directions;
- consultation de la liste des agents autorisés à s'absenter (par service, par direction, de tous les services de toutes les directions) ;
- notification automatique en cas de demande à traiter ;
- listing de l'historique des congés et absences des agents par service, par direction... ;

- consultation des états de compteurs (nombre de jours pris et restants) pour chaque agent ;
 - consultation des statistiques des congés et absences pour chaque agent.
 - consultation des statistiques des départs en congé par type de congé, par service, par direction... ;
 - consultation des statistiques des autorisations d'absences par type d'absence, par service, par direction... ;
 - gestion de compte et des droits utilisateurs.
- Le service de la solde
- consultation de la liste de départ en congé ;
 - calcul des allocations de congés payés.

III.1.2 Langages de programmation

Un langage de programmation est un langage informatique, permettant à un être humain d'écrire un code source qui sera analysé par une machine, généralement un ordinateur.

Nous avons mené une étude comparative de langage de programmation.

III.1.2.1 Etude comparative de langages de programmation web

Tableau 18: Etude comparative de langages de programmation web [7]

Nom	Points forts	Points faibles
<p>ASP.NET</p> 	<ul style="list-style-type: none"> ✓ Il est compilé, rapidité d'exécution une fois qu'il est exécuté ✓ Le langage a déjà une bonne structure et architecture pour la programmation : Bibliothèque située à une place, les pages Web à une autre, le code à une autre. ✓ Le langage est soutenu par Microsoft possibilité de développer en VB.NET ou en C# (C Sharp). ✓ Solution utilisée par de nombreux gouvernements et institutions financières. ✓ Beaucoup de documentation par les sites Web altruiste 	<ul style="list-style-type: none"> ✓ Le coût est nécessairement plus élevé que les autres puisqu'il n'est pas gratuit. ✓ L'hébergement se fait exclusivement sous Windows server 2003, 2008, ✓ A la merci des choix technologiques de Microsoft : Ce veut un remplacement du ASP 3.0, lequel révélait de nombreuses limitations, manques de structures, rapidité médiocre
<p>JEE 5[8]</p> 	<ul style="list-style-type: none"> ✓ Très grande portabilité ✓ Plus grande stabilité du code à travers le temps ✓ Le langage est soutenu par Oracle (anciennement Sun Microsystems). ✓ Stabilité du JFC ✓ Sécurité excellente avec Java/CORBA ✓ Solution utilisée par de nombreuses grandes entreprises et institutions financières 	<ul style="list-style-type: none"> ✓ Il est plus demandant au niveau du microprocesseur ✓ L'hébergement du JSP n'est pas disponible chez tous les hébergeurs

Nom	Points forts	Points faibles
<p>PHP 5</p> 	<ul style="list-style-type: none"> ✓ Il est gratuit ✓ Il existe une grande variété de projets opensource disponible pour celui-ci ✓ Il ne nécessite pas beaucoup de code pour obtenir un résultat ✓ L'hébergement du PHP est supporté presque partout ✓ Beaucoup de documentation par les sites Web altruiste 	<ul style="list-style-type: none"> ✓ Les failles de sécurité se sont révélées très nombreuses au fil des années ✓ Le PHP 6 tarde depuis déjà 8 ans : la version PHP5 date de 2004, c'est très long 8 ans en informatique !!! ✓ Il n'y pas de structure prédéfini, il faut donc ajouter des cadres d'applications pour atteindre des niveaux comme le MVC (<i>Zend, Symfony,...</i>) ✓ Les fonctions sont biodégradables : environ 500 fonctions sont obsolètes depuis sa version 3, sachant qu'il y a environ 5 000 fonctions, ceci indiquant une volatilité de 10%
<p>ColdFusion</p>  <p><small>MADE WITH THE NEW COLD FUSION</small></p>	<ul style="list-style-type: none"> ✓ Code simple et basé sur des balises ✓ Ne nécessite pas un IDE sophistiqué pour le programmer ✓ Grand nombre de fonctionnalités avancées ✓ Le langage est soutenu par <i>Adobe</i> 	<ul style="list-style-type: none"> ✓ Le coût est nécessairement plus élevé que les autres puisqu'il n'est pas gratuit ✓ Très peu d'hébergement disponible ✓ Peu documenté par les sites Web altruistes

III.1.2.2 Choix du langage de programmation

Pour sa souplesse, sa portabilité ainsi que son intégration complète du modèle objet, le groupe de projet en accord avec le groupe de pilotage a choisi JEE 5 comme langage de programmation pour mener ce projet.

