

UNIVERSITE DE YAOUNDE I

ECOLE NORMALE SUPERIEURE DE
YAOUNDE

DEPARTEMENT D'INFORMATIQUE ET
DE TECHNOLOGIES EDUCATIVES



UNIVERSITY OF YAOUNDE I

HIGHER TEACHER'S TRAINING
COLLEGE

DEPARTMENT OF COMPUTER AND
INSTRUCTIONAL TECHNOLOGIES

AGREGATION EN LIGNE DES DONNEES STATISTIQUES SUR LES ENSEIGNANTS DU SECONDAIRE AU CAMEROUN.

*Mémoire présenté et soutenu en vue de l'obtention du Diplôme de Professeur
d'enseignement Secondaire Deuxième Grade (DIPES II)*

Par :

BELIBI NDJIE Albert

Licence en Informatique

EBONG BEYEKE Justine Diane

Licence en Lettres Bilingues

ESSOMO MBOUTCHOUANG Eric Steve

Licence en Géologie

Sous la direction de :

Dr. AZEGUE ONDOA RENE

Année Académique : 2015-2016

Seul On Fait Si Peu, Ensemble On Peut Faire Tellement Beaucoup.

HELEN KELLER

DEDICACE

*A nos familles,
Pour tout votre amour et votre soutien
indéfectible.*

REMERCIEMENTS

Nous tenons à remercier vivement toutes les personnes qui nous ont accompagnés dans cette formation, précisément durant la rédaction de ce mémoire, pour en faire un des moments les plus enrichissants :

*Tous les enseignants du département d'informatique et des technologies éducatives, pour la qualité de leurs enseignements et de nos échanges avec eux, et particulièrement le **Pr. Fouda Ndjodo Marcel** qui, par la qualité de ses enseignements sur les Systèmes d'Information, nous a donné envie de nous intéresser à ce sujet,*

*Le **Dr. Azegue Ondoa Réné**, pour son écoute et pour le temps qu'il a accepté de nous consacrer pour nous conseiller dans le choix du sujet de mémoire et dans le suivi de celui-ci.*

*Nos remerciements vont également à l'endroit de **Mr. ZOBO** pour ses conseils pratiques et son expertise.*

Nous remercions aussi toutes les personnes qui nous ont apporté leur aide pour l'élaboration de ce mémoire qui ont participé indirectement à la réalisation de ce mémoire, notamment pour leur relecture et leurs critiques qui nous ont permis de réaliser au mieux ce travail de recherche.

*Nos camarades de la promotion « **visionnaire** » qui par leurs personnalités et leurs connaissances, mais aussi par leurs complicités, nous ont offert des échanges forts tant sur le plan académique que personnel,*

Nous ne pouvons conclure cette liste sans adresser de chaleureux remerciements à nos proches et amis, qui nous ont soutenus tout au long de cet intense périple.

TABLE DES MATIERES

DEDICACE	i
REMERCIEMENTS	ii
TABLE DES MATIERES	iii
RESUME	vii
ABSTRACT	viii
LISTE DES ABREVIATIONS	ix
LISTE DES FIGURES	x
LISTE DES TABLEAUX	xi
INTRODUCTION GENERALE	1
I. CONTEXTE ET JUSTIFICATION	3
II. PROBLEMATIQUE	6
III. OBJECTIFS DE L'ETUDE	6
IV. LE PLAN	7
CHAPITRE 1 : REVUE DE LITTERATURE	8
I. BASES DE COMPREHENSION DES SIME	9
1. Description des SIME	9
2. Architecture d'un SIGE/SIME	12
3. Intérêt pour le public cible	13
II. ETUDE DE CAS	14
1. Le SIME sénégalais.....	14
2. L'initiative Ethiopienne : Teacher's Management Information System (TMIS)	16
3. OpenEMIS : un système d'information pour la gestion de l'éducation.....	16
III. CADRE METHODOLOGIQUE	18
1. La collecte et le traitement des données statistiques	18
2. Réalisation du système de gestion des enseignants.....	18
CHAPITRE 2 : METHODOLOGIE	21
I. METHODE, MODELE D'ANALYSE ET DE CONCEPTION DU SYSTEME	22

1.	La méthode d'analyse et de conception : l'approche orientée objet	22
2.	Le langage de modélisation du système : UML	23
II.	LE MODELE DE DEVELOPPEMENT DU SYSTEME	24
1.	Le modèle de développement en V	25
2.	Le modèle de développement en cascade	26
3.	Le modèle de développement en spirale	26
4.	Le modèle de développement par incrément.....	27
III.	L'ARCHITECTURE DU SYSTEME	28
1.	L'architecture 3 tiers.	28
2.	l'architecture observateur/observable	29
3.	L'architecture MVC	29
V.	LES OUTILS DE DEVELOPPEMENT.....	31
1.	Le système de gestion de bases de données (SGBD).....	31
2.	Le langage de programmation	34
CHAPITRE 3 : CONCEPTION ET IMPLEMENTATION DU SYSTEME . 37		
I.	CONCEPTION ET MODELISATION STATIQUE.....	38
1.	Les diagrammes de classe	38
II.	CONCEPTION ET MODELISATION DYNAMIQUE	41
1.	Les diagrammes de cas d'utilisation	41
2.	Les diagrammes de séquence	43
3.	Le diagramme de déploiement	45
III.	IMPLEMENTATION DU SYSTEME.....	46
1.	Les pages d'accueil	47
1.	Le trajet d'une requête.....	49
CHAPITRE 4 : RESULTATS ET IMPLICATIONS..... 52		
I.	LES RESULTATS.....	53
1.	Description de la solution.....	53
2.	Rapport problématique/ SETEMIS : les fonctionnalités.....	54
3.	L'évaluation	58

II. DISCUSSION	59
CONCLUSION ET PERSPECTIVES	64
BIBLIOGRAPHIE	67
ANNEXES	70
ANNEXE A : LA DESCRIPTION DES SCENARIOS	I
ANNEXE B : LE CAHIER DE CHARGE	II
ANNEXE C : LE GUIDE D'UTILISATION	III

RESUME

La gestion des enseignants du secondaire se heurte à certaines difficultés dues à la lenteur et à lourdeur des processus de traitement et d'édition des données des informations. C'est dans ce cadre que s'inscrit ce projet de fin d'études portant sur le développement d'une plateforme de collecte et de production en ligne des statistiques sur les enseignants du secondaire au Cameroun afin d'assurer la consultation et le suivi de ce personnel et de garantir un accès illimité aux informations de celui-ci. Cet outil, SETEMIS entendu *SEcondary TEacher's Management Information System*, se positionne comme est un instrument de collecte, traitement, de promulgation en ligne mais aussi comme un outil d'aide à la décision pour les planificateurs et gestionnaires de l'éducation. Ce mémoire se compose dans une première partie d'une introduction générale qui renseigne sur le contexte, la problématique, les objectifs de l'étude et le plan du mémoire. La deuxième partie baptisée revue de littérature évoque les travaux antérieurs dans le domaine des Systèmes d'Information pour le Management de l'Education (SIME). Les questions méthodologiques sont abordées en troisième point tandis que le quatrième point lui, laisse entrevoir le travail de conception et d'implémentation du système. Les résultats et les diverses implications de ce travail sont présentés dans le cinquième point ; et le travail se termine sur la conclusion et les perspectives pour l'amélioration de SETEMIS.

Mots clés : Management, Système d'Information, SIME, système d'aide à la décision, Planification.

ABSTRACT

The management of secondary teachers faces some difficulties due to failures of the actual system because of slowness and heaviness within the processes of treatment and edition of data. In this light, lays our end of course project which consists on the development of a platform that enables the collection and production of statistics over secondary teachers in Cameroon in order to ensure the consultation of teachers' statistics and therefore guarantee an unlimited access to their information. This tool, SETEMIS understand as *SEcondary TEacher's Management Information System*, stand as an implement of gathering, treatment and publication using internet connexion but also as a system which helps in decision-making the managers and educator administratives. This work is composed of a general introduction which explains the choice of this topic. The second part is about previous works on the domain of Education Management Information System (EMIS), followed by methodological questions. The fourth point concerns the design and implementation processes while the next point is about the results and the differents implications of the merger of SETEMIS in the system.this work ends with the conclusion and the future prospects for the improvements of SETEMIS.

Key words : management, information system, EMIS, decision-making system, planning.

LISTE DES ABREVIATIONS

- BD : bases de données
- BREDA : bureau régional pour l'éducation en Afrique
- CapEFA : programme de renforcement des capacités pour l'Education pour Tous
- CSS : cascading style sheet
- DIPES I : diplôme de professeur de l'enseignement secondaire 1^{er} grade
- DIPET I : diplôme de professeur de l'enseignement technique 1^{er} grade
- EMIS : education management information system
- ENIEG : école normale des instituteurs de l'enseignement général
- ENIET : école normale des instituteurs de l'enseignement technique
- ENS : école normale supérieure
- ENSET : école normale supérieure de l'enseignement technique
- EPT : éducation pour tous
- HTML : hypertext markup language
- HTTP : hypertext transfert protocol
- IIRCA : Institut International Pour Le Renforcement Des Capacités en Afrique
- MVC : modèle vue contrôleur
- OMD : Objectifs du Millénaire pour le Développement
- ONG : Organisation Non Gouvernemental
- ONU :Organisation des Nations Unies
- PHP : php hypertext preprocessor
- SADT : structured analysis design technic
- SETEMIS : Secondary Teacher's Management Information System
- SGBD : système de gestion de base de données
- SGBDR: système de gestion de base de données relationnelle
- SI : système d'information
- SIGE : système d'information de gestion de l'éducation
- SIGIPES : système informatisé de gestion intégrée du personnel de l'état et de la solde
- SIME : système d'information pour le management de l'éducation
- SQL : structured query language
- TMIS : teacher's management information system
- UML : unified modeling language
- UNESCO : organisation des nations unies pour l'éducation, la science et la culture

LISTE DES FIGURES

<i>Figure 1 : Diagramme du flux de données actuel au Ministère des enseignements secondaires</i>	4
<i>Figure 2: Cycle de vie de l'information dans un SIME</i>	11
<i>Figure 3 : Schéma du SIGE Camerounais</i>	13
<i>Figure 4: Schéma du système d'information de gestion de l'éducation du Sénégal,</i>	15
<i>Figure 5: Flux d'information dans le SIGE Sénégalais,</i>	15
<i>Figure 6 : Modèle de développement en V,</i>	25
<i>Figure 7: Schéma du modèle MVC,</i>	30
<i>Figure 8: Diagramme des classes gérer formulaire établissement</i>	39
<i>Figure 9 : Diagramme des classes gérer transfert enseignant</i>	40
<i>Figure 10 : Diagramme des classes valider formulaire</i>	40
<i>Figure 11 : Diagramme général des cas d'utilisation.</i>	41
<i>Figure 12 : Diagramme des cas d'utilisation gérer formulaire enseignant.</i>	42
<i>Figure 13 : Diagramme des cas d'utilisation valider formulaire enseignant</i>	42
<i>Figure 14 : Diagramme des cas d'utilisation gérer compte établissement</i>	43
<i>Figure 15 : Diagramme de séquence créer compte enseignant.</i>	44
<i>Figure 16 : Diagramme de séquence remplir formulaire</i>	44
<i>Figure 17 : Diagramme de séquence valider formulaire</i>	45
<i>Figure 18 : Diagramme de déploiement du système</i>	46
<i>Figure 19 : La page d'accueil du système</i>	47
<i>Figure 20 : La page de connexion</i>	48
<i>Figure 21 : Menu</i>	49
<i>Figure 22 : Item</i>	49
<i>Figure 23 : Page de statistiques</i>	50
<i>Figure 24 : Page de statistique nationale par genre, secteur et grade.</i>	51
<i>Figure 25 : Diagramme du flux des données du Ministère des Enseignements Secondaires après l'intégration de SETEMIS</i>	58

LISTE DES TABLEAUX

<i>Tableau 1 : Le public cible du système</i>	<i>54</i>
<i>Tableau 2 : Les fonctionnalités du système.....</i>	<i>57</i>
<i>Tableau 3 : La grille d'évaluation du système.....</i>	<i>59</i>

INTRODUCTION GENERALE

INTRODUCTION GENERALE

Le développement croissant des Technologies de l'Information et de la Communication pour l'Education (*TICE*) associé à l'informatisation tend à rationaliser les processus et tâches quotidiennes. Ce qui laisse envisager un renouvellement et une amélioration des services et outils mis à la disposition des différentes parties concernées. L'heure est à la rapidité et la vitesse d'exécution, la conservation et l'accès permanent aux données, la limitation maximale voire la suppression des tâches manuelles et du recours au système « *papier* » ainsi que la sécurité des dites données. Cet intérêt pour la recherche de solutions informatiques capables de répondre à leurs besoins pousse les structures éducatives à se doter de systèmes d'information répondant destinés au management de l'éducation (*SIME*). Notre intérêt pour « *l'agrégation en ligne des données statistiques sur les enseignants du secondaire au Cameroun* » illustre notre volonté d'apporter une réponse adéquate à la préoccupation suivant laquelle la gestion informatisée et automatique des enseignants au Cameroun présente des lacunes. Pour parvenir à des conclusions objectives et en vue de mener à bien cette investigation, nous avons procédé à un travail méthodique et méthodologique qui portera tour à tour sur la raison d'être de ce sujet, la problématique qu'il soulève, les objectifs visés par son étude.

I. CONTEXTE ET JUSTIFICATION

Le développement perpétuel de l'informatique et des technologies de l'information et de la communication permet tout autant le développement de différents secteurs de la vie : santé, économie, sécurité et même le domaine de l'éducation. Ce changement peut déjà être observé dans plusieurs services et administrations notamment à travers les différents Systèmes d'Informations (SI) déployés dans les établissements scolaires afin d'en faciliter la gestion (*à l'instar de GEPI utilisé par le Collège François Xavier Vogt, SIACO utilisé par l'Université adventiste de Nanga Eboko et qui est spécialisé dans la gestion des enseignements et des notes des étudiants*); ou encore le SIGIPES (*Système Informatisé de Gestion Intégré des Personnels de l'Etat et de la Solde*) qui, lui, permet la gestion salariale des fonctionnaires du Cameroun. Un regard sur la population enseignante camerounaise, en particulier celle du secondaire, nous a permis de nous rendre compte de l'absence totale d'un système d'information numérisé pour le management des enseignants.

Le choix de ce sujet se justifie ainsi premièrement par le fait que la publication de statistiques crédibles sur l'éducation et leur transmission au moment opportun est un facteur déterminant de l'analyse des interventions des politiques éducatives existantes, l'élaboration de politiques et de stratégies éducatives appropriées au contexte et l'évaluation des progrès accomplis vers la réalisation de certaines initiatives mondiales. Pour ce faire, et conformément aux textes de l'ONU, « *L'Afrique doit développer ses propres données, à l'aide de bases de données complètes et comparables entre pays. Ce domaine prioritaire vise à promouvoir des SIGE pérennes aux échelons continental, régional et des États membres de l'ONU, tout en s'assurant que sont en place un suivi et une évaluation rigoureux des activités éducatives.* » Le développement de SIME se révèle donc être une nécessité dans la mesure où il est le parfait indicateur de qualité sur les performances et capacités d'un système éducatif. Plusieurs pays africains (*Côte-d'Ivoire, Sénégal, ...*) ont de ce fait entrepris de mettre sur pied des SIGE pour le suivi de leurs systèmes éducatifs respectifs. Ceux-ci en grande partie axés sur l'élève et la gestion d'établissement scolaire, ne prennent pas en compte les enseignants qui eux constituent une part importante du système éducatif. C'est sur cette observation que l'Ethiopie a entrepris de développer un système d'information pour la gestion des enseignants : le TMIS (*Teacher's Management Information System*) sous l'initiative du programme CapEFA et suivant l'exemple du SIME développé par l'ONU : OpenEMIS. Les initiateurs de ce projet sont d'avis que « *les projections de l'offre et de la demande d'enseignants, par exemple, sont un bon moyen d'évaluer le manque d'effectifs, de prévoir une pénurie ou un excédent éventuels de*

personnels et d'apprécier l'ampleur des défis des enseignants d'un pays. Malgré cela, les systèmes d'information pour la gestion des enseignants (TMIS) tendent à être négligés dans bien des pays, notamment dans ceux qui sont le plus à la traîne dans la réalisation des objectifs de l'EPT. » Ainsi, à l'exemple de l'Ethiopie, avons-nous opté pour la mise sur pieds d'un système d'information axé sur la gestion des enseignants du secondaire.

Par ailleurs, des insuffisances portant sur le traitement des dossiers des enseignants à plusieurs échelles ont été relevées. Certaines sont dues à la lenteur et à la lourdeur des procédures et processus : collecte, traitement, analyse, publication et diffusion des données des enseignants sur les enseignants. Le schéma suivant illustre cet état des choses.

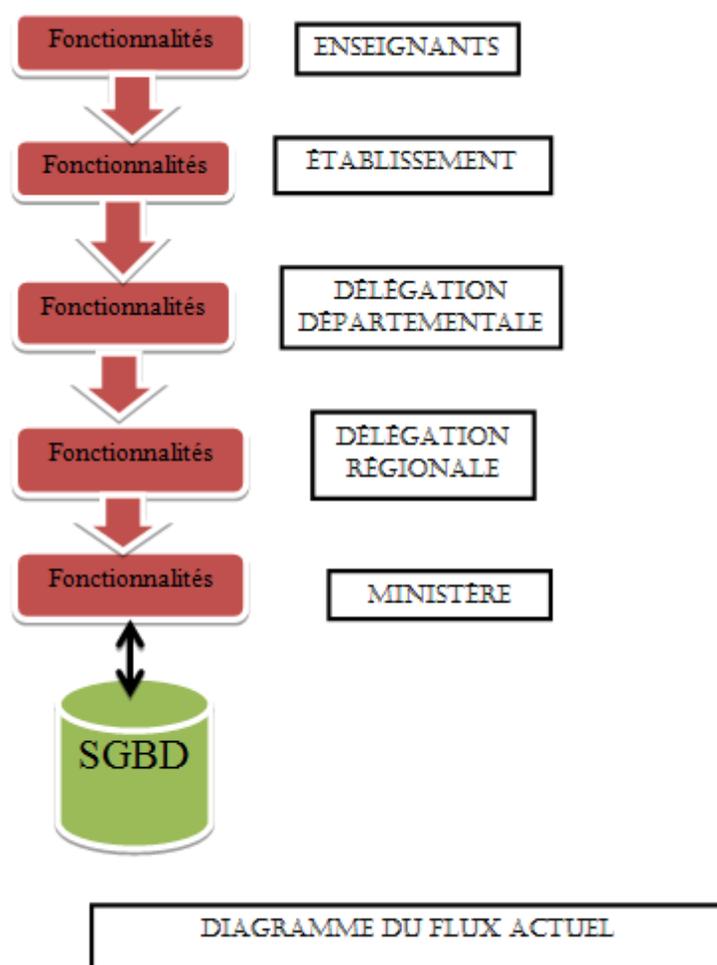


Figure 1 : diagramme du flux de données actuel au Ministère des enseignements secondaires

Le système étant essentiellement manuel, il s'écoule plusieurs mois (3 à 6) pour la collecte des données aux différents niveaux de l'échelle et plusieurs autres mois (2 à 4) pour le traitement de ces données. Ce qui implique que les décideurs doivent souvent se contenter de données fragmentaires et obsolètes causant par là un préjudice considérable dans leur travail. De plus, l'enseignant, comme tout autre personnel de l'Etat, est sujet à une prise en charge qui débute dès sa

sortie d'une quelconque école de formation des enseignants, de l'enseignement normal appelée ENIEG(*Ecole Normale des Instituteurs de l'Enseignement Général*), ENIET(*Ecole Normale des Instituteurs de l'Enseignement Technique*), du secondaire appelée ENS (*Ecole Normale Supérieure*) pour l'enseignement général et ENSET (*Ecole Normale Supérieure de l'Enseignement Technique*) pour l'enseignement technique, toutes localisées principalement dans les villes de Yaoundé, Maroua, Bamili, Douala, Kumba. Cette prise en charge à laquelle l'enseignant est assujéti comprend plusieurs étapes et prend fin à la retraite. En effet, il est tout d'abord affecté suivant des critères définis par l'autorité hiérarchique ; durant l'exercice de ses fonctions, il est sujet à des fluctuations/déplacement et bénéficie des avancements qui ont une incidence sur sa carrière. La gestion de tout ce parcours nécessite une connaissance profonde du circuit ministériel et du corps enseignant lui-même, et demande des informations claires et actualisées qui ne sont pas toujours disponibles dans les délais et de ce fait handicape le travail à effectuer. Par ailleurs, de grands défis se profilent concernant :

- La détermination exacte de son patrimoine, c'est-à-dire définir le nombre qui constitue le corps enseignant afin d'entraîner une meilleure répartition du capital à travers le pays.
- La gestion efficace des carrières des enseignants, notamment les promotions et augmentations octroyées au personnel indépendamment des critères fixés par les décrets et autres textes de lois réglementant le profil de carrière au Cameroun...
- L'amélioration de la transparence dans la gestion des ressources enseignantes
- Le renforcement des capacités de planification du capital enseignant
- La détermination des besoins en enseignant suivant les spécialités et en fonction des régions, particulièrement les Zones d'Education Prioritaire (ZEP).

