

Ministère des Enseignements Secondaire et Supérieur
(M.E.S.S)

Secrétariat Général

Université Polytechnique 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 : INFORMATISATION DU PROGRAMME NATIONAL
D'APPROVISIONNEMENT EN EAU POTABLE ET D'ASSAINISSEMENT
(PN-AEPA) DES HAUTS-BASSINS**

Période du 06 août au 07 novembre 2013

Auteurs : HIEN Oho Isabelle
SANON Ardjata

Maître de stage

Mr Hèba Marius SAMA
Directeur Régional de l'Eau, des
Aménagements Hydrauliques et de
l'Assainissement des Hauts-bassins

Superviseur

Dr Borlli Michel Jonas SOME
Enseignant chercheur à l'Ecole
Supérieure d'Informatique

Année académique 2012-2013

DEDICACE

A ma **Mère**,
Qui n'a ménagé aucun effort
pour mon avancée scolaire.

HIEN Oho Isabelle

A mes chers **Parents**

SANON Ardjata

REMERCIEMENTS

Nous ne saurions parler de notre travail sans faire mention des personnes vers qui se portent nos reconnaissances sincères. Au nombre de ces personnes, nous citons :

- Mr le Directeur de l'école supérieure d'informatique(ESI) ;
- Dr Borlli Michel Jonas SOME, enseignant chercheur à l'ESI qui nous a supervisés ;
- Tous les enseignants et personnels de l'ESI ;
- Mr Héba Marius SAMA, Directeur Régional de l'Eau, des Aménagements Hydrauliques et de l'Assainissement des Hauts-Bassins (DREAHA-HB), notre maître de stage qui a bien voulu nous donné un thème pour compléter notre formation ;
- Tout le personnel de la DREAHA –HB pour l'accueil et la bonne ambiante de travail ;
- Mr Tifori SOMA, Responsable du centre de documentation de la Direction Générale de l'Agence de l'Eau et du Mouhoun ;
- Mr Yembi Nestor Fiacre COMPAORE, Assistant Technique du Projet PADI-BF101 ;
- Mr Daouda SANON, Sociologue à l'Association pour le Développement des Adductions d'Eau Potable(ADAE) ;
- Mr Karim Aimé ZOUNGRANA , statisticien à l'ADAE
- Mr Hamadou SANOU, Juriste à l'ADAE.

A tous ceux dont les noms n'ont pas été cités et qui ont contribué de près ou de loin au bon déroulement de ce stage, nous disons MERCI !

TABLE DES MATIERES

DEDICACE	ii
REMERCIEMENTS.....	iii
TABLE DES MATIERES	iv
SIGLES-ABREVIATIONS	vi
LISTE DES FIGURES.....	viii
LISTE DES TABLEAUX	x
PREAMBULE.....	xi
INTRODUCTION GENERALE	1
Chapitre I : Contexte générale du stage	2
I-Présentation de la structure d'accueil	4
II-Présentation du thème.....	7
II.1-Problématique.....	7
II.2-Résultats attendus	8
III-Acteurs de la conduite du projet.....	9
III.1-Le groupe de pilotage.....	9
III.2-Le groupe de projet.....	9
III.3-Le groupe des utilisateurs	9
IV-APPROCHE DE RESOLUTION.....	10
IV.1-Unified Modeling Language (UML)	10
IV.2-Processus Unifié.....	10
V-Planning	14
V.1- Planning Prévisionnel.....	14
V.2-Planning réel	15
V.3-Explication des écarts	16
Chapitre II : Etude préliminaire.....	18
I-Présentation du projet	20
II-Choix techniques.....	20
II.1-Langage de programmation	20
II.2-Système de gestion de base de données (SGBD)	21
II.3-Design pattern.....	22
II.4-L'Environnement de développement intégré(EDI)	24
III- Recueils des besoins fonctionnels.....	25
III.1- Interviews et consultation de documents.....	25
III.2- Les fonctionnalités du futur système	29

III.3- Identification des acteurs du système	29
III 4- Identification des messages	30
III.5- Modélisation du contexte	31
Chapitre III : Capture des besoins fonctionnels	33
I-Détermination des cas d'utilisation	35
II- Description de certains cas d'utilisation	36
Chapitre IV : Analyse et Conception	54
I- Diagramme de cas d'utilisations	56
II- Diagramme de classes	57
II.1- Les règles de gestion	57
II.2- Le diagramme de classe	58
III- Etude des scénarii architecturaux	63
III.1-Description du premier scénario.....	63
III.2- Description du deuxième scénario.....	68
III.3- Scénario retenu.....	72
Chapitre V : La Réalisation	74
I- Présentation de prototypes	76
I.1- L'accueil.....	76
I.2- La connexion	77
I.3- L'interface d'enregistrement d'un « acteur projet ».....	78
II- Politique de sécurité	80
III- Procédure Transitoire	80
CONCLUSION GENERALE	82
Bibliographie	83
Webographie.....	83
ANNEXES	84
Annexe 1 : Présentation de GanttProject	84
Annexe 2 : Présentation de COCOMO (CONstructive COSt Model)	86

SIGLES-ABREVIATIONS

Sigle	Définition
2TUP	Two Tracks Unified Process
« Acteur Projet »	Acteur d'un Projet au sein de la DREAHA-HBS
ADAE	Association pour le Développement des Adductions d'Eau Potable
AEP	Approvisionnement en Eau Potable
AEPA	Approvisionnement en Eau Potable et à l'Assainissement
AEPS	Adduction d'Eau Potable Simplifiée
CICI	Cycle d'Ingénieur de Conception en Informatique
CITI	Cycle des Ingénieurs en Travaux Informatiques
COCOMO	<i>Constructive COst Model</i>
CUI	Cadre Unifié d'Intervention
DREAHA-HBS DREAHA	Direction Régionale de l'Eau, des Aménagements Hydrauliques et de l'Assainissement des Hauts-Bassins
ESI	Ecole Supérieure d'Informatique
IDE	Integrated Development Environment
ONG	Organisation Non-Gouvernementale
PCD	Plan Communal de Développement
PEA	Poste d'Eau Autonome
PN-AEPA	Programme National d'Approvisionnement en Eau Potable et d'Assainissement
PME	Point d'Eau Moderne
PTF	Partenaires Techniques et Financiers
RUP	Rational Unified Process
SGBD	Système de Gestion de Base de Données
sysAEPA	Nom donné à au nouveau système à développer
TCM	Toilettes à Chasse d'Eau Mécanique
UML	Unified Modeling Language

UP	Unified Process
UPB	Université Polytechnique de Bobo-Dioulasso
XP	eXtreme Programming

LISTE DES FIGURES

Figure 1: Organigramme de la DREAHA-HBS	6
Figure 2: Le processus de développement en Y	13
Figure 3: Liste des tâches effectuées du planning prévisionnel	14
Figure 4: Diagramme de GANTT prévisionnel	15
Figure 5: Liste des tâches effectuées du planning réel	15
Figure 6: Diagramme de GANTT réel.....	16
Figure 7: Architecture MVC	23
Figure 8: Diagramme d'activité du CU1 « S'authentifier »	38
Figure 9: Diagramme de séquence du CU1 « S'authentifier »	38
Figure 10: Diagramme d'activité du CU2 « Enregistrer un « acteur projet »».....	40
Figure 11: Diagramme de séquence du CU2 « Enregistrer un « acteur projet »».....	40
Figure 12: Diagramme d'activité du CU8 « Créer un programme ».....	42
Figure 13: Diagramme de séquence du CU8 « Créer un programme ».....	43
Figure 14: Diagramme d'activité du CU9 « Enregistrer une activité d'un programme »	45
Figure 15: Diagramme de séquence du CU9 « Enregistrer une activité d'un programme»	46
Figure 16: Diagramme d'activité du CU10 « Effectuer une recherche autre que sur le financement ».....	48
Figure 17: Diagramme de séquence du CU10 « Effectuer une recherche autre que sur le financement ».....	49
Figure 18: Diagramme d'activité du CU11 « Effectuer des recherches sur un financement » .	51
Figure 19: Diagramme de séquence du CU11 « Effectuer des recherches sur un financement »	52
Figure 20: Diagramme des cas d'utilisations	56
Figure 21: Diagramme de classe	59
Figure 22: Les symboles d'architecture réseau du premier scénario	63
Figure 23: Architecture réseau du premier scénario	64
Figure 24: Les symboles de l'architecture du deuxième scénario	68
Figure 25: Architecture réseau du deuxième scénario	69
Figure 26: Interface d'accueil	76
Figure 27: Interface de connexion	77
Figure 28: Interface d'enregistrement d'un « acteur projet ».....	78

Figure 29: Interface de recherche d'un « acteur projet » 79

LISTE DES TABLEAUX

Tableau 1: Acteurs et fonctions.....	31
Tableau 2: Modèle de description des cas d'utilisation	36
Tableau 3: CU1 « S'authentifier ».....	37
Tableau 4: CU2 « Enregistrer un « acteur projet » ».....	39
Tableau 5: CU8 « Créer un programme ».....	41
Tableau 6: CU9 « : Enregistrer une activité d'un programme».....	44
Tableau 7: CU10 « Effectuer une recherche autre que sur le financement ».....	47
Tableau 8 : CU11 « Effectuer des recherches sur un financement ».....	50
Tableau 9: Description de la classe « ActeurProjet »	60
Tableau 10: Description de la classe « Projet »	61
Tableau 11: Description de la classe « EAU ».....	61
Tableau 12 : Description de la classe « Assainissement ».....	62
Tableau 13: Equipement nécessaire au premier scénario.....	65
Tableau 14: Evaluation du coût de l'équipement du premier scénario à fournir.....	65
Tableau 15: Coût de développement de l'application du premier scénario.....	66
Tableau 16: Coût de formation des utilisateurs pour le premier scénario	66
Tableau 17: Coût total du premier scénario.....	67
Tableau 18: Equipement nécessaire au deuxième scénario.....	69
Tableau 19: Evaluation du coût de l'équipement du deuxième scénario.....	70
Tableau 20 : Coût de développement de l'application du deuxième scénario.....	71
Tableau 21: Coût de formation des utilisateurs pour le deuxième scénario	71
Tableau 22: Coût total du deuxième scénario.....	71
Tableau 23 : Tableau comparatif des deux scénarii.....	72

PREAMBULE

L'Université Polytechnique de Bobo-Dioulasso (UPB) qui est située à une quinzaine de kilomètres de Bobo-Dioulasso regroupe à son sein plusieurs établissements dont l'Ecole Supérieure d'Informatique (ESI) où nous avons été formées.

La formation pédagogique proposée par l'ESI comporte deux cycles qui sont :

- Le Cycle des Ingénieurs des Travaux Informatiques (CITI) avec les options Analyse et Programmation (AP) et Réseau et Maintenance Informatique (REMI) ;
- Le Cycle des Ingénieurs de Conception en Informatique (CICI).

La formation suivie au cours du Cycle des Ingénieurs de travaux informatiques option Analyse et Programmation a pour objectif de former des cadres moyens opérationnels, évolutifs et aptes à participer efficacement à la conception, à la réalisation et à la maintenance d'applications informatiques ainsi que de tout ce que ces notions impliquent. Cette formation est sanctionnée par un diplôme d'ingénieur de travaux informatiques.

Dans le but d'obtenir de ce diplôme, tous les étudiants doivent valider les cinq semestres et faire un stage de trois mois en entreprise. C'est dans le souci de mieux professionnaliser l'enseignement dispensé aux étudiants que le stage est organisé. Ce stage qui a ainsi un but complémentaire à la formation reçue permet l'application pratique des connaissances acquises et l'insertion à la vie professionnelle.

C'est dans ce cadre que nous avons été admises à la DREAHA des Hauts-Bassins.

INTRODUCTION GENERALE

Il est primordial pour toute structure de nos jours de garder des traces, preuves de toute action effectuée. Cela revient à gérer énormément d'informations qui n'intéressent pas les mêmes personnes et qui ne sont pas nécessaires au même moment. Sans outils informatiques, cette masse d'informations est difficilement gérable et cause une perte de temps considérable en consultation.

Dans ce contexte, chaque administration tente d'organiser l'information pour servir au mieux ses intérêts dans le cadre de ses fonctions. Or, informatiser sa structure revient à identifier et rendre accessible les informations nécessaires aux bonnes personnes au bon moment.

C'est dans cette logique que la DREAHA-HBS nous a soumis la conception d'une application pour un meilleur suivi des « acteurs projet » et des actions que ceux-ci exécutent dans la cadre du Programme National d'Approvisionnement en Eau Potable et Assainissement (PN-AEPA) des Hauts-Bassins.

Le présent document s'articule autour de cinq chapitres. Dans le premier chapitre, nous donnerons un aperçu de la structure d'accueil et expliciterons le thème que celle-ci nous a accordé. Pour la bonne appréhension de la conduite du projet, le deuxième chapitre montrera les décisions techniques prises et les différentes fonctionnalités du futur système à considérer. Cela débouche aux cas d'utilisation de ce système qui feront l'objet du troisième chapitre. Après avoir connu ce qu'implique chacun de ces cas d'utilisation, nous établirons des diagrammes présentant le système globalement pour ensuite faire un choix d'architecture réseau de celui-ci dans le quatrième chapitre. Le cinquième chapitre qui correspond à la partie réalisation présentera des prototypes des interfaces réalisés et les décisions prises pour le déploiement de l'application.

CHAPITRE I : CONTEXTE GENERALE DU STAGE

Introduction

Le pilotage de tout projet dans toute structure passe nécessairement par une bonne connaissance de la structure en question, du thème à étudier ainsi que du choix d'une méthode à suivre. Pour chacune des étapes de la méthode qui sera choisie, une durée de réalisation sera définie. Il sera donc question dans ce chapitre de donner une vue générale du contexte de ce projet.

Partant de la présentation de la structure d'accueil, nous dégagerons la problématique suscitée par ce projet pour ainsi déterminer les résultats visés afin d'adopter une meilleure approche de résolution.

I-PRESENTATION DE LA STRUCTURE D'ACCUEIL

La Direction Régionale de l'Eau, des Aménagements Hydrauliques et de l'Assainissement des Hauts-Bassins a pour mission de mettre en œuvre les politiques et stratégies du Ministère de l'Eau, des Aménagements Hydrauliques et de l'Assainissement dans la région. Ces politiques visent notamment à collecter des données pour assurer le suivi-évaluation des activités de tous les intervenants dans les domaines de l'eau, des aménagements hydrauliques et de l'assainissement des régions. Ces données collectées depuis la base (Communes -> Provinces -> Région), permettent de suivre les activités et leurs impacts au moyen d'indicateurs (taux d'accès aux services d'eau et d'assainissement, taux de pannes des ouvrages réalisés, etc...).

Dans cette optique, la DREAHA-HBS doit :

- constituer et gérer une banque de donnée de projets ;
- collecter, traiter et capitaliser les données concernant tous les projets passés, en cours ou à réaliser dans la région ;
- élaborer les programmes et les rapports périodiques de mise en œuvre des activités;
- proposer les améliorations et ajustements nécessaires ;
- apporter un appui-conseil aux collectivités territoriales et aux structures déconcentrées de l'Etat en matière de planification, de réalisation et de maintenance des infrastructures et équipements ;
- participer à la mise à jour de la banque de données en eau, aménagements hydrauliques et assainissement ;
- veiller à la bonne tenue des documents, des archives et la mise à jour du fichier de la documentation ;
- élaborer un état des lieux et constituer une base de données des ouvrages d'AEP et des ouvrages d'assainissement ;
- superviser les études et travaux des ouvrages d'AEP et d'assainissement ;
- contribuer au suivi quantitatif et qualitatif des ressources en eau et à la planification de leur exploitation;
- suivre l'exploitation et la maintenance des ouvrages de mobilisation et de régulation des ressources en eau ;
- assurer l'assistance technique à la maîtrise d'ouvrage communale en matière d'Approvisionnement en Eau Potable et d'Assainissement (AEPA) ;
- mener des actions de sensibilisation sur l'hygiène et l'assainissement ;

- suivre, contrôler et appuyer la gestion des ouvrages d'AEP et d'assainissement ;
- promouvoir en relation avec les services partenaires, l'organisation des acteurs du domaine des aménagements hydrauliques ;
- assurer le suivi et la supervision des études et des travaux de réalisation, maintenance et de réhabilitation des aménagements et équipements hydrauliques et hydro agricoles;
- recenser, synthétiser et analyser les besoins en aménagements en collaboration avec les directions provinciales;
- Réceptionner les fournitures d'équipements hydrauliques ;
- suivre la fonctionnalité des ouvrages hydrauliques et hydro agricoles.

Vu toutes ces multiples fonctions à assurer par la DREAHA, il va de soi qu'il y ait une organisation adéquate. D'où l'organigramme est présenté en *Figure 1*.

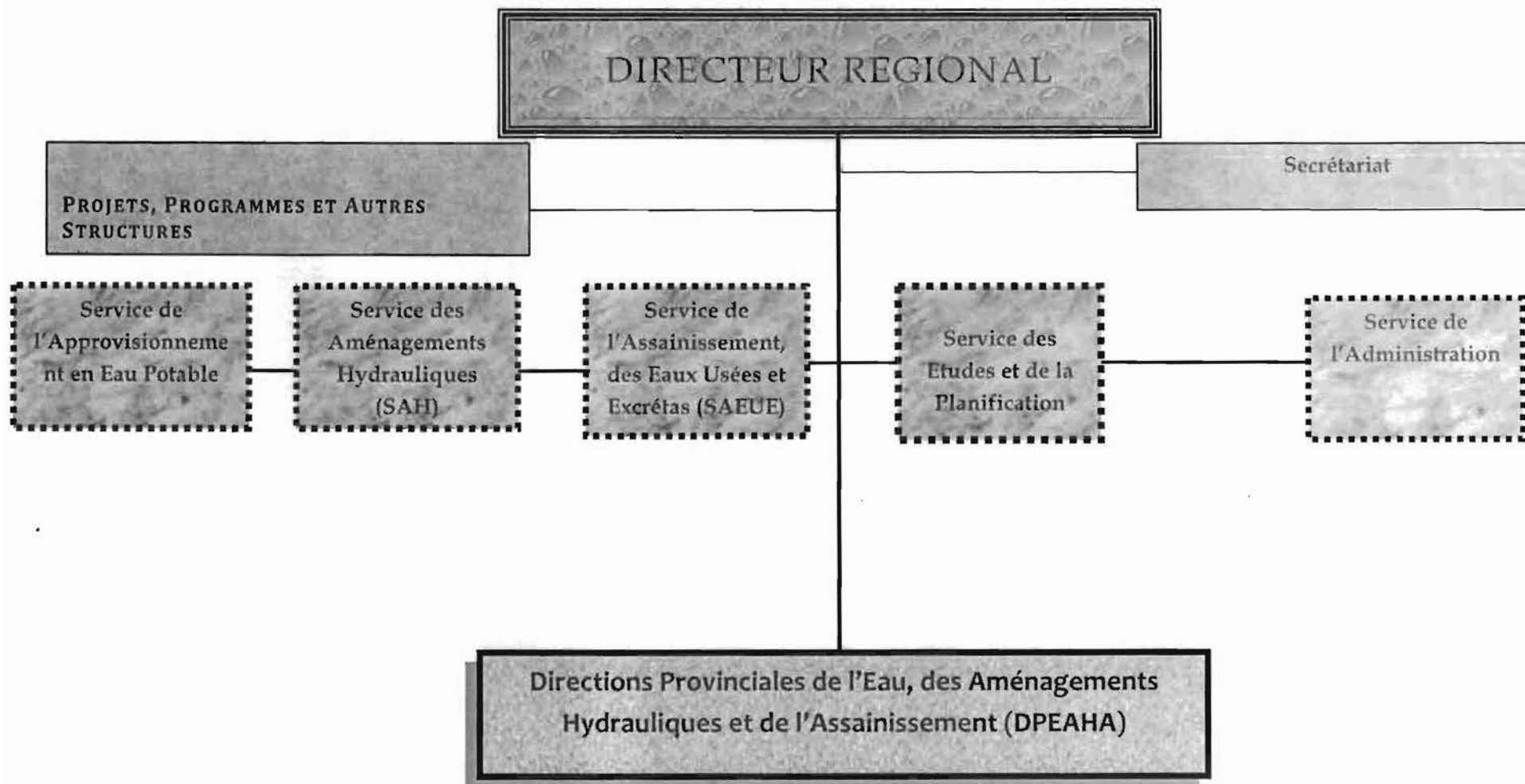


FIGURE 1: ORGANIGRAMME DE LA DREAHA-HBS

II-PRESENTATION DU THEME

Le thème de notre étude s'intitule comme suit : « Informatisation du Programme National d'Approvisionnement en Eau Potable et Assainissement des Hauts-Bassins ».