III.1.2.3 Framework JEE

En programmation informatique, un framework est un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel (architecture).

Un framework est conçu en vue d'aider les programmeurs dans leur travail. L'organisation du framework vise la productivité maximale du programmeur qui va l'utiliser.

Le tableau 20 présente la liste des framework que le groupe de projet en accord avec le groupe de pilotage a choisi d'utiliser pour ce projet.

Tableau 19: Framework JEE utilisés

Framework	Description
 JSF 2.2 [10]	<ul style="list-style-type: none"> ✓ Framework de vue ✓ Utilisation de composants et de la gestion d'événements ✓ Intégration forte dans certains IDE (Sun Studio Creator, Netbeans ...) pour permettre des développements de type RAD
 Primefaces 4.0 [11]	<ul style="list-style-type: none"> ✓ Framework de vue ✓ Jeu de composants open-source supportant Ajax, Jsf 2, Push ✓ Complète les limites et insuffisances de JSF
 Hibernate 3.2.1 [12]	<ul style="list-style-type: none"> ✓ Framework open source ✓ ORM (Object Relational Mapping) ✓ Apporte une solution aux problèmes d'adaptation entre le paradigme objet et les SGBD en remplaçant les accès à la base de données par des appels à de méthodes objets de haut niveau

III.1.2.4 Autres langages

Dans le but de rendre plus ergonomique l'application, le groupe de projet a décidé d'ajouter à JEE d'autres langages de programmation web :

- **JavaScript** (avec utilisation du framework **JQUERY**) : langage de programmation de scripts principalement utilisé dans les pages web interactives;
- **CSS** langage informatique qui sert à décrire la présentation des documents HTML, JSP, XHTML,...

III.1.3 L'environnement de développement intégré

Un IDE est un environnement de développement intégré réunissant tout ou presque tous les outils nécessaires à la création d'applications, aussi complexe qu'elles soient.

Le groupe de projet en accord avec le groupe de pilotage a utilisé **NetBeans 7.4 [13]** comme IDE.

III.1.4 Présentation de quelques écrans de GCAP

Les écrans suivants sont des captures d'écran de CGAP.



Figure 18: Page de connexion



Figure 19 : Page d'accueil du DRH



Figure 20 : Ecran de demande de congé annuel



Figure 21 : Ecran de notification envoyée à un agent



Figure 22 : Ecran de traitement de demande de congé annuel par un directeur de direction

III.1.5 Politique de sécurité

III.1.5.1 Notion de sécurité

La sécurité informatique est l'ensemble des moyens techniques, organisationnels, juridiques et humains nécessaires et mis en place pour conserver, rétablir et garantir la sécurité des systèmes d'informations. Elle est intrinsèquement liée à la sécurité de l'information et des systèmes d'information [14].

Les objectifs visés par l'adoption d'une politique de sécurité sont entre autres la garantie de :

- l'intégrité des données ;
- la confidentialité des données ;
- la disponibilité des données ;
- la traçabilité des données ...

Une politique de sécurité est l'ensemble des orientations suivies par une entité en matière de sécurité. À ce titre, elle se doit d'être élaborée au niveau de la direction de l'organisation concernée, car elle concerne tous les utilisateurs du système.

III.1.5.2 Sécurité applicative

La sécurité applicative fait partie des domaines de la sécurité informatique. Elle définit les règles sécuritaires à respecter dans le développement logiciel.

Le codage de l'application a été faite en respectant les règles de la sécurité applicative :

- les triggers ont été créés pour le contrôle d'accès à la base de données ;
- un utilisateur n'ayant pas droit à une ressource ne le verra pas sur son écran ;
- les mots de passe seront cryptés.

III.1.5.3 La politique de sécurité contre les virus

Un virus est un programme malveillant pouvant altérer gravement le fonctionnement d'un système informatique.