Enfin, en notre qualité d'étudiants en 5^{ème} année informatique de l'Ecole Normale Supérieure de Yaoundé et futurs enseignants de la même discipline, nous nous proposons de mettre les enseignements reçus et les fruits de nos recherches au service du développement de notre corps en mettant sur pied un système d'information de management des enseignants du secondaire au Cameroun (*SETEMIS*) qui faciliterait désormais la gestion et le suivi des enseignants du secondaire dans un premier temps et qui pourrait être étendu à d'autres corps. De plus, produire cette application permettra de nous arrimer à la vague de modernisme et surfer sur le torrent de nouvelles technologies pour le développement et le progrès. La mise en place de ce SIME pourrait résoudre ce problème à long terme et servir de guide ou encore d'indicateur dans la prise de décision en matière de gestion des enseignants.

Conscients que notre projet est une véritable gageure, nous avons suivi une méthodologie et une démarche strictes et appliquées pour atteindre les objectifs que nous nous sommes fixés.

II. PROBLEMATIQUE

Les développements qui précèdent démontrent à souhait combien le sujet est important au vue de l'objet sur lequel il porte. On pourrait s'interroger sur la raison d'être des Systèmes d'Information de Management de l'Education ce qui serait superflu et proprement inconvenant étant donné la multitude de texte qui les encensent et qui démontrent qu'ils ont déjà fait leurs preuves. La préoccupation centrale dans ce travail repose sur l'efficacité et l'efficience du système d'information que nous envisageons mettre sur pied. De ce fait, comment améliorer la gestion et le suivi du personnel enseignant Camerounais ? Le SETEMIS peut-il résoudre les problèmes actuels du système existant tout en apportant une avancée significative au développement du SIGE camerounais conformément aux objectifs de l'OMD, EPT? Ce mémoire a pour objectif de fournir un outil décisionnel aux politiques et à toutes les couches de bénéficiaire.

III. OBJECTIFS DE L'ETUDE

La mise en place d'un système d'information est basée sur un postulat simple : tout système, toute organisation quels qu'ils soient doivent produire de l'information pour s'informer et informer sur leur état (*et ses caractéristiques*), leur fonctionnement et leurs résultats. Sans données, aucun système ne peut fonctionner rationnellement et par conséquent aucune décision opérationnelle ne peut être prise. A la base de tout système d'information, se pose la question de la finalité des informations à recueillir (*pour la gestion et le contrôle du système, en l'occurrence ici SETEMIS*), la définition des priorités, la planification et la formulation des politiques, enfin pour le suivi et l'évaluation de ces dernières. En d'autres termes, le système éducatif est composé d'une multitude d'acteurs qui se trouvent et agissent à différents niveaux. Bien qu'en interaction les uns avec les autres, ces acteurs ont des besoins divers et variés en information que le système d'information se doit de prendre en compte et de renseigner. SETEMIS est orienté vers les utilisateurs et entend faciliter l'analyse détaillée et la synthèse des données afin d'en extraire les informations les plus saillantes et pertinentes pour aider à la planification de l'éducation et aux décisions politiques.

Le nœud de ce travail est donc de pourvoir les gestionnaires, politiques, décideurs et planificateurs de l'éducation, d'un matériel de travail, véritable outil d'aide à la décision en matière de planification, suivi et gestion des enseignants. Ce qui ne peut être effectif qu'en mettant à la disposition de ces utilisateurs des informations précises, pertinentes, fiables et actualisées sur l'état

du corps enseignant pour l'enseignement secondaire. C'est pourquoi nous avons mené une opération statistique nous permettant de collecter, traiter et publier des données sur les enseignants. Les niveaux de scission/séparation des données concernent aussi bien la localisation géographique (*région, département, arrondissement*), le genre (*femmes et hommes*), le statut des établissements (*public et privé*), le type de formation reçu (*école de formation fréquentée*) et les différents types d'enseignement (*enseignement général et enseignement technique et professionnel*). Ce qui permettra de mettre sur pied un véritable Système d'Information, fiable, sécurisé et d'un accès rapide et illimité pour la gestion de l'enseignant. De plus, la mise en place de ce système d'information sera une vitrine permettant d'évaluer les avancés et progrès conformément aux Objectifs du Millénaire pour le développement(*OMD*) dans sa 2^e condition, au forum mondial sur l'Education (*Dakar 2000*), à la 2^{ième} décennie de l'Education (*2006-2015*), conditions de l'Education Pour Tous. Ce travail permettra de manière spécifique de :

- ❖ Mettre sur pied un mécanisme de consultation et de suivi du corps enseignant au niveau d'un arrondissement tout d'abord mais qui pourrait être étendu au département, à la région et même au niveau national.
- ❖ Faire de la gestion et du suivi des enseignants un exercice automatique, pratique, simple, accessible.
- ❖ Conserver et publier des données nécessaires à la prise des décisions stratégiques en matière d'éducation.

IV. LE PLAN

Ce mémoire est un collectif de six chapitres explicites qui montrent le parcours suivi pour la réalisation de cette application. Le chapitre 1 est une prise de connaissance du sujet pour savoir de quoi il est question et sur quelles bases il repose. Le deuxième chapitre présente un examen de l'existant par une étude de cas de certains systèmes d'information pour la gestion de l'éducation. Dans le chapitre 3, il est question des méthodes et outils de conception et de développement de l'application. Le chapitre 4 laisse voir le travail de conception et d'implémentation du système par des diagrammes et captures d'écran. Ces différentes étapes débouchent sur l'analyse des données et les résultats obtenus et le travail se clôt par une conclusion et un énoncé des perspectives.

CHAPITRE 1

REVUE DE LITTERATURE

CHAPITRE 1: REVUE DE LA LITTERATURE

Ce chapitre présente une vue d'ensemble pratique et du cadre méthodologique des Systèmes d'Information de Gestion de l'Education (*SIGE*) ou Systèmes d'Information de Management de l'Education (*SIME*).

La première partie a pour but de poser les bases des SIME. Dans une première sous partie, nous apporterons une description des SIME, ensuite une présentation de l'architecture des systèmes d'information. Pour finir, il sera question de ressortir l'intérêt de ce travail pour le public cible.

La seconde partie présente une étude de cas basée sur quelques systèmes d'information pour le management de l'éducation déployés dans certains pays pour finir par un système d'information développé par l'UNESCO OpenEMIS.

Dans la dernière partie, nous nous intéresserons au cadre méthodologique de ce travail à travers le processus de collecte et de traitement de données et la réalisation du module pour la gestion des enseignants.

I. BASES DE COMPREHENSION DES SIME

1. DESCRIPTION DES SIME

Le concept de SIME est large et ne répond pas à une définition standard dans la mesure où chacun y va de son entendement. Cette situation constitue un handicap certain car elle cause ainsi un développement anarchique et incompréhensible des systèmes d'informations qui ne répondent pas toujours aux besoins et attentes de l'éducation. Le développement d'un SIME exploitable dépend largement de sa compréhension, du succès de l'intégration des différents types de modules et données issues des différentes sources internes et externes, et des différents niveaux du système éducatif d'une part, mais également de la cohésion entre les différents sous-systèmes d'informations qui composent le SIME. La mise sur pied des plans et politiques d'éducation a pour but d'améliorer la qualité de l'éducation ainsi que de ses différents services et repose de ce fait sur des informations fiables et pertinentes issues du SIME. Le noyau d'un SIGE est souvent une base de données centralisant les données collectées via un recensement scolaire annuel, des données transactionnelles des parties prenantes du système éducatif, ainsi que d'autres sources de données (*données du recensement de la population*). Il est utilisé par les ministères/départements d'éducation, les ONG, les chercheurs, les bailleurs de fonds, et autres intervenants dans le secteur de l'éducation pour la planification, le suivi et la prise de décision politique.

Les SIME font face toutefois à certains problèmes communs qui entravent leur fonctionnement :

- une couverture insuffisante avec des données manquantes (*les écoles ne complétant pas leur recensement*)
- Le manque de données historiques
- La qualité des données et leur validation
- Les différences entre SIGE nationaux complexifient les comparaisons internationales
- Un besoin de modifications permanentes du système
- Des infrastructures et un soutien technique inadéquats dans certains pays
- L'insuffisance d'infrastructures adaptées pour la diffusion des données

❖ Objectifs d'un SIGE/SIME

Un SIME n'est pas un système de gestion des ressources humaines, ou des finances, ni moins un système de gestion administrative, mais il peut être appliqué à la gestion de tel module. Il s'agit d'un système de données et d'informations qui incorpore plusieurs types de données de diverses sources, de multiples niveaux. En résumé, le SIME a pour but de renforcer les capacités de gestion, de planification et de diffusion de l'information à tous les niveaux du système éducatif pour tous les centres de réflexion et de décision. Ainsi, conformément à l'UNESCO, il a pour objectif de :

- *améliorer les capacités de collecte, de traitement, de stockage, d'analyse et de diffusion des données afin que les décideurs, administrateurs et gestionnaires puissent se baser sur des données fiables et actualisées ;*
- *centraliser et coordonner les efforts dispersés dans l'acquisition, le traitement, l'analyse et la diffusion des informations sur la gestion de l'éducation ;*
- *rationaliser la nature et le flux d'informations nécessaires aux prises de décision en réduisant et en éliminant les duplications, et en comblant les lacunes dans l'information ;*
- *relier et rassembler les différents systèmes d'information existants ;*
- *intégrer et synthétiser dans un seul système les données quantitatives et qualitatives;*
- *améliorer la collecte, l'utilisation et la diffusion des informations de gestion de l'éducation pour répondre à l'évolution constante des besoins en informations.*

Les objectifs des politiques éducatives et les résultats attendus à la fin de la mise en œuvre de plans bien pensés, appellent des informations pertinentes pour les planificateurs et les gestionnaires du système éducatif.

❖ Données et outils d'un SIGE/SIME

Un système d'information pour la gestion de l'éducation (*SIGE, anglais: EMIS, education management information system*) est un système de management d'information conçu pour la gestion d'information au sein d'un système d'éducation. Un SIGE facilite la collecte des données, le traitement, l'analyse et le report des informations sur l'éducation ceci incluant les écoles, élèves,

professeurs et autre personnel. L'information du SIGE est utilisée par les ministères en charge de l'éducation, par les ONG, chercheurs, donateurs et des autres parties prenantes de l'éducation pour la recherche, planification, monitoring et évaluation et surtout la prise de décision. L'information provenant du SIGE est particulièrement utilisé pour créer des indicateurs qui surveillent la performance d'un système d'éducation et pour gérer la distribution et allocation des services et ressources de l'éducation.

❖ Cycle de vie de l'information dans un SIGE/SIME

L'élaboration et le suivi des programmes nécessitent un SIGE répondant aussi bien à la multiplicité des niveaux de prise de décision (*politique, planificateur et gestionnaire*) qu'au paradigme de la gestion axée sur les résultats. Le système d'information est un processus systématique de collecte et de transformation de données en informations fiables, valides, opportunes et adaptées aux besoins des différents utilisateurs. Il peut être schématisé dans ses étapes principales :

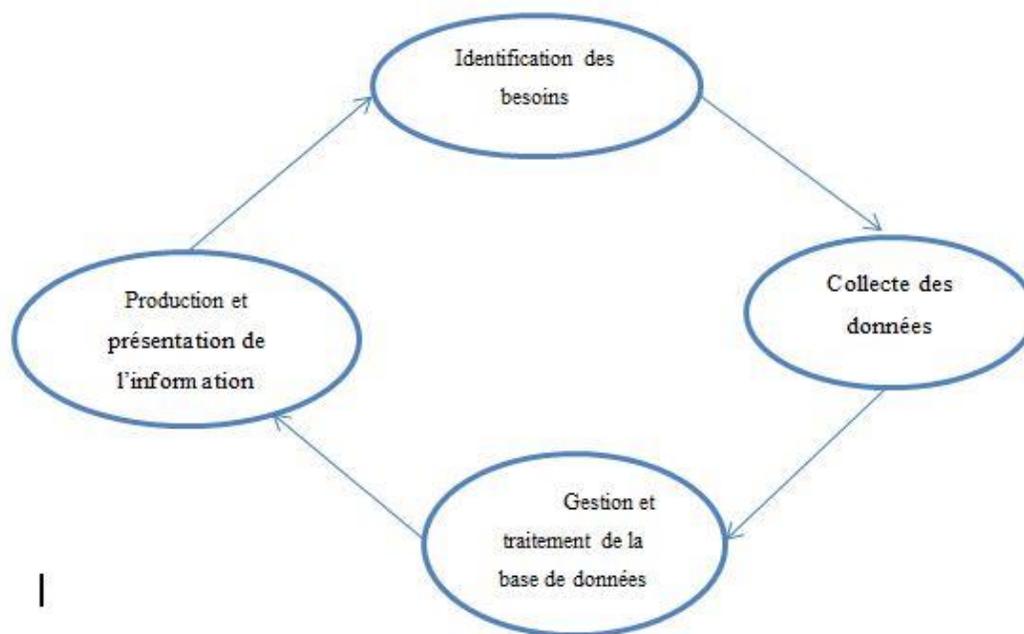


Figure 2: Cycle de vie de l'information dans un SIME.

➤ **Identification des besoins**

Les besoins en informations claires, précises, actualisées, vérifiables et atemporelles ont été exprimés dans les programmes.

➤ **Collecte des données**

La collecte des données se fait à travers une campagne statistique qui est un recensement portant sur l'ensemble des Établissements: Elle va de l'élaboration des questionnaires à la distribution et collecte des données.

- L'élaboration ou la révision des documents de collecte : Il s'agit de préparer les instruments de collecte des données correspondant aux besoins d'information définis.
- La multiplication, distribution et remontée des documents de collecte
- La saisie des données

➤ **Gestion et traitement de la base de données**

C'est la phase de traitement des données ; elle comprend l'organisation, traitement, vérification et nettoyage des données. Son effectivité nécessite de :

- S'assurer que les données de tous les enseignants sont bien saisies.
- Corriger les données incohérentes;
- Faire l'imputation des données manquantes;
- Stabiliser les données;

➤ **Production et présentation de l'information**

Les données collectées, traitées et analysées doivent dès lors être produites : c'est la phase du rapport des données qui implique la publication et la diffusion des informations. L'information produite est présentée sous la forme d'un annuaire statistique, d'un support électronique (*site WEB*).

2. ARCHITECTURE D'UN SIGE/SIME

Le SIME est multi niveaux, multi sources et atemporel. Il dispose d'une base de données qui emmagasine des données précieuses pour la planification de l'éducation. Pour l'élaboration des plans d'éducation, les responsables disposent des données stockées sur plusieurs années et suivant différents critères, qui rendent possible la comparaison des données et informations sur des périodes. Le ministère en charge de l'éducation est porteur du SIME central qui est en relation avec les différents « *petits SIME* ». Son pilotage est assuré par une équipe diversifiée qui est chargée de la collecte et du stockage dans la base de données du SIME.

Les sources de données sont les établissements d'enseignement secondaire. La base de données centrale se situe au ministère de l'éducation nationale. De manière progressive, les données sont collectées au niveau de l'établissement, ensuite remontées vers le niveau supérieur. Le niveau

supérieur de l'établissement est fonction de la structure administrative du pays. Au Cameroun, il s'agit du département, ensuite de la région, enfin le ministère.

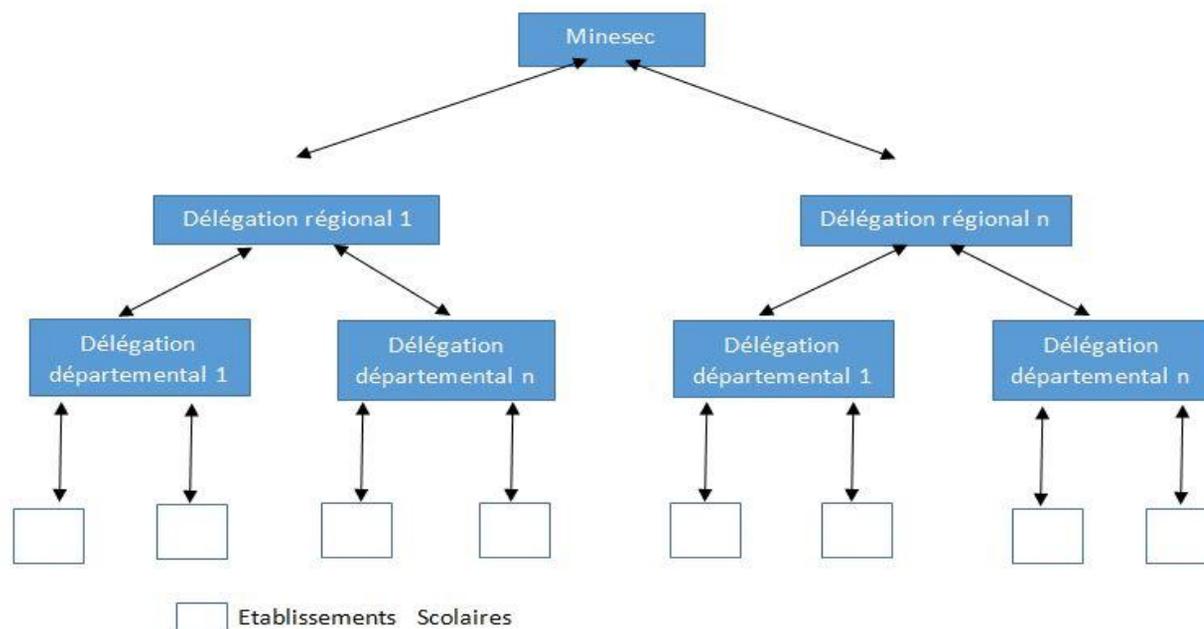


Figure 3 : Schéma du SIGE Camerounais

3. INTERET POUR LE PUBLIC CIBLE

Ce travail loin d'être un SIME, se positionne comme une composante du SIME dans la mesure où il se concentre exclusivement sur les enseignants et non sur la totalité des composantes du système éducatif. Il s'adresse de ce fait à un public aussi riche que varié de décideur, planificateur et d'une manière plus large aux techniciens chargés du recrutement et de la formation, du suivi et de l'évaluation des enseignants camerounais. Il a pour vocation de leur fournir des éléments de réflexion pour leur permettre de comprendre le fonctionnement d'un système d'information statistique (SIS), et de devenir ainsi capable de porter un regard critique sur le fonctionnement des structures en charge du suivi des enseignants et de proposer des solutions pour l'améliorer. Il aide ainsi :

➤ **Le planificateur à:**

- Faire un diagnostic fondée sur une analyse précise des données éducatives passées et présentes de manière à appréhender les lacunes du système pour formuler des politiques éducatives;
- Planifier le développement futur de l'éducation : fixation d'objectifs quantitatifs (*effectifs d'élèves, d'enseignants...*), projection des besoins en infrastructures et des coûts ;
- Suivre le développement du secteur ;
- Préparer des programmes et des projets pour l'amélioration du système éducatif ;

- Evaluer les résultats des plans et des projets d'éducation.
- **Le gestionnaire et/ou l'administrateur à:**
 - Gérer le personnel, le parc immobilier (*bâtiments scolaires*) et les équipements,
 - Identifier les problèmes (*déficit en salles de classe, tables-bancs, manuels scolaires...*),
 - Evaluer les investissements à réaliser et les dépenses de fonctionnement.

Un certain nombre de notions et de concepts nécessaires à la compréhension de l'architecture d'une base de données relationnelle sont expliqués, afin de faciliter le dialogue avec les techniciens chargés du développement du système informatique.

L'intention de ce travail est également de servir de document de référence aux organisations qui, conscientes des insuffisances de leur SIS, cherchent des solutions pour l'améliorer ou le refondre. Des outils et des stratégies ayant prouvé leur efficacité et leur efficacité seront présentés ainsi qu'un modèle informatique capable de s'adapter à des situations diverses.

II. ETUDE DE CAS

Dans le souci d'améliorer la gestion de l'éducation dans leurs secteurs respectifs, plusieurs pays se sont lancés dans le développement de leur système d'éducation et ont pour ce faire conçu ou adopté des SIME.