II.1-PROBLEMATIQUE

A la suite d'études menées en 2005 au Burkina-Faso, il a été identifié une grande proportion des populations, surtout des milieux ruraux qui n'ont pas un accès adéquat à l'eau potable et assainissement. Pour pallier à ce problème, le PN-AEPA a été adopté en 2006. Le PN-AEPA vise ainsi à réduire d'ici 2015 cette proportion de la population qui n'a pas un accès adéquat à l'eau potable. Ce programme s'inscrit dans le cadre des Objectifs du Millénaire pour le Développement (OMD) et de la Stratégie de Croissance Accélérée pour le Développement Durable (SCADD).

Les acteurs du PN-AEPA interviennent au niveau de trois composantes qui sont:

- La réalisation des ouvrages d'Approvisionnement en Eau Potable;
- La réalisation des ouvrages d'Assainissement, des Eaux Usées et Excréta;
- Le Cadre Unifié d'Intervention (CUI) qui regroupe les actions autres que les réalisations physiques et qui sont l'ensemble des procédures de mise en œuvre comme la sensibilisation et la formation des bénéficiaires des ouvrages d'eau et d'assainissement.

La gestion des acteurs et des projets menés par ceux-ci se fait jusque-là manuellement. Cette gestion manuelle des acteurs et projets au sein du PN-AEPA pose de nombreux problèmes à savoir :

- le problème de lenteur dans le traitement de l'information ;
- la difficulté de trouver l'information sur un acteur et son projet en temps réel ;
- le problème de sauvegarde des données ;
- la redondance de l'information ;
- la difficulté de suivi des projets des « acteurs projet ».

Afin de pallier à ces erreurs et optimiser le suivi des projets, la DREAHA-HBS nous a confié la mission de développer une application qui lui permette de suivre chaque « acteur projet » et l'évolution du projet qu'il mène.

II.2-RESULTATS ATTENDUS

Face à cette problématique, le système à développer devra être capable de :

- Permettre le traitement rapide des informations ;
- Favoriser l'accès à l'information en temps réel ;
- Assurer la pérennité de l'information ;
- Eviter la redondance de l'information ;
- Faciliter le suivi des projets des « acteurs projet ».

III-ACTEURS DE LA CONDUITE DU PROJET

Un acteur d'un projet est une personne qui intervient pour le bon déroulement du projet en question. Nous pouvons citer trois (03) types d'acteurs sont : Le groupe de pilotage, le groupe de projet et le groupe d'utilisateurs.

III.1-LE GROUPE DE PILOTAGE

C'est un groupe qui guide la prise de décisions pour l'atteinte des objectifs visés. Il définit les orientations générales, les délais et les moyens nécessaires à la réalisation du projet. Aussi valide-t-il les différents choix technique. Il est constitué de **Monsieur Hèba Marius SAMA**, Directeur de la DREAHAH-HBS et le **Docteur Borlli Michel Jonas SOME**, Enseignant à l'ESI.

III.2-LE GROUPE DE PROJET

Le groupe de projet est le groupe qui exécute le projet, c'est-à-dire l'analyse, la conception, la réalisation, et le déploiement de l'application. Il est également chargé de la rédaction d'un rapport sur le déroulement du projet auprès du groupe qui pilote. Il se compose de **HIEN Oho Isabelle** et de **SANON Ardjata**.

III.3-LE GROUPE DES UTILISATEURS

C'est un groupe de personnes qui vont potentiellement utiliser l'application future. C'est en fait auprès des personnes qui forment ce groupe que celui de projet recueilleront les informations nécessaires pour le déroulement du projet. Ce groupe est formé de tous les membres du personnel de la DREAHAH-HBS.

IV-APPROCHE DE RESOLUTION

IV.1-UNIFIED MODELING LANGUAGE (UML)

UML (traduit par Langage de Modélisation Unifié) est une notation permettant de modéliser un problème de façon standard. Il unifie à la fois les notations et les concepts orientés objet. Il ne s'agit pas d'une simple notation mais les concepts transmis par un diagramme ont une sémantique précise et sont porteurs de sens au même titre que les mots d'un langage. Ce langage est né de la fusion de plusieurs méthodes existant auparavant, et est devenu désormais la référence en termes de modélisation objet.

UML est un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier, concevoir des solutions et communiquer des points de vue (Pitman, 2006). En plus de cette déclaration, UML présente d'autres avantages qui ont guidé notre choix. UML est valable pour n'importe quel langage de programmation. En fait la modélisation objet consiste à créer une représentation informatique des éléments du monde réel auxquels on s'intéresse, sans se préoccuper de l'implémentation, c'est-à-dire indépendamment du langage de programmation. Aussi offre-t-il une manière élégante de présenter le système selon différentes vues complémentaires grâce aux diagrammes.

UML n'étant qu'un langage, ne préjuge pas une démarche, un processus à suivre. Ainsi dans l'objectif du bon déroulement du projet nous suivrons un processus itératif fortement axé sur UML qui est PU (Processus Unifié).

IV.2-PROCESSUS UNIFIE

Le Processus Unifié (UP en anglais pour Unified Process) est une méthode de développement logiciel construite sur UML ; elle est itérative et incrémentale, centré sur l'architecture, conduite par les cas d'utilisation et piloté par les risques. Son approche itérative est fondée sur la croissance et l'affinement successifs d'un système par le biais d'itérations multiples. Ses activités de développement fondamentales sont la capture des besoins, la modélisation métier, l'analyse et la conception, l'implémentation, le test et le déploiement qui peuvent se dérouler en plusieurs phases. Par conséquent, le PU est une trame des meilleures pratiques de développement. Il a l'avantage de permettre de personnaliser ses critères de développement afin d'avoir un processus plus adapté aux besoins exprimés.

Il existe plusieurs méthodes de développement qui répondent aux caractéristiques d'UP. Parmi ces méthodes nous avons 2TUP (Two Tracks Unified Process), XP (eXtreme Programming) et RUP (Rational Unified Process).

Notre choix s'est porté sur 2TUP (*Figure 2*).

2TUP répond aux contraintes de changements continus imposés aux systèmes d'information des structures. En ce sens, il renforce le contrôle des capacités d'évolution et de correction de tels systèmes. Le 2TUP propose un cycle de développement qui dissocie les aspects techniques des aspects fonctionnels et propose une étude parallèle des deux branches : fonctionnelle (étude de l'application) et technique (étude de l'implémentation).

Le processus 2TUP s'articule autour de trois phases :

- ✓ Une branche fonctionnelle
- ✓ Une branche technique
- ✓ Et une branche de conception-réalisation

Branche fonctionnelle : Les principales étapes de la branche fonctionnelle se présentent comme suit :

- l'étape capture des besoins fonctionnels produit le modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie, au plus tôt le risque de produire un système inadapté aux utilisateurs. Cette phase a pour objectif de définir :
 - La frontière fonctionnelle entre le système considéré comme une boîte noire et son environnement, c'est le niveau contexte.
 - Les activités attendues des différents utilisateurs par rapport au système toujours envisagé comme une boîte noire, c'est le niveau cas d'utilisation.
- L'étape d'analyse consiste à étudier précisément les spécifications fonctionnelles de manière à obtenir une idée de ce que réalisera le système en termes de métier.

Branche technique : Les principales étapes de la branche technique se présentent comme suit :

- L'étape capture des besoins techniques recense toutes les contraintes sur les choix de dimensionnements et de conception du système. Les outils et le matériel sélectionnés ainsi que la prise en compte des contraintes d'intégration avec l'existant (pré requis d'architecture technique). Cette étape permet de définir le modèle d'analyse technique. Le rôle de ce dernier est d'établir les couches logicielles et y spécifier les activités techniques attendues.
- L'étape conception générique définit ensuite les composants nécessaires à la construction de l'architecture technique. Cette conception est complètement indépendante des aspects fonctionnels. Elle permet de générer le modèle de conception technique ou design pattern (aspect qui sera développé ultérieurement) qui définit les Framework. Ces derniers, délivrant les services techniques, assurent la réponse aux exigences Opérationnelles du système.

Branche Conception-Réalisation : Les principales étapes de cette branche sont :

- L'étape conception préliminaire est une étape délicate, car elle intègre le modèle d'analyse fonctionnelle dans l'architecture technique de manière à tracer la cartographie des composants du système à développer. Cette étape permet de produire le modèle de conception système. Ce dernier organise le système en composants, délivrant les services techniques et fonctionnels. Ce modèle regroupe les informations des branches techniques et fonctionnelles.
- L'étape conception détaillée permet d'étudier comment réaliser chaque composant. Cette étape produit le modèle de conception des composants. Ce modèle fournit l'image prête à fabriquer du système complet. C'est dans l'étape de codage que s'effectue la production des composants et les tests des unités de code au fur et à mesure de leur réalisation.
- L'étape de recette consiste à valider les fonctionnalités du système développé.

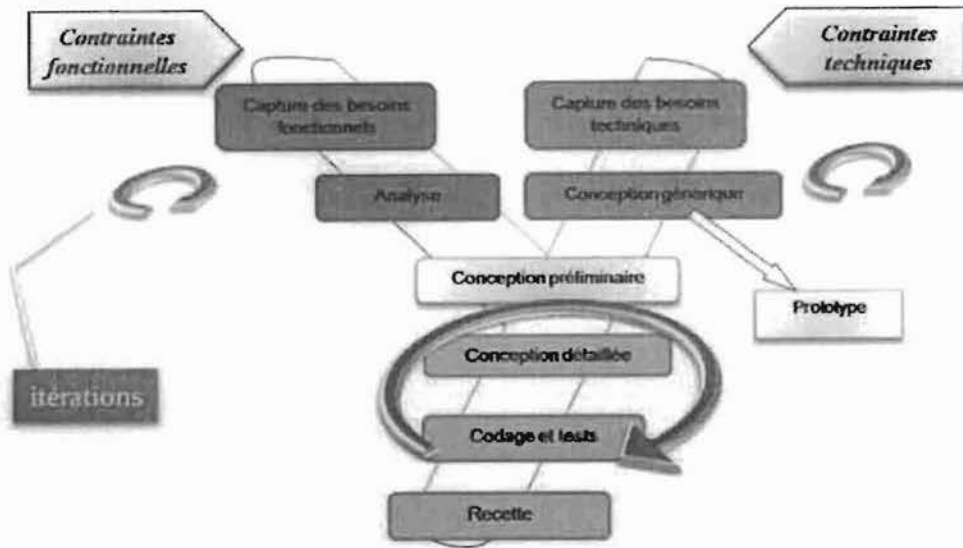


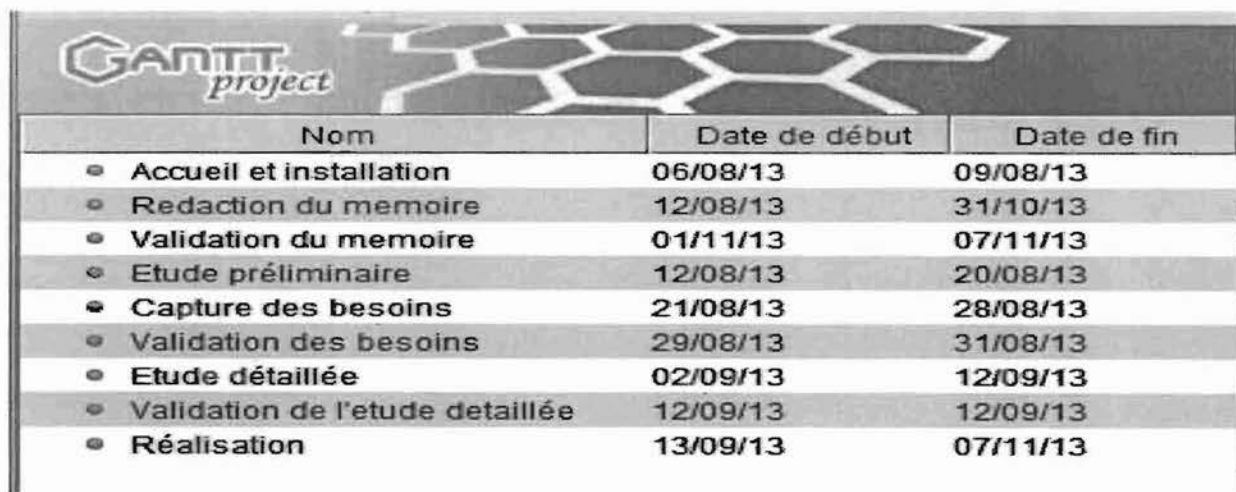
FIGURE 2: LE PROCESSUS DE DEVELOPPEMENT EN Y

V-PLANNING

Le bon déroulement de tout projet nécessite l'établissement et surtout le respect d'un planning bien défini. Ce planning doit tenir compte des contraintes dont celles de l'organisation interne de la structure d'accueil, du temps accordé au groupe de projet et à la méthode d'analyse choisie. La bonne avancée de notre projet exige donc un planning prévisionnel.

V.1- PLANNING PREVISIONNEL

Sachant les différentes tâches à effectuer, nous prévoyons pour chacune d'elle une date de début et une de fin comme suit :



Nom	Date de début	Date de fin
• Accueil et installation	06/08/13	09/08/13
• Redaction du memoire	12/08/13	31/10/13
• Validation du memoire	01/11/13	07/11/13
• Etude préliminaire	12/08/13	20/08/13
• Capture des besoins	21/08/13	28/08/13
• Validation des besoins	29/08/13	31/08/13
• Etude détaillée	02/09/13	12/09/13
• Validation de l'étude détaillée	12/09/13	12/09/13
• Réalisation	13/09/13	07/11/13

FIGURE 3: LISTE DES TACHES EFFECTUEES DU PLANNING PREVISIONNEL

Pour mieux visualiser le temps d'exécution de chacune des tâches définies, nous proposons la représentation graphique de la Figure 4 :

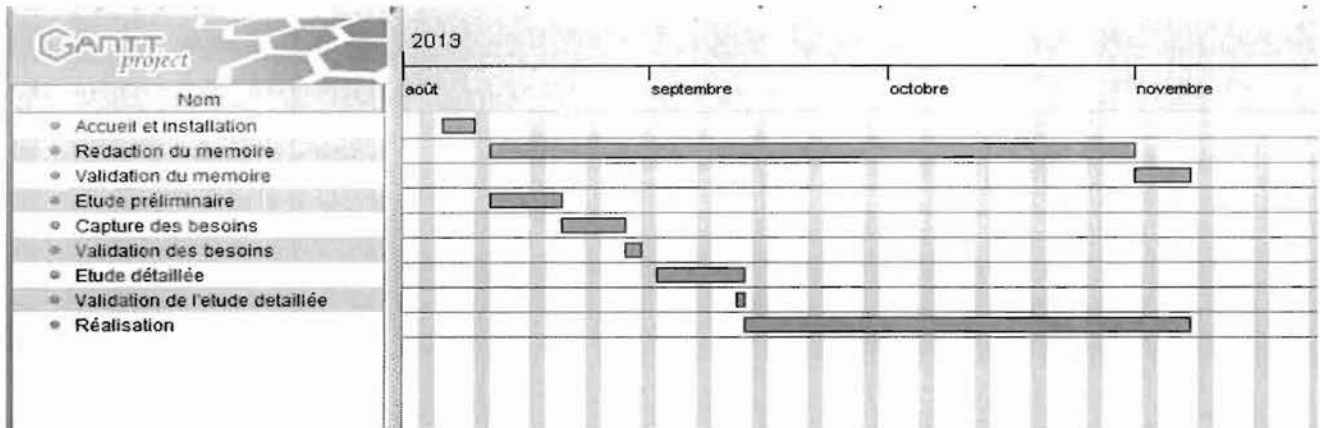


FIGURE 4: DIAGRAMME DE GANTT PREVISIONNEL

V.2-PLANNING REEL

A la fin des tâches, nous nous rendons compte que le temps prévu pour chaque tâche n'a pas été respecté. D'où la nouvelle figure des dates suivantes :

Nom	Date de début	Date de fin
• Accueil et installation	06/08/13	10/08/13
• Rédaction du mémoire	12/08/13	09/01/14
• Validation du mémoire	09/01/14	09/01/14
• Etude préliminaire	12/08/13	20/09/13
• Capture des besoins	23/09/13	11/10/13
• Validation des besoins	14/10/13	14/10/13
• Etude détaillée	15/10/13	25/10/13
• Validation de l'étude détaillée	25/10/13	25/10/13
• Réalisation	28/10/13	09/01/14

FIGURE 5: LISTE DES TACHES EFFECTUEES DU PLANNING REEL

La représentation graphique de ce planning suivi en réalité donne cette figure (Figure 6) :

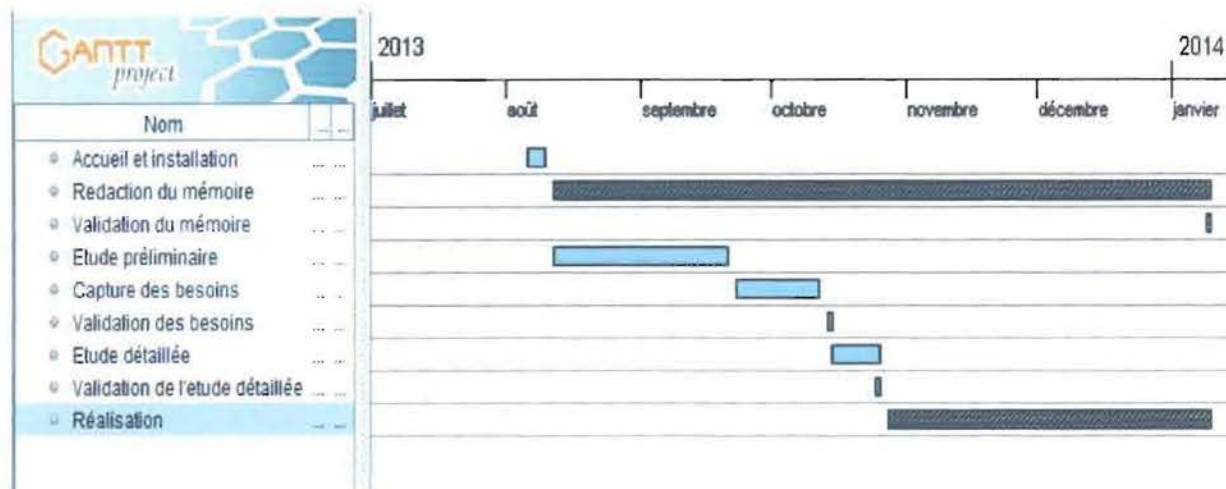


FIGURE 6: DIAGRAMME DE GANTT REEL

V.3-EXPLICATION DES ECARTS

Nous remarquons énormément d'écarts entre le planning prévu et celui réel. Ces écarts s'expliquent entre autre par la difficulté d'appréhension du thème et à cerner la limite du contexte. Cela a entrainé une révision importante dans la validation des besoins fonctionnels. A cela s'ajoute les difficultés rencontrées dans la conception.

Conclusion

Il ressort à la fin de ce chapitre que le groupe du projet doit informatiser le PN-AEPA de la DREAHA-HBS dans un temps donné. 2TUP a été choisi comme méthodologie d'analyse.

Par conséquent le groupe aura à cerner les différentes fonctionnalités du système avec toutes les décisions nécessaires à prendre dans le chapitre suivant.