Pour une protection contre les virus un logiciel antivirus en occurrence **Kaspersky Lab 2006** sera installé sur tous les postes clients.

III.1.5.4 Politique de sécurité pour la gestion des mots de passe et des accès non autorisés

Concernant les mots de passe utilisateurs, leur durée de validité est de trois mois, autrement dit après trois mois chaque utilisateurs doit renouveler son mot de passe. Aussi, tous les mots de passe

sont cryptés par l'algorithme **md5**.

La protection contre les accès non autorisés vise la confidentialité des données. Ainsi, un administrateur à savoir le DRH se chargera de la gestion des comptes utilisateurs.

III.1.5.5 Politique de sécurité contre les catastrophes

Les différentes catastrophes susceptibles d'altérer le fonctionnement du système sont la foudre, les incendies, les inondations...

La **CARFO** dispose déjà d'un système de détection de fumée, d'extincteurs, et de parafoudre.

III.1.5.6 Mise en place des sauvegardes et des restaurations [15]

En informatique, une sauvegarde ou « **backup** » en anglais, est l'opération qui consiste à dupliquer et à mettre en sécurité les données contenues dans un système informatique.

Dans le choix de la stratégie de sauvegarde, seuls les trois types de techniques ont été pris en compte.

- La première est la technique de **sauvegarde complète** (en anglais **full backup**). Comme son nom l'indique, cette méthode consiste à sauvegarder l'intégralité des données. Cependant, pour de gros volumes de données, la sauvegarde complète peut poser des problèmes de lenteur (si les données sont modifiées en cours de sauvegarde).
- La deuxième technique est la **sauvegarde incrémentale** et incrémentielle (en anglais **incremental backup**) consiste à copier tous les éléments modifiés depuis la sauvegarde précédente. Ce type de sauvegarde est plus performant qu'une sauvegarde totale car elle permet de se focaliser uniquement sur les fichiers modifiés avec un espace de stockage plus faible, mais nécessite en contrepartie de posséder les sauvegardes précédentes pour reconstituer la sauvegarde complète.
- La troisième technique est la **sauvegarde différentielle** (en anglais **differential backup**) se focalise uniquement sur les fichiers modifiés depuis la dernière sauvegarde complète, ce qui la rend plus lente et plus coûteuse en espace de stockage qu'une sauvegarde incrémentale mais également plus fiable car seule la sauvegarde complète est nécessaire pour reconstituer les données sauvegardées.

Comme stratégie de stockage, le groupe de projet en accord avec le groupe de pilotage a choisi de faire des sauvegardes incrémentales du lundi au jeudi et une sauvegarde complète les vendredis soir à **18H00**.

III.2 TRANSITION

III.2.1 Tests et déploiement

Avant la mise en place du système futur, celui-ci sera soumis à cinq (5) types de tests afin de valider sa qualité. Ces tests seront effectués par des informaticiens. Il s'agit :

- d'un test **fonctionnel** : il consiste à vérifier que les résultats produits par le système sont ceux attendus; ce test prendra en compte les scénarii nominaux, alternatifs et exceptionnels des différentes fonctionnalités du système ;
- d'un test **structurel** : il vise à contrôler le mode et les normes métiers de réalisation des différentes fonctionnalités.
- d'un test de **vulnérabilité** : il vise à vérifier la sécurité du logiciel ;
- d'un test de **performance** : ce test vise à vérifier que les performances annoncées dans la spécification sont bien atteintes.

A la suite de ces tests, viendra le déploiement du système. Pour permettre la continuité des services des départements couverts par le système, le groupe de projet préconise un fonctionnement en parallèle du nouveau système avec le système actuel pendant une période de trois (3) mois.

Cette période de couplage des deux systèmes sera mise à profit pour l'identification d'éventuelles discordances ou disfonctionnement du système mais aussi permettre au groupe de projet d'apporter d'éventuelles corrections à ces bugs dans le but d'offrir aux utilisateurs un produit qui répond au mieux à leur attente.