1. LE SIME SENEGALAIS

Les objectifs des politiques éducatives et les résultats attendus à la fin de la mise en œuvre de plans bien pensés, appellent des informations pertinentes pour les planificateurs et les gestionnaires du système éducatif sénégalais. Le sénégalais dispose d'un SIME évolué et répondant qui s'inscrit suivant la politique éducative du pays traduite sous forme de programme, PDEF entre 2000 et 2011 et PAQUET-EF de 2013 à 2025. La Lettre de Politique Sectorielle (*LPS*) a défini les orientations, les priorités ; le modèle de projection et de simulation donne une valeur cible à l'horizon 2025 aux principaux indicateurs nécessaires au pilotage des objectifs du PAQUET. Le Cadre de Mesure de Rendement (*C.M.R.*), s'adossant sur un modèle logique, répertorie un ensemble d'indicateurs facilitant le suivi des performances des sous-secteurs, tous en cohésion totale avec les programmes de l'UNESCO. Le SIME sénégalais englobe tout le secteur de l'enseignement petite enfance, primaire, moyen-secondaire (*le secondaire est complété par le cycle : 1^{er} cycle et 2nd cycle*) et se présente comme suit :

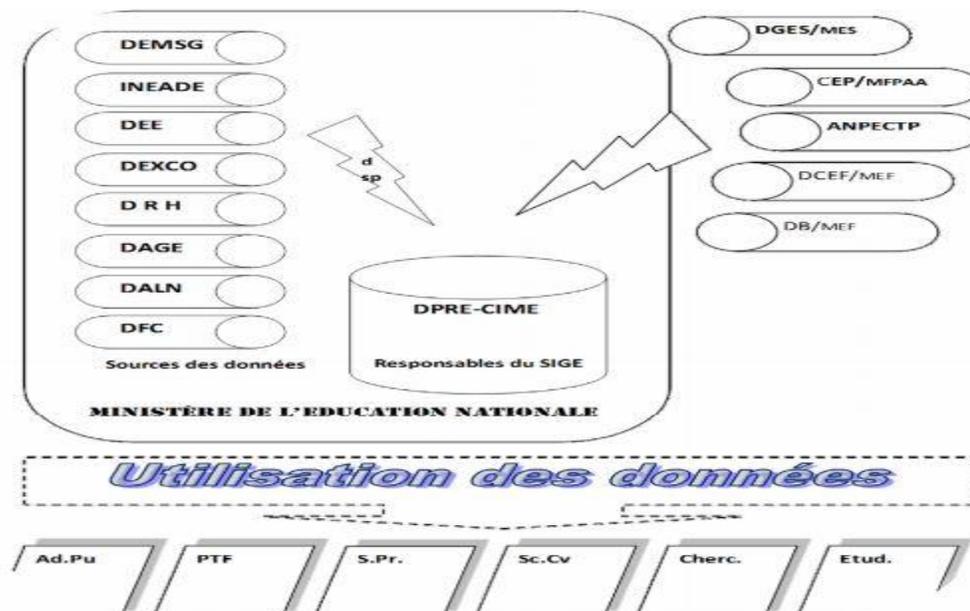


Figure 4: du système d'information de gestion de l'éducation du Sénégal,
Source : système d'information de gestion de l'éducation du Sénégal(SIGE), 2013

L'information au sein du SIGE Sénégalais circule suivant le schéma suivant :

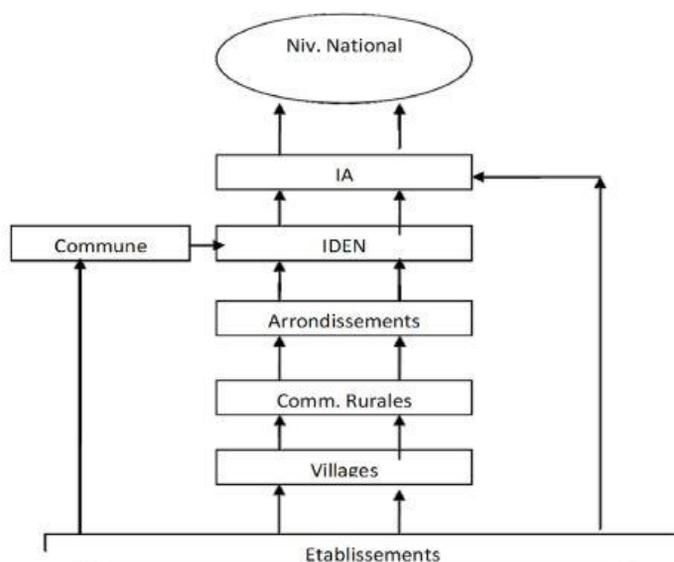


Figure 5: Flux d'information dans le SIGE Sénégalais,
Source : système d'information de gestion de l'éducation du Sénégal, 2013.

2. L'INITIATIVE ETHIOPIENNE : TEACHER'S MANAGEMENT INFORMATION SYSTEM (TMIS)

Les systèmes d'information sont essentiels pour la gestion des enseignants. Les projections de l'offre et de la demande d'enseignants, par exemple, sont un bon moyen d'évaluer le manque d'effectifs, de prévoir une pénurie ou un excédent éventuels de personnels et d'apprécier l'ampleur des défis enseignants d'un pays. Malgré cela, les systèmes d'information pour la gestion des enseignants (*TMIS*) tendent à être négligés dans bien des pays, notamment dans ceux qui sont le plus à la traîne dans la réalisation des objectifs de l'EPT.

L'amélioration de la gestion des instituts de formation des enseignants en même temps que des systèmes d'information pour la gestion des enseignants se trouve donc au cœur de l'initiative CapEFA en Éthiopie. Une estimation des besoins, indispensable pour concevoir le TMIS éthiopien a été menée. CapEFA souhaite contribuer pleinement à ce système dont le but est d'évaluer la répartition des enseignants dans les écoles du pays, en jugeant la répartition des plus qualifiés et en appréciant leurs évolutions de carrière. Des expériences pilotes réalisées avec les personnels régionaux et fédéraux des SIGE en différents points du pays et dans un choix d'établissements ont montré qu'un TMIS améliorerait le suivi de la performance des formations enseignantes, ainsi que l'élaboration des systèmes d'assurance qualité.

Parallèlement, l'IIRCA, l'institut de l'UNESCO, a été chargé de former des gestionnaires des instituts de formation des enseignants aux niveaux tant fédéral que régional. Comme l'ont signalé ces cadres de haut niveau, les activités de développement des capacités facilitées par l'UNESCO leur ont permis de mieux comprendre comment ils gèrent leurs établissements et quelles méthodes ils appliquent. Ils sont, dit-on, mieux au fait des possibilités de financement, se sentent mieux à même de gérer les ressources humaines et ont une meilleure maîtrise de leurs capacités.

3. OPENEMIS : UN SYSTEME D'INFORMATION POUR LA GESTION DE L'EDUCATION.

OpenEMIS est un Système d'Information Open-Source pour la Gestion de l'Education conçu pour recueillir des données sur les écoles, les élèves, et les enseignants. Le système a été initialement conçu par l'UNESCO pour être facilement personnalisable afin de répondre aux besoins spécifiques des systèmes éducatifs des États Membres.

Capable de fonctionner de manière autonome sur des postes de travail aussi bien que via le web ou des terminaux mobiles, OpenEMIS est conçu pour faciliter la collecte, le traitement, l'analyse et la diffusion de données sur les systèmes éducatifs. Avec toutes ces fonctionnalités, OpenEMIS constitue un outil sectoriel souple et rentable conçu pour être facilement et rapidement

adapté aux besoins des producteurs et utilisateurs d'information aux niveaux national et sous-national. Cet outil peut gérer un large éventail d'information, que ce soit les effectifs des élèves, des personnels enseignants et non-enseignants, les classes, les manuels scolaires, les infrastructures, les données financières ou encore les résultats des apprentissages. Afin de répondre aux besoins nationaux, OpenEMIS est en mesure de gérer à la fois des données désagrégées par individus ou bien des tableaux de données agrégées (*de type recensement*) pour les élèves, le personnel enseignant et non-enseignant.

➤ **Objectifs et caractéristiques de OpenEMIS**

L'objectif de OpenEMIS est de renforcer la formulation de politiques et de stratégies de l'éducation fondées sur des faits avérés dans les pays en développement à travers la mise en place aisée d'un Système d'Information sur la Gestion de l'Education abordable, complet, et facile à utiliser. Cet outil vise à améliorer l'information statistique par le recours à un SIGE simple, flexible, et rentable qui soit facilement adaptable aux besoins des producteurs et des utilisateurs de l'information de l'éducation. La flexibilité de OpenEMIS permet son utilisation dans une variété de situations dans les pays en développement (*Ministères de l'Education, Bureau des Statistiques, unités administratives aux niveaux régional, local, et scolaire*).

OpenEMIS est compatible avec DevInfo, le système de base de données approuvé par les Nations Unies pour le suivi des progrès des pays vers les Objectifs du Millénaire pour le Développement et les autres priorités nationales. Ainsi, les données d'éducation saisies dans OpenEMIS peuvent être facilement diffusées via des tableaux, des graphiques, et des cartes qui fourniront aux décideurs des tableaux de bord synthétiques pour identifier rapidement les disparités. L'initiative OpenEMIS est soutenue par l'UNESCO et par une solide équipe de soutien technique équipée pour aider les pays sur tous les aspects de la mise en œuvre au niveau national, y compris la personnalisation, l'intégration, la migration, l'installation, la configuration, et les mises à niveau. Les principales caractéristiques d'OpenEMIS sont les suivantes:

- *Système d'Information sur la Gestion de l'Education (SIGE) open source et libre de redevances*
- *Fonctionne hors ligne sur un ordinateur de bureau, sur un réseau local, ou comme une application cloud hébergée sur un serveur sécurisé OpenEMIS*
- *Entièrement personnalisable pour répondre aux exigences de systèmes éducatifs variés*
- *Prise en charge multilingue*
- *Calcule automatiquement les indicateurs standards de l'éducation et agrège les données*
- *Système de sécurité puissant qui permet aux utilisateurs de créer des rôles accordant des autorisations d'accès à différents modules, différentes zones géographiques, ou différentes institutions*

- *Soutenu par un réseau mondial de spécialistes de l'éducation et de techniciens disponibles par le biais d'un service d'assistance 24/7*

Pour l'atteinte de ses objectifs, OpenEMIS dispose d'outils adaptés à différents paliers de travail:

- *OpenEMIS core : c'est un système d'information au niveau ministériel*
- *OpenEMIS survey : c'est un outil de collecte de données sur l'éducation*
- *OpenEMIS school : c'est un système d'information qui facilite la collecte, le traitement et la gestion de l'information au niveau d'un établissement.*
- *OpenEMIS classroom est une application mobile pour la gestion de la salle de classe et conçue pour l'enseignant afin de faciliter sa gestion de l'élève dans la discipline, le travail scolaire,...*
- *OpenEMIS visualizer est une application mobile qui analyse et contrôle les données sur l'éducation en appui aux chercheurs, décideurs, planificateurs,...*
- *OpenEMIS analyzer*

III. CADRE METHODOLOGIQUE

Il est question dans ce chapitre des procédés utilisés pour la mise sur pied de ce système d'information. Les processus sont nombreux mais échelonnés et suivent une certaine organisation.

1. LA COLLECTE ET LE TRAITEMENT DES DONNEES STATISTIQUES

La nécessité de disposer de données statistiques solides, fiables. Fondées sur les résultats exacts de recensement, de manière à pouvoir mesurer avec précision la taille réelle du capital enseignant ainsi que la disponibilité et le déploiement de celui-ci. L'importance fondamentale des statistiques et la nécessité de disposer d'institutions crédibles et indépendantes chargées de les établir se fait donc ressentir. Les carences observées à l'occasion du recensement effectué, incitent les pouvoirs publics à :

- s'intéresser à la gestion qui est faite des enseignants camerounais depuis leurs affectations dans les établissements scolaires jusqu'au suivi de leur retraite,
- développer les moyens qui permettront de collecter et de produire en temps utile des données précises, tant qualitatives que quantitatives, aux responsables de la planification, de la gestion, de l'évaluation et du suivi.

2. REALISATION DU SYSTEME DE GESTION DES ENSEIGNANTS

Le recensement est le principal mode de collecte qui permet de constituer les bases de données exhaustives sur les produits des actions menées dans le système éducatif et principalement en ce qui concerne le capital enseignant. A travers ce type d'enquête, des informations sur les enseignants sont recueillies des personnes ressources. La mesure des décisions est donc établie à

partir de leur niveau direct d'application, ce qui conforte la validité des données recueillies. Cet inventaire implique d'une part, les responsables des structures scolaires pour le renseignement des fiches statistiques et d'autre part, les Inspections de l'Enseignement secondaire et des Délégations de l'Éducation pour l'organisation pratique au niveau déconcentré. La mise sur pieds de ce système s'est faite en huit étapes à savoir:

(i) Le dispatching des questionnaires

Processus de diffusion des questionnaires et fiches de collecte de données.

(ii) L'entretien direct

Il est question des différentes personnes que nous avons rencontrées pour la compréhension et l'élaboration de ce travail. Il s'agit du personnel du Ministère des enseignements secondaires, des délégations et des inspections des enseignements secondaires qui n'ont pas souhaité être cités.

(iii) La collecte des données

En vue de l'aménagement et du développement des statistiques, nous avons associé les données sur le terrain et les données administratives pour en faire une compilation harmonieuse.

- ❖ Collecte sur le terrain des données dans les établissements scolaires, délégations et inspections.
- ❖ Collecte des données au Ministère. Cette étape a permis de vérifier la conformité des informations reçues par rapport aux fiches renseignées.

(iv) La saisie et la compilation des données

Il est question d'indiquer le logiciel utilisé pour remplir cette étape.

(v) L'apurement des données

L'apurement a essentiellement consisté à une vérification et un traitement des données et à la production des tableaux en vue de l'élaboration de l'annuaire.

(vi) La tabulation, l'analyse, et la production des rapports d'analyse et de l'annuaire.

Les bases de données sous format de fichier Excel conçues lors du traitement des données permettront de produire les rapports d'analyse et l'annuaire statistique.

(vii) Le contenu de l'annuaire statistique

A l'issue de ce travail, cet annuaire statistique sera composé de toutes les informations relatives au capital enseignant qui compose le département de la LEKIE et ce avec des chiffres et données concrètes et fiables et dont la mise à jour se fera de façon automatique à toutes les échelles.

(viii) Les limites

Durant le processus de collecte et de traitement des données, plusieurs problèmes ont été rencontrés. Ce sont entre autres :

- ❖ le manque de moyens financiers et matériels ;
- ❖ le non-respect, par certains établissements, des consignes de remplissage des fiches et des délais fixés pour déposer les fiches de collecte.
- ❖ la mauvaise qualité des fiches remplies (absence de certaines informations, etc.)
- ❖ les informations contenues dans les fiches déposées non désagrégées.

D'où un taux de couverture assez faible qui altère par conséquent l'exhaustivité des données dans le document.

CHAPITRE 2

METHODOLOGIE

CHAPITRE 2 : METHODOLOGIE

Ce chapitre fera l'objet de la présentation de la méthodologie adoptée pour la réalisation de ce travail. La première section portera sur les méthode et modèle d'analyse et de conception du système. Dans la deuxième section il sera question du modèle de développement du système ; ce qui permettra aisément d'aborder la question du type d'architecture du système en troisième point. La dernière section, pour sa part, dévoilera les outils de développement du SETEMIS.

I. METHODE, MODELE D'ANALYSE ET DE CONCEPTION DU SYSTEME

Les méthodes d'analyse et de conception fournissent un modèle, une méthodologie et des notations standards qui aident à concevoir des logiciels de qualité. Elles sont regroupées suivant les critères composition et décomposition, et fonctionnel et orienté objet. Le choix d'une méthode d'analyse impose conséquemment un modèle de conception ou langage de modélisation.

Il existe plusieurs types de modélisation des systèmes d'information, nous pouvons citer : la modélisation structurée, la modélisation systémique et la modélisation orientée-objet. Chaque technique de modélisation utilise un langage de modélisation bien approprié. Un langage de modélisation est un ensemble de concepts et de règles permettant de construire des modèles décrivant les systèmes d'information.

1. LA METHODE D'ANALYSE ET DE CONCEPTION : L'APPROCHE ORIENTEE OBJET

L'objet est le cœur de cette approche. Tout objet donné possède deux caractéristiques : son état courant (*attributs*) et son comportement (*méthodes*). En approche orientée objet, on utilise le concept de classe qui permet de regrouper des objets de même nature. Une classe est un prototype qui permet de définir les attributs (*champs*) et les méthodes (*comportement*) à tous les objets de cette classe. L'avantage majeur de cette approche est la réutilisabilité des éléments (*objets*), utiliser un objet de base afin de produire un autre qui peut être une amélioration de cet objet d'où le phénomène d'héritage. L'approche objet permet de :

- Faciliter la réutilisation du code, encapsulation et abstraction
- Faciliter de l'évolution du code
- Améliorer la conception et la maintenance des grands systèmes
- Effectuer une programmation par « composants », (*Conception d'un logiciel à la manière de la fabrication d'une voiture*).

➤ **Différence avec l'approche systémique**

L'approche systémique perçoit le système d'information du point de vue de ses relations avec d'une part le système opérationnel de l'organisation et d'autre part son système de pilotage. MERISE est la méthode la plus connue ; elle s'inspire de la théorie des systèmes et part du général au particulier : c'est à dire, commence par comprendre l'organisation, puis délimite le domaine, ensuite conçoit le projet pour enfin réaliser l'application.

➤ **Différence avec l'approche structurée**

L'approche structurée est le résumé de la méthode cartésienne et de l'approche fonctionnelle : elle assimile le système d'information à une fonction globale de gestion et s'appuie sur les concepts et techniques de décomposition hiérarchiques. Cette approche utilise une démarche arborescente pour l'analyse et la conception des systèmes d'information. Elle met en pratique une technique axée sur les fonctions à remplir par le système et les différents rôles à jouer. La conception est alors assimilée à l'analyse des fonctions du système. La modélisation utilise un procédé descendant « *top-down* », du haut vers le bas, qui va du général au particulier. Ce qui revient à fragmenter le système en sous-système et le présenter sous forme d'un arbre généalogique. L'analyse et la conception reposent ainsi sur une technique de raffinements successifs suivant le principe d'abstraction. Le modèle courant ici est SADT (*Structured Analysis Design Technic*). Il permet non seulement de décrire les tâches du projet et leurs interactions, mais aussi de décrire le système que le projet vise à créer ou modifier, en mettant notamment en évidence les parties qui constituent le système, la finalité et le fonctionnement de chacune, ainsi que les interfaces entre ces diverses parties.

2. LE LANGAGE DE MODELISATION DU SYSTEME : UML

La description de la programmation par objets fait ressortir l'étendue du travail conceptuel nécessaire : définition des classes, de leurs relations, des attributs et méthodes, des interfaces etc. la programmation d'une application nécessite préalablement l'agencement des idées, de la documentation, puis une organisation claire de la réalisation en définissant les modules et étapes de celle-ci. Cette démarche antérieure à l'écriture, modélisation, produit un modèle qui tient lieu de référence. UML n'est pas une méthode (*i.e. une description normative des étapes de la modélisation*) mais un métalangage car il fournit les éléments permettant de construire le modèle qui, lui sera le langage du projet. Merise et UML sont les deux grands principes de « *traduction* »

ou modélisation d'un système d'information. Néanmoins, ils ne sont pas aussi proches qu'on pourrait le penser. Le choix de l'un ou de l'autre se fait selon trois critères: l'accessibilité, la précision et l'exploitabilité.

- ❖ Parlant de l'accessibilité, MERISE présente l'intérêt d'avoir des modèles logiques moins détaillés, mais facilement compréhensibles par un utilisateur moins avisé. Tandis qu'UML est conçu pour s'adapter à n'importe quel langage de programmation orientée objet (*POO*), et présente plusieurs modèles (*diagrammes*) dont la compréhension nécessite une grande attention.
- ❖ En ce qui concerne la précision, MERISE est inapproprié. Malgré sa clarté, il manque de précision du fait qu'il s'éloigne du langage et par conséquent difficile à implémenter alors qu'UML intègre les éléments communs des différents langages, sa volonté étant d'être fidèle à la réalisation finale. Toutefois, UML est beaucoup plus complet avec ses différents diagrammes.
- ❖ Pour ce qui est de l'exploitabilité, MERISE est une méthode plus généraliste. Elle donne une vue globale de la solution sans entrer dans les petits détails. Contrairement à UML qui est conçu pour l'implémentation objet avec ses différents détails et sa portabilité (*s'adapte à n'importe quelle plateforme*) elle est donc plus exploitable.

L'une ou l'autre présente des avantages et des inconvénients. Si on cherche la précision et l'exploitabilité, UML se trouve être le meilleur choix. La conception de cette application demandant bien une grande précision et une exploitabilité maximale, c'est la raison pour laquelle UML a été retenu.