CHAPITRE II : ETUDE PRELIMINAIRE

Introduction

L'étude préliminaire est une phase importante pour mettre en route notre processus de développement. Elle favorise en effet une bonne compréhension du thème et une vision des limites du futur système. Pour ce faire, nous présenterons clairement le projet d'abord, nous définirons ensuite nos choix techniques. Et enfin nous déterminerons les fonctionnalités du système par les moyens nécessaires.

I-PRESENTATION DU PROJET

De nos jours, une des tendances qui concerne tous les secteurs de développement est l'informatisation. Ainsi les entreprises et les entités publiques aspirent à optimiser et à rendre fiable la gestion de leur structure interne. C'est dans ce contexte que la DREAHA des Hauts-Bassins a jugé bon d'informatiser le PN-AEPA.

La solution que nous proposerons à la DREAHA facilitera le suivi du PN-AEPA à travers la conception et la réalisation d'un logiciel. Pour cela, il nous faut utiliser des outils informatiques.

II-CHOIX TECHNIQUES

II.1-LANGAGE DE PROGRAMMATION

Il existe plusieurs langages de programmation dont Java, C++, C, C#. Il sera prétentieux de notre part de faire une comparaison de tous les langages de programmation existant. Mais nous affronterons deux d'entre eux afin d'adopter celui qui nous convient le plus : Java et C++.

II.1.1-Java

Java est un langage orienté objet dont la syntaxe est similaire à celle du C et qui possède une grande portabilité. En plus d'être gratuit et simple, il autorise le multitâches (multithreading; un thread: processus qui s'exécutent simultanément à l'intérieur d'un unique programme) et contient une très riche bibliothèque de classes (packages) qui permettent de créer des interfaces graphiques, d'utiliser des données multimédia, de communiquer à travers les réseaux, d'accéder à des BD, etc. Tout ceux-ci sont des véritables atouts pour le futur système.

II.1.2-C++

Le langage C++ est une extension du langage C. Le plus grand avantage de ces deux langages est sans doute l'extraordinaire quantité de bibliothèques disponibles. Il est notamment très utilisé pour les applications commerciales et les jeux vidéo grâce à sa capacité d'adaptation à tout type de programmation. Cela le rend captivant pour beaucoup de programmeur.

I.1.3-Justification du choix de langage de programmation

Notre choix s'est donc porté sur **Java**. Ses caractéristiques, en effet, ainsi que la richesse de son écosystème et de sa communauté lui ont permis d'être très largement utilisé pour le développement d'applications de type très disparates. Java fait notamment le bonheur de beaucoup de développeurs d'applications d'entreprises.

II.2-SYSTEME DE GESTION DE BASE DE DONNEES (SGBD)

Le SGBD, comme son nom l'indique est un logiciel permettant de gérer, interagir avec une base de données. Il existe plusieurs SGBD. Il sera prétentieux de faire un tableau comparatif de tous les SGBD existants. Ainsi notre étude portera sur deux d'entre eux, PostgreSQL et Oracle, qui ont des critères assez plaisant pour notre application future.

II.2.1- PostgreSQL

PostgreSQL est un très bon SGBD, fiable et relativement performant, tout en restant simple d'utilisation. Il supporte la majorité du standard SQL. Il fournit les fonctions nécessaires pour garantir l'intégrité des données (transactions, trigger, procédures stockées, clefs étrangères, etc.). PostgreSQL est gratuit et est un très bon choix pour gérer des bases de données de taille moyenne.

II.2.2- Oracle

Oracle est un SGBD relationnel figurant parmi les plus performants du marché. C'est un système lourd à administrer et donc réservé aux entités (grandes ou moyennes entreprises, laboratoires, universités, etc.) nécessitant de fortes contraintes de disponibilité. Oracle n'est pas un SGBD relationnel optimisé pour de petites bases de données, ce qui signifie que sur de petits volumes et peu d'utilisateurs les performances seront comparables à d'autres SGBD plus légers.

I.2.3-Justification du choix du SGBD

Il nous sera plus avantageux d'utiliser un SGBD gratuit, simple d'utilisation et d'administration. Ainsi nous choisissons PostgreSQL qui a les caractéristiques essentielles dont on pourrait avoir besoin telles qu'il respecte le standard SQL, possède des extensions comme Java et PL/SQL et intègre la notion d'héritage de tables.

II.3-DESIGN PATTERN

Le code source de l'application se doit d'être compréhensible pour toute personne qui voudra en ajouter des fonctionnalités. Pour cela, il faut une bonne structuration du code. D'où notre choix de modèle de conception MVC.

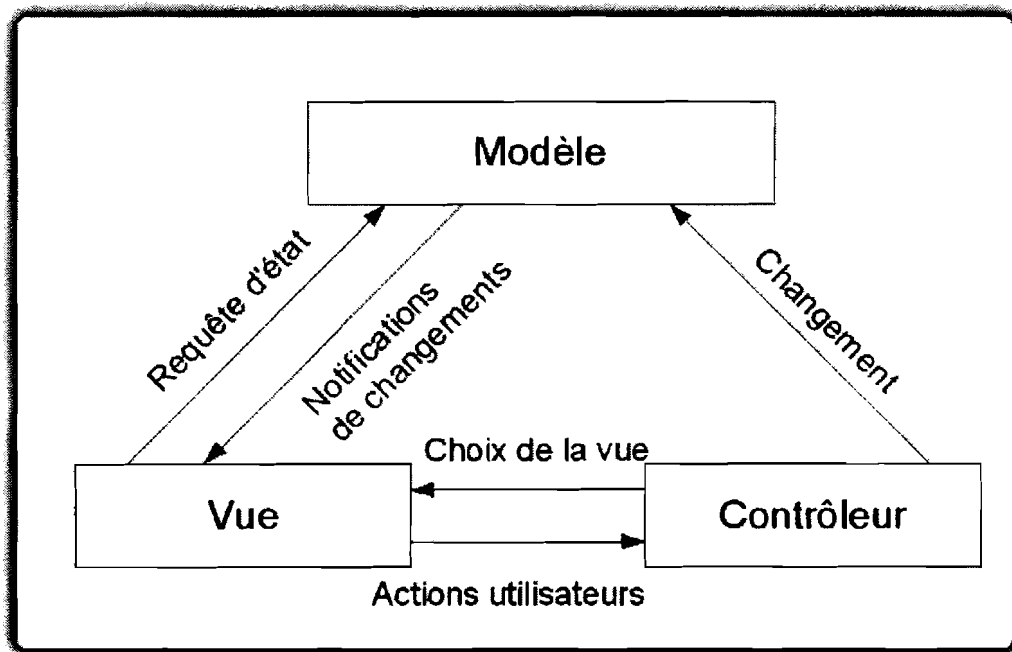
MVC est un design pattern ou patron (modèle de conception) de conception d'interface utilisateur permettant de découpler le modèle (logique métier et accès aux données) des vues (interfaces utilisateur [présentation des données et interface de saisie pour l'utilisateur]). Des modifications de l'un n'auront ainsi, idéalement, aucune conséquence sur l'autre ce qui facilitera grandement la maintenance. Son but principal est de proposer une solution générale aux problèmes d'utilisateurs manipulant des données volumineuses et complexes.

Le **modèle** gère les données et reprend le **logique métier** (il peut être décomposé en plusieurs couches mais cette décomposition n'intervient pas au niveau de MVC). Le **modèle ne prend en compte aucun élément de présentation!**

La **vue** affiche les données, provenant exclusivement du modèle, pour l'utilisateur et/ou reçoit ses actions. **Aucun traitement – autre que la gestion de présentation - n'y est réalisée.**

Le **Contrôleur**, quant à lui, a pour rôle le **traitement les événements** en provenance de l'interface utilisateur et les transmet au modèle pour le faire évoluer ou à la vue pour modifier son aspect visuel (pas de modification des données affichées mais des modifications de présentation (couleur de fond, affichage ou non de la légende d'un graphique, ...)).

Le Design pattern présente des atouts considérables dans l'écriture du code de notre future application avec le langage de programmation Java choisi. Son schéma (*Figure 7*) montre les différents liens entre les trois couches.



Interactions entre les couches

FIGURE 7: ARCHITECTURE MVC

II.4-L'ENVIRONNEMENT DE DEVELOPPEMENT INTEGRE(EDI)

Les EDI sont des programmes qui regroupent un ensemble d'outils pour le développement de logiciels. De façon générale, un EDI contient un éditeur de texte, un compilateur, des outils automatiques de fabrication, et très souvent un débogueur. Il existe des EDI pour de nombreux langages, cependant il est très courant qu'un IDE soit conçu pour un seul langage de programmation. Nous examinerons Eclipse et NetBeans qui sont des EDI très utilisés et appropriés pour le développement Java pour voir lequel sied mieux pour notre développement. En effet ces deux sont openSource.

II.4.1-Eclipse

Eclipse est un IDE performant qui a su trouver sa place comme l'un des environnements de développement Java les plus populaires. C'est une plateforme de développement écrite en Java, fruit du travail d'un consortium de grandes entreprises (IBM, Borland, Rational Rose, HP...). Pour ce qui est de l'ergonomie, Eclipse n'a rien à envier à ses concurrents. La version "packagée" d'Eclipse contient un certain nombre de plugins préconfigurés visant à synchroniser la sortie de plusieurs projets pour en assurer la compatibilité. Renouvelée chaque année fin juin, la nouvelle version d'Eclipse propose toujours davantage de projets en standard grâce à sa communauté très active.

II.4.2-NetBeans

Cet IDE est un environnement fortement car repose sur un noyau robuste, la plateforme NetBeans, que vous pouvez également utiliser pour développer vos propres applications Java, et un système de plugins performant, qui permet vraiment d'avoir un EDI sur mesure. Ainsi ayant la déclinaison concernant le développement Java SE, il ne sera pas livré avec le support Java EE, EJB, Jsp, Tomcat, Glassfish, ... A côté de la version complète, nous avons différentes déclinaisons (6 au total, sans compter la version complète) qui se concentre sur une plateforme ou langage bien précis. Il en résulte un IDE performant et Open Source qui a su trouver sa place comme l'un des environnements de développement Java les plus populaires.

II.4.3-Justification du choix de l'EDI

Dans le cadre de notre projet, nous choisissons Eclipse. En effet, Eclipse est un environnement de développement intégré libre, extensible, universel et polyvalent, permettant potentiellement de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Cela est favorisé par les plugins qu'il contient. Il est probablement plus simple à utiliser vu les avantages qu'il offre aux débutants dans la programmation Java. Il permet ainsi un meilleur apprentissage.

III- RECUEILS DES BESOINS FONCTIONNELS

III.1- INTERVIEWS ET CONSULTATION DE DOCUMENTS

Nous avons été auprès de certains responsables du domaine de l'eau et de l'assainissement dont les noms suivent :

- Mr Héba Marius SAMA, Directeur Régional de la DREAHA ;
- Mr Daouda SANON, Sociologue à l'ADAE ;
- Mr Karim Aimé ZOUNGRANA, Statisticien à l'ADAE ;
- Mr Amadou SANOU, Juriste à l'ADAE.

En plus des informations recueillies auprès d'eux, nous avons consulté quelques documents dont « *La réforme du système de gestion des infrastructures hydrauliques d'alimentation en eau potable en milieu rural et semi-urbain* » et « *Le rapport 2010 du PN-AEPA* ».

À la sortie de cette recherche nous pouvons résumer ces informations comme suit :

Fonctionnement du PN-AEPA

Un projet ou un programme est toute action d'intervention publique ou de coopération technique exécutée pendant une période déterminée dans le domaine de l'AEPA dans le but de produire des biens et services, de réaliser des infrastructures socio-économiques, de renforcer les capacités institutionnelles de l'administration centrale, déconcentrée et locale de la société civile et de renforcer les capacités et compétences des ressources humaines.

Un acteur est une structure chargée de la mise en œuvre d'une étape ou de la totalité du projet ou du programme. Ainsi les acteurs se différencient par leur rôle. On a donc des acteurs financiers, des acteurs réalisateurs, des acteurs exécuteurs, et des acteurs bénéficiaires ou des acteurs qui jouent plusieurs rôles à la fois. Les financiers vont donner les financements nécessaires. Les réalisateurs s'occupent de la mise en place de l'activité en faisant appel à un bureau d'étude et à une entreprise qui exécuteront concrètement sur le terrain. Le bureau d'étude se chargera d'étudier la faisabilité de l'activité et de la formation des futurs usagers tandis que l'entreprise se chargera des travaux physiques. Les collectivités territoriales qui sont les bénéficiaires veilleront au maintien et à la gestion des ouvrages ainsi construits. Un agent communal avec la supervision de la DREAHA fera un rapport semestriel sur l'état des ouvrages existants et sur l'avancement des projets en cours.

En cas de panne majeure ou du vieillissement d'un ouvrage on procède à sa réhabilitation. Les étapes nécessaires à cette réhabilitation sont les mêmes que celles de la construction d'un ouvrage neuf excepté la foration.

Procédure d'intervention

Pour monter un projet il faut que le besoin soit exprimé par une localité. Ainsi tout acteur désirant réaliser un projet doit d'abord se faire connaître par les autorités (régionales, provinciales ou communales) de cette localité pour avoir une idée de ses besoins à travers le PCD (Plan Communal de Développement) que chaque commune aura établi. Ensuite il se rend à la DREAHA ayant en tête une localité précise où il interviendra et précise son type d'intervention. Enfin il donne toutes les informations sur son activité. Ces informations seront en fonction du type d'intervention qu'il désire mener. La connaissance d'une localité et de son PCD (Plan Communal de Développement) est essentielle pour tout acteur désirant monter un projet.

Si l'acteur a les financements nécessaires il a la liberté de les donner à la DREAHA pour qu'elle se charge du reste du projet ou de réaliser tout le travail soi-même. En effet, Dans le cas où l'acteur propose de financer seulement le projet c'est la DREAHA qui trouvera des opérateurs nécessaires pour réaliser les différentes étapes qui sont l'étude du milieu, la construction de l'ouvrage et la formation des bénéficiaires. Pour ce faire elle fera des appels d'offre à l'issue desquels elle retiendra les meilleurs opérateurs. Dans le cas contraire il cherchera lui-même les opérateurs qui accompliront le travail jusqu'à sa finalité. Dans tous les cas la DREAHA supervisera le projet dans son entière réalisation.

Mais si l'acteur désire seulement faire une prestation il se soumettra à un appel d'offre à concurrence ouvert.

Les différents types d'ouvrages

Ouvrages spécifiques à l'eau potable sont :

Forage : ouvrage de petit diamètre destiné à capter des aquifères profonds.

Puits modernes : ouvrage de grand diamètre destiné à capter l'eau de la nappe phréatique.

Adduction d'eau potable simplifiée(AEPS) : comprennent les mini-réseaux d'AEP (approvisionnement en eau potable) et les postes d'eau autonomes.

Mini réseau d'AEP : système d'approvisionnement en eau potable adapté aux petites agglomérations.

Postes d'eau autonome(PEA) : système compact d'équipements hydrauliques.

Réalisation d'un ouvrage d'AEP : tout ouvrage nouvellement réalisé et mise en service.

Réhabilitation d'un ouvrage d'AEP : la remise en état de l'ouvrage dans les conditions proches de son état initial.

Réhabilitation d'un forage : elle porte sur des travaux de pannes sérieuses.

Réhabilitation d'une AEPS : elle porte sur tout ou une partie des organes de l'ouvrage.

Extension d'AEPS : elle porte sur les canalisations, l'ajout de forage ou l'augmentation de la capacité de stockage.

Ouvrages spécifiques à l'assainissement sont :

Latrine VIP (ventilated improved pit) : la latrine améliorée à fosse ventilée. Il existe des latrines VIP à fosse unique, à double fosses, ou à fosses multiples.

Latrine EcoSan (Ecological sanitation) : Type de latrine écologique à fosse ventilée (une ou deux fosses).

Toilette à chasse d'eau manuelle (TCM): Type de latrine comprenant une superstructure, un siphon à faible volume d'eau empêchant les remontées d'odeurs dans la cabine.

Toilettes à chasse d'eau mécanique : elles sont composées d'un réservoir d'eau et d'une cuvette à la turque, reliée à une fosse septique.

Puisard domestique : est un dispositif d'infiltration consistant en un puits destiné à recevoir les eaux usées pour éviter le colmatage de l'ouvrage.

Blocs de latrines : c'est un ensemble constitué d'une ou plusieurs cabines moyennes.

Cabine ou poste : c'est un compartiment délimité par des murs et qui assure l'intimité de l'utilisateur d'une latrine.

Construction d'un ouvrage d'assainissement : elle consiste en la construction complète d'un nouvel ouvrage.

Réhabilitation d'une latrine : elle consiste essentiellement à la réfection de composants d'un ouvrage éventuellement endommagés ou enlever au niveau de service.

III.2- LES FONCTIONNALITES DU FUTUR SYSTEME

Les différentes fonctionnalités qui ont été retenues pour le système à développer sont :

- Gestion des acteurs : enregistrer un « acteur projet » et créer son profil
- Gestion des paramètres de début projet : pour chaque nouveau projet qui commence, enregistrer les informations globales sur celui-ci
- Suivi des projets : enregistrer l'avancée du projet (faire des mises à jour et donner le pourcentage déjà exécuté).
- Suivi des ouvrages dans les communes : l'inventaire de chaque commune
- Edition du rapport annuel qui comportera:
 - o Nombre de projet réussi ;
 - o Nombre de projet en cours ;
 - o Nombre de projet abandonné et leur état.

Différentes communes et leur taux d'accès à l'eau potable et à l'assainissement.

- Recherches : sur un « acteur projet », un projet, une commune et un ouvrage.
- Gestion de financement : montrer la part de l'Etat et des partenaires financiers (ressources extérieures) en donnant le GAP et le montant global.
- Programmation d'un projet : enregistrer les activités pour la réalisation d'un ouvrage, donner le coût de l'ouvrage par rapport à chaque activité et trouver ainsi le coût unitaire à la programmation de l'ouvrage.

III.3- IDENTIFICATION DES ACTEURS DU SYSTEME

Avant d'énumérer les acteurs susceptibles d'interagir avec le système, donnons d'abord une définition de ce qu'est un acteur.

Un acteur en UML représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié.

L'équipe de suivi de projet : l'équipe de projet peut enregistrer les informations sur un projet, un « acteur projet », un financement, faire des mises à jour, établir le programme d'un projet et effectuer des recherches.

Les autres utilisateurs : ils peuvent effectuer des recherches sur les autres aspects différents du financement.

III.4- IDENTIFICATION DES MESSAGES

Le système et les acteurs s'envoient des messages auxquels chacun répond à travers un message retour. Pour une bonne vision de ces messages, nous les classons comme suit :

Messages émis par le système

- L'état d'avancement d'un projet ;
- L'état d'un financement (donner le GAP) ;
- L'état d'un ouvrage ;
- l'édition de rapport ;
- la liste de projet/commune ;
- l'inventaire des ouvrages ;
- Informations sur un « acteur projet » ;
- Informations sur une commune;
- Informations d'une activité sur la réalisation d'un ouvrage ;
- Informations sur un programme d'un projet.

Messages reçus par le système

- Les créations/modifications/suppressions d' « acteur projet » ;
- Les créations/modifications /suppressions de projets ;
- Les créations/modifications d'ouvrages ;
- Ajout d'un financement ;
- Les créations/modifications /suppressions d'un programme ;
- Les créations/modifications /suppressions d'une activité ;

III.5- MODELISATION DU CONTEXTE

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

TABLEAU 1: ACTEURS ET FONCTIONS

Utilisateurs finaux	Actions
Equipe de suivi de projets	<ul style="list-style-type: none"> -S'authentifier -Enregistrer un « acteur projet » -Créer un projet -Enregistrer un financement -Rechercher les informations sur un financement -Enregistrer un ouvrage -Modifier les informations sur un « acteur projet », un ouvrage, un projet -Créer un programme et les activités liées à ce programme
Tous utilisateurs	<ul style="list-style-type: none"> -S'authentifier -Effectuer une recherche autre que sur le financement

Conclusion

En somme, dans ce chapitre nous avons choisi le langage Java, PostgreSQL, Eclipse et MVC pour le bon déroulement du projet. A travers les interviews, nous avons déterminés les différentes fonctionnalités du futur système dans leur contexte.