III.2.2 Formation des utilisateurs

Une chose est de mettre en place un système informatique, une autre est son utilisation effective et efficace par les utilisateurs. Aussi, un système n'est sûr que si les utilisateurs prennent conscience de certaines règles sécuritaires et donc du respect de la politique de sécurité adoptée.

En effet, les utilisateurs doivent être formés à bien utiliser les services du système en évitant les opérations qui pourraient le déstabiliser ou présenter des failles de sécurité et en privilégiant les opérations qui participent au mieux à son maintien et à sa sécurité.

Pour ce faire, nous préconisons former les différents utilisateurs sur l'utilisation du système, sur la politique de sécurité et leur apporter une assistance rapprochée durant les tout premiers mois de déploiement.

En outre, un manuel d'utilisation de l'application sera mis à leur disposition.

CONCLUSION

Dans ce chapitre, il a été question de montrer les aspects liés au développement effectif de l'application et à son déploiement.

En plus, les fonctionnalités de l'application ont été présentées ainsi que la politique de sécurité qui va régir son fonctionnement.

CONCLUSION GENERALE

A la **CARFO**, il nous été soumis le problème de gestion manuelle des demandes et traitements de congés et des autorisations d'absences.

Pour résoudre ce problème nous avons proposé trois solutions dont une a été retenue.

Pour mener le projet nous avons utilisé le processus unifié comme méthode d'analyse et UML comme langage de modélisation.

Le choix du langage pour le développement s'est porté sur JEE et celui du serveur de base de données sur Mysql 5.7.17. Apache Tomcat 7.0.42 a été le serveur d'application utilisé.

De l'étude qui précède, il apparaît clairement que la mise en place de l'application de gestion de congés et d'absences professionnels va être un grand apport pour la **CARFO** dans la gestion des ressources humaines car elle va résoudre les problèmes liés à la gestion manuelle des demandes et traitement de congés et des autorisations d'absences.

L'analyse et la conception du système ont été entièrement réalisées, l'application étant toujours en développement. Nous estimons le taux d'avancement du projet à 60%.

L'expérience a été très enrichissante. Un tel résultat est à mettre au crédit de l'Université Polytechnique de Bobo-Dioulasso, plus précisément de l'Ecole Supérieure d'Informatique pour la qualité de la formation reçue.

Comme perspective nous comptons achever le développement de **GCAP** et procéder à son déploiement effectif.

REFERENCES BIBLIOGRAPHIQUES ET WEBOGRAPHIQUES

Index	Documentation	Date de consultation
[0]	Mission d'étude pour la réalisation du schéma directeur informatique de la Caisse Autonome	-
[1]	http://thieum22.free.fr/Quest_RUP.htm	20/10/2013
[2]	http://thieum22.free.fr/Quest_RUP.htm	20/10/2013
[3]	http://www.banque-pdf.com/fr_methode-cocomo.html	21/02/2014
[4]	http://www.banque-pdf.com/fr_methode-cocomo.html	17/01/2014
[4]	http://www.phpindex.com/index.php/2007/10/26/3885-comparaison-des-sgbd-oracle-mysql-et-postgresql-partie-1	30/10/2013
[5]	http://www.materiel.net/serveur-application/	
[6]	http://www.idealofr.com/cat/3099F1483965/routeurs.html	03/12/2013
[7]	http://www.gladir.com/LEXIQUE/ARTICLES/comparelangagewebp hpvsaspnetvsjsp.htm	15/01/2014
[8]	Les cahiers du programmeur JAVA EE 5 d'Antonio Goncalves aux Editions Eyrolles	-
[9]	http://www.developpez.com/actu/65954/Classement-TIOBE-2013-C-reste-le-langage-le-plus-populaire-Transact-SQL-creer-la-surprise-en-s-invitant-dans-le-top-10/	17/11/2013
[10]	http://www.objis.com/formation-java/tutoriel-JSF-premier-application-jsf.html	20/01/2014
[11]	http://www.primefaces.org	25/01/2014
[12]	http://www.hibernate.org	30/01/2014
[13]	http://docs.oracle.com/cd/E40938_01/doc.74/e40142/toc.htm	10/02/2014
[14]	http://fr.wikipedia.org/wiki/S%C3%A9curit%C3%A9_des_syst%C3%A8mes_d%27information	10/03/2014
	http://fr.wikipedia.org/wiki/Politique_de_s%C3%A9curit%C3%A9_du_syst%C3%A8me_d%27information	10/03/2014