II. LE MODELE DE DEVELOPPEMENT DU SYSTEME

Dans le souci de réaliser un système suivant une méthodologie claire, et qui soit adaptée à la fois aux bénéficiaires et au concepteur, des modèles de cycle de vie définissant les étapes du développement du logiciel ont été mis au point. A la fin de chaque étape, des documents spécifiques sont produits afin de valider chacune d'elle avant de passer à l'étape suivante. Les modèles de développement ont pour objectifs de:

- Organiser les différentes phases du cycle de vie pour l'obtention d'un logiciel fiable, adaptable et efficace
- Guider le développeur dans ses activités techniques
- Fournir des moyens pour gérer le développement et la maintenance (*ressources, délais, avancement, etc.*)

1. LE MODELE DE DEVELOPPEMENT EN V

Le modèle de cycle en V a été conçu pour pallier au problème de réactivité du modèle en cascade qui permet, en cas d'anomalie, de limiter un retour aux étapes précédentes. Les phases de la partie montante doivent renvoyer de l'information sur les phases en vis-à-vis (*en face*) lorsque des défauts sont détectés afin d'améliorer le logiciel. Le modèle de cycle de vie en V part du postulat que les procédures de vérification de la conformité du logiciel aux spécifications doivent être élaborées dès les phases de conception. Il s'agit d'un modèle dans lequel le développement des tests et du logiciel sont effectués de manière synchrone. Le principe de ce modèle est qu'avec toute décomposition doit être décrite la recombinaison et que toute description d'un composant est accompagnée de tests qui permettront de s'assurer qu'il correspond à sa description. Ce qui rend explicite la préparation des dernières phases (*validation-vérification*) par les premières (*construction du logiciel*), et permet ainsi d'éviter un écueil bien connu de la spécification du logiciel : énoncer une propriété qu'il est impossible de vérifier objectivement après la réalisation. Cependant, ce modèle souffre toujours du problème de la vérification trop tardive du bon fonctionnement du système.

D'un point de vue technique, le cycle en V permet de limiter le scope de validation de chaque étape, en ne regardant la conformité de celle-ci « *que* » par rapport à la précédente sur le V. On peut avoir, « *en cascade* », des reprises à faire dans l'étape précédente ; dans ce cas on fera alors attention de reproduire cela d'abord sur les étapes précédentes (*de bas en haut*), puis, de haut en bas, sur les étapes correspondantes dans la branche de droite, et enfin, vérifier de bas en haut la compatibilité des changements avec le reste de l'avancement des travaux.

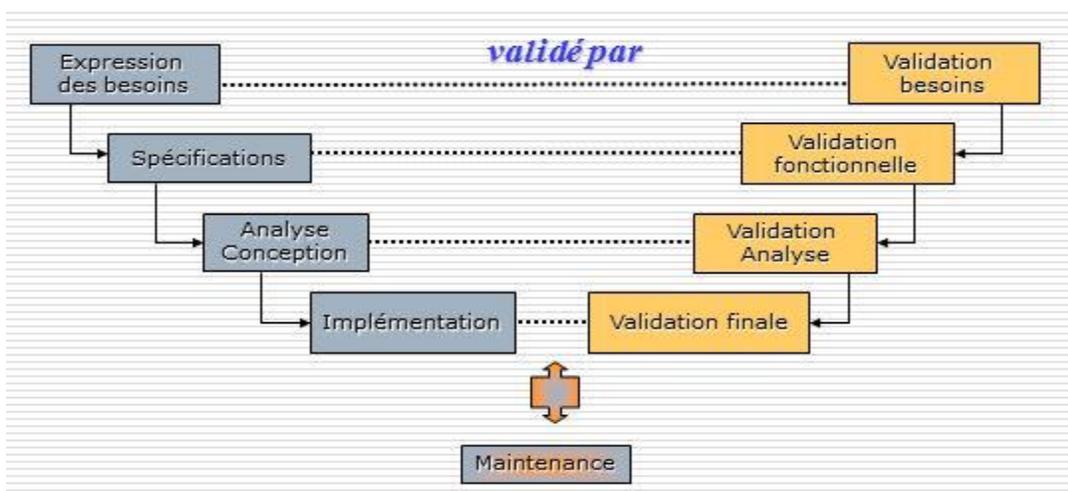


Figure 6 : modèle de développement en V.
Source : support de cours de génie logiciel ENS, 2015.

2. LE MODELE DE DEVELOPPEMENT EN CASCADE

Le modèle de cycle de vie en cascade définit successivement des phases séquentielles à l'issue desquelles des documents sont produits pour en vérifier la conformité avant de passer à la phase suivante. Dans ce modèle, le principe est très simple : chaque phase se termine à une date précise par la production de certains documents ou logiciels. Les résultats sont définis sur la base des interactions entre étapes, ils sont soumis à une revue approfondie et on ne passe à la phase suivante que s'ils sont jugés satisfaisants. Le modèle original a été revisité et la possibilité de retour en arrière a été rajoutée ultérieurement sur la base qu'une étape ne remet en cause que l'étape précédente, ce qui, dans la pratique, s'avère insuffisant.

Le principal avantage de ce modèle est que le planning est établi à l'avance et le maître d'ouvrage sait précisément ce qui va lui être livré et quand il pourra en prendre livraison.

Comme inconvénients on peut relever une très faible tolérance à l'erreur ainsi que la vérification du bon fonctionnement du système qui est réalisée trop tardivement : lors de la phase d'intégration, ou pire, lors de la mise en production ce qui induit automatiquement un coût important en cas d'anomalie.

3. LE MODELE DE DEVELOPPEMENT EN SPIRALE

Ce modèle, beaucoup plus général que le modèle en V, met l'accent sur l'activité d'analyse des risques ; chaque cycle de la spirale se déroule en quatre phases :

- détermination, à partir des résultats des cycles précédents, ou de l'analyse préliminaire des besoins, des objectifs du cycle, des alternatives pour les atteindre et des contraintes
- analyse des risques, évaluation des alternatives et, éventuellement maquettage
- développement et vérification de la solution retenue, un modèle « classique » (cascade ou en V) peut être utilisé ici
- revue des résultats et vérification du cycle suivant.

L'analyse préliminaire est affinée au cours des premiers cycles. Le modèle utilise des maquettes exploratoires pour guider la phase de conception du cycle suivant. Le dernier cycle se termine par un processus de développement classique. Le but premier de ce modèle étant la gestion des risques, ceux-ci sont logiquement limités. L'expertise du client croit à chaque itération du cycle et l'apprentissage se fait par couche et pas d'un seul bloc. Enfin, ce modèle est très adaptatif : si chaque prototype apporte des fonctionnalités indépendantes, il est possible de changer l'ordre de livraison des versions. Ce qui représente des avantages non négligeables.

Toutefois, le principal défaut du cycle en spirale est qu'il n'est adapté qu'aux projets importants. De plus, l'évaluation des risques en elle-même et la stricte application du cycle de

développement peuvent engendrer plus de coûts que la réalisation du logiciel. Enfin, ce type de cycle de développement est complexe, entre les étapes prévues en théorie et celles mises en pratique il y a une grande différence.

4. LE MODELE DE DEVELOPPEMENT PAR INCREMENT

Dans les modèles précédents un logiciel est décomposé en composants développés séparément et intégrés à la fin du processus. Contrairement au modèle en V, le modèle par incrément développe un seul ensemble de composants à la fois ; des incréments viennent s'intégrer à un noyau de logiciel développé au préalable. Chaque incrément est par la suite développé selon l'un des autres modèles (*spirale, cascade, ...*). Les avantages de ce type de modèle sont les suivants :

- chaque développement est moins complexe ;
- les intégrations sont progressives ;
- il est ainsi possible de livrer et de mettre en service chaque incrément ;
- il permet un meilleur lissage du temps et de l'effort de développement grâce à la possibilité de recouvrement (parallélisme) des différentes phases.

Les risques de ce modèle sont :

- remettre en cause les incréments précédents ou pire le noyau ;
- ne pas pouvoir intégrer de nouveaux incréments.
- Le modèle incrémental est risqué car il ne donne pas beaucoup de visibilité sur le processus complet

Les noyaux, les incréments ainsi que leurs interactions doivent donc être spécifiés globalement, au début du projet. Les incréments doivent être aussi indépendants que possibles, fonctionnellement mais aussi sur le plan du calendrier du développement.

Au vue des caractéristiques des différents modèles, le choix s'est porté sur le modèle de développement en V. En effet, l'énorme intérêt du cycle en V est qu'il est un excellent support à la formalisation des relations entre le client et l'équipe de développement. Il oblige le client à réfléchir aux différents aspects de sa demande. La phase de spécification permet à l'équipe de développement de vérifier qu'elle a bien compris la demande du client qui lui se doit de valider la spécification. Tel que présenté, le cycle en V ne permet de lever les interrogations technologiques tard. On ajoute souvent en parallèle de la phase de spécification, une phase de validation technologique afin de lever les écueils techniques (*performance, difficulté algorithmique spécifique, ...*). Dans le cas d'un développement web ou d'une interface graphique, la spécification est accompagnée d'une maquette graphique. Le cycle en V, très critiqué dans son application, est une base pédagogique très importante. Ses concepts de bases se retrouvent dans tous les autres cycles de développement.

III. L'ARCHITECTURE DU SYSTEME

L'étude d'une application amène à identifier le rôle du système d'exploitation, éventuellement associé au logiciel navigateur, en tant qu'environnement d'exécution. Celui-ci est à même de prendre en charge le stockage et la sécurité des fichiers, l'hébergement et l'exécution des applications et des services, le dialogue avec l'utilisateur via une interface graphique ou textuelle. Les moteurs d'exécution permettent l'exécution d'applications qui s'appuient sur un composant logiciel spécifique (*machine virtuelle*). L'analyse de la logique de fonctionnement d'une application permet de repérer ses niveaux fonctionnels : présentation (*interface utilisateur, logique de présentation*), traitements ou logique applicative (*prise en charge des règles de gestion*), données (*accès et gestion des données*). Pour visualiser l'architecture d'un logiciel, le génie logiciel utilise les patrons ou motifs de conception. Ils décrivent des solutions standard pour résoudre des problèmes d'architecture et de conception des logiciels. On peut considérer un patron de conception comme une formalisation de bonnes pratiques, ce qui signifie que le privilège est accordé aux solutions éprouvées. Il n'est pas question de code ici, mais d'une manière formalisée de résoudre un problème donné.

1. L'ARCHITECTURE 3 TIERS.

L'architecture trois tiers est un modèle en couches, c'est-à-dire, que chaque couche communique seulement avec ses couches adjacentes (*supérieures et inférieures*) et le flux de contrôle traverse le système de haut en bas. Les couches supérieures contrôlent les couches inférieures, c'est-à-dire que les couches supérieures sont toujours sources d'interaction (*clients*) alors que les couches inférieures ne font que répondre à des requêtes (*serveurs*).

Dans le modèle MVC, il est généralement admis que la vue puisse consulter directement le modèle (*lecture*) sans passer par le contrôleur. Par contre, elle doit nécessairement passer par le contrôleur pour effectuer une modification (*écriture*). Ici, le flux de contrôle est inversé par rapport au modèle en couches, le contrôleur peut alors envoyer des requêtes à toutes les vues de manière qu'elles se mettent à jour. Dans l'architecture trois tiers, si une vue modifie les données, toutes les vues concernées par la modification doivent être mises à jour, d'où l'intérêt de l'utilisation du MVC au niveau de la couche de présentation. Celle-ci permet alors d'établir des règles du type « *mettre à jour les vues concernant X si Y ou Z sont modifiés* ». Mais ces règles deviennent rapidement trop nombreuses et ingérables si les relations logiques sont trop élevées. Dans ce cas, un simple rafraîchissement des vues à intervalle régulier permet de surmonter aisément ce problème. Il s'agit

d'ailleurs de la solution la plus répandue en architecture trois tiers, l'utilisation du MVC étant moderne et encore marginale.

2. L'ARCHITECTURE OBSERVATEUR/OBSERVABLE

Les observables envoient des notifications à leurs observateurs en cas de changement. En cas de notification, les observateurs exécutent leur méthode *update* ; ils peuvent obtenir des informations complémentaires par l'appel à la méthode *getState* de l'observable. Les objets de base de ce modèle sont le sujet et l'observateur. Un sujet peut avoir un nombre quelconque d'observateurs sous sa dépendance. Tous les objets reçoivent une notification chaque fois que le sujet subit une modification de son état. En réponse, chaque observateur interrogera le sujet sur son état afin d'y adapter le sien propre. Ce type d'interaction est également connu sous le nom de *Diffusion – Souscription*. Dans ce modèle :

- Le sujet diffuse les notifications
- Il les expédie, sans avoir à connaître ses observateurs
- Des observateurs, en nombre quelconque, peuvent souscrire pour recevoir les notifications

D'après le patron de conception observateur/observable, la vue est un « *observateur* » du modèle qui est lui « *observable* ».

3. L'ARCHITECTURE MVC

Le Modèle-Vue-Contrôleur(en abrégé MVC, de l'anglais *Model-View-Controller*) est une architecture et une méthode de conception logicielle qui impose la séparation entre les données, les traitements et la présentation. L'organisation d'une interface graphique est délicate. L'architecture "MVC" ne prétend pas en éliminer tous les problèmes, mais fournit une certaine approche dans cet objectif. Offrant un cadre normalisé pour structurer une application, elle facilite aussi le dialogue entre les concepteurs ; ce modèle est utilisé uniquement pour la couche présentation (*côté serveur*) dans l'architecture d'une application, la couche métier n'étant en aucun cas concernée. Ce paradigme regroupe les fonctions nécessaires en trois catégories :

- un modèle (*modèle de données*) ;
- une vue (*présentation, interface utilisateur*) ;
- un contrôleur (*logique de contrôle, gestion des événements, synchronisation*).

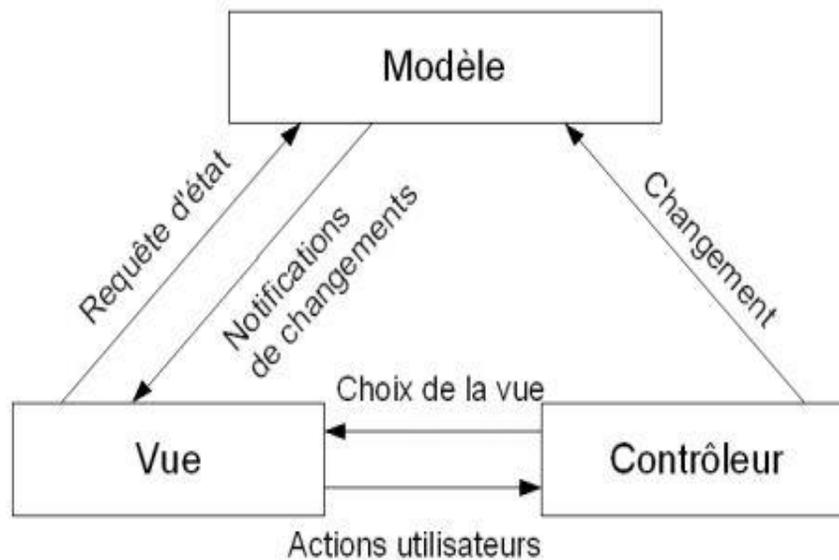


Figure 7: Schéma du modèle MVC.

Source : IUT Bordeaux 1 2013-2014 Département Informatique OMGL3 ACSI Objet

❖ Le modèle

Le modèle représente le comportement de l'application : traitements des données, interactions avec la base de données, etc. Il peint les données manipulées par l'application, assure la gestion de ces données et en assure l'intégrité. Dans le cas précis d'une base de données, c'est le modèle qui la contient. Le modèle offre des méthodes pour mettre à jour ces données (*insertion, suppression, changement de valeur*). Il offre aussi des méthodes pour récupérer ces données. Les résultats renvoyés par le modèle ne s'occupent pas de la présentation. Le modèle ne présente aucun lien direct vers le contrôleur ou la vue, sa communication avec cette dernière s'effectue au travers du patron observateur. Le modèle peut autoriser plusieurs vues partielles des données. Par exemple, le programme manipulant une base de données sur les enseignants d'un arrondissement, le modèle développe des méthodes pour avoir tous les enseignants d'un lycée, tous les enseignants d'une matière (*français, anglais, droit, ...*), le pourcentage d'enseignant féminin ou masculin du lycée.

❖ La vue

La vue correspond à l'interface avec laquelle l'utilisateur interagit. Sa première tâche consiste à présenter les résultats renvoyés par le modèle. Sa seconde tâche est de recevoir les différentes actions de l'utilisateur (*hover, clic de souris, sélection d'un bouton, sélection d'une case, entrée de*

texte, de mouvements, de voix, etc.). Ces différents événements sont envoyés au contrôleur. La vue n'effectue pas de traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle et d'interagir avec l'utilisateur. Plusieurs vues peuvent afficher des informations partielles ou non d'un même modèle. Par exemple dans le cas de cette application, le volume d'enseignants titulaires d'un DIPES I et d'un DIPET I peut être affiché de multiples façons (*en texte sous forme de liste numérotée, avec des boutons à cocher, avec des curseurs ou encore sous forme de graphique statistique*). La vue peut aussi offrir à l'utilisateur la possibilité de changer de vue. Ce qui permet une certaine récursivité du modèle.

❖ **Le contrôleur**

Il prend en charge la gestion des événements pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle et ensuite avertit la vue que les données ont changé pour qu'elle se mette à jour. Certains événements de l'utilisateur ne concernent pas les données mais la vue. Dans ce cas, le contrôleur demande à la vue de se modifier. Le contrôleur n'effectue aucun traitement, ne modifie aucune donnée : il analyse la requête du client et se contente d'appeler le modèle adéquat et de renvoyer la vue correspondant à la demande. Par exemple, dans le cas d'une base de données gérant les professeurs d'un arrondissement, une action de l'utilisateur peut être l'entrée (*saisie*) d'un nouvel enseignant. Le contrôleur ajoute cet enseignant au modèle et demande sa prise en compte par la vue. Une action de l'utilisateur peut aussi être de sélectionner une nouvelle personne pour visualiser tous ses enseignants. Ceci ne modifie pas la base des enseignants mais nécessite simplement que la vue s'adapte et offre à l'utilisateur une vision des enseignants de cet arrondissement.

Lorsqu'un même objet contrôleur reçoit les événements de tous les composants, il lui faut déterminer l'origine de chaque événement. Ce tri des événements peut s'avérer fastidieux et peut conduire à un code assez touffu (*un énorme switch*). C'est pourquoi le contrôleur est souvent scindé en plusieurs parties (*sous contrôleurs*) chacune assurant la gestion des événements des composants dont elle a la charge.

V. LES OUTILS DE DEVELOPPEMENT

1. LE SYSTEME DE GESTION DE BASES DE DONNEES (SGBD)

La gestion et l'accès à une base de données sont assurés par un ensemble de programmes qui constituent le *Système de gestion de base de données (SGBD)*. Un SGBD doit permettre la création,

l'ajout, la modification et la recherche de données. Il héberge généralement plusieurs bases de données, qui sont destinées à des logiciels ou des thématiques différents. La plupart des SGBD fonctionnent selon un mode client/serveur. Le serveur (*sous-entendu serveur de base de données qui stocke les données*) reçoit des requêtes de plusieurs clients (*sous-entendu logiciel client*) et ceci de manière concurrente. Le serveur analyse la requête, la traite et retourne le résultat au client. En gérant les requêtes des utilisateurs, le SGBD garantit l'intégrité des données (*s'assure qu'elles soient toujours disponibles et qu'elles soient organisées de façon cohérente*) ainsi que leur sécurité (*seuls ceux disposant des privilèges d'accès peuvent accéder aux données*). Les autres principaux objectifs d'un SGBD sont l'indépendance des programmes aux données, l'accès aux données, l'administration centralisée des données, la non redondance des données, le partage des données, la résistance aux pannes. Quelque soit le modèle, un des problèmes fondamentaux à prendre en compte est la *cohérence des données*.

MySQL est un SGBDR (*Système de Gestion de Base de Données Relationnelle*) qui utilise le langage *SQL*. Le langage *SQL* sera utilisé par tous ceux (*manuellement ou par un outil*) travaillant sur la base de données (*administrateur, développeur, utilisateur*). Le langage procédural de MySQL permet d'incorporer nativement tout ordre *SQL* dans un programme. Ce serveur de bases de données relationnelles *SQL* fonctionne sur de nombreux systèmes d'exploitation (*dont Linux, Windows,...*) et est accessible en écriture par de nombreux langages de programmation, incluant notamment *PHP, Java,...* L'une des spécificités de MySQL est qu'il inclut plusieurs moteurs de bases de données et qu'il est par ailleurs possible au sein d'une même base de définir un moteur différent pour les tables qui composent la base. Cette technique est astucieuse et permet de mieux optimiser les performances d'une application. La réplication est possible avec MySQL et permet ainsi de répartir la charge sur plusieurs machines, d'optimiser les performances ou d'effectuer facilement des sauvegardes des données. Etant un SGBD relationnel, il permet d'organiser, de gérer et d'exploiter des données enregistrées dans des tables grâce au maintien de «relations» entre celles-ci. C'est un logiciel léger et peu gourmand en ressource et espace mémoire contrairement à Oracle. Il est très utilisé avec le langage de création de pages web dynamiques (*PHP*), ainsi que pour le déploiement de site *web dynamique*.