Les fonctionnalités ainsi connu seront déterminantes dans la perception des besoins du futur système. Nous ferons aussi une description de ces besoins pour mieux les appréhender.

CHAPITRE III : CAPTURE DES BESOINS FONCTIONNELS

Introduction

Le développement du système passe forcément par la détection des différentes utilisations dont il peut être l'objet. La détermination des cas d'utilisation est donc une étape incontournable dans notre processus. Dans ce quatrième chapitre, nous identifierons tous les cas d'utilisation et ferons des descriptions détaillées de certains d'entre eux.

I-DETERMINATION DES CAS D'UTILISATION

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

Nous avons les cas d'utilisations suivants :

- CU1 : S'authentifier ;
- CU2 : Enregistrer un « acteur projet » ;
- CU3 : Enregistrer les informations sur un projet ;
- CU4 : Mettre à jour les informations sur un projet ;
- CU5 : Enregistrer un ouvrage ;
- CU6 : Mettre à jour les informations sur un ouvrage ;
- CU7 : Enregistrer un financement d'un projet ;
- CU8 : Créer un programme ;
- CU9 : Enregistrer l'activité d'un programme ;
- CU10 : Effectuer une recherche autre que sur le financement ;
- CU11 : Effectuer des recherches sur un financement ;
- CU12 : Etablir la liste de projet de chaque commune ;
- CU13 : Editer un rapport semestriel d'un projet ;
- CU14 : Editer un rapport annuel ;
- CU15 : Faire l'inventaire.

II- DESCRIPTION DE CERTAINS CAS D'UTILISATION

Dans le but d'avoir une plus meilleure appréhension des cas d'utilisation que nous allons décrire, nous établirons des diagrammes (diagramme d'activité et celui des séquences) de ceux-ci. Ces différents diagrammes ont été réalisés par l'outil de dessin dénommé *Edraw Max* à sa version 7. Le *Tableau 2* donne le modèle suivi pour la description des cas d'utilisation.

TABLEAU 2: MODELE DE DESCRIPTION DES CAS D'UTILISATION

Identification du cas
Nom : Nom du cas
Objectif : Une description résumée permettant de comprendre l'intention principal du cas d'utilisation
Acteurs : ceux qui vont réaliser le cas d'utilisation
Description du fonctionnement du cas
Pré conditions : Elles définissent ce qui doit être vrai en amont du cas d'utilisation pour que celui-ci puisse être déclenché.
Scénario nominal : Décrit le déroulement du cas quand il n'y a pas d'erreur
Scénarii d'exception : Décrivent les cas d'erreurs qui mettent fin au cas d'utilisation
Post conditions : Elles définissent ce qui doit être vrai lorsque le cas d'utilisation se termine avec succès

Le tableau ci-dessous présente l'enchaînement des opérations à effectuer pour s'authentifier.

TABLEAU 3:CU1 « S'AUTHENTIFIER »

Identification du cas
<p>Nom : S'authentifier</p> <p>Acteurs : l'équipe de suivi de projet</p> <p>Objectif : permettre à tous les utilisateurs de s'identifier afin d'avoir accès aux fonctionnalités de l'application</p>
Description du fonctionnement du cas
<p>Pré conditions : le système est démarré et l'utilisateur est non connecté</p> <p>Scénario nominal :</p> <ol style="list-style-type: none">1-l'utilisateur demande à être connecter2-Le système affiche la fenêtre d'identification3-l'utilisateur saisit son identifiant et son mot de passe4-le système vérifie les informations saisies5- le système valide6-le système affiche l'espace de travail <p>Scénario d'exception :</p> <p>Exception1 : l'identifiant et/ou le mot de passe est incorrect</p> <p>5-le système affiche un message signalant que l'identifiant et le mot de passe est incorrect.</p> <p>Le scénario reprend à son point 2.</p> <p>Post conditions : l'acteur est connecté et il a accès aux fonctionnalités dont il a droit.</p>

L'enchaînement permettant de réaliser le cas d'utilisation « s'authentifier » ayant été connu, nous obtenons les diagrammes suivants :

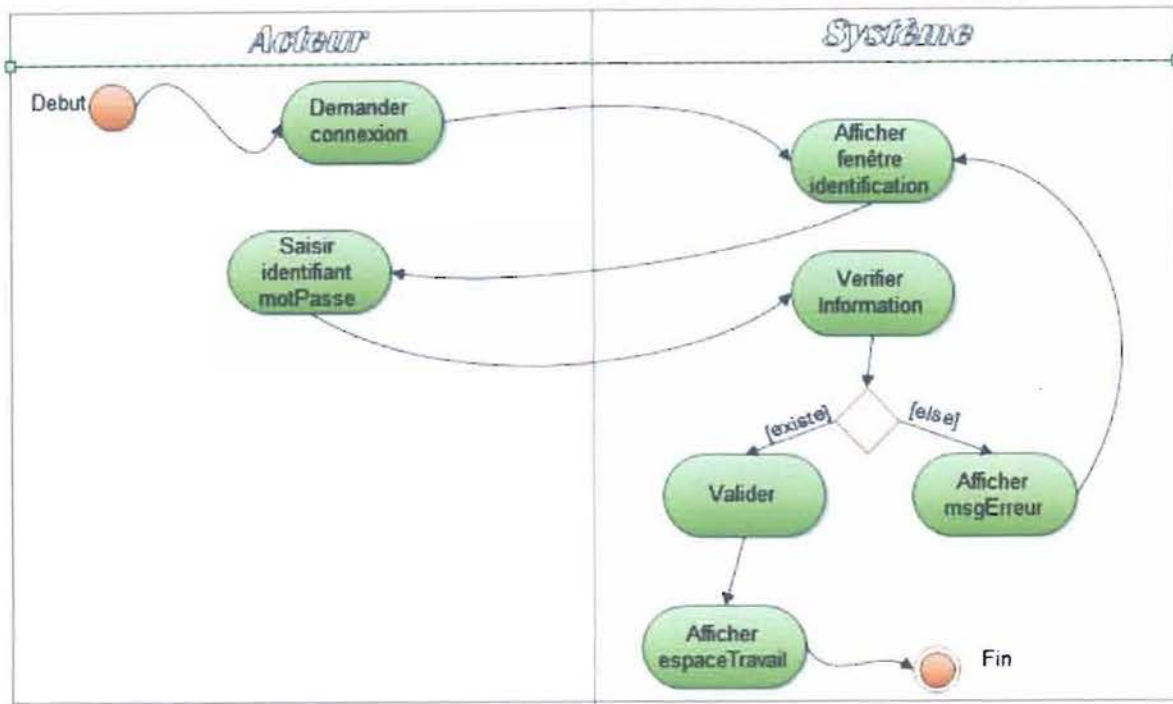


FIGURE 8: DIAGRAMME D'ACTIVITE DU CU1 « S'AUTHENTIFIER »

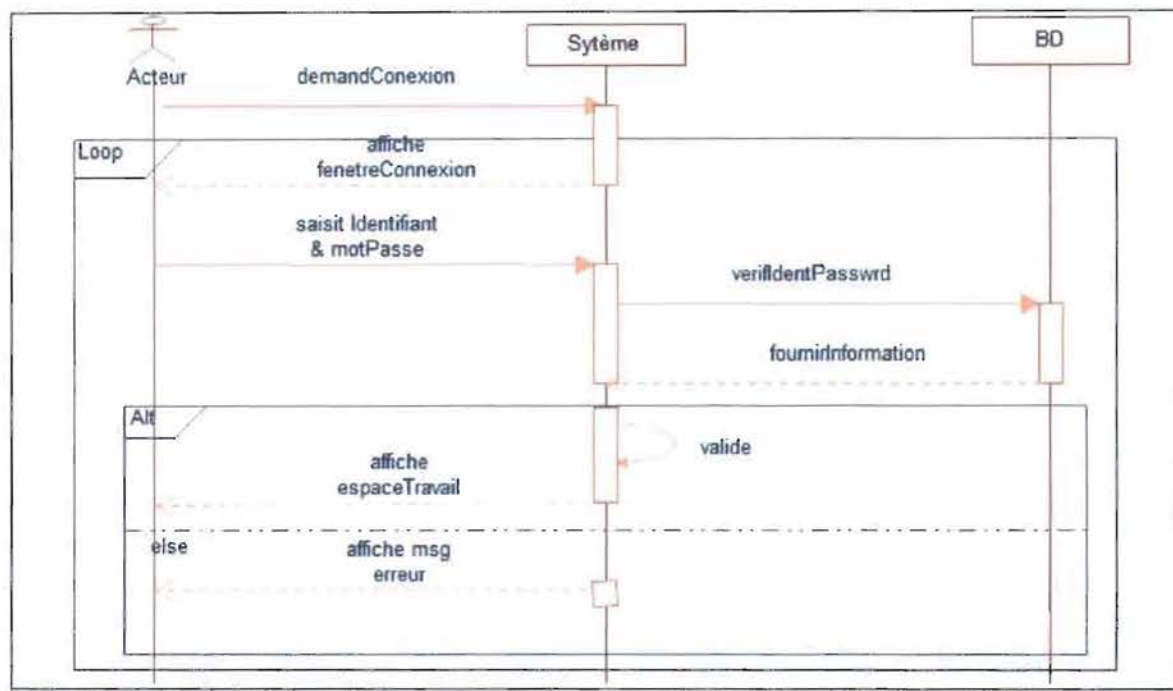


FIGURE 9: DIAGRAMME DE SEQUENCE DU CU1 « S'AUTHENTIFIER »

Le tableau suivant donne l'enchaînement des opérations pour « enregistrer un « acteur projet » ».

TABLEAU 4: CU2 « ENREGISTRER UN « ACTEUR PROJET » »

Identification du cas
<p>Nom : Enregistrer « un acteur projet »</p> <p>Acteurs : Equipe de suivi de projet</p> <p>Objectif : pouvoir enregistrer toutes les informations possibles sur un « acteur projet » donné.</p>
Description du fonctionnement du cas
<p>Pré conditions : l'acteur est connecté</p> <p>Scénario nominal :</p> <p>1-l'acteur saisi les informations générales sur un « acteur projet » dont le type d' « acteur projet ».</p> <p>2-le système affiche une interface en fonction du type D'« acteur projet » spécifié.</p> <p>3-l'acteur saisi les informations propre à l'« acteur projet »</p> <p>4-le système enregistre et affiche un message de Confirmation de l'enregistrement</p> <p>Scénario d'exception : Néants</p> <p>Post conditions : un « acteur projet » est enregistré</p>

Les figures ci-dessus qui se succèdent présenteront l'enchaînement du cas d'utilisation « Enregistrer un « acteur projet » » par des diagrammes.

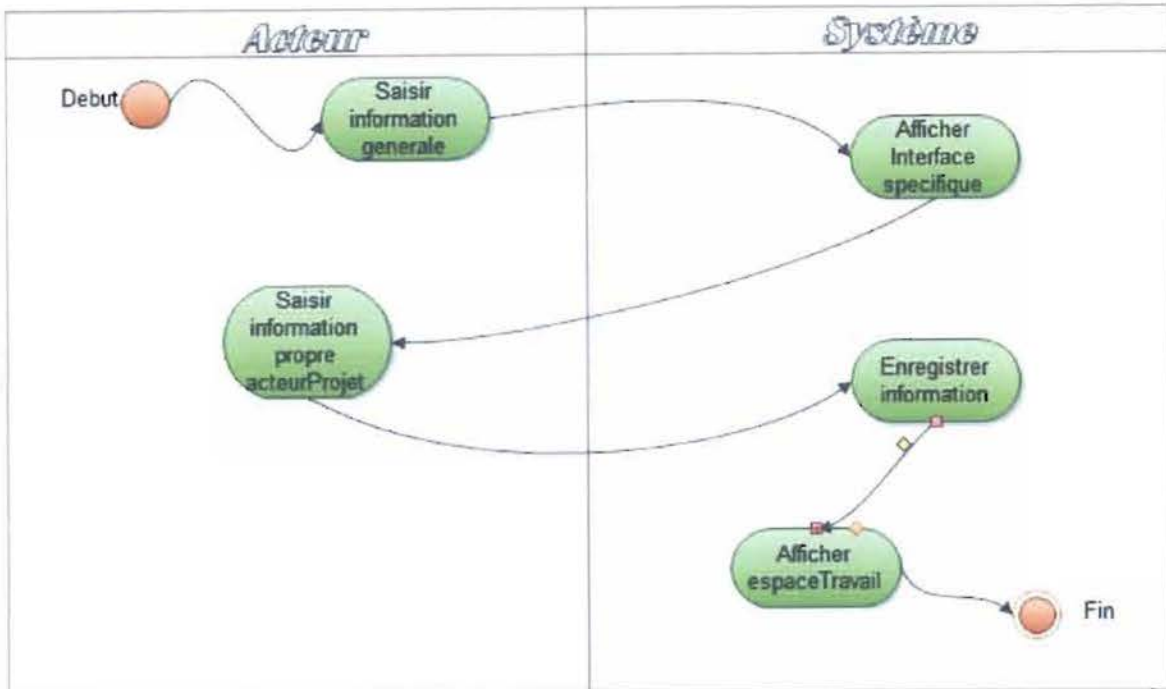


FIGURE 10: DIAGRAMME D'ACTIVITE DU CU2 « ENREGISTRER UN « ACTEUR PROJET » »

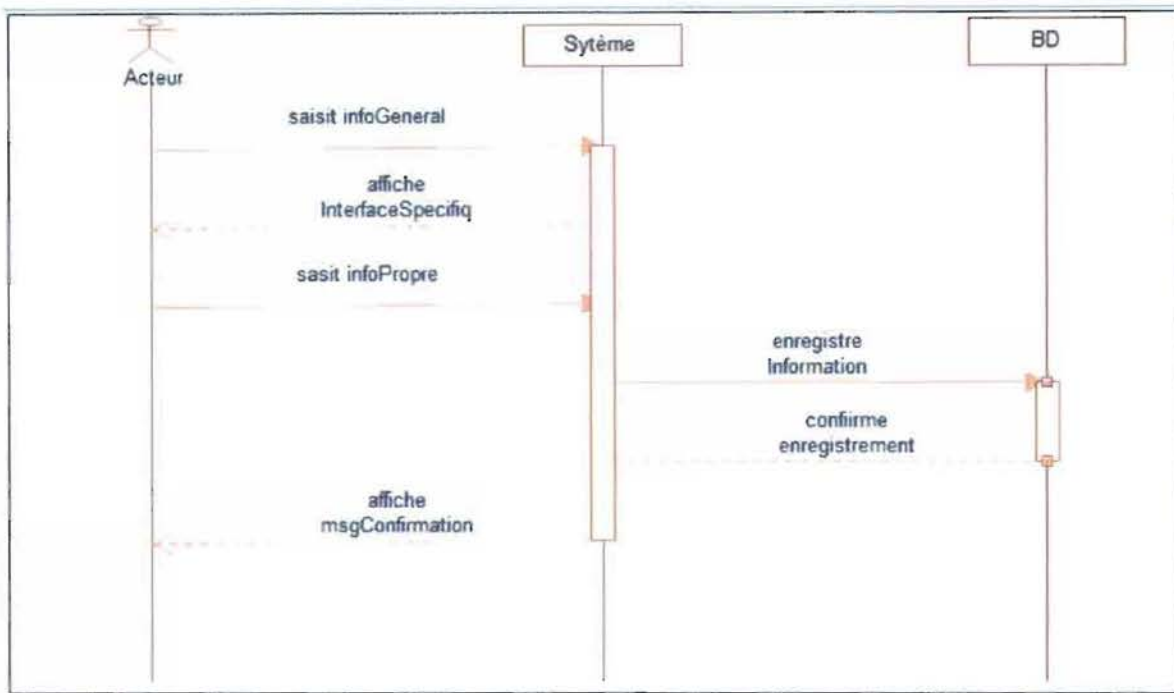


FIGURE 11: DIAGRAMME DE SEQUENCE DU CU2 « ENREGISTRER UN « ACTEUR PROJET » »

L'enchaînement des opérations pour la création d'un programme peut se présenter comme suit :

TABLEAU 5: CU8 « CREER UN PROGRAMME »

Identification du cas
<p>Nom : Créer un programme</p> <p>Objectif : permettre d'établir un programme d'activités pour la réalisation d'un projet</p> <p>Acteurs :Equipe de suivi du projet</p>
Description du fonctionnement du cas
<p>Pré conditions : L'acteur est connecté</p> <p>Scénario nominal :</p> <ol style="list-style-type: none"> 1-l'acteur demande à créer un programme 2-le système lui présente l'interface de création de programme 3-le système demande le choix d'un projet dont on veut programme 4-l'acteur choisit un projet 5-le système vérifie dans la base si le projet choisi est déjà programme 6-le système permet la saisie des autres informations concernant le programme et la création de nouvelles activités du programme 7-L'acteur saisit les informations nécessaires et crée les activités pertinentes du programme 8-Le système enregistre et affiche un message de confirmation <p>Scénarii d'exception :</p> <p>Exception2: le projet choisit est déjà programmé</p> <ol style="list-style-type: none"> 6- Le système affiche un message disant que le projet est déjà programmé et demandant la suppression du programme existant 7- L'acteur fait un choix 8-Le système affiche l'interface de création en fonction du choix <p>Le scenario reprend en son point 3</p> <p>Post conditions : L'acteur est connecté</p>

Les diagrammes d'activité et de séquence de l'enchaînement qui réalise le cas d'utilisation « Créer un programme » sont :

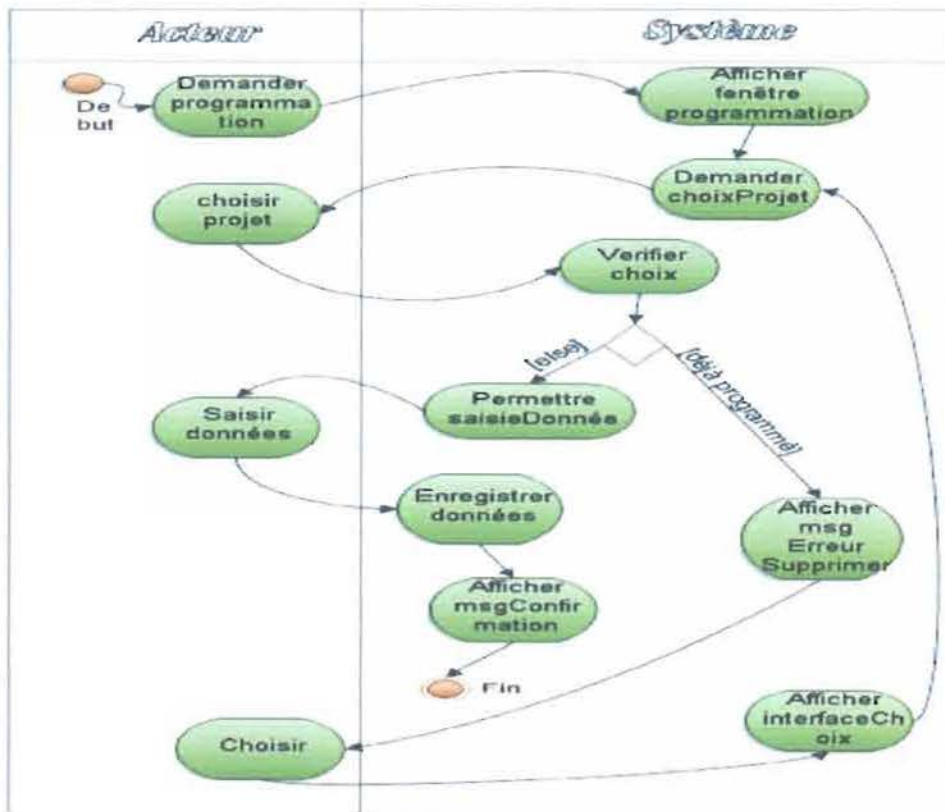


FIGURE 12: DIAGRAMME D'ACTIVITE DU CU8 « CREER UN PROGRAMME »