Index	Documentation	Date de consultation
[15]	http://www.materiel-informatique-entreprise.fr/sauvegarde-de-donnees/#0	15/03/2014
	http://fr.wikipedia.org/wiki/Sauvegarde_%28informatique%29	15/03/2014
[16]	http://fr.wikipedia.org/wiki/MySQL	02/04/2014
[17]	http://fr.wikipedia.org/wiki/NetBeans	03/01/2014
[18]	Rapport Seydou OUEDRAOGO et Souleymane BELEM (année 2012-2013)	-
<i>Autre</i>	http://www.s2s-onduleur.fr/	17/02/2014
	http://programmation.developpez.com/actu/58624/Quel-est-votre-environnement-de-developpement-integre-EDI-prefere-en-2013-Pourquoi-lui-et-pas-un-autre/	20/01/2014

ANNEXES

Annexe 1 : Présentation de MySQL [16]

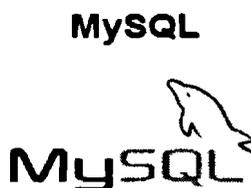
Présentation générale

MySQL est un système de gestion de base de données relationnelle (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde autant par le grand public (application web principalement) que par des professionnels en concurrence avec Oracle, Informix et Microsoft SQL Server.

Son nom vient du prénom de la fille du cocréateur Michael Widenius, My. SQL fait allusion au *Structured Query Language*, le langage de requête utilisé.

Principales caractéristiques

MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multi-thread et multi-utilisateur. C'est un logiciel libre développé sous double licence selon qu'il est distribué avec un produit libre ou avec un produit propriétaire.



Une session MySQL dans une Konsole de KDE

Une session MySQL dans une Konsole de KDE.

Apparu en	1995
Développeur	MySQL AB / Oracle
Dernière version	5.6.18 [11 avril 2014] [---]
Version en développement	5.7.3 [3 décembre 2013] [---]
Système d'exploitation	Windows, Linux, Mac OS X, et d'autres (voir plus bas)
Licence	Licence publique générale GNU ou licence propriétaire (voir Caractéristiques)
Site web	mysql.com/ [E]
modifier	Consultez la documentation du modèle

Dans ce dernier cas, la licence est payante, sinon c'est la licence publique générale GNU (GPL) qui s'applique. Un logiciel qui intègre du code MySQL ou intègre MySQL lors de son installation devra donc être libre ou acquérir une licence payante. Cependant, si la base de données est séparée du logiciel propriétaire qui ne fait qu'utiliser des API tierces (par exemple en C# ou PHP), alors il n'y a pas besoin d'acquérir une licence payante.

Langages informatiques supportés

MySQL supporte deux langages informatique, le langage de requête *SQL* et le *SQL/PSM* (Persistent Stored Modules), une extension procédurale standardisée au SQL incluse dans la norme *SQL:2003*, SQL/PSM, introduit dans la version 5 de MySQL, permet de combiner des requêtes SQL et des instructions procédurales (boucles, conditions...), dans le but de créer des traitements complexes destinés à être stockés sur le serveur de base de données (objets serveur), par exemple des procédures stockées ou des déclencheurs (rudimentaires).

Systèmes d'exploitation supportés

MySQL fonctionne sur de nombreux systèmes d'exploitation différents, incluant AIX, IBM i-5, *BSDi*, *FreeBSD*, *HP-UX*, *Linux*, *Mac OS X*, *NetWare*, *NetBSD*, *OpenBSD*, *OS/2 Warp*, *SGI IRIX*, *Solaris*, *SCO OpenServer*, *SCO UnixWare*, *Tru64 Unix*, *Windows*.

Les bases de données sont accessibles en utilisant les langages de programmation *C*, *C++*, *VB*, *VB .NET*, *C#*, *Delphi/Kylix*, *Eiffel*, *Java*, *Perl*, *PHP*, *Python*, *Windev*, *Ruby* et *Tcl* ; une API spécifique est disponible pour chacun d'entre eux. Une interface ODBC appelée *MyODBC* est aussi disponible. En Java, MySQL peut être utilisé de façon transparente avec le standard *JDO*.