➤ **Différence avec PostgreSQL**

PostgreSQL est un système multiplateforme connu pour son comportement stable et pour être très respectueux des normes ANSI *SQL*. Il apporte une puissance additionnelle substantielle en incorporant les quatre concepts de base suivants afin que les utilisateurs puissent facilement étendre

le système : classes, héritage, types, fonctions. D'autres fonctionnalités accroissent la puissance et la souplesse : contraintes, déclencheurs, règles, intégrité des transactions. Ces fonctionnalités placent PostgreSQL dans la catégorie des bases de données relationnelles objet mais bien que PostgreSQL possède certaines fonctionnalités orientées objet, il appartient avant tout au monde des SGBDR. L'une des principales qualités de PostgreSQL est d'être un logiciel libre, c'est-à-dire gratuit et dont les sources sont disponibles. Il est possible de l'installer sur les systèmes Unix/Linux et Win32. PostgreSQL fonctionne selon une architecture client/serveur, il est ainsi constitué :

- d'une partie serveur, c'est-à-dire une application fonctionnant sur la machine hébergeant la base de données (*le serveur de bases de données*) capable de traiter les requêtes des clients ;
- d'une partie client (*psql*) devant être installée sur toutes les machines nécessitant d'accéder au serveur de base de données (*un client peut éventuellement fonctionner sur le serveur lui-même*).

➤ **Différence avec Oracle**

Oracle n'est pas un SGBDR optimisé pour de petites bases de données, ou appliqué sur de petits volumes de traitements (*2 Go par exemple*) et pour peu d'utilisateurs (*une trentaine*). Oracle dispose des assistants performants via Oracle Manager Server, qui ont la possibilité de gérer en interne des tâches et des alarmes. Des réglages fins sont possibles dans la mesure où connaissant suffisamment le moteur, presque tout est paramétrable. L'accès aux données système de fait via des vues, plus facilement manipulables que des procédures stockées. Oracle aussi des services Web, un support XML de même qu'un ordonnanceur intégré. Toutefois, il se révèle défaillant sur de nombreux points : un coût exorbitant, tant au point de vue des licences que des composants matériels (*RAM, CPU*) à fournir pour de bonnes performances. Fort demandeur de ressources, Oracle est bien plus gourmand en espace disque et ressource mémoire que ses concurrents, ce qui implique un investissement matériel supplémentaire non négligeable. La connexion utilisateur nécessite par exemple près de 700 Ko/utilisateur, contre une petite centaine sur des serveurs MS-SQL. L'on retrouve une quantité de bugs proportionnelle à la richesse fonctionnelle, surtout sur les dernières versions et une gestion irrégulière des rôles et privilèges (*pas possible de donner des droits sur des schémas particuliers sans passer par leurs objets, désactivation des rôles lors d'exécution de packages...*). De plus, elle est sujette à de nombreuses failles de sécurité liées à l'architecture elle-même.

➤ **Différence avec Access**

Access est aussi bien un outil grand public que professionnel, selon les besoins qu'on a. Il est assez performant en tant que SGBD allié à un outil de développement intégré qui en facilite l'utilisation. Access peut, en tant qu'outil de développement, être utilisé conjointement avec un véritable Serveur de base de données SQL pour bénéficier des avantages du Client/serveur, sous certaines conditions. Seulement, Access en tant que format de données n'est pas un SGBD client/serveur mais seulement un SGBD fichier. Le trafic qu'il génère sur le réseau en utilisation réseau multiposte peut fortement perturber ses performances. Les performances chutent rapidement lorsque plusieurs utilisateurs sont connectés ou si la base dépasse une certaine taille (*100000 lignes*). Cependant Access en tant qu'outil de développement peut être utilisé conjointement avec un véritable Serveur de base de données SQL pour bénéficier des avantages du Client/serveur. MS-Access est recommandé si l'on souhaite avoir une base de donnée de petite taille et facilement gérable.

Le système manipulant une grande quantité de données, il est judicieux de faire usage d'un SGBD capable de les accueillir et qui pourra faciliter l'insertion, l'ajout, bref la manipulation de celles-ci en limitant les risques de fuite ni de perte de données. Ce qui justifie le choix qui s'est porté sur MySQL.

2. LE LANGAGE DE PROGRAMMATION

La mise sur pied de l'application a nécessité l'utilisation de plusieurs langages qui cohabitent aisément et contribuent à l'harmonie et l'unité du système.

➤ **Le PHP**

PHP est un langage interprété indépendant de la plate-forme d'exécution et qui s'exécute sur le serveur et qui permet de générer les pages web. Il est un langage de script utilisé pour illustrer le mécanisme de construction dynamique des pages web. Il est principalement utilisé pour écrire des scripts de collecte de données issues de formulaires, de stockage éventuel de ces données dans une base de données, de génération dynamique de contenu. Ce langage est à même de communiquer et d'exploiter des données gérées par MySQL ; les deux formant ainsi une combinaison appropriée aux applications utilisant Internet et ses technologies comme c'est le cas pour celle-ci. Une autre caractéristique de PHP est qu'il permet facilement l'intégration de code HTML, Javascript, CSS. L'intérêt d'un langage de script tient à ce qu'il permet une certaine interactivité de même qu'il autorise un certain contrôle des interactions qui seraient trop lourd pour le serveur. Il est facilement intégré au HTML et représente la partie créative et dynamique du document HTML envoyé par le

serveur et que le navigateur transforme en page web. PHP a des avantages certains. Le client n'a pas accès au code source puisque celui-ci est interprété avant envoi (*Javascript*); de ce fait, il ne reçoit que le résultat de l'exécution du script. Le code n'est pas alourdi par des commandes destinées à générer la page HTML (*CGI*) et le serveur HTTP peut être configuré pour masquer complètement la qualité dynamique des pages. PHP a à sa disposition de nombreuses bibliothèques/modules pour: le calcul mathématique, la création dynamique d'images, la gestion de sessions, les connexions sécurisées (*SSL*), l'accès à la plupart des SGBD, l'accès aux bases LDAP ... de plus il offre une portabilité certaine. Toutefois, il rencontre des problèmes du point de vue de la rapidité et de la vitesse; pas aussi simple que *ASP-VBScript*, il est plus respectueux des standards web et est plus facilement extensible. PHP ne permet pas d'interactivité au niveau du client mais cette tare est vite rattrapée par son association à Javascript.

➤ **Le HTML**

HTML est d'un langage de description (et non pas d'un langage de programmation) qui permet de décrire l'aspect d'un document, d'y inclure des informations variées (textes, images, sons, animations etc.) et d'établir des relations cohérentes entre ces informations grâce aux liens hypertextes. Son rôle est d'organiser et de gérer le contenu d'une page web. HTML inclut des éléments qui permettent la communication entre un client web et un serveur capable de recevoir des informations sur ce client, de les stocker et/ ou d'effectuer des traitements indépendants. La réalisation de sites web et la création de pages web dynamiques exigent l'effectivité de cette communication. Les avantages du langage HTML sont nombreux ; peu coûteux, (*un simple éditeur de texte suffit à écrire ses premiers documents HTML*) et relativement facile à aborder, il représente en outre un bon moyen de dépasser les problèmes de compatibilité entre des systèmes et des formats informatiques différents.

➤ **Le CSS**

Pour créer un site web, on doit donner des instructions à l'ordinateur. Il ne suffit pas simplement de taper le texte qui devra figurer dans le site (*comme on le ferait dans un traitement de texte Word, par exemple*), il faut aussi indiquer où placer ce texte, insérer des images, faire des liens entre les pages, etc. encore appelé *feuilles de style*, CSS est un langage qui permet de réaliser le design et l'apparence de la page web (*agencement, décoration, positionnement, couleurs, taille du texte, harmonie, ...*). Il vient ainsi en complément au HTML qui lui se charge de l'organisation du contenu car tandis que HTML définit le contenu, CSS lui se charge de la

présentation (*mise en forme et mise en page*) de ce contenu. On comprend donc que CSS nécessite une page HTML pour fonctionner.

➤ **Le JavaScript**

JavaScript est un langage de programmation permettant de développer du « Web Dynamique » et qui peut être inclus dans des pages HTML destinées aux navigateurs WWW les plus courants. Grâce à ce langage, il est possible d'écrire des pages interactives. Contrairement à l'utilisation du CGI, les réponses aux événements déclenchés par le lecteur de la page se font au niveau du navigateur, sans nécessité d'une transmission vers le serveur d'où provient la page : le code JavaScript est exécuté par le client et non par le serveur. JavaScript (*JS*) s'utilise dans le contexte du Web principalement au travers d'un navigateur Internet et permet l'exécution de code informatique intégré à des pages Web. Les principales caractéristiques de JavaScript relèvent du fait qu'il est interprété, faiblement typé et sans notion de pointeurs. Il manipule des types de base simples (*nombre, chaîne et booléen*), les fonctions et les variables peuvent être déclarées n'importe où et admettent un nombre de paramètre variable. Ce langage est indépendant du matériel et donc portable, il est en général intégré dans d'autres logiciels (*navigateurs*) qui fournissent des données sous forme d'objets utilisés par les scripts JavaScript. Le réel intérêt de JS tient de son étroite liaison avec les navigateurs Internet ; il peut effectuer le contrôle du contenu et de l'apparence de page Web, contrôle de fonctionnalités du Navigateur Internet, la gestion du contenu d'une page Web, gestion d'interactions avec l'utilisateur, manipulation et gestion d'images. JavaScript ne doit pas être confondu avec Java, qui lui est un langage beaucoup plus complexe permettant de créer des applications autonomes.

CHAPITRE 3

CONCEPTION ET IMPLEMENTATION DU

SYSTEME

CHAPITRE 3 : CONCEPTION ET IMPLEMENTATION DU SYSTEME.

Il est impossible de donner une représentation graphique complète d'un logiciel, ou de tout autre système complexe, de même qu'il est impossible de représenter entièrement et de façon détaillée un monument où une statue par des photographies. Mais il est possible de donner sur un tel système des vues partielles, chacune semblable à une photographie d'une statue, et dont la combinaison donnera une idée utilisable en pratique sans risque d'erreur grave. Le modèle UML permet de parvenir à ce résultat et ce par le biais de diagrammes dont le rôle est d'utiliser autant de vues distinctes pour représenter des concepts particuliers du système d'information. UML propose plusieurs diagrammes réparti en deux grands groupes : les diagrammes structurels ou statiques et les diagrammes comportementaux ou dynamiques. Ces diagrammes, d'une utilité variable selon les cas, ne sont pas nécessairement tous produits à l'occasion d'une modélisation. Ainsi, pour la conception de ce système nous avons choisi pour chaque groupe quelques-uns notamment les diagrammes de cas d'utilisation, de classes, de séquence et de déploiement à travers lesquels il est possible de montrer les différentes interactions ayant cours dans le système et de formaliser les contraintes de la réalisation et la solution technique. Les deux premières parties de ce chapitre présenteront successivement les diagrammes structurels et les diagrammes dynamiques tandis que la troisième partie, elle portera sur le rendu de ce travail de conception et de construction du système : l'implémentation.

I. CONCEPTION ET MODELISATION STATIQUE

1. LES DIAGRAMMES DE CLASSE

Le diagramme de classes est généralement considéré comme le plus important dans un développement orienté objet. Il représente l'architecture conceptuelle du système : il décrit les classes utilisées par le système ainsi que leurs liens, que ceux-ci représentent une imbrication conceptuelle (*héritage*) ou une relation organique (*agrégation*). Le diagramme de classes présente un système du point de vue de sa structure interne, il permet de fournir une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisation. Un diagramme de classes n'est donc pas adapté (sauf cas particulier) pour détailler, décomposer, ou illustrer la réalisation d'un cas d'utilisation particulier. Le diagramme de classes modélise les concepts du domaine d'application ainsi que les concepts internes créés de toutes pièces dans le cadre de l'implémentation d'une application. Il permet ainsi de modéliser les classes du système et leurs relations indépendamment d'un langage de programmation particulier.

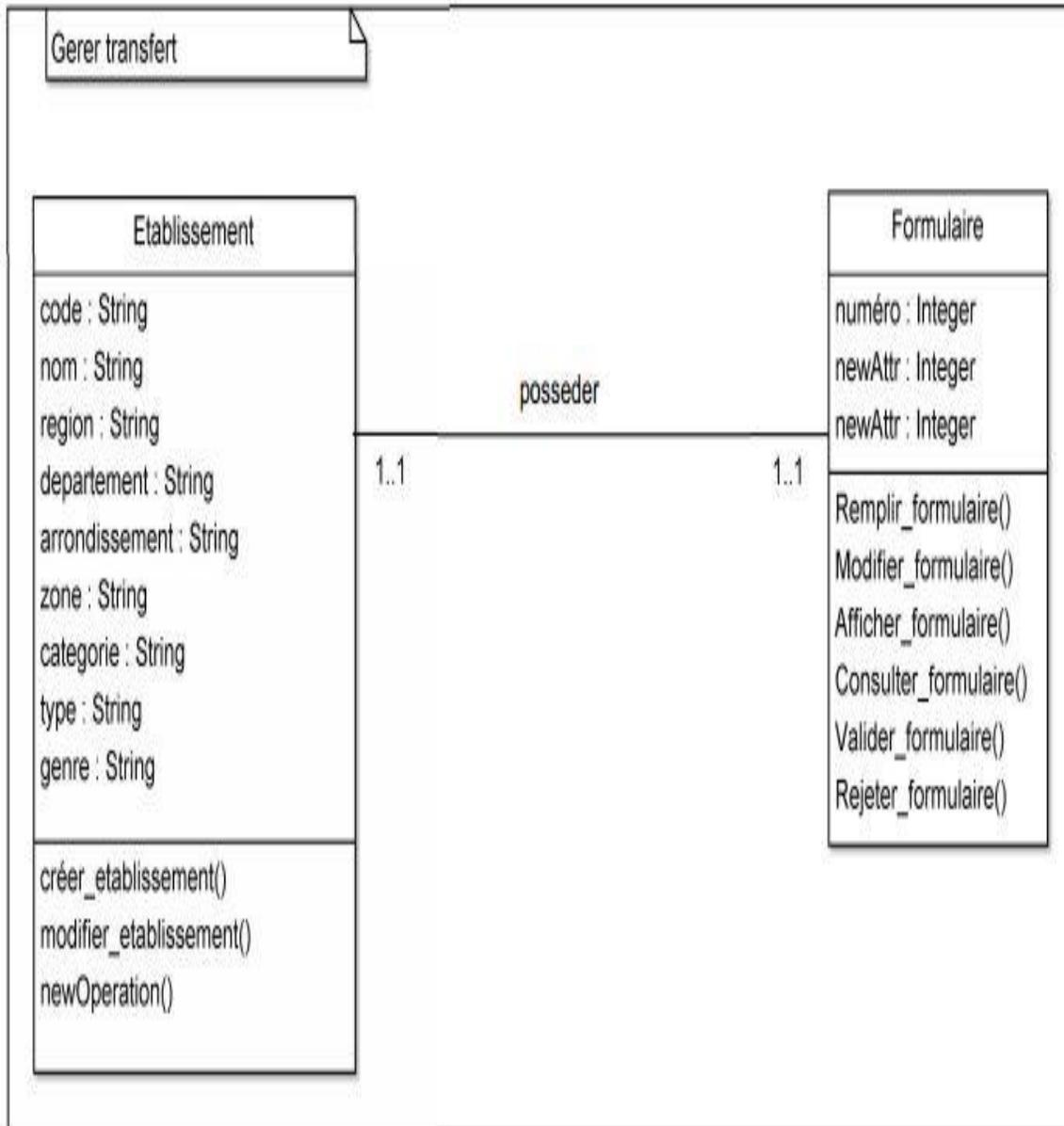


Figure 8: diagramme des classes gérer formulaire établissement

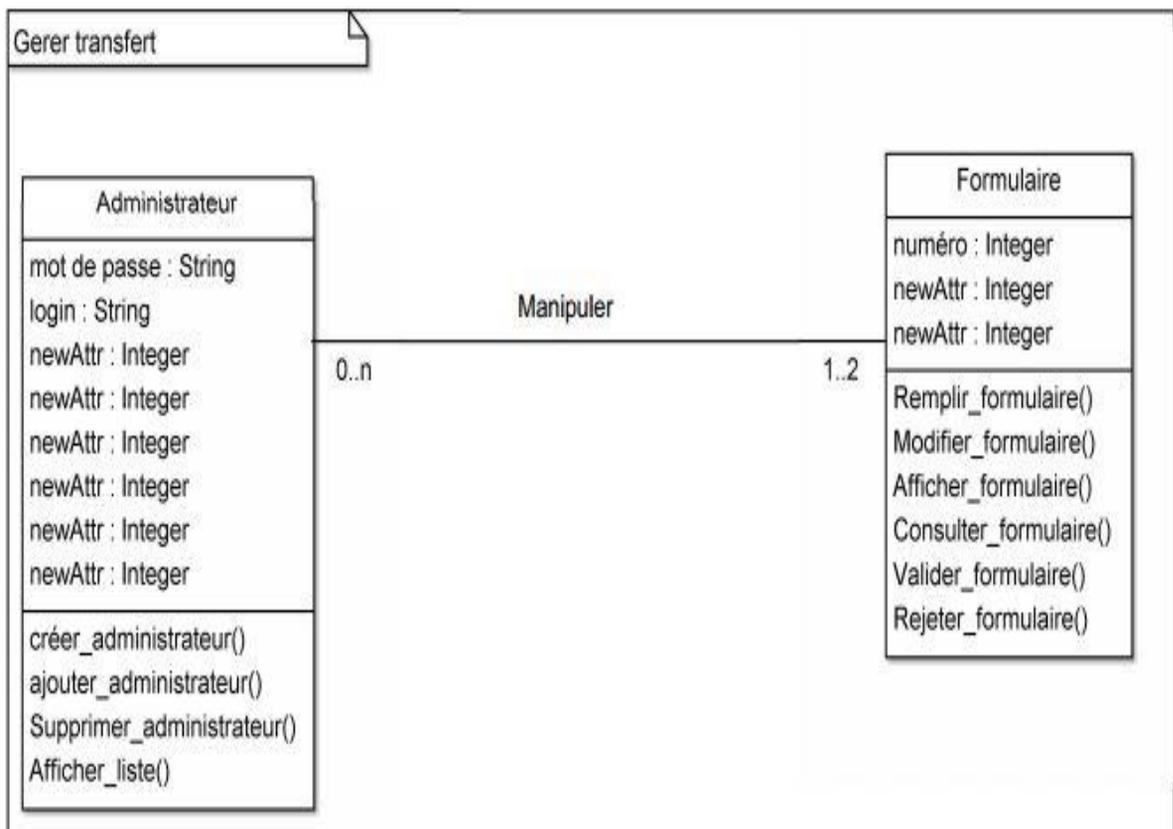


Figure 9 : diagramme des classes gérer transfert enseignant

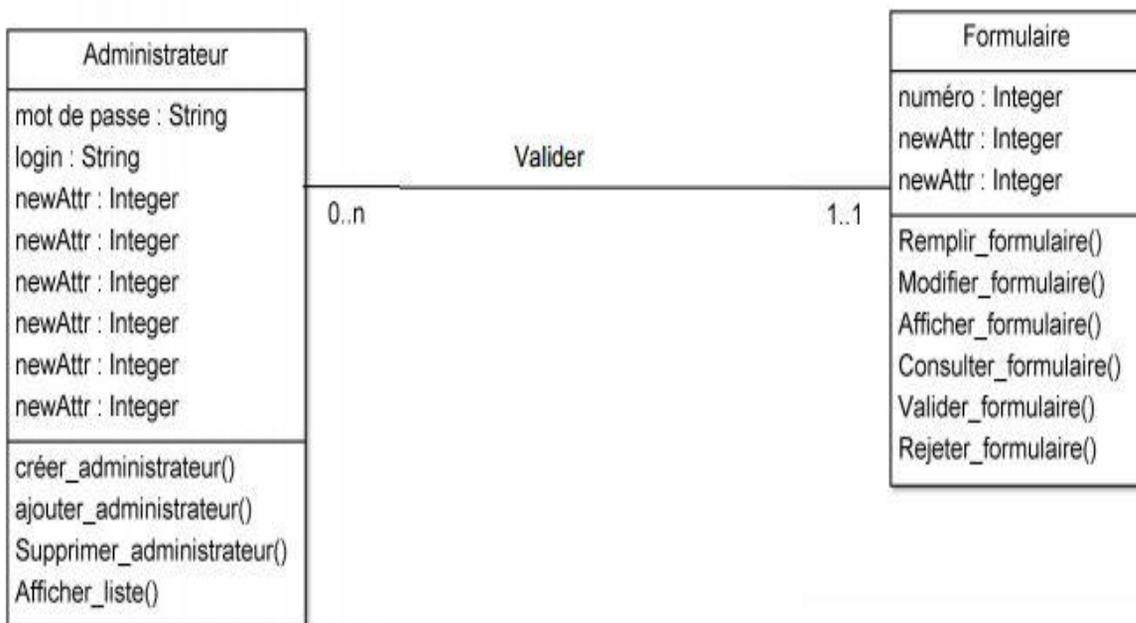


Figure 10 : diagramme des classes valider formulaire

II. CONCEPTION ET MODELISATION DYNAMIQUE

1. LES DIAGRAMMES DE CAS D'UTILISATION

Le diagramme de cas d'utilisation représente la structure des grandes fonctionnalités nécessaires aux utilisateurs du système. C'est le premier diagramme du modèle UML, celui où s'assure la relation entre l'utilisateur et les objets que le système met en œuvre. Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en différentes unités cohérentes ayant un sens pour les acteurs. Les cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée utilisateur de ce besoin loin d'une vision informatique.