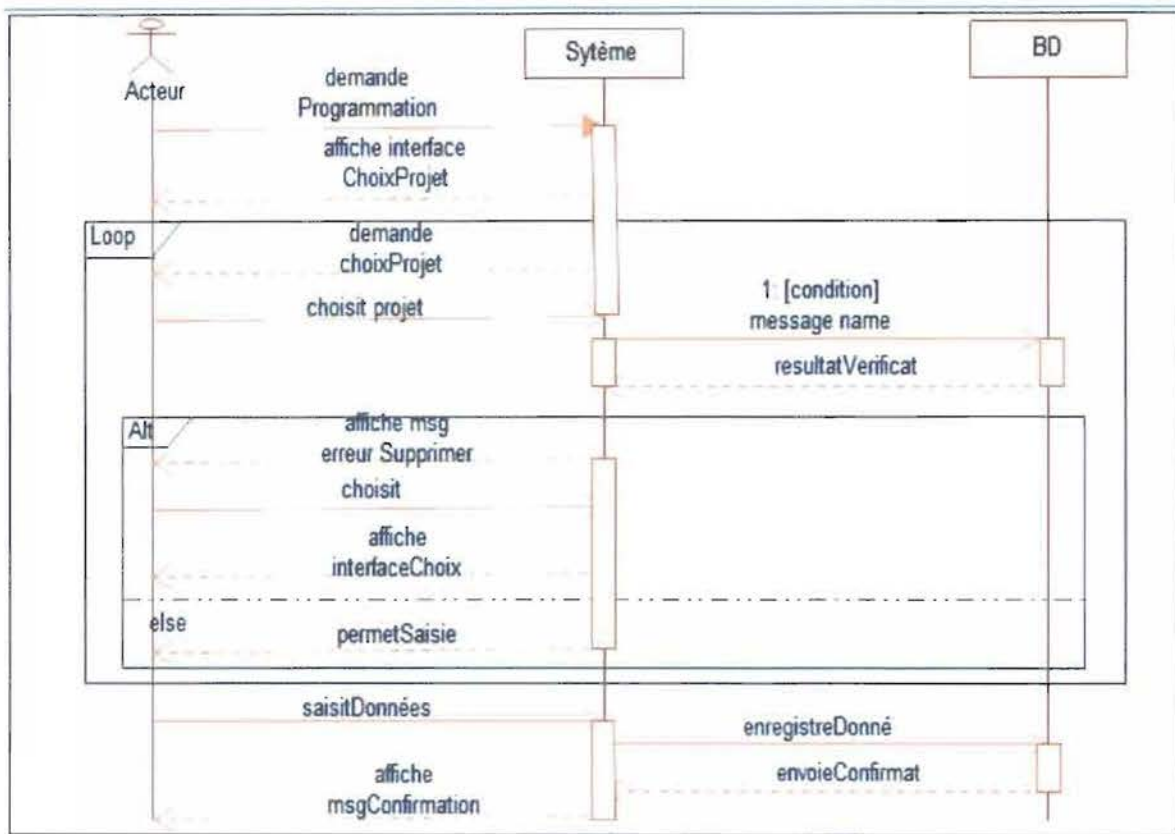


FIGURE 13: DIAGRAMME DE SEQUENCE DU CUB « CREER UN PROGRAMME »

Le cas d'utilisation « Enregistrer une activité d'un programme » à pour enchainement celui suivant .

TABLEAU 6: CU9 « : ENREGISTRER UNE ACTIVITE D'UN PROGRAMME»

Identification du cas
<p>Nom : enregistrer une activité d'un programme</p> <p>Objectif : permettre d'enregistrer une activité d'un programme</p> <p>Acteurs : Equipe de suivi de projet</p>
Description du fonctionnement du cas
<p>Pré conditions : Le programme est créé</p> <p>Scénario nominal :</p> <p>1-l'acteur demande à créer une activité</p> <p>2-Le système affiche l'interface de saisie des données de l'activité</p> <p>3-L'acteur saisit les données dont le nom de l'activité</p> <p>4-Le système vérifie dans la base s'il existe une activité du même nom du même programme</p> <p>5-Le système enregistre les données dans la base et affiche un message de confirmation</p> <p>Scénarii d'exception :</p> <p>Exception3 : le nom de l'activité existe déjà dans ce programme</p> <p>5-Le système affiche un message spécifiant que le nom existe déjà</p> <p>Le scénario reprend en son point 2.</p> <p>Post conditions : L'activité est créée.</p>

« Enregistrer une activité d'un programme » en diagrammes d'activité et de séquence se présente ainsi :

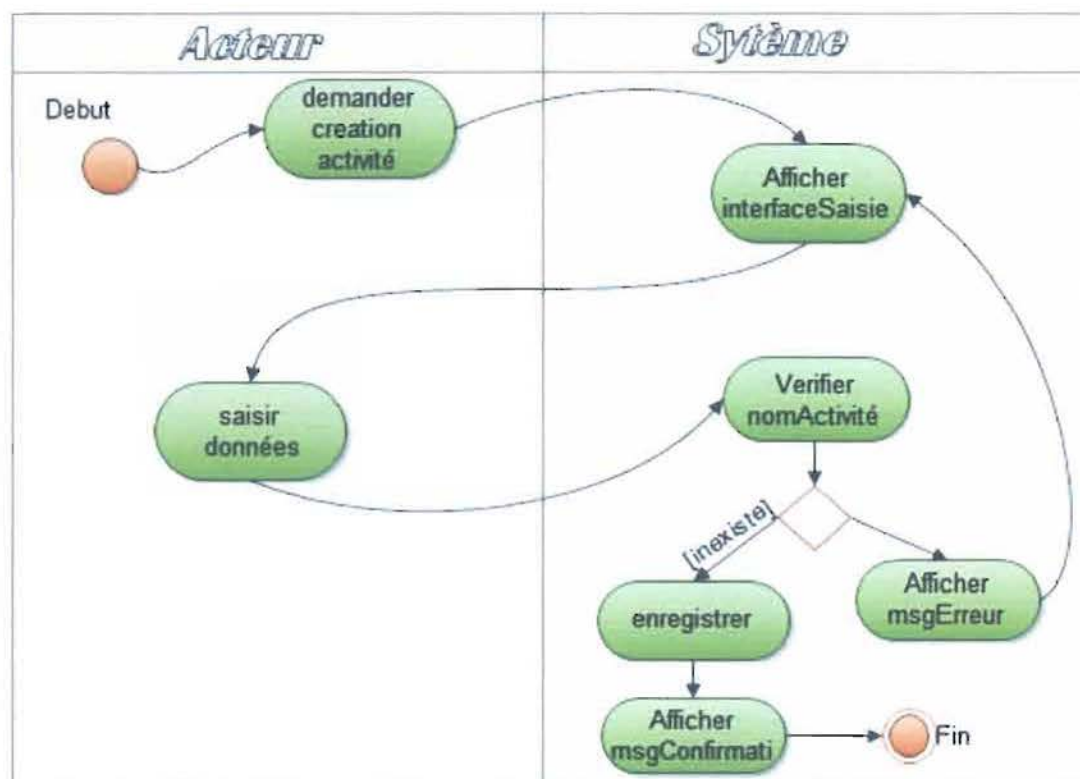


FIGURE 14: DIAGRAMME D'ACTIVITE DU CU9 « ENREGISTRER UNE ACTIVITE D'UN PROGRAMME »

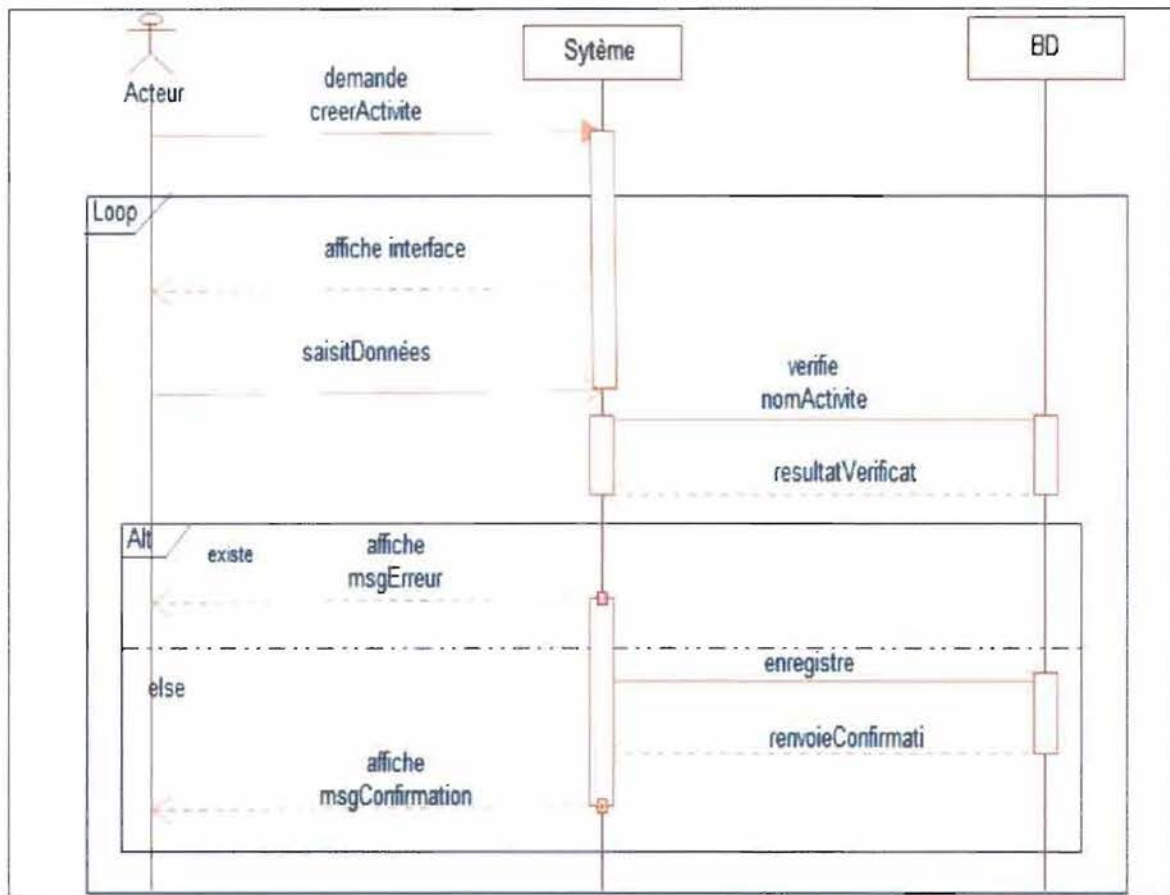


FIGURE 15: DIAGRAMME DE SEQUENCE DU CU9 « ENREGISTRER UNE ACTIVITE D'UN PROGRAMME »

L'enchaînement des opérations pour « Effectuer une recherche autre que sur le financement » est le suivant :

TABLEAU 7: CU10 « EFFECTUER UNE RECHERCHE AUTRE QUE SUR LE FINANCEMENT »

Identification du cas
<p>Nom : Effectuer une recherche</p> <p>Acteurs : tous les acteurs</p> <p>Objectif : pouvoir rechercher les informations sur un« acteur projet », une commune, un ouvrage et un projet.</p>
Description du fonctionnement du cas
<p>Pré conditions : l'acteur est connecté</p> <p>Scénario nominal :</p> <p>1-le système affiche une zone de saisi</p> <p>2-l'acteur entre l'appellation d'un « acteur projet » ou l'appellation d'un ouvrage.</p> <p>3-le système recherche les données correspondant</p> <p>4-le système affiche toutes les informations retrouvées</p> <p>Scénario d'exception :</p> <p>Exception 4 : La recherche a été infructueuse</p> <p>4-Le système signale l'inexistence des données recherchées</p> <p>Le scénario reprend en son point 1</p> <p>Post conditions : l'acteur a terminé ses recherches</p>

Pour le cas d'utilisation « Effectuer une recherche autre que sur le financement », nous obtenons les diagrammes ci-dessous.

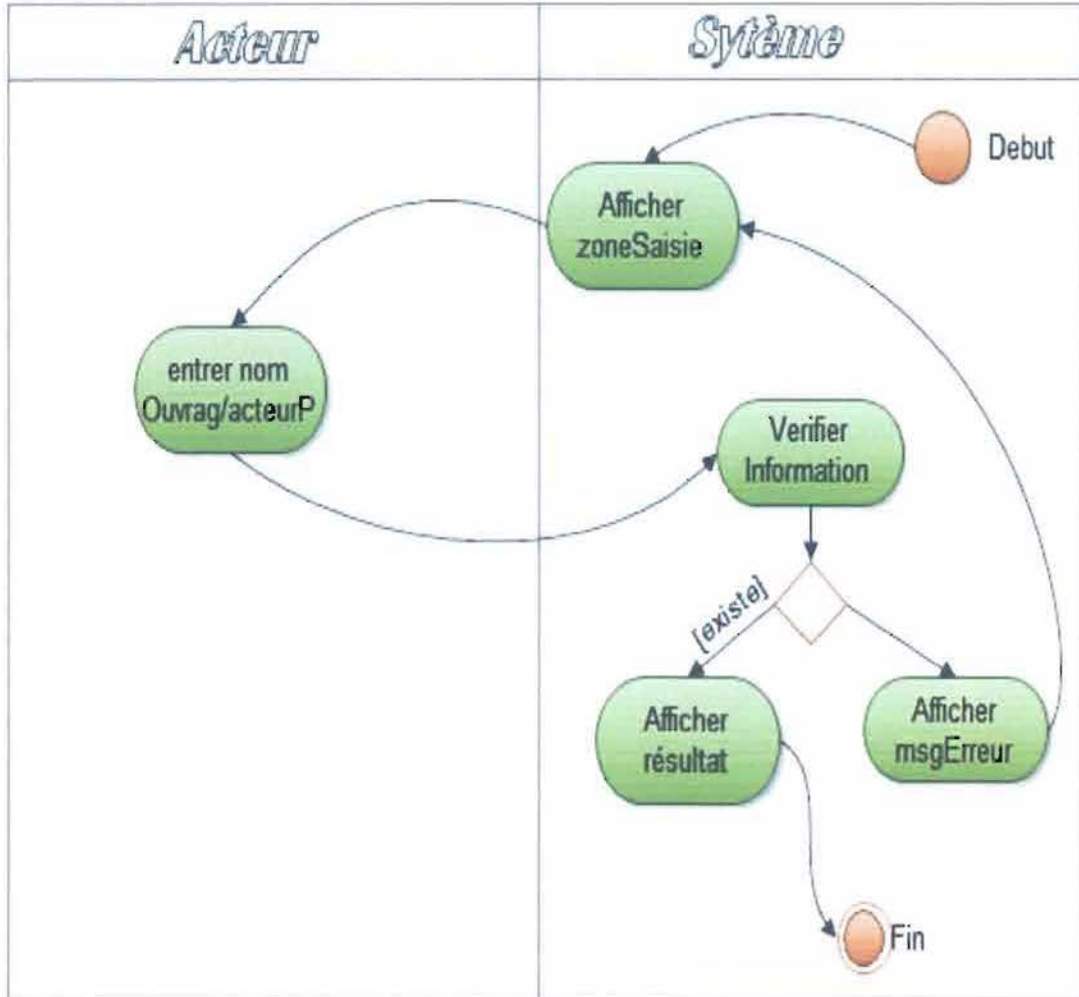


FIGURE 16: DIAGRAMME D'ACTIVITE DU CU10 « EFFECTUER UNE RECHERCHE AUTRE QUE SUR LE FINANCEMENT »

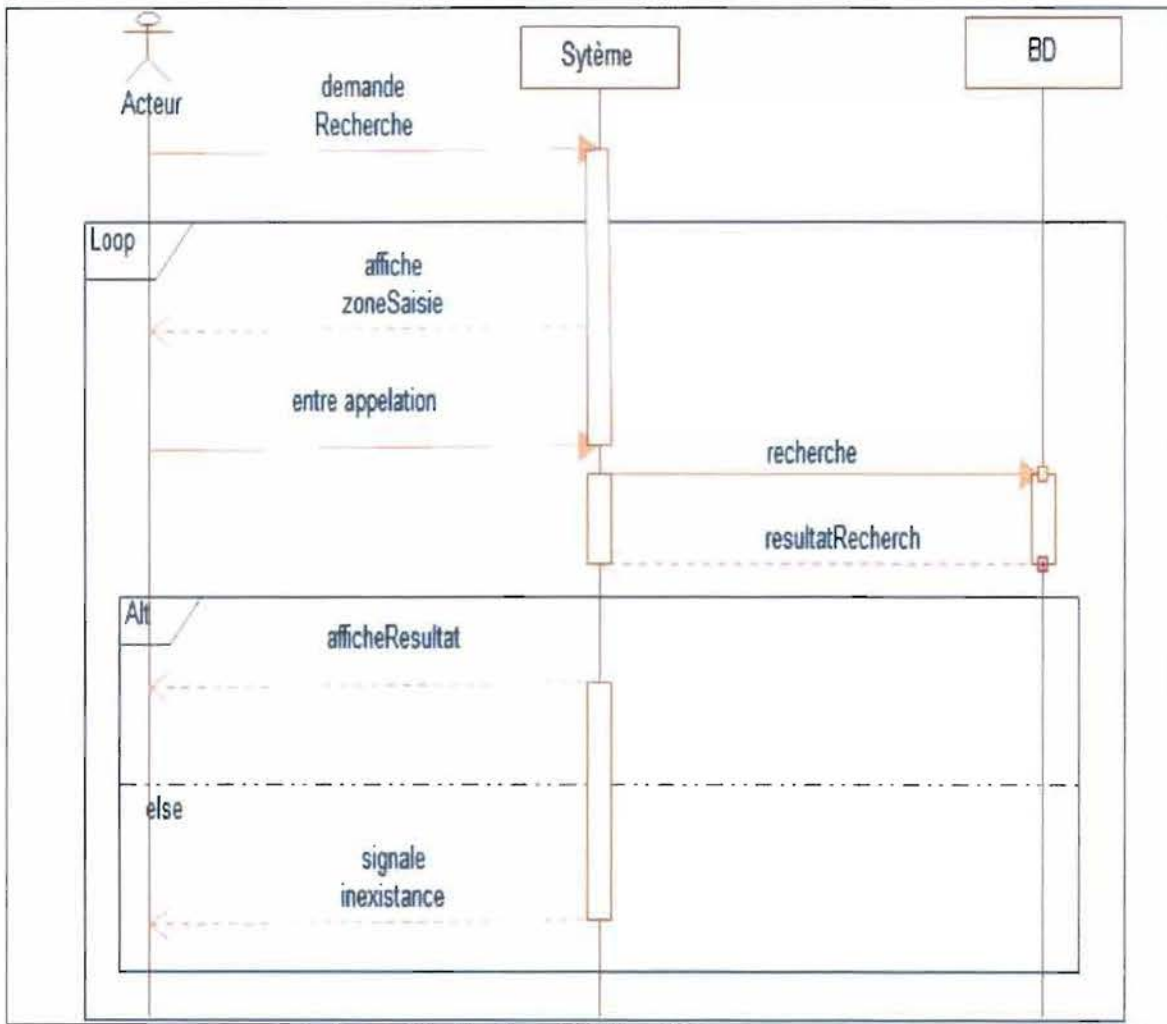


FIGURE 17: DIAGRAMME DE SEQUENCE DU CU10 « EFFECTUER UNE RECHERCHE AUTRE QUE SUR LE FINANCEMENT »

Le tableau de l'enchaînement des opérations pour « Effectuer des recherches sur un financement » se présente ainsi :

TABLEAU 8 : CU11 « EFFECTUER DES RECHERCHES SUR UN FINANCEMENT »