Annexe 2 : Présentation de NetBeans [17]

Présentation générale

NetBeans est un environnement de développement intégré (EDI), placé en open source par Sun en juin 2000 sous licence CDDL et GPLv2 (Common Development and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages,

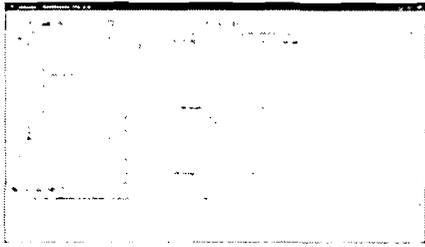
Langages supportés

- ✓ Java (Java SE, Java ME, Java FX, Java EE), Javadoc ;
- ✓ Groovy et Grails ;
- ✓ PHP (dont les environnements Zend et Symfony);
- ✓ JavaScript;
- ✓ C, C++, Fortran. NetBeans ne requiert pas l'utilisation d'un compilateur particulier. À noter le support des bibliothèques Qt. Les plates formes supportées sont Microsoft Windows, Linux, Mac OS, Solaris 10 et OpenSolaris;
- ✓ Python;
- ✓ HTML, XHTML, RHTML, XML, CSS;
- ✓ JSP, JSF;
- ✓ Ruby (via un greffon depuis la version 7.0 de NetBeans);

NetBeans



Logo



Développeur	Oracle
Première version	1996, sous le nom de Xelfi
Dernière version	7.4 [P] (15 octobre 2013) [...]
Environnements	Plateforme Java
Langues	Multilingue (français : partiellement)
Type	IDE pour Java, PHP, C/C++, Fortran, JavaScript, Python, Ruby
Licence	CDDL/ GPL
Site web	netbeans.org [P]

modifier 

Systèmes d'exploitation supportés

NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Development Kit JDK est requis pour les développements en Java.

Points forts

- ✓ multilingue : NetBeans est disponible en 23 langues ;
- ✓ documentation intégrée.

Annexe 3 : Tableau comparatif de quelques implémentations d'UP [18]

Méthode	Description	Points forts	Points faibles
RUP : <i>Rational Unified Process</i>	<ul style="list-style-type: none"> ✓ Il est à la fois une méthodologie et un outil prêt à l'emploi ✓ Il est la cible des projets de plus de 10 personnes 	<ul style="list-style-type: none"> ✓ Il est itératif, ✓ Il spécifie le dialogue entre les différents intervenants du projet ✓ Il propose aussi des modèles de documents, et des canevas pour des projets 	<ul style="list-style-type: none"> ✓ Il est coûteux à personnaliser, ✓ Il est très axé processus, au détriment du Développement
2TUP : <i>Two Track Unified Process</i>	<ul style="list-style-type: none"> ✓ Il s'articule autour de l'architecture et propose un cycle de Développement en Y, ✓ Il est la cible des projets de toutes tailles 	<ul style="list-style-type: none"> ✓ Il est itératif, ✓ Il donne une large place à la technologie et à la gestion du risque, ✓ Il définit les profils des intervenants, les livrables, les plannings, les prototypes 	<ul style="list-style-type: none"> ✓ Il se révèle plutôt superficiel sur les phases situées en amont et en aval du développement
XP : <i>Extreme Programming</i>	<ul style="list-style-type: none"> ✓ C'est un ensemble de " Bests Practices" de développement ✓ Il est la cible des projets de moins de 10 personnes. 	<ul style="list-style-type: none"> ✓ Il est itératif ✓ Il est simple à mettre en œuvre ✓ Il donne une large place aux aspects techniques, ✓ Il est assez innovant: programmation en duo, kick-off matinal debout 	<ul style="list-style-type: none"> ✓ Il ne couvre pas les phases en amont et en aval au développement ✓ Il élude la phase d'analyse, si bien qu'on peut dépenser son énergie à faire et défaire

Annexe 4 : Organigramme de la CARFO