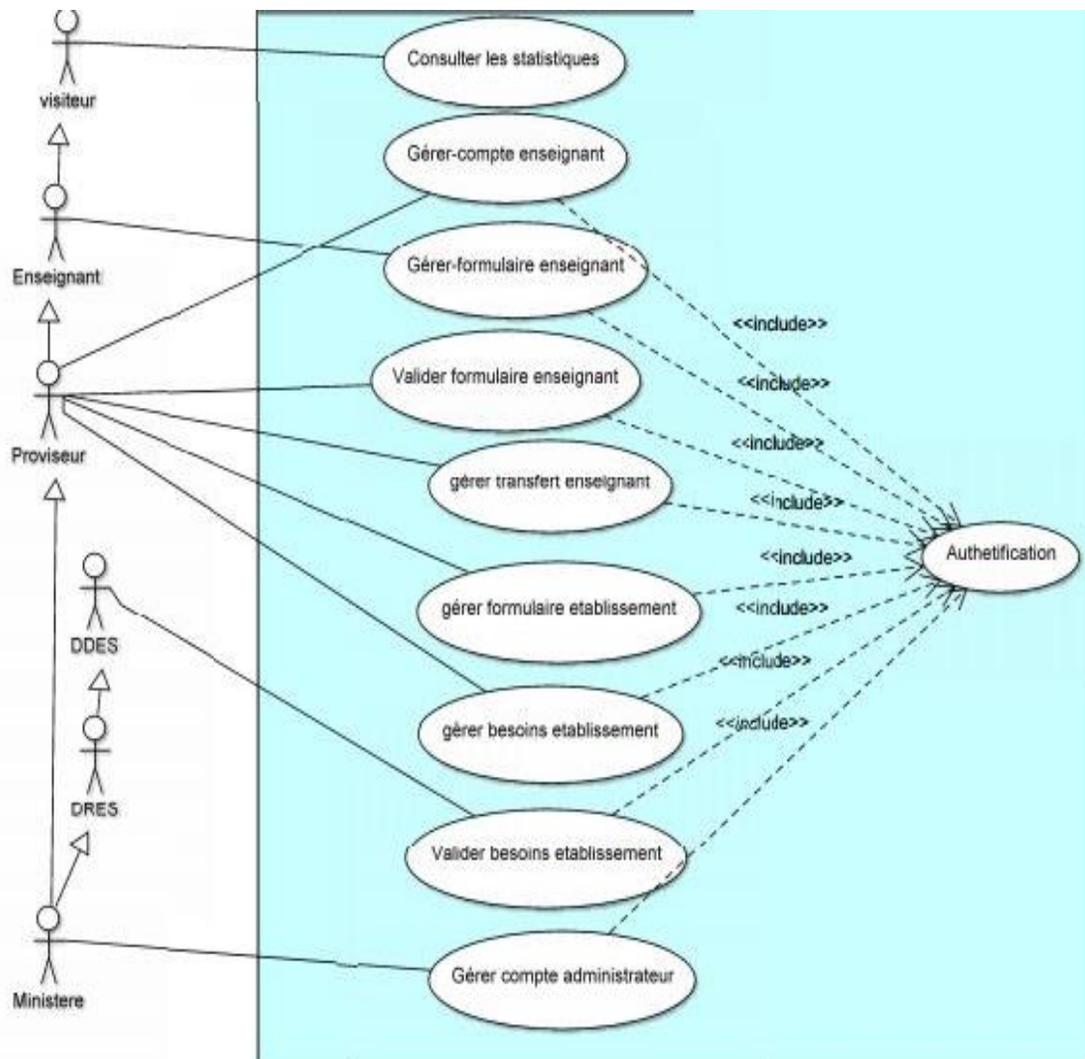


Figure 11 : Diagramme général des cas d'utilisation.

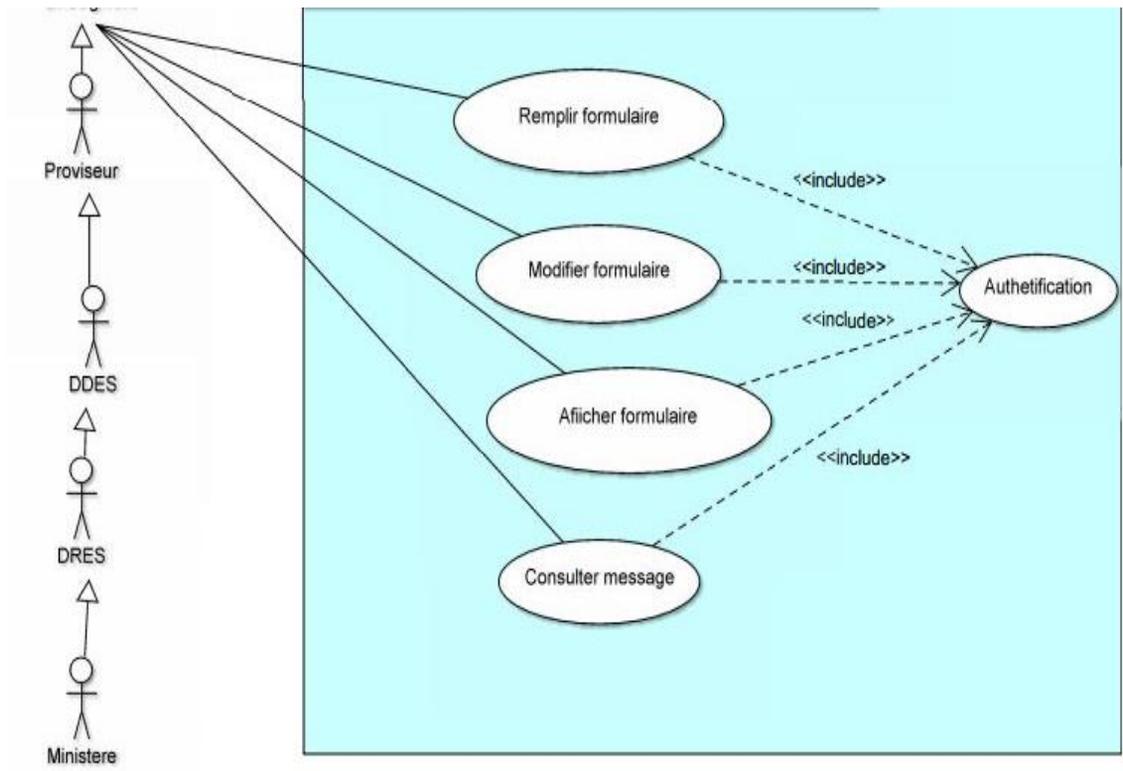


Figure 12 : Diagramme des cas d'utilisation gérer formulaire enseignant.

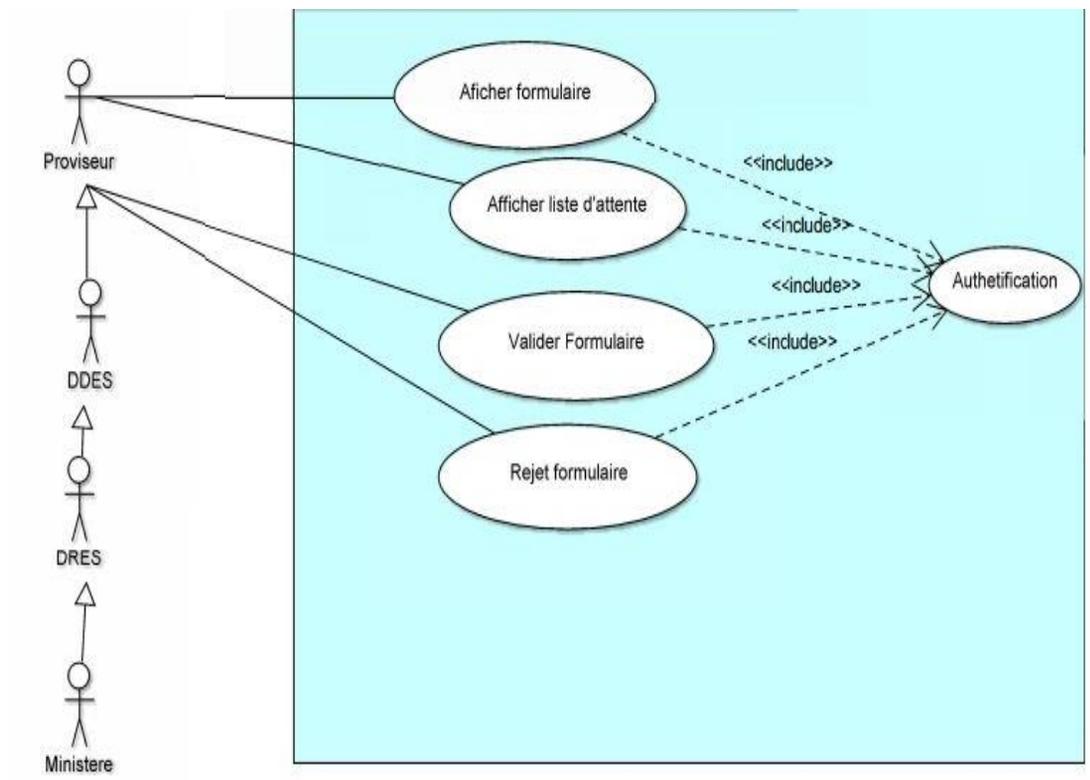


Figure 13 : Diagramme des cas d'utilisation valider formulaire enseignant

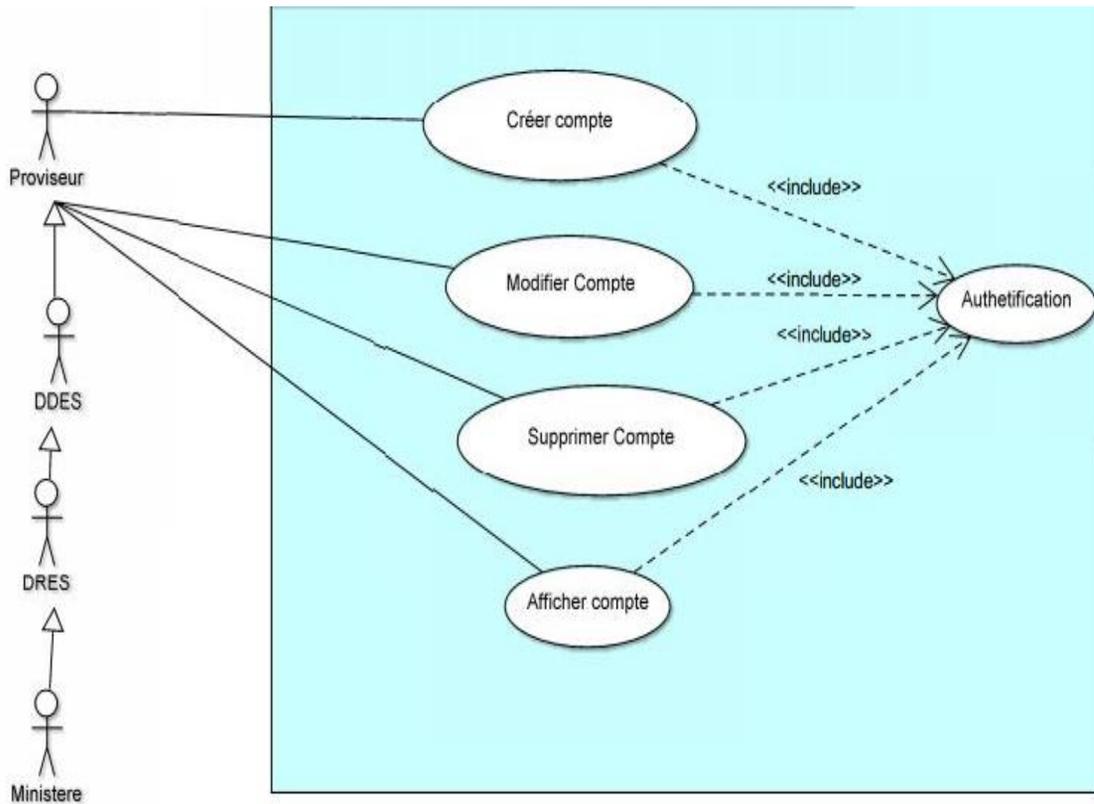


Figure 14 : Diagramme des cas d'utilisation gérer compte établissement

2. LES DIAGRAMMES DE SEQUENCE

Le diagramme de séquence est un diagramme d'interaction qui représente la succession chronologique des opérations réalisées par un acteur. Il indique les objets que l'acteur va manipuler et les opérations qui font passer d'un objet à l'autre. Les principales informations contenues dans un diagramme de séquence sont les messages échangés entre les lignes de vie, présentés dans un ordre chronologique. Ainsi, contrairement au diagramme de communication, le temps y est représenté explicitement par une dimension (*verticale*) et s'écoule de haut en bas.

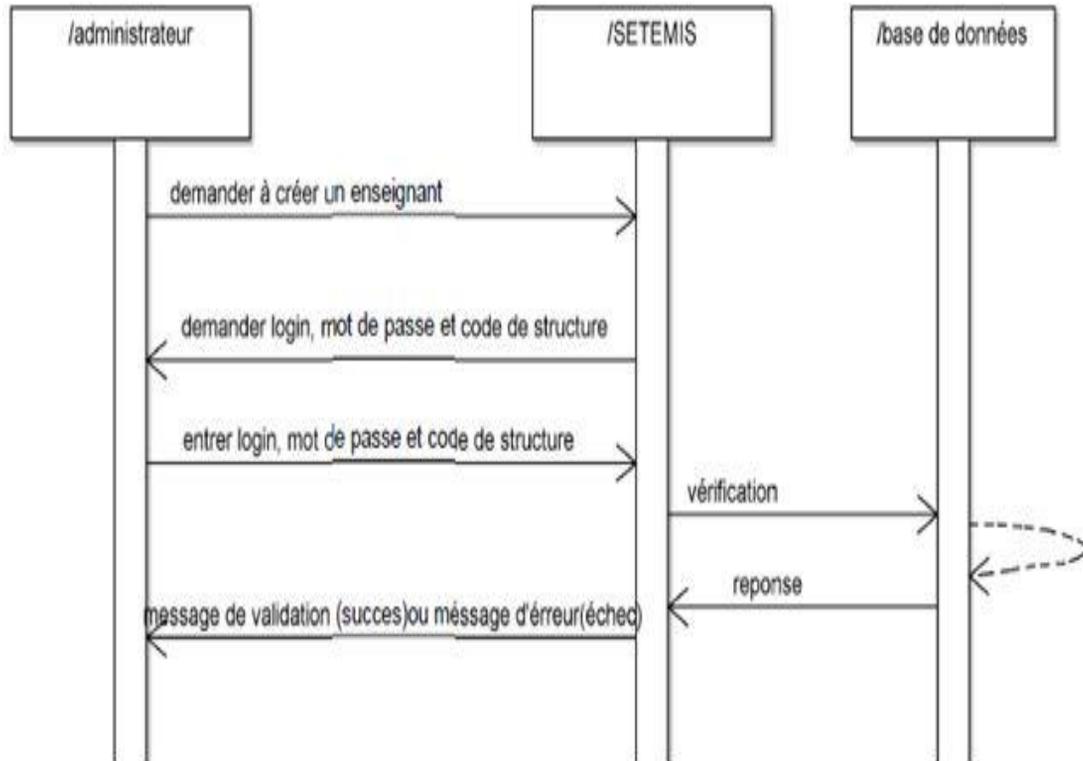


Figure 15 : diagramme de séquence créer compte enseignant.

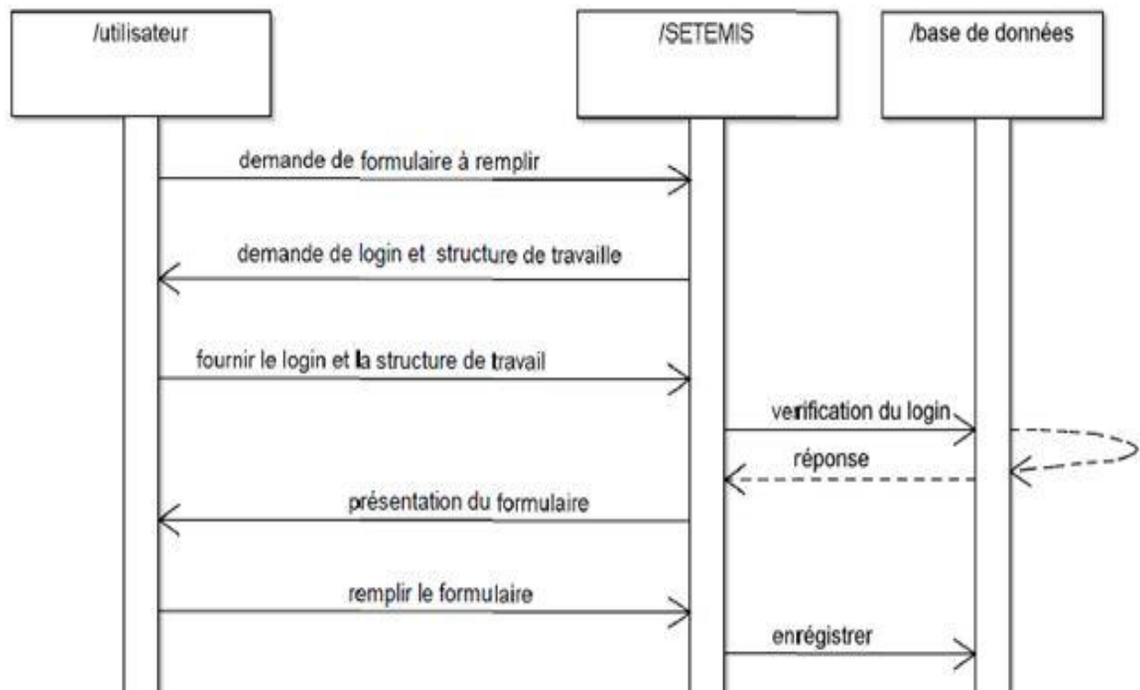


Figure 16 : diagramme de séquence remplir formulaire

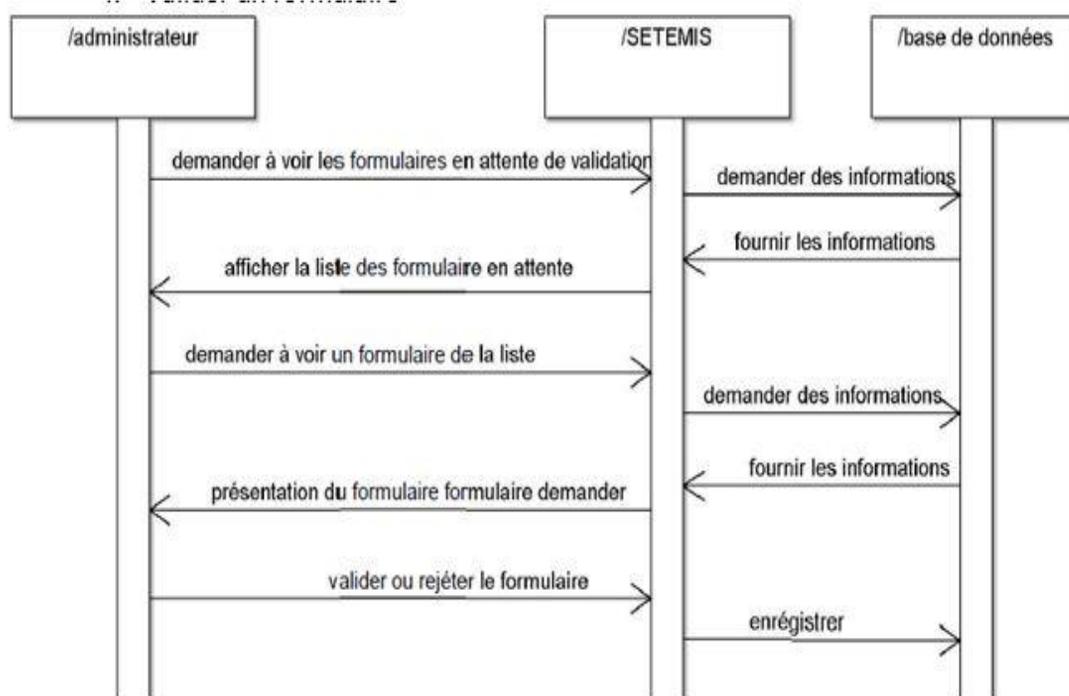


Figure 17 : diagramme de séquence valider formulaire

3. LE DIAGRAMME DE DEPLOIEMENT

Un diagramme de déploiement décrit la disposition physique des ressources matérielles qui composent le système et montre la répartition des composants sur ces matériels. Chaque ressource étant matérialisée par un nœud, le diagramme de déploiement précise comment les composants sont répartis sur ceux-ci et quelles sont les connexions entre les composants ou les nœuds. Dans un diagramme de déploiement, les associations entre les nœuds sont des chemins de communication qui permettent l'échange d'informations.