Identification du cas
Nom : Rechercher les informations sur un financement
Acteurs : l'équipe de suivi de projet
Objectif : permettre de rechercher toutes les informations sur un financement acquis
Description du fonctionnement du cas
Pré conditions : l'acteur est connecté
Scénario nominal :
1-le système demande d'abord à l'acteur de choisir une année
2-l'acteur fait son choix
3-le système vérifie si l'année choisie existe dans la base
4-le système affiche une interface permettant de choisir un moteur de recherche qui est soit le CUI soit une commune soit une province
5-l'acteur fait un choix
6-le système recherche dans la base
7-le système affiche les résultats de la recherche en fonction du choix
Scénario d'exception :
Exception 5 : l'année choisit est inexistante
4-le système affiche un message demandant de ressaisir une autre année
Le scénario reprend en son point 1
Post conditions : l'acteur a terminé ses recherches

Avec le cas d'utilisation « Effectuer des recherches sur un financement », nous obtenons les diagrammes d'activité et de séquence suivants :

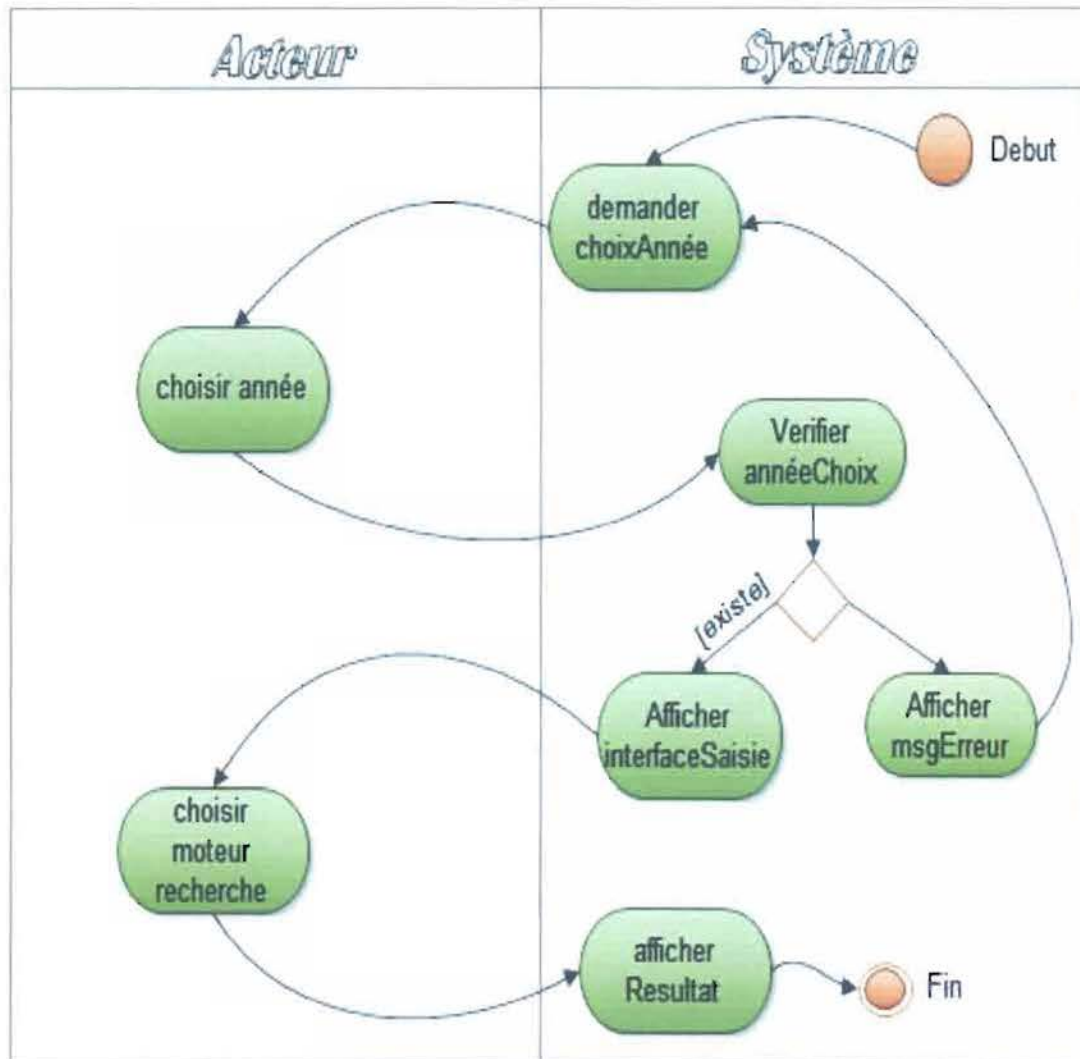


FIGURE 18: DIAGRAMME D'ACTIVITE DU CU11 « EFFECTUER DES RECHERCHES SUR UN FINANCEMENT »

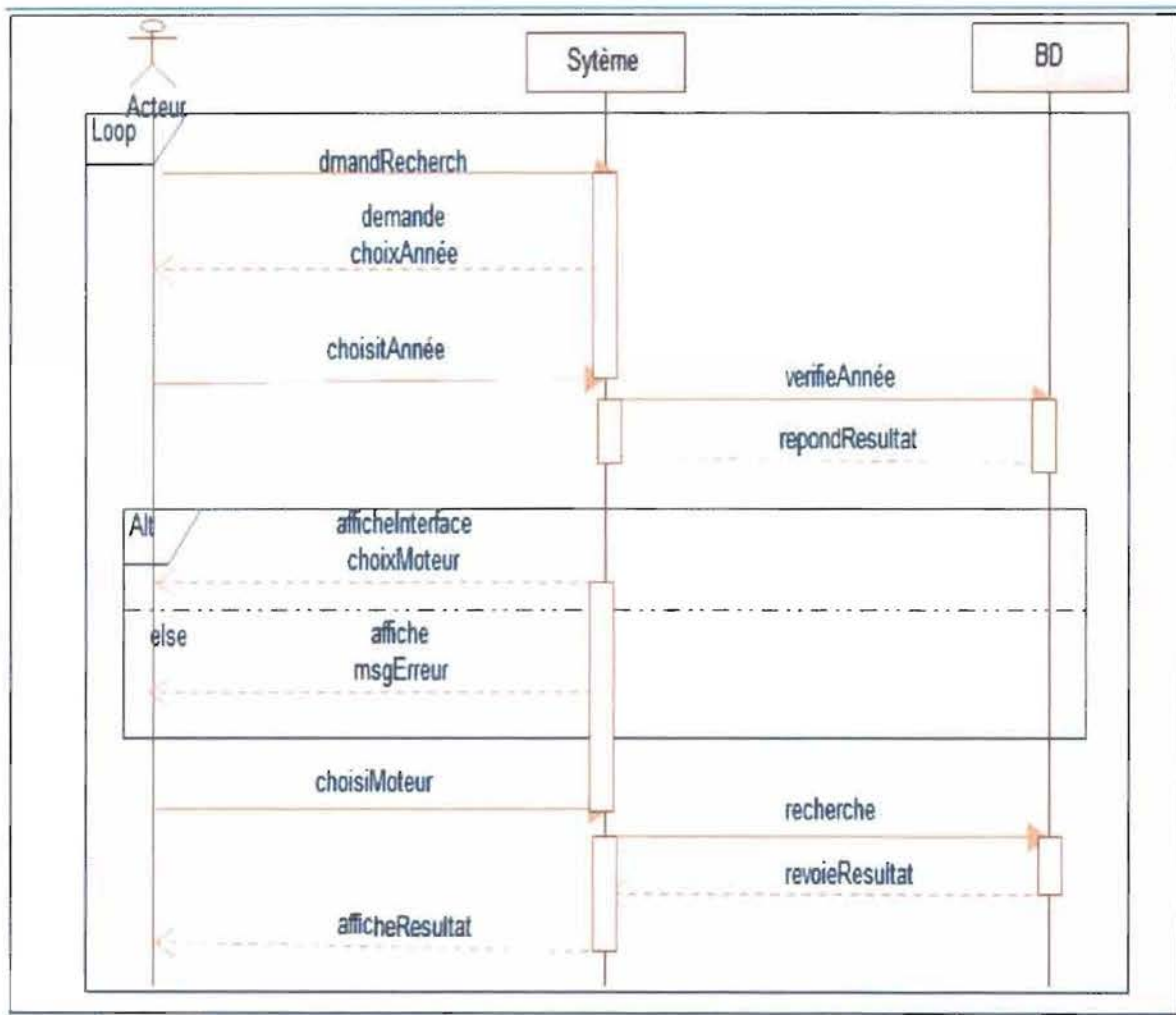


FIGURE 19: DIAGRAMME DE SEQUENCE DU CU11 « EFFECTUER DES RECHERCHES SUR UN FINANCEMENT »

Conclusion

Cette phase nous a permis d'énumérer les différents cas d'utilisation et de prendre conscience de ce que chaque cas d'utilisation implique grâce aux différents diagrammes d'activité et de séquence. Ces différentes descriptions favoriseront une meilleure analyse du travail. Nous terminerons par une conception du système.

CHAPITRE IV : ANALYSE ET CONCEPTION

Introduction

Toute bonne analyse aboutit à une bonne conception en ce sens que l'analyse permettra d'avoir une vision globale du système. La conception quant à elle décrira de façon précise et concise le fonctionnement du système futur afin d'en faciliter la réalisation. Ainsi tout au long du chapitre nous donnerons d'abord le diagramme de cas d'utilisation du système. Ensuite, nous construirons diagramme de classe et pour finir nous ferons un choix d'architecture réseau pour notre système.

I- DIAGRAMME DE CAS D'UTILISATIONS

La figure suivante présente le diagramme de cas d'utilisation définissant les exigences fonctionnelles attendues, les acteurs (utilisateurs du système) ainsi que les relations qui unissent ces acteurs et fonctionnalités.

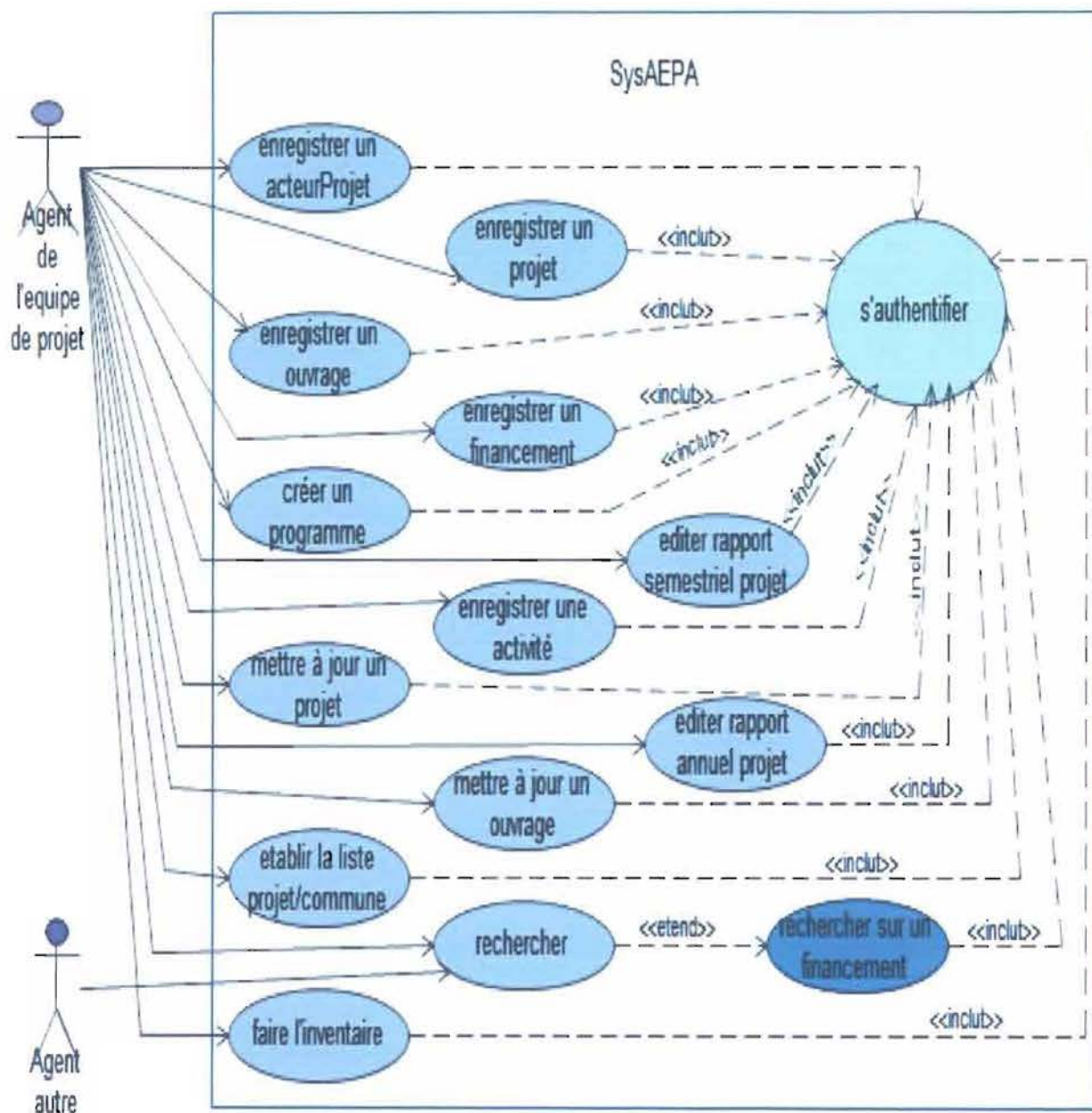


FIGURE 20: DIAGRAMME DES CAS D'UTILISATIONS

II- DIAGRAMME DE CLASSES

II.1- LES REGLES DE GESTION

Pour une bonne structuration des données, nous avons établi des règles de gestion. Ces règles permettront d'éviter au maximum les redondances de données dans la base de données. Nous pouvons citer entre autre :

RG1 : Un projet est réalisé par au moins un « acteur projet ».

RG2 : Tout projet est réalisé dans une localité bien définie.

RG3 : Un projet réalise soit un ouvrage d'eau, soit un ouvrage d'assainissement, soit un type de CUI.

RG4 : Tout projet pris à un instant donné a une étape bien déterminée.

RG5 : Un programme est formé d'au moins une activité.

RG6 : Un programme peut être utilisé pour plusieurs projets différents.

RG7 : Pour un projet qui a un programme, il est utilisé un type de matériels donnés.

RG8 : Tout financement est fait par un « acteur projet ».

RG9 : Un besoin est exprimé par une localité bien donné.

RG10 : Tout « acteur projet » intervient dans au moins un domaine.

II.2- LE DIAGRAMME DE CLASSE

La figure suivante présente la structure statique du système c'est-à-dire la partie du système toujours présente lors des interactions définies par les cas d'utilisation.

- Description de certaines classes

TABLEAU 9: DESCRIPTION DE LA CLASSE « ACTEURPROJET »

CLASSE : ActeurProjet		
Attributs		
Code	Type	Description
id_acteur_p	Int	identifiant
appellation	String	Nom porté par l'acteurProjet
contact	String	L'adresse téléphonique permettent de joindre à tout moment l'acteurProjet
adresse_siege_bf	String	L'adresse du siège officiel de l'acteurProjet au Burkina-Faso
nom_representant	String	Nom de la personne représentant l'acteurProjet
Num_type_acteur	int	C'est un numero que nous avons lié à aux types d'acteurProjet existant pour une reconnaissance aisée des types données.
Methodes		
Nom	Type	Description
enregistrer_acteur()	void	Permet d'enregistrer un nouvel acteurProjet
maj_acteur()	void	Permet de mettre à jour les renseignements sur l'acteurProjet
liste_acteur()	void	Permet d'établir la liste des acteurProjets existants
supprimer_acteur()	void	Permet de supprimer des informations sur un acteurProjet

TABLEAU 10: DESCRIPTION DE LA CLASSE « PROJET »

CLASSE : Projet		
Attributs		
Code	Type	Description
id_projet	Int	Identifiant du projet
Nom	String	Nom donné au projet
duree_prevue	Int	Durée projet prevue
Methodes		
Nom	Type	Description
créer_projet()	void	Permet d'enregistrer un nouveau Projet
maj_projet()	void	Permet de mettre à jour les renseignements sur un Projet
lister_projet()	void	Permet d'établir la liste des Projets existants
supprimer_projet()	void	Permet de supprimer des informations sur un Projet

TABLEAU 11: DESCRIPTION DE LA CLASSE « EAU »

CLASSE : Eau		
Attributs		
Code	Type	Description
id_ouvrage_e	Int	Identifiant de l'ouvrage d'eau
date_fonction	Date	Date à laquelle l'ouvrage a été mise en service
source_eau	String	Source d'eau qui alimente le forage
système_protection	String	Système mis en place pour protéger l'ouvrage
tete_forage	Int	Montre si le forage a une tête ou non
compteur_forage	Int	Montre si le forage à un compteur ou non
index	Int	Permet de connaitre le débit du forage
nom_ouvrage	String	Titre ou denomination de l'ouvrage
id_projet	Int	Identifiant du projet qui a réalisé l'ouvrage
Methodes		
Nom	Type	Description
enregistrer_ouvrage_e	void	Permet d'enregistrer un nouvel ouvrage d'eau
maj_ouvrage_e	void	Permet d'actualiser les informations sur un ouvrage d'eau
liste_ouvrage_e	void	Permet d'avoir la liste de tous les ouvrages d'eau d'un type donné

TABLEAU 12 : DESCRIPTION DE LA CLASSE « ASSAINISSEMENT »

CLASSE : Assainissement		
Attributs		
Code	Type	Description
id_ouvrage_a	Int	Identifiant de l'ouvrage assainissement
nbr_trou_cabine	Int	Permet de determiner le nombre de trou
en_bloc	Int	Montre si l'ouvrage est en bloc
population_destine	String	Population bénéficiaire de l'ouvrage
date_fonction	Date	Date de mise en fonction
nom_ouvrage	String	Titre ou denomination de l'ouvrage
id_projet	Int	Identifiant du projet qui a réalisé l'ouvrage
Methodes		
Nom	Type	Description
enregistrer_ouvrage_a	Void	Permet d'enregistrer un nouvel ouvrage d'assainissement
maj_ouvrage_a	Void	Permet d'actualiser les informations sur un ouvrage d'assainissement
liste_ouvrage_a	void	Permet d'avoir la liste de tous les ouvrages d'assainissement d'un type donné

III- ETUDE DES SCENARI ARCHITECTURAUX

Dans le but faire un meilleur développement, nous allons étudier deux scénarii qui mettent en évidence des architectures de réseaux et de logiciels. Cette étude nous aidera à faire un choix optimal qui convient à nos besoins.

III.1-DESCRIPTION DU PREMIER SCENARIO

Ce premier scénario met en place une architecture logicielle 2-tièrs client lourd c'est-à-dire que les parties présentation et applicative seront du côté client et partie gestion des données côté serveur. Les données seront ainsi centralisées sur un seul serveur de données. Les différentes machines de la structure pourront y accéder via un réseau soit par des câbles soit par wifi depuis les différents postes respectifs. L'application sera installée sur chaque ordinateur, mais chaque utilisateur pourra travailler simultanément avec les autres.