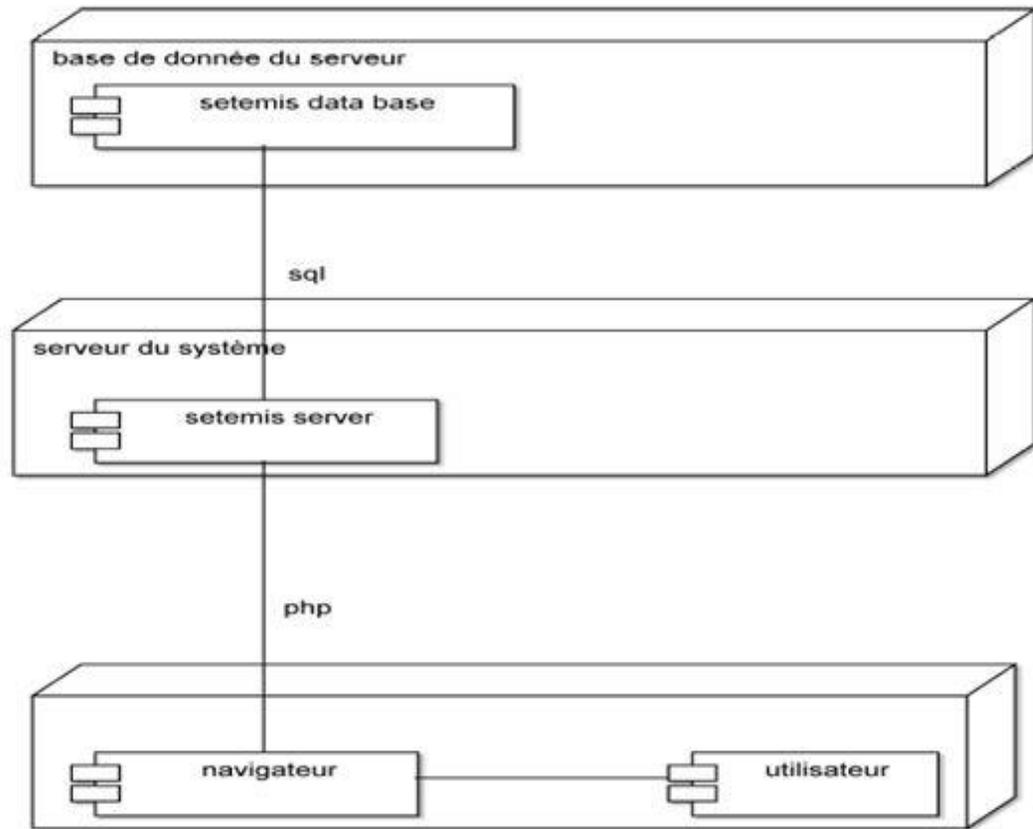


Figure 18 : diagramme de déploiement du système.

III. IMPLEMENTATION DU SYSTEME

Il est question ici des différentes interfaces qui montrent l'exécution du code et témoignent de l'application des diagrammes. C'est la phase de présentation du système développé qui se fera par le biais des captures d'écran réalisées à plusieurs niveaux de l'utilisation du système depuis les pages d'accueil jusqu'aux interfaces minimales.

1. LES PAGES D'ACCUEIL

SETEMIS (SESecondary Teaching Management Information System)

Accueil Connexion

Aujourd'hui, nous sommes le 02-06-2016

SETEMIS (SESecondary Teaching Management Information System)

SETEMIS compte actuellement 21 personnel enseignant dont 8 Femme(s) et 13 Homme(s).

Avec ce système d'agrégation statistique du personnel enseignant (Camerounais), vous avez la possibilité de voir à temps réel, les statistiques d'ordre national, régional, départemental et même pour un établissement sous forme de chiffre (tableau), sous forme graphique (pourcentage) et sous forme de liste pour savoir quel enseignant est dans tel ou tel région, département ou établissement. ces statistiques sont par Genre, Grade, Corps de métier, école de formation, et toutes les combinaisons possible par exemple Grade/Genre, Statut/Genre, Statut/école de formation etc.

©Copyright Administration SETEMIS 2016

SETEMIS (SESecondary Teaching Management Information System)

statistiques du personnel enseignant

- NATIONAL
- REGION
- DEPARTEMENT
- ETABLISSEMENT

Informations sur le formulaire

- FORMULAIRE

Informations generales

tous les professeurs sortent de l'ENS de jaooude sont informés que les revele de note ...
[Lire la suite](#)

tous les professeurs sortent de l'ENS de jaooude sont informés que les revele de note ...
[Lire la suite](#)

Figure 19 : La page d'accueil du système

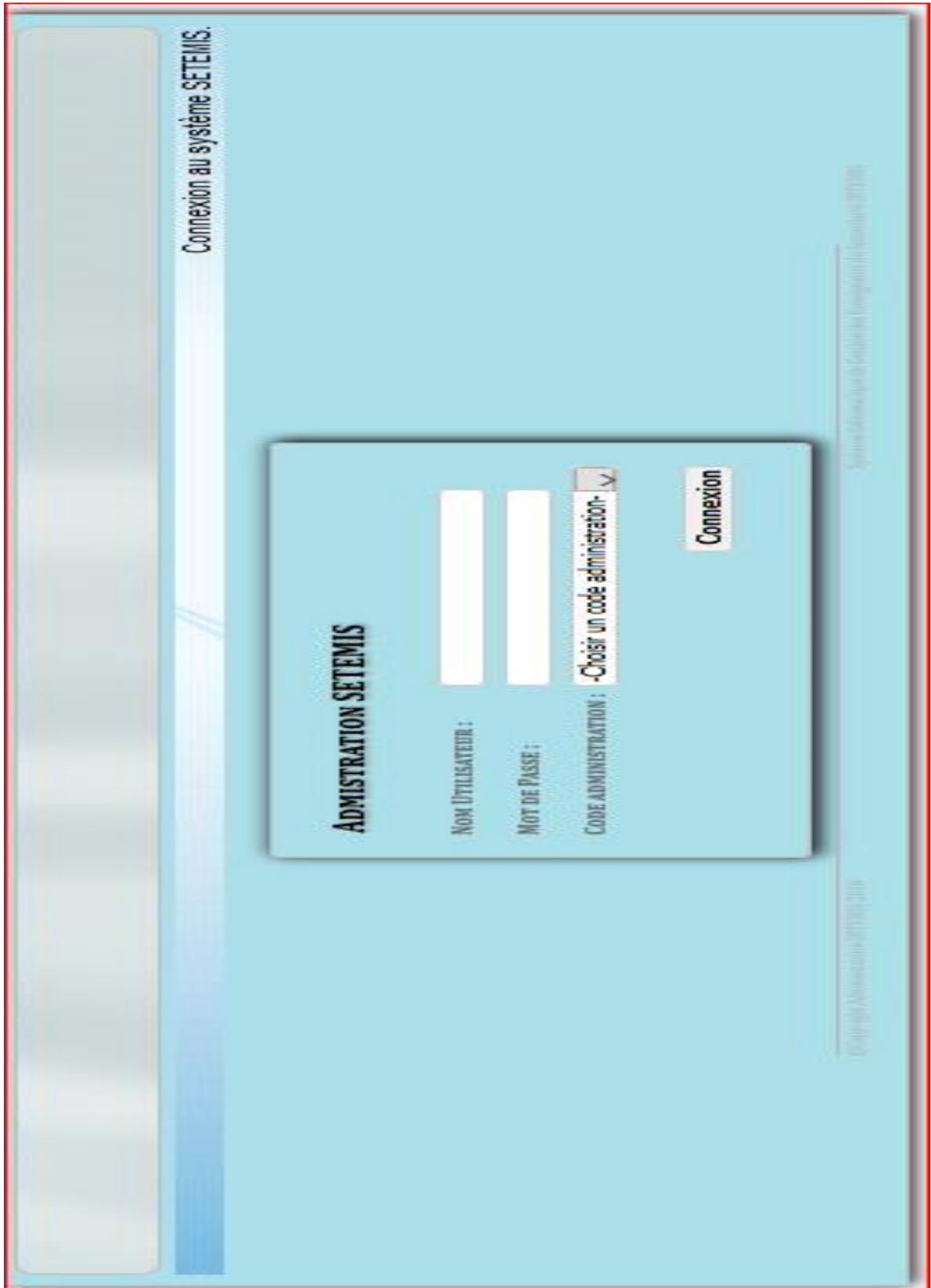
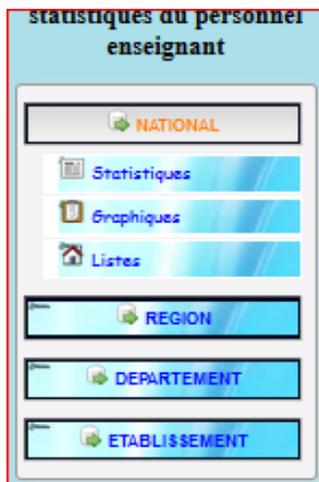


Figure 20 : La page de connexion

1. LE TRAJET D'UNE REQUETE

Nous voulons consulter la répartition à l'échelle nationale des enseignants du secondaire en fonction du genre, leur secteur d'activité et leur grade sous forme de tableau. Après avoir ouvert la page d'accueil :



- Posons le curseur de la souris sur « national » en haut à droite dans la page. En survolant « national », le système nous permet de choisir la représentation souhaitée des statistiques.

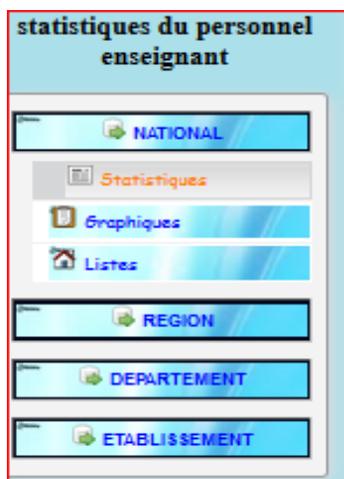
- « statistiques » : nous donne les statistiques sous forme de tableau

- « graphiques » : nous donne les statistiques sous forme graphique

- « listes » : nous donne une liste d'enseignant en fonction des paramètres préalablement choisis.

Figure 21 : menu

1.1. Pour les besoins de notre exemple, nous allons choisir « statistiques »



En posant le curseur sur notre choix celui-ci s'éclaircit.

Figure 22 : item

1.2. Après avoir cliqué sur « Statistiques » on obtient la page ci-dessous :

SETEMIS

SETEMIS (Secondary Teaching Management Information System)

[Accueil](#) [Connexion](#)

statistiques du personnel enseignant

[NATIONAL](#)

[REGION](#)

[DEPARTEMENT](#)

[ETABLISSEMENT](#)

Informations sur le formulaire

[FORMULAIRE](#)

Informations generales

tous les professeurs sortient de l'ENS de yaounde sont informés que les revele de note ...

Lire la suite

tous les professeurs sortient de l'ENS de yaounde sont informés que les revele de note

CHOIX DE L'ETAT

Choisir l'Etat à afficher

[STATISTIQUES PAR GENRE, SECTEUR ET GRADE](#)

[STATISTIQUES SECTER/GENRE ET SECTEUR/GRADE](#)

[STATISTIQUES PAR SITUATION ET PAR CORPS DE METIER](#)

[STATISTIQUES PAR SITUATION/SEXE](#)

[STATISTIQUES PAR CORPS DE METIER/SEXE](#)

Figure 23 : page de statistiques

Cette page nous présente une liste de statistique et il revient à l'utilisateur de choisir celle qui l'intéresse. Pour les besoins de notre exemple nous allons cliquer sur : « STATISTIQUES PAR GENRE, SECTEUR ET GRADE ». On obtient la page suivante :



SETEMIS (Secondary Teaching Management Information System)

[Accueil](#) [Connexion](#)

Statistique Nationale par Genre, Secteur et Grade [Imprimer](#)

REGION	GENRE		SECTEUR		GRADE									
	FEMMES	HOMMES	Public	Privé	TOTAL	PLEG	PCEG	PLET	PCET	PENI	PAENI	PEP\$	PAEP\$	TOTAL
ADAMAOUA	1	0	1	0	1	1	0	0	0	0	0	0	0	1
CENTRE	3	10	5	8	13	6	5	0	0	0	0	0	0	13
EST	0	1	1	0	1	0	0	1	0	0	0	0	0	1
EXTREME-NORD	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LITTORAL	1	0	1	0	1	1	0	0	0	0	0	0	0	1
NORD	1	0	1	0	1	1	0	0	0	0	0	0	0	1
NORD-OUEST	0	1	1	0	1	0	1	0	0	0	0	0	0	1
OUEST	1	0	1	0	1	0	0	1	0	0	0	0	0	1
SUD	0	1	1	0	1	0	0	0	0	1	0	0	0	1
SUD-OUEST	1	0	1	0	1	1	0	0	0	0	0	0	0	1
TOTAL	8	13	13	8	21	10	6	2	0	1	0	0	0	21

statistiques du personnel enseignant

- [NATIONAL](#)
- [REGION](#)
- [DEPARTEMENT](#)
- [ETABLISSEMENT](#)

Informations sur le formulaire

- [FORMULAIRE](#)

Informations generales

tous les professeurs sortent de l'ENS de Yaounde sort informés que les revele de note ... [Lire la suite](#)

tous les professeurs sortent de l'ENS de Yaounde sort informés que les revele de note ... [Lire la suite](#)

©Copyright Administration SETEMIS 2016 SETEMIS (Secondary Teaching Management Information System)

Figure 24 : page de statistique nationale par genre, secteur et grade.

CHAPITRE 4

RESULTATS ET IMPLICATIONS

CHAPITRE 4 : RESULTATS ET IMPLICATIONS

Dans ce travail, nous avons proposé une méthodologie globale de conception au chapitre 3 ainsi qu'un ensemble d'information sur les diagrammes de conception et une implémentation du système dans le chapitre 4. Après avoir matérialisé nos propositions dans le système, nous sommes en mesure d'en faire une première évaluation afin de ressortir les résultats et les implications.

Dans la première partie de ce chapitre, nous allons discuter des différentes caractéristiques de SETEMIS que nous voulons évaluer et des méthodes d'évaluation. Nous détaillerons la méthode d'évaluation pour mesurer l'utilité de SETEMIS. Celle-ci permettait de comparer la qualité des services avec et sans l'utilisation de SETEMIS. Notre objectif était d'étudier l'impact de SETEMIS sur les bénéficiaires. Pour cela, chacun d'entre eux a eu à utiliser le système et à l'éprouver de tous les points de vue. Pour des raisons de coût et de temps, nous ne déroulerons pas le film des tests, mais nous nous basons sur l'appréciation générale qui a été faite sous forme de grille.

Dans la deuxième partie de ce chapitre, nous proposons une analyse des observations faites pendant l'évaluation de SETEMIS ainsi qu'une analyse de l'impact concret de l'utilisation de SETEMIS sur la qualité de la gestion des enseignants.

I. LES RESULTATS

1. DESCRIPTION DE LA SOLUTION

➤ Les caractéristiques du système

SETEMIS est une application conçue pour améliorer la qualité de la gestion du capital enseignant du Cameroun. Notre travail a consisté à développer un système informatique qui permet de gérer automatiquement les enseignants de l'enseignement secondaire au Cameroun.

Il s'agit d'un système informatisé qui collecte en ligne les informations sur le personnel enseignant, les stocke dans une base de donnée et fournit des statistiques relatives à ces informations. Tout ceci est effectué dans le strict respect de la hiérarchie administrative.

➤ **Le public cible**

PUBLIC CIBLE	IMPORTANCE	CENTRE D'INTERET
- Enseignants (enseignant, censeur, surveillants généraux, enseignants travaillant hors des établissements)	Primaire	- Fournir leurs informations - recherche d'information
- Les administrateurs secondaires : chef d'établissement délégué départemental délégué régional	primaire	- fournir leurs informations - recherche d'information - administration de leurs structures respectives
- L'administrateur principal (qui ici pourrait être le ministre des enseignants du secondaire)	Primaire	- fournir leurs informations - recherche d'information - administration du système à l'échelle national
- Les visiteurs (chercheurs, ONG,...)	Secondaire	- recherche d'information

Tableau 1 : Le public cible du système

➤ **La langue**

Le système produit a été conçu et réalisé en langue française uniquement pour l'instant. Ce qui a permis certaines facilités à l'équipe de réalisation de celui-ci.

2. RAPPORT PROBLEMATIQUE/ SETEMIS : LES FONCTIONNALITES

Rappelons tout d'abord la problématique de ce travail : améliorer la gestion et le suivi du personnel enseignant Camerounais et dégager une stratégie de résolution des problèmes actuels du

Le système existant tout en apportant une avancée significative au développement du SIGE camerounais conformément aux objectifs de l'OMD, EPT. Dans le but de satisfaire à cette problématique, nous avons mis au point un système répondant (SETEMIS) afin de satisfaire au maximum les attentes des différentes parties. Ainsi, concernant l'amélioration de la gestion et du suivi des enseignants, le système a été doté de fonctionnalités qui permettent une gestion informatisée et beaucoup plus aisée des enseignants loin du système jusqu'ici utilisé. Le travail autrefois rudimentaire et fastidieux se trouve ainsi amoindri et automatisé par l'utilisation de ce système.

Nom rubrique	Fonctionnalités	Descriptif
<p>➤ Gestion des utilisateurs (enseignant, censeur, surveillant général, personnel de délégation, etc)</p>	<ul style="list-style-type: none"> - Créer un compte utilisateur - Supprimer un compte - Valider un formulaire (profil) - rejeter un formulaire (profil) - Transférer un utilisateur à un autre service - Consulter la liste des transferts reçus - Valider un transfert - Afficher la liste des utilisateurs sous son administration - Consulter les statistiques 	<p>Chaque administrateur est chargé de la gestion des utilisateurs qui se trouvent dans sa structure. Et ne peuvent pas gérer ceux qui ne se trouvent pas sous ses ordres.</p>
<p>➤ Gestion des formulaires (profils)</p>	<ul style="list-style-type: none"> - Remplir le formulaire - Modifier son formulaire - Afficher son formulaire - Consulter les statistiques 	<p>Chaque détenteur de compte obtient un formulaire lorsque son compte est créé. Il le remplit et l'envoie à son administrateur pour validation. Ce formulaire pourra le modifier après.</p>

<p>➤ Gestion des structures (délégations régional, délégation départemental, établissement)</p>	<ul style="list-style-type: none"> - Créer une structure - Supprimer une structure - Transférer un administrateur à une structure - Changer l'administrateur d'une structure - Consulter les statistiques 	<p>Les structures sont gérées par l'administrateur principal.</p>
<p>➤ Gestion des besoins d'un établissement</p>	<ul style="list-style-type: none"> - Remplir le formulaire d'un établissement - Modifier le formulaire d'un établissement - Consulter les statistiques 	<p>Un établissement est géré par son administrateur qui est le chef d'établissement.</p>
<p>➤ Gestion de département</p>	<ul style="list-style-type: none"> - Valider le formulaire d'un établissement - Rejeter le formulaire d'un établissement - Afficher la liste des établissements - Afficher la liste des utilisateurs - Afficher la liste des chefs d'établissement - Consulter les statistiques 	<p>Un département est géré par son l'administrateur qui est le délégué départemental.</p>
<p>➤ Gestion de région</p>	<ul style="list-style-type: none"> - Afficher la liste des établissements - Afficher la liste des utilisateurs - Afficher la liste des chefs d'établissement - Afficher la liste des délégués départementaux - Consulter les statistiques 	<p>Une région est gérée par son l'administrateur qui est le délégué régional.</p>

<p>➤ Gestion des administrateurs</p>	<ul style="list-style-type: none"> - Créer compte administrateur - Transformer un utilisateur en administrateur - Supprimer un administrateur - Consulter les statistiques 	<p>Les administrateurs secondaires sont gérés par l'administrateur principal</p>
<p>• Gestion des visiteurs</p>	<ul style="list-style-type: none"> - Consulter les statistiques 	<p>Ce sont tous ceux qui peuvent avoir accès</p>

Tableau 2 : Les fonctionnalités du système.