❖ Architecture réseau

Les différents éléments de cette proposition d'architecture sont :

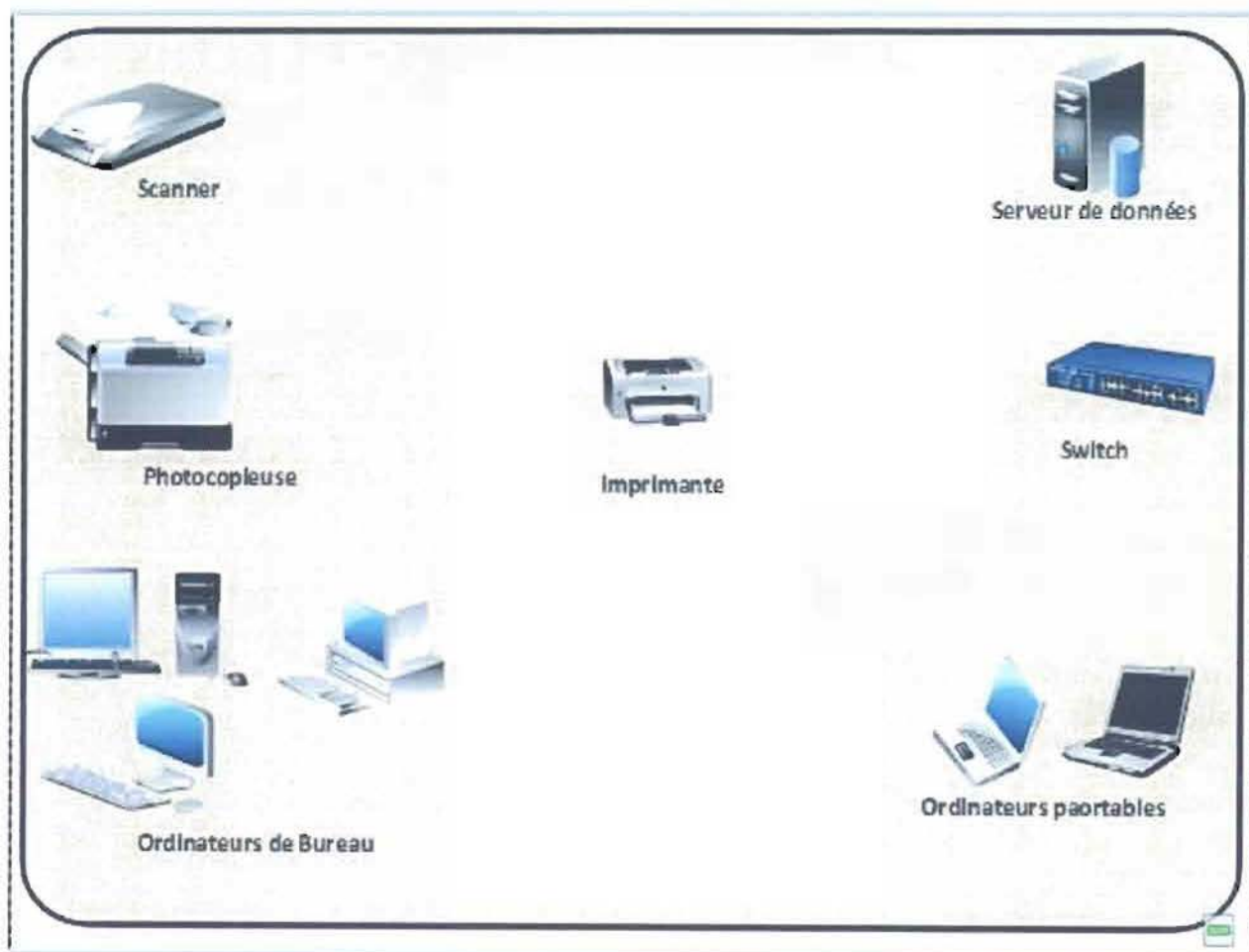


FIGURE 22: LES SYMBOLES D'ARCHITECTURE RESEAU DU PREMIER SCENARIO

L'architecture de ce premier scénario de réseau se présente comme suit :

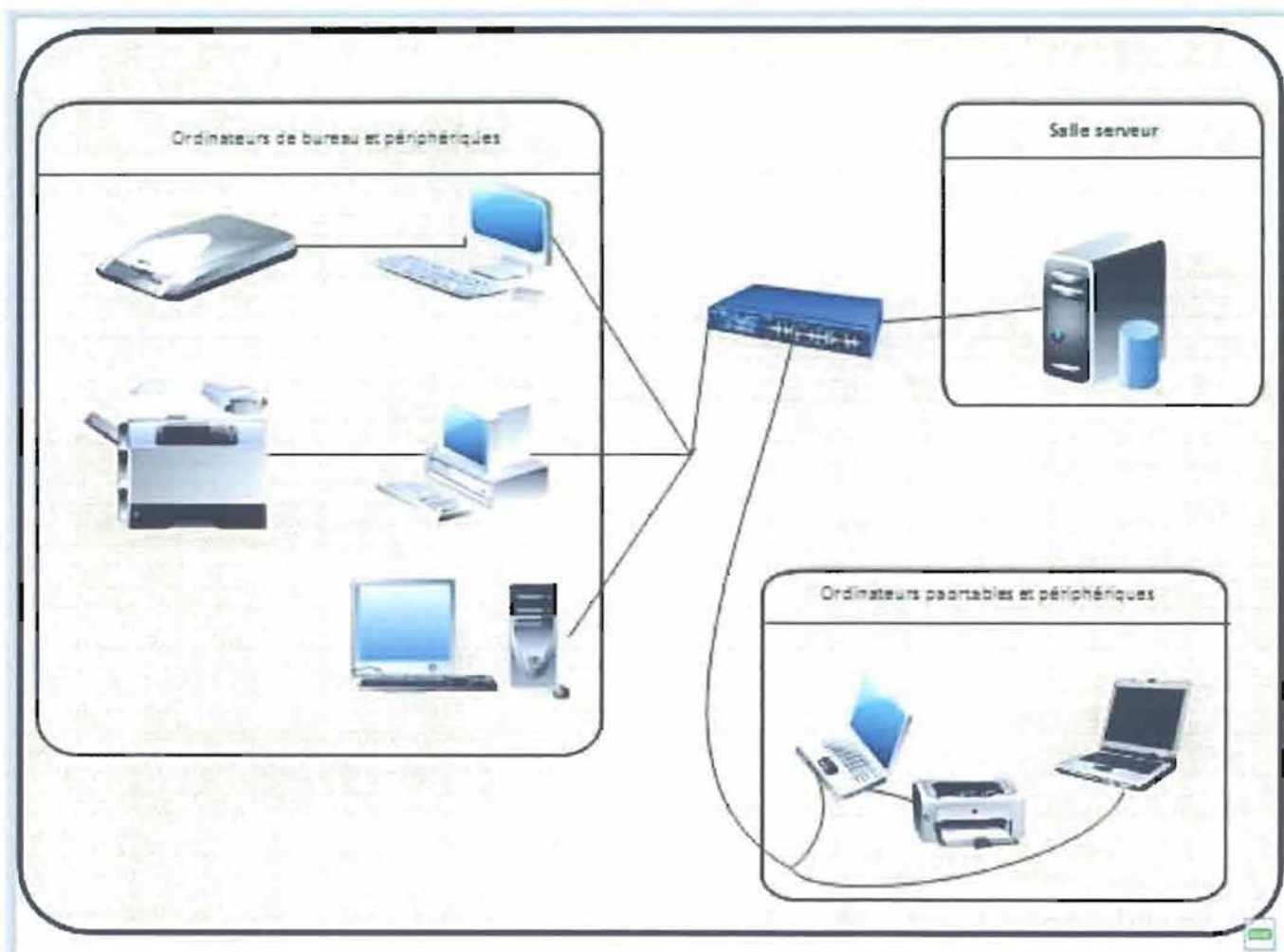


FIGURE 23: ARCHITECTURE RESEAU DU PREMIER SCENARIO

❖ Matériels et logiciels à fournir

TABLEAU 13: EQUIPEMENT NECESSAIRE AU PREMIER SCENARIO

Désignation	Caractéristiques	Quantités
Serveur BD	Serveur HP Proliant DL 380 G7 Disque dur 300Go RAM 6Go Processeur Intel Quad-Core Xeon E5649 à 2,53Ghz	01
Switch	Switch-PT Link 10/100 RJ45 24Ports	01
SGBD	PostgreSql 9.2	---
IDE	Eclipse	---

Evaluation du coût des matériels et des logiciels**TABLEAU 14: EVALUATION DU COUT DE L'EQUIPEMENT DU PREMIER SCENARIO A FOURNIR**

Désignation	Quantité	Prix unitaire (en F CFA)	Prix total (en F CFA)
Serveur BD	01	2 832 000	2 832 000
Switch	01	99000	99000
SGBD	-	-	Gratuit
IDE	-	-	Gratuit
Total (en F CFA)			2 931 000

Coût du développement de l'application

COCOMO (acronyme de l'anglais *Constructive COst MOdel*) est un modèle permettant de définir une estimation de l'effort à fournir dans un développement logiciel et la durée que ce dernier prendra en fonction des ressources allouées. Vu l'application à développer, nous estimons sa complexité de niveau P (en anglais *semi detached*). Ainsi la formule de calcul de coût proposée par COCOMO est la suivante :

$$\text{Effort en Homme/Mois (Effort HM)} = 3 * KILS^{1,12}$$

KILS (pour Kilo Ligne Source) représente le nombre de milliers d'instructions que contiendra l'application. L'application est estimée à 10000 lignes de codes sources.

$$\text{Donc KILS} = 8000/1000 = 8.$$

$$\text{Temps de Développement (TDEV EN MOIS)} = 2,5 * (\text{Effort HM})^{0,35}$$

TABLEAU 15: COUT DE DEVELOPPEMENT DE L'APPLICATION DU PREMIER SCENARIO

Intitulé	Formule	Valeur
Effort à consentir (HM)	$3(8\ 000/1\ 000)^{1,12}$	30,80 Homme-mois
Temps de développement (TDev)	$2,5 (30,80)^{0,35}$	8,30 ~ 8 mois
Nombre de personnes nécessaires	Effort / TDev : 30,80/8,30	3,71 ~ 4 personnes
Coût financier de l'application	Effort*Salaire moyen : $30,80*200\ 000$	6 160 000 F CFA

Coût de formation des utilisateurs

TABLEAU 16: COUT DE FORMATION DES UTILISATEURS POUR LE PREMIER SCENARIO

Prix de l'heure (F)	Nombre de séances	Nombre d'heures par séance	Nombre d'utilisateurs	Coût total de formation (F CFA)
5000	3	4	6	360 000

Coût total de mise en œuvre**TABLEAU 17: COUT TOTAL DU PREMIER SCENARIO**

Désignation	Prix (F CFA)
Coût du matériel et des logiciels	2 931 000
Coût de formation des utilisateurs	360 000
Coût de développement	6 160 000
Coût total	9 451 000

Critiques du premier scénario**○ Avantages**

- L'accès à l'application est réduit à seulement ceux qui en ont le droit
- Cette architecture est adaptée aux applications dont les types de données sont non évolutives dans le temps.
- Les données sont plus en sécurité.

○ Inconvénients

- L'application est non souple à l'extension de la structure

III.2- DESCRIPTION DU DEUXIEME SCENARIO

Ce deuxième scénario quant à lui décrit une architecture 3-tiers. Il y aura un serveur de données et un serveur d'application. Les données seront gérées par le serveur de données grâce à la base de données. Mais l'application sera uniquement sur le serveur d'applications d'où elle communiquera avec la base de données via les requêtes. L'utilisateur aura ainsi accès à l'application grâce à un navigateur web.

❖ Architecture réseau

Les éléments nécessaires à la réalisation de cette architecture sont :

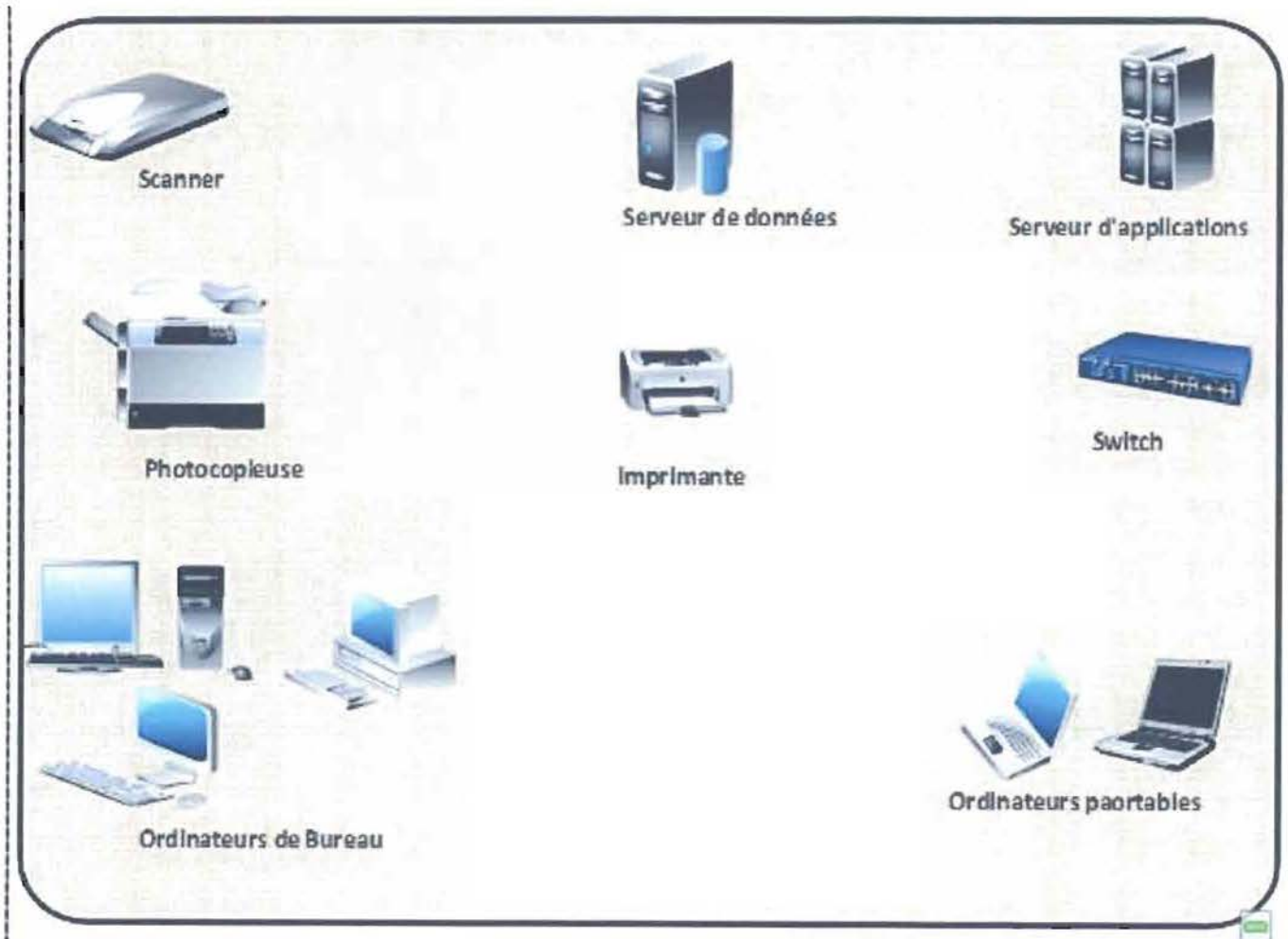


FIGURE 24: LES SYMBOLES DE L'ARCHITECTURE DU DEUXIEME SCENARIO

Nous pouvons schématiser l'architecture réseau de ce deuxième scénario ainsi:

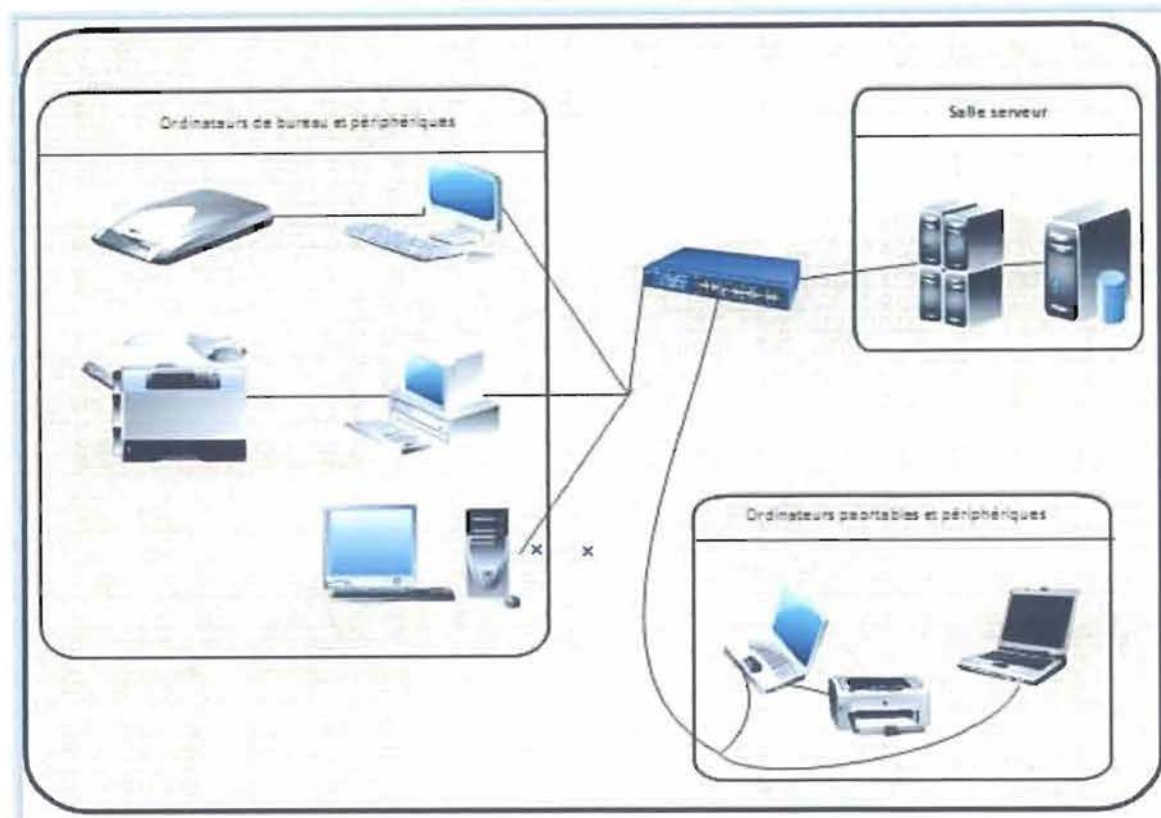


FIGURE 25: ARCHITECTURE RESEAU DU DEUXIEME SCENARIO

❖ Matériels et logiciels à fournir

TABLEAU 18: EQUIPEMENT NECESSAIRE AU DEUXIEME SCENARIO

Désignation	Caractéristiques	Quantités
Serveur BD	Serveur HP Proliant DL 380 G7 Disque dur 300Go RAM 6Go Processeur Intel Quad-Core Xeon E5649 à 2,53Ghz	01
Serveur d'Applications	Serveur HP Proliant ML 350 Gen8 Disque dur 146Go RAM 4Go Processeur Intel Dual-Core Xeon E5-2609 à 2,53Ghz	01
Switch	Switch-PT Link 10/100 RJ4524Ports	01
SGBD	PostgreSql 9.2	-
IDE	Eclipse	-

Evaluation du coût des matériels et des logiciels

TABLEAU 19: EVALUATION DU COUT DE L'EQUIPEMENT DU DEUXIEME SCENARIO

Désignation	Quantité	Prix unitaire (en F CFA)	Prix total (en F CFA)
Serveur BD	01	2 832 000	2 832 000
Serveur d'applications	01	1 855 550	1 855 550
Switch	01	99000	99000
SGBD	-	-	Gratuit
IDE	-	-	Gratuit
Total (F CFA)			4786550

Coût du développement de l'application

COCOMO (acronyme de l'anglais *Constructive Cost MOdel*) est un modèle permettant de définir une estimation de l'effort à fournir dans un développement logiciel et la durée que ce dernier prendra en fonction des ressources allouées. Vu l'application à développer, nous estimons sa complexité de niveau P (en anglais *semidetached*). Ainsi la formule de calcul de coût proposé par COCOMO est la suivante :

$$\text{Effort en Homme/Mois (Effort HM)} = 3 * \text{KILS}^{1.12}$$

KILS (pour Kilo Ligne Source) représente le nombre de milliers d'instructions que contiendra l'application.

$$\text{Temps de Développement (TDEV EN MOIS)} = 2,5 * (\text{Effort HM})^{0.35}$$

I

TABLEAU 20 : COUT DE DEVELOPPEMENT DE L'APPLICATION DU DEUXIEME SCENARIO

Intitulé	Formule	Valeur
Effort à consentir (HM)	$3(8\ 000/1\ 000)^{1,12}$	30,80Homme-mois
Temps de développement (TDev)	$2,5(30,80)^{0,35}$	8,30 ~ 8mois
Nombre de personnes nécessaires	Effort / TDev : 30,80/8,30	3,71 ~ 4 personnes
Coût financier de l'application	Effort*Salaire moyen : $30,80*200\ 000$	6 160 000 F CFA

Coût de formation des utilisateurs

TABLEAU 21: COUT DE FORMATION DES UTILISATEURS POUR LE DEUXIEME SCENARIO

Prix de l'heure (F CFA)	Nombre de séances	Nombre d'heures par séance	Nombre d'utilisateurs	Coût total de formation (F CFA)
5000	3	4	6	360 000

Coût total de mise en œuvre

TABLEAU 22: COUT TOTAL DU DEUXIEME SCENARIO

Désignation	Prix (F CFA)
Coût du matériel et des logiciels	4 786 550
Coût de formation des utilisateurs	360 000
Coût de développement	6 160 000
Coût total	11 306 550

Critiques du deuxième scénario

o Avantages

- Une haute disponibilité de l'application ;
- L'application est souple à l'extension de la structure.

o Inconvénients

- Le coût de développement de l'application est élevé ;
- Les données sont plus vulnérables.

III.3- SCENARIO RETENU

TABLEAU 23 : TABLEAU COMPARATIF DES DEUX SCENARII

Critères	Scénario 1	Scénario 2
Disponibilité	Faible	Haute
Souplesse	Non	Oui
Sécurité	Assez forte	Assez faible
Accessibilité	Restreinte	Large
Coût	Moyen	Élevé

Au regard des types de données gérées par l'application, du genre d'utilisation dont elle sera l'objet et du nombre de personnes relativement bas qui auront fréquemment à travailler dessus, le scénario qui sied le mieux est le premier. En fait, l'accessibilité restreinte proposée par cette solution est un avantage en soi en plus de son aspect sécuritaire.

Cette solution jugée adaptée et réalisable a ainsi été retenue avec l'accord du groupe de pilotage.

Conclusion

Grâce aux différents diagrammes, nous avons vu le fonctionnement global du système. En effet, le diagramme de cas d'utilisation a montré de façon globale les fonctionnalités attendues de sysAEPA tandis que le diagramme de classe nous a fait cas de toutes les classes d'objet existants du système. Des deux (02) scénarii présentés, les besoins ont joué en faveur du premier scénario qui décrit une architecture 2-tiers. Cette conception du système nous guidera dans la phase réalisation.

CHAPITRE V : LA REALISATION

Introduction

Pour notre système la réalisation consiste à l'écriture du code Java permettant d'obtenir les interfaces de l'application. Cette partie du document présentera dans un premier temps quelques prototypes de l'application, montrera la politique de sécurité de l'application dans un second temps. Et dans un troisième nous expliquerons la procédure transitoire.

I- PRESENTATION DE PROTOTYPES

Ce chapitre présentera quelques interfaces réalisées dans le but d'expliquer les principales fonctionnalités du système.

I.1- L'ACCUEIL

L'interface d'accueil est la première de l'application présentée à n'importe quel utilisateur, utilisateur non authentifié. Elle présente un menu permettant de se connecter et d'avoir des informations sur l'application. Elle fait défiler également des images d'ouvrages d'eau et d'assainissement, et jusqu'à ce que l'utilisateur s'authentifie.



FIGURE 26: INTERFACE D'ACCUEIL

I.2- LA CONNEXION

La figure suivante représente l'interface de connexion. Tout utilisateur doit se connecter pour avoir accès aux fonctionnalités dont il a droit. L'interface se présente tant que l'utilisateur n'entre pas les mots justes.



FIGURE 27: INTERFACE DE CONNEXION

I.3- L'INTERFACE D'ENREGISTREMENT D'UN « ACTEUR PROJET »

Elle présente la possibilité d'entrer toutes les informations possible sur un « acteur projet ». Etant donné qu'il y a plusieurs types d' « acteur projet », elle comporte une zone qui change en fonction du type choisi.

Application Recherche Inventaire A Propos

Projet Acteur Financements Ouvrage d'Eau CUI

Denomination : Barika

Nom du Représentant : HIEN Marie

Siège : Gaoua

Contact : 78906534

Domaine d'Intervention : Eau

Durée d'Intervention : 24 (en Mois)

Date debut : 12/08/2013

Date fin :

Période d'Intervention :

Type d'Acteur : Association

Nature de l'Association : ... Non lucrative

Numero de Recplissée : ... A2013/2015

Modifier Supprimer Mise à Jour Enregistrer Annuler Rechercher

FIGURE 28: INTERFACE D'ENREGISTREMENT D'UN « ACTEUR PROJET »

I.4- L'interface de recherche d'un « acteur projet »

Elle permet de rechercher un « acteur projet » en tapant son appellation et une liste de tous les acteurs dont l'appellation est semblable au mot entré s'affiche. A la sélection d'un des acteurs, les informations sur lui s'affichent à côté.

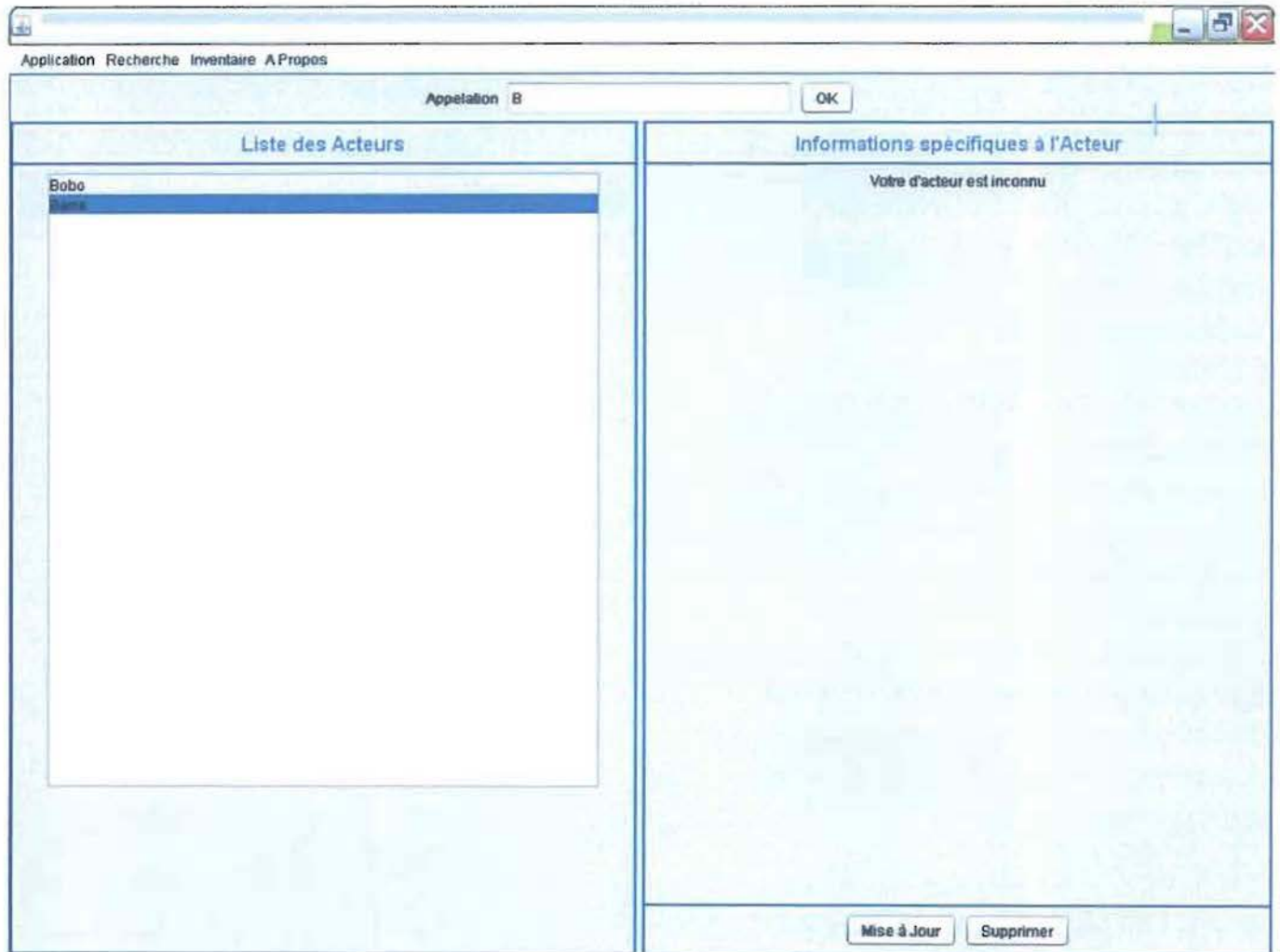


FIGURE 29: INTERFACE DE RECHERCHE D'UN « ACTEUR PROJET »

II- POLITIQUE DE SECURITE

La sécurité du système mise en place se résume à une politique de confidentialité. Il existe des données qui ne doivent être visible que par certaines personnes. Ainsi il a été décidé que les personnes s'occupant de projets d'eau n'auront accès qu'aux données sur ceux-ci et ceux s'occupant de l'assainissement qu'aux données qui les concernent. Mais le Directeur Régional aura accès à tout.

Nous prévoyons qu'à l'inscription d'une personne à l'utilisation de cette application, la possibilité de définir son type. C'est le Directeur qui aura cette possibilité d'inscription de tout utilisateur, ainsi que sa suppression. Mais l'utilisateur aura la possibilité de changer son mot de passe sans possibilité de changer d'identifiant.

III- PROCEDURE TRANSITOIRE

Il est très souvent difficile de changer les coutumes d'une communauté, d'une structure. Pour établir ce changement, il faut donc le faire progressivement sans provoquer d'irritations. De plus, il faut souvent du temps pour tester la robustesse d'une application afin d'en maîtriser les points forts et les faiblesses.

Pour ce faire, il a été décidé d'utiliser l'application en parallèle avec les anciennes méthodes de gestion de données. Cela favorisera ainsi son insertion facilement au sein de la DREAHA, et son acceptation au près du personnel.

Conclusion

Ce chapitre a fait cas de quelques prototypes de l'application réalisée. La politique sécuritaire qui sera mise en place a été adoptée. Cela a favorisé le choix de la procédure transitoire au sein de la structure. Tous ces choix permettront la bonne gestion des données du PN-AEPA.

CONCLUSION GENERALE

Nous avons été accueillis au sein de la Direction Régional de l'Eau, des Aménagements Hydrauliques et de l'Assainissement des Hauts-Bassins dans le cadre de notre stage de fin de Cycle des Ingénieurs de Travaux Informatiques option Analyse et Programmation. Il nous a été confié l'informatisation du Programme National de l'Approvisionnement en Eau Potable et Assainissement des Hauts-Bassins. Cette informatisation revient à connaître tous les « acteurs projet » et les différents projets qu'ils exécutent. L'application devrait permettre un traitement rapide en temps réel des informations sur ceux-ci.

Pour une bonne gestion de ce projet il a été nécessaire de choisir un langage de modélisation qui est UML, et d'un processus de développement qui est 2TUP. Ce processus nous a permis de déterminer les différentes fonctionnalités du système à développer. A cela s'est ensuivi le décernement des besoins pour en faire une analyse afin de concevoir le système en vue de sa réalisation.

Au sortir de notre travail, nous admettons que ce stage nous a placées dans les conditions réelles de la vie professionnelle. Nous avons ainsi eu l'occasion de mettre en application des connaissances acquises à l'école et d'apprendre nouvelles techniques de développement de logiciels.

Il est aussi nécessaire de souligner que le développement de l'application est à sa phase de réalisation. Ce qui laisse entendre que le travail n'est pas totalement fini. Il sera osé de notre part de dire que le travail accompli a été parfait. Par contre nous pouvons admettre que les parties développées font l'objet d'une certaine satisfaction. Pour une plus grande satisfaction des acteurs de la conduite du projet, nous veillerons à finaliser la tâche initialement demandée.

BIBLIOGRAPHIE

(s.d.) « *La réforme du système de gestion des infrastructures hydrauliques d'alimentation en eau potable en milieu rural et semi-urbain.*

(2010). *Rapport 2010 du PN-AEPA.*

Cariou, E. (s.d.). *Architectures client/serveur.* Université de Pau et des Pays de l'Adour, UFR Sciences Pau – Département Informatique: Université de Pau et des Pays de l'Adour.

Claude, G. (2010 – 2011). *UML et les Bases de Données.* ÉDITIONS EYROLLES.

FLEURE, C. (Septembre 2011). *LE MODELE COCOMO, BONNES PRATIQUES V3.*

Gérard, P. (2007/2008). *Processus de Développement Logiciel, Cours M14.* Université de Paris 13.

Roques, P. (s.d.). *UML2 par la pratique.* ÉDITIONS EYROLLES.

Rosmorduc, S. (2000–2005). *jdbc.*

WEBOGRAPHIE

Facade. (s.d.). *Quel SGBD choisir ?* Consulté le 08 06, 2013, sur www.developpez.net:www.developpez.net/forums/u8420/facade

Processus de développement 2TUP.htm. (s.d.). Consulté le 09 05, 2013, sur www.e-bancel.com:www.e-bancel.com/Processus de développement 2TUP.htm

Tutoriel Java. (s.d.). Consulté le 08 20, 2013, sur <http://www.objjs.com/>.

UML en français. (s.d.). Consulté le 09 11, 2013, sur free.fr:www.free.fr/UML%20en%20français.htm

Une conception sur le modèle MVC. (s.d.). Consulté le 09 10, 2013, sur [www.perso.telecom-paristech.fr:www.perso.telecom-paristech.fr/~hudry/coursJava/Une conception sur le modèle MVC.html](http://perso.telecom-paristech.fr:www.perso.telecom-paristech.fr/~hudry/coursJava/Une conception sur le modèle MVC.html)

ANNEXES

ANNEXE 1 : PRESENTATION DE GANTTPROJECT



GanttProject est un logiciel né d'un projet d'étude, à l'Université de Marne La Vallée en 2003, et proposé sous licence libre. Il est utilisable sur de nombreuses plateformes, telles que Windows, Mac et Linux.

GanttProject permet la planification de projets à l'aide du diagramme de Gantt. Il propose les fonctionnalités de base de ce type d'outil, comme la création des tâches, l'affectation des ressources, la gestion des dépendances et de l'avancement, mais dispose également de propriétés plus avancées comme l'exportation des documents en HTML/PDF et le travail collaboratif à distance sur internet. Développé en Java, il nécessite une machine virtuelle JAVA installée.

GanttProject affiche les tâches du projet dans un tableau hiérarchique. De cette façon, un grand nombre d'informations peut être visible rapidement. Par défaut, seuls les Nom, Date de début et Date de fin de la tâche sont affichés. Cependant, il est possible d'afficher :

- Identifiant ;
- Nom ;
- Date de début ;
- Date de fin ;
- Durée (affichée en jours) ;
- Avancée (pourcentage) ;
- Type (tâche mère, simple tâche, ...) ;
- Priorité ;
- Responsable ;
- Information (pour l'instant seuls les éventuels retards) ;
- Liste de prédécesseurs.

GanttProject permet aussi de gérer simultanément plusieurs projets. L'utilisation la plus commune est d'importer successivement plusieurs projets et d'indiqué à GanttProject ces projets.

Facile à prendre en main, il bénéficie d'une qualité graphique remarquable. On peut aisément créer des tâches, les découper, définir les ressources, les allouer aux tâches, construire et modifier le diagramme de Gantt à la souris. On peut aussi gérer, manuellement, le pourcentage de réalisation de chaque tâche

ANNEXE 2 : PRESENTATION DE COCOMO (CONSTRUCTIVE COST MODEL)

COCOMO est un modèle permettant de définir une estimation de l'effort à fournir dans un développement logiciel et la durée que ce dernier prendra en fonction des ressources allouées

COCOMO est divisé en trois modèles :

- ✓ Le modèle de base effectue un simple calcul de l'effort et de la durée en fonction du nombre d'instructions que l'application doit contenir et la complexité de cette dernière. Une ventilation est également possible, permettant de déterminer le temps de développement et l'effort nécessaire pour chaque partie du cycle de développement.
- ✓ Le modèle intermédiaire reprend l'effort et la durée du modèle de base, en appliquant cette fois-ci des coefficients prenant en compte des facteurs de coût (compétence de l'équipe, complexité de l'environnement technique, etc.).
- ✓ Le modèle détaillé reprend les données du modèle intermédiaire en affinant notamment les facteurs de coût en fonction de chaque phase du cycle de développement. Ce modèle n'est véritablement nécessaire que pour de très gros projets.

a-Complexité

Les complexités des applications à développer peuvent être de trois types :

- S (en anglais organic) : Ce sont des applications simples, n'ayant que peu de cas particuliers et de contraintes. Elles sont parfaitement déterministes.
- P (en anglais semidetached) : Ce sont des applications intermédiaires, plus complexes que les applications de type S, elles restent tout de même déterministes, bien que le nombre de cas particuliers et de tests doivent être plus important que pour les applications de type S
- E (en anglais embedded) : Ce sont des applications très complexes, que ce soit au niveau de leurs contraintes (comme un système temps réel) ou au niveau des données saisies (comme certaines interfaces graphiques où l'on ne peut envisager toutes les possibilités de saisies qu'un utilisateur pourrait effectuer). Elles ne sont pas déterministes.

La catégorisation d'une application dans un type de complexité reste une des choses la plus compliqué à définir dans le modèle de base de COCOMO. En cas de doute et pour ne pas avoir de surprise (comme une sous-estimation de l'effort et donc du temps de développement), il vaut mieux surestimer la complexité d'une application, sans tomber dans l'excès... En fonction de la complexité de l'application, on utilisera différents coefficients prenant en compte les différentes complexités et forcément les efforts différents à fournir.

b- Formules

Complexité	Effort (en mois homme)	Temps de développement (en mois)
S	Effort = $2,4 \cdot KLS^{1,05}$	TDev = $2,5 * KLS^{0,38}$
P	Effort = $3 \cdot KLS^{1,12}$	TDev = $2,5 * KLS^{0,35}$
E	Effort = $3,6 \cdot KLS^{1,2}$	TDev = $2,5 * KLS^{0,32}$

KLS (pour Kilo Ligne Source) représente le nombre de millier

Le plus complexe est de déterminer le nombre de KLS. À première vue, on peut se dire que c'est une chose impossible ou avec une très grande marge d'erreur. Cependant, pour être valable le modèle COCOMO ne doit être utilisé que lorsque la phase de conception est déjà bien avancée, de manière à avoir une idée assez précise du travail à réaliser. De plus, l'expérience de la personne utilisant le modèle est déterminante, car il sera ainsi en mesure de s'approcher au plus près du bon nombre de KLS.

c- La ventilation

Comme nous l'avons vu plus haut, la ventilation, aussi appelé distribution par phase, permet de déterminer le temps de développement et l'effort nécessaire pour chaque phase du cycle de développement. COCOMO divise en 4 grandes phases le cycle de développement :

- Expression des besoins et planification (Plans et requirements) ;
- Conception générale (Product design) ;
- Programmation (Programming) ;
- Conception détaillée (Detailed design) ;
- Programmation et tests unitaires (Code and unit test) ;
- Tests et intégration (Integration and test).

Selon la complexité et la taille (en KLS) de l'application, l'effort et le temps de développement varie sur chacune des phases. Le modèle COCOMO exprime cela sous la forme d'un coefficient représentant le pourcentage d'effort à réaliser et le temps passé.