Au vue de ces différentes fonctionnalités, l'on peut se rendre compte des solutions que le système apporte notamment l'amélioration de la qualité et du taux de recouvrement des données des enseignants. L'on constate ainsi que SETEMIS est plus qu'un simple site web. La mise en place de ce système contribue ainsi à l'amélioration du système de la gestion du système éducatif camerounais et concourt de ce fait à l'atteinte des objectifs de l'OMD et de l'EPT pour qui un SIME efficace et répondant est un élément important de consultation, de suivi et de perspectives d'un système éducatif. Ainsi, SETEMIS se positionne comme un outil mieux un mécanisme de gestion et de prise en charge de la condition enseignante et surtout s'inscrit en droite ligne de la politique de l'ONU en fournissant un instrument stratégique de collecte, traitement, analyse, publication et de diffusion des données. La différence dans le fonctionnement du système après l'intégration de SETEMIS est très nette et perceptible comme l'illustre le schéma suivant.

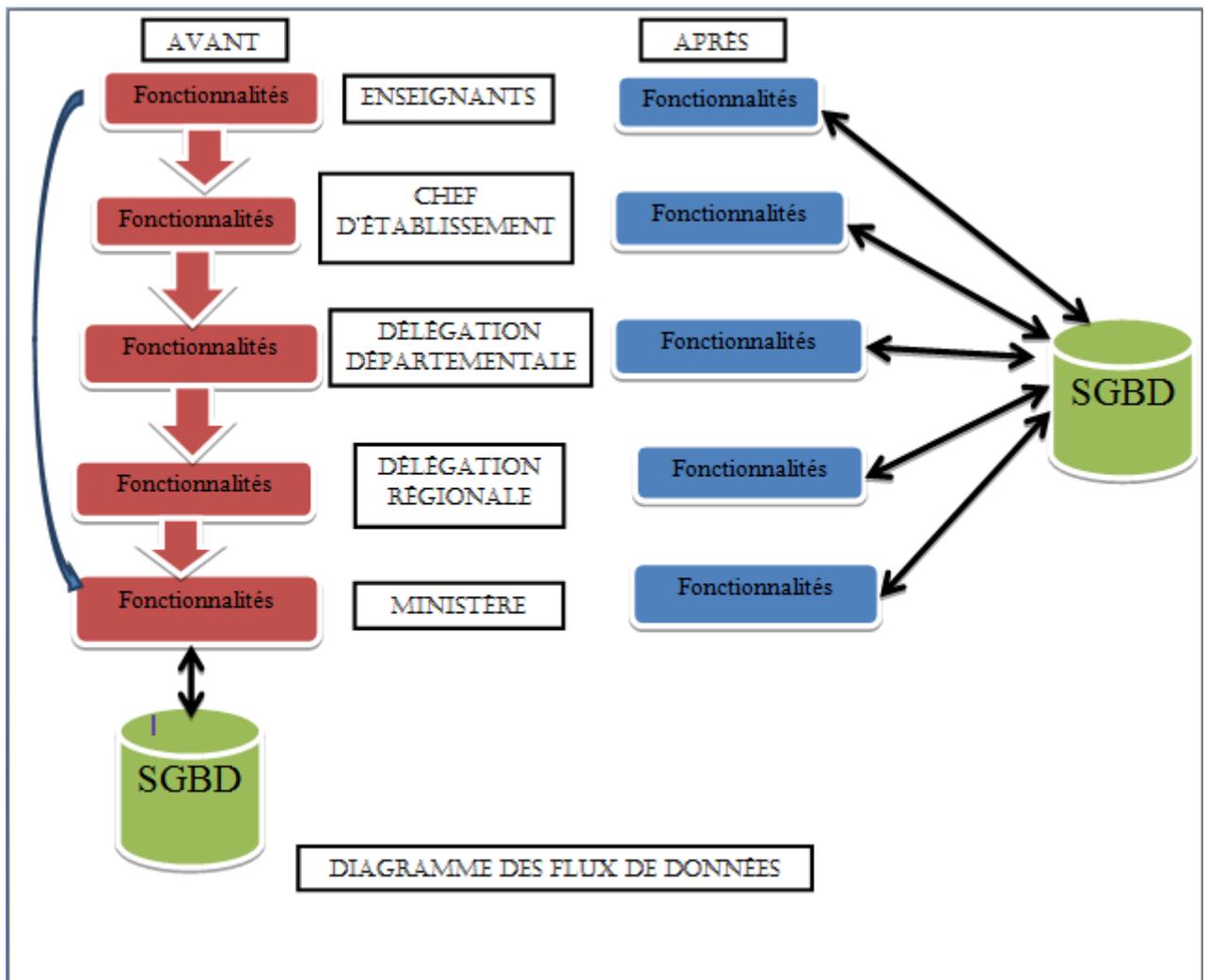


Figure 25 : diagramme du flux des données du ministère des enseignements secondaires après l'intégration de SETEMIS.

3. L'ÉVALUATION

Pour évaluer la qualité de SETEMIS, nous avons administré un questionnaire aux différents bénéficiaires sous la forme de spécifications apportées par SETEMIS et une grille avec les critères de qualité à évaluer. Questionnaires qui, évidemment, ont été remplis après utilisation du système. De plus, nous les avons priés d'indiquer un indice de confiance (*de 1 à 3*) pour chaque évaluation qu'ils faisaient afin de recueillir plus de précision sur la valeur de leurs appréciations. Nous présentons la grille de critères qualité utilisée lors de l'évaluation pour estimer la qualité de SETEMIS. Cette grille donne des précisions sur la qualité et la quantité d'information traitées, la robustesse de SETEMIS, sa portabilité, le critère célérité et accessibilité, son potentiel ainsi que son

utilité et sa pertinence dans le contexte du management. La grille permet également de mesurer le respect du cahier des charges initial. La grille de notation se constitue comme suit :

n°	Questions	Réponses (cochez une case)		Indice de confiance (cochez une case)		
		oui	non	1	2	3
1	Avez-vous trouvé SETEMIS facile à prendre en main ?					
2	Pensez-vous que SETEMIS vous permette de mettre à jour vos informations plus aisément ?					
3	D'après vous, SETEMIS apporte-t-il un plus dans la gestion des enseignants ?					
4	Pensez-vous qu'il soit un bon instrument de management des enseignants ?					
6	Considérant les difficultés rencontrées avec le système actuel pensez-vous que SETEMIS les résolve ?					
7	Caractérissez SETEMIS en 5 mots :					

Tableau 3 : La grille d'évaluation du système.

II. DISCUSSION

Après l'implémentation du système, il est opportun de le tester ou encore de le présenter aux futurs utilisateurs qui en sont les juges. Le retour observé lors de cette phase nous a permis de nous rendre compte que SETEMIS a atteint ses objectifs. En effet, de manière générale, les utilisateurs se sont dits enclins à utiliser cet outil novateur qui apporte un plus dans leur gestion. De plus l'indice de confiance assez haut (*une moyenne de 2.5*) nous conforte dans cette idée. Les adjectifs qui reviennent sont novateur, simple, pratique.

Toutefois, la mise en application du système énoncé dans ce travail se heurterait à des difficultés considérables lors du déploiement sur le terrain. En effet, les éventuels obstacles seraient :

- La disponibilité des infrastructures et des hommes : l'absence de matériel informatique et le problème d'électrification qui prive certaines zones rurales de l'énergie électrique et donc de connexion internet. Le système n'étant pas déployé sous androïde (*pour l'instant*), les utilisateurs ne pourront y avoir accès.
- Le coût du déploiement de ce système se révèle énorme dans la mesure où c'est un aspect novateur qui demande un investissement certain pour son déploiement tant au niveau du ministère que des établissements.
- Le recyclage et la formation des responsables est nécessaire pour une prise en main aisée du système. Il est de ce fait primordial d'organiser des séminaires de formation, des colloques,... qui devront mettre l'accent sur l'introduction du système dans les usages quotidiens et ses avantages principalement chez les sujets réfractaires au changement.

CONCLUSION ET PERSPECTIVES

CONCLUSION ET PERSPECTIVES

L'objet de ce mémoire a porté sur la gestion en ligne des enseignants du secondaire au Cameroun par le développement d'un système d'information destiné à la faciliter. Afin de mener à bien ce travail de recherche, nous avons commencé ce mémoire par une introduction générale afin de fournir le contexte et les éléments justificatifs de ce travail de même que ses objectifs, son intérêt pour le public cible ainsi que le cadre théorique.

Une revue de littérature pour expliquer le concept de système d'information pour le management de l'éducation a été faite en deuxième partie pour ressortir ses caractéristiques. Par la suite, il a été question d'une étude de cas portant sur les SIME existant et sur le cas éthiopien spécifiquement car il est l'un des pionniers en matière de système d'information pour le management des enseignants (*TMIS*). C'est un ensemble de programmes ou d'applications informatiques permettant d'automatiser les différents processus qui composent la gestion du capital enseignant. SETEMIS va avoir pour objectif d'aider les bénéficiaires dans leur travail au quotidien. Puis, nous avons étudié les cadres méthodologique et conceptuel pour asseoir les bases de ce système.

Dans le troisième moment de ce travail, nous nous sommes intéressés à la méthodologie suivie pour la conception et l'implémentation du système. Nous avons présenté les méthode et modèle d'analyse et de conception, le modèle de développement, l'architecture du système et les outils de développement.

Le chapitre quatre montre la conception et l'implémentation du système à travers les diagrammes de conception statique et dynamique. Le volet implémentation est reflété par la présentation du système au travers des captures d'écran. Pour finir, le chapitre cinq explicite les résultats dans une première sous partie à travers la description du système, la présentation du rapport problématique/SETEMIS et une évaluation du système. Dans la seconde sous partie, il est question de la critique de la solution apportée dans la discussion.

Le travail que nous nous sommes assignés dans ce mémoire consistait à mettre sur pied un outil de gestion informatisée des enseignants du secondaire. Cependant, le système réalisé peut être amélioré de bien de manière par l'ajout de certaines fonctionnalités.

PERSPECTIVES

SETEMIS a été développé de sorte qu'il soit assez facile d'y ajouter des fonctionnalités et des palettes. Nous ambitionnons de ce fait de le mettre en liaison avec le SIGIPES et les services de la CNPS afin d'assurer un meilleur suivi de la communauté enseignante de sorte que ces derniers puissent à partir de cette plateforme avoir accès à leur bulletin de solde, suivre l'évolution de leur carrière et même de leur dossier de retraite.

Dans le même sens, nous envisageons gérer la dynamique et la traçabilité du système et des enseignants en donnant la possibilité d'avoir le parcours intégral d'un enseignant de sa sortie d'une école de formation à sa position actuelle tout en relevant les cas de déperdition professionnel de même que les promotions hors respect du schéma administratif règlementaire.

Par ailleurs, un accent particulier sera mis sur les aspects sécurité et fiabilité du système pour lesquels l'apport de la biométrie sera fondamental.

Il serait intéressant d'agrèger SETEMIS avec un système d'information pour la gestion des élèves du secondaire afin d'obtenir un Système d'Information pour le Management du Secondaire (SIMS).

BIBLIOGRAPHIE

BIBLIOGRAPHIE

ARTICLES

Bulletin d'information de l'ISU n°9, ressources scolaires et pédagogiques en Afrique subsaharienne : analyse des données régionales collectées par l' ISU en 2011 sur l'éducation, avril 2012.

LIVRES

- Laudon, k., et Laudon, j., 2010, management des systèmes d'information, éditions pearson.
- GILLET, M., GILLET, P., 2010, SIRH: Système d'information des ressources humaines, Editions Dunod.
- PERETTI, j-m., 2012, Gestion des ressources humaines, Editions eyrolles.

MEMOIRES

- Ophélie CASTILLO, « Le Système d'Information des Ressources Humaines au sein des Petites et Moyennes Entreprises. », Mémoire de fin d'études 1^{ère} année de Master Management Stratégique des Organisations, Parcours Gestion des Ressources Humaines, 2011-2012.

PROGRAMMES

- Adaptation des systèmes d'information à la gouvernance du premier degré et au pilotage des écoles, rapport n°2015-054, septembre 2015
- Annuaire statistique de l'Enseignement secondaire, Burkina Faso (2011-2012), juillet 2012
- Annuaire statistique de l'Enseignement Technique et de la Formation Professionnelle (2011-2012)
- Développement des capacités pour l'Éducation pour tous (2011) UNESCO
- Examen national 2015 de l'Éducation pour tous : Cameroun
- Examen national 2015 de l'Éducation pour tous : Côte d'Ivoire
- Institut de statistique de l'UNESCO (ISU) et UNICEF (2015). Réaliser la promesse non tenue de l'Éducation pour tous : Résultats de l'Initiative mondiale en faveur des enfants non scolarisés. Montréal : ISU. <http://dx.doi.org/10.15220/978-92-9189-163-4-fr>

- L'éducation pour tous en AFRIQUE subsaharienne en 2010 : Rapport pour la réunion du Groupe de Haut Niveau (Jomtien 22-24 mars 2011)
- MENET/DPES/SDS&E/ANNUAIRE STATISTIQUE DU SECONDAIRE 2013-2014, république de Côte-d'ivoire.
- OpenEMIS Core User Guide, 3 July 2014
- Programme UNESCO-CapEFA
- Projet de Perspectives de l'Education en Afrique, 2014
- RECUEIL DE DONNÉES MONDIALES SUR L'ÉDUCATION 2011 Statistiques comparées sur l'éducation dans le monde, UNESCO-UIS 2011
- Système d'Information pour la Gestion de l'Education (S.I.G.E.), 2013, Sénégal

SUPPORTS DE COURS

Support de cours Génie logiciel ENS 2014-2015

Support de cours bases de données, ENS 2014-2015

Support de cours systèmes d'information ENS 2014-2015

IUT Bordeaux 1 2013-2014 Département Informatique OMGL3 ACSI Objet

ANNEXES

ANNEXE A

LA DESCRIPTION DES SCENARIOS

DESCRIPTION DES SCENARIOS EN FONCTON DES CAS DUTILISATION

I. CAS D'UTILISATION GESTION DES ENSEIGNANTS

<u>Cas d'utilisation</u> : gestion des enseignants
<u>Acteurs principaux</u> : Proviseur, Administrateur système
<u>Description</u> : le Proviseur est responsable de la gestion des enseignants. Il procède à l'ajout des nouveaux enseignants, la modification de certaines informations et la suppression des enseignants. On peut afficher la liste des enseignants inscrit dans une période, voir la fiche d'un enseignant, les statistiques des enseignants par niveau et par genre...

DESCRIPTION DES SCENARIOS

- *Ajouter un enseignant*

<u>Cas d'utilisation</u> : Ajouter un enseignant
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'ajouter un nouvel enseignant dans un établissement
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à ajouter un enseignant (1)➤ Le système lui demande de fournir le matricule de l'enseignant (2)➤ L'utilisateur fournit le matricule de l'enseignant et passe à suivant (3) Le système vérifie le matricule (4) <ul style="list-style-type: none">➤ Le système présente le formulaire d'inscription à l'utilisateur (nom et prénom, date de naissance, nom du père, nom de la mère, adresse, téléphone, email, lieu de naissance, religion, ancien établissement, image, genre, grade, statut,...) (5)➤ L'utilisateur remplit le formulaire et valide (6)➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<u>Description alternative 1</u> : le matricule existe déjà <ul style="list-style-type: none">➤ (1), (2), (3)➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre matricule car celui-ci existe déjà (4)➤ (5), (6), (7)

<p><u>Description alternative 2</u> : les champs obligatoires non remplis</p> <ul style="list-style-type: none"> ➤ (1), (2), (3), (4), (5), (6) ➤ Le système renvoi un message d’alerte à l’utilisateur lui demandant de remplir tous les champs obligatoires (7’) ➤ (7)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met l’enseignant dans la liste d’attente pour validation par le proviseur

- ***Modifier un enseignant***

<p><u>Cas d’utilisation</u> : Modifier un enseignant</p>
<p><u>Acteurs</u> : Proviseur, Administrateur Système</p>
<p><u>Description</u> : ce cas d’utilisation permet de modifier (les informations) un enseignant dans l’établissement</p>
<p><u>Pré-condition</u> : l’utilisateur doit être un proviseur ou l’administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L’utilisateur demande à modifier un enseignant (1) ➤ Le système lui demande de choisir l’enseignant à modifier (2) ➤ L’utilisateur choisit l’enseignant et valide (3) ➤ Le système lui présente le formulaire d’inscription de l’enseignant avec les anciens paramètres (nom et prénom, date de naissance, nom du père, nom de la mère, adresse, téléphone, email, lieu de naissance, religion, ancien établissement, image, sexe, statut, grade,...) (5) ➤ L’utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l’opération (7)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met à jour les informations de l’enseignant

- ***Supprimer un enseignant***

<p><u>Cas d’utilisation</u> : Supprimer un enseignant</p>
<p><u>Acteurs</u> : Proviseur, Administrateur Système</p>
<p><u>Description</u> : ce cas d’utilisation permet de supprimer un enseignant dans l’établissement</p>

<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à supprimer un enseignant (1) ➤ Le système lui demande de choisir l'enseignant à supprimer (2) ➤ L'utilisateur choisit l'enseignant et valide (3) ➤ Le système demande confirmation de l'opération de suppression de cet enseignant (4) ➤ L'utilisateur confirme la suppression (5) ➤ Le système valide l'opération (6)
<u>Post-condition</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'enseignant ne fait plus partie de l'établissement

- ***valider un enseignant***

<u>Cas d'utilisation</u> : valider un enseignant
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet l'inscription effective d'un enseignant en lui associant un établissement.
<u>Pré-conditions</u> : <ul style="list-style-type: none"> ➤ l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe ➤ L'enseignant doit être nouveau dans l'établissement.
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un enseignant (1) ➤ Le système lui demande de choisir l'enseignant (2) ➤ L'utilisateur choisit l'enseignant puis appuie sur suivant (3) ➤ Le système valide l'opération (4)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'enseignant est ajouté dans la liste des enseignants de cette classe ➤ L'enseignant est de ce fait retiré de la liste d'attente

- ***Transférer un enseignant.***

<u>Cas d'utilisation</u> : Transférer un enseignant
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de Transférer un enseignant d'un

établissement à un autre.
<p><u>Pré-conditions</u> :</p> <ul style="list-style-type: none"> ➤ l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe. ➤ L'enseignant doit être inscrit dans un établissement
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à transférer un enseignant (1) ➤ Le système lui demande de choisir l'enseignant à transférer (2) ➤ L'utilisateur choisit l'enseignant et valide (3) ➤ Le système lui présente le formulaire de transfert (4) ➤ L'utilisateur remplit le formulaire et valide (5) ➤ Le système valide l'opération (6)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'enseignant est supprimé dans la liste des enseignants de cet établissement ➤ L'enseignant est désormais dans la liste d'attente de l'établissement d'accueil

II. CAS D'UTILISATION GESTION ETABLISSEMENT

<u>Cas d'utilisation</u> : gestion d'établissement
<u>Acteurs principaux</u> : délégué d'arrondissement, Administrateur système
<p><u>Description</u> : le délégué d'arrondissement est responsable de la gestion d'établissement. Il procède à l'ajout des nouveaux établissements, la modification de certaines informations et la suppression des établissements. On peut afficher la liste des établissements inscrits dans une période, voir la fiche d'un établissement, les statistiques des établissements.</p>

DESCRIPTION DES SCENARIOS

- *Ajouter un établissement*

<u>Cas d'utilisation</u> : Ajouter un établissement
<u>Acteurs</u> : délégué arrondissement, Administrateur Système
<p><u>Description</u> : ce cas d'utilisation permet d'ajouter un nouvel établissement dans un arrondissement.</p>

<p><u>Pré-condition</u> : l'utilisateur doit être un délégué arrondissement ou l'administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à ajouter un établissement (1) ➤ Le système lui demande de fournir le code de l'établissement (2) ➤ L'utilisateur fournit le code de l'établissement et passe à suivant (3) ➤ Le système vérifie le code (4) ➤ Le système présente le formulaire d'inscription à l'utilisateur (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5) ➤ L'utilisateur remplit le formulaire et valide (6) ➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<p><u>Description alternative 1</u> : le code existe déjà</p> <ul style="list-style-type: none"> ➤ (1), (2), (3) ➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre code car celui-ci existe déjà (4) ➤ (5), (6), (7)
<p><u>Description alternative 2</u> : les champs obligatoires non remplis</p> <ul style="list-style-type: none"> ➤ (1), (2), (3), (4), (5), (6) ➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant de remplir tous les champs obligatoires (7') ➤ (7)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met l'établissement dans la liste d'attente pour validation par le délégué

- ***Modifier un établissement***

<p><u>Cas d'utilisation</u> : Modifier un établissement</p>
<p><u>Acteurs</u> : délégué arrondissement, Administrateur Système</p>
<p><u>Description</u> : ce cas d'utilisation permet de modifier les informations d'un établissement</p>
<p><u>Pré-condition</u> : l'utilisateur doit être un délégué d'arrondissement ou l'administrateur système qui est authentifié par un login et un mot de passe.</p>
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à modifier un établissement (1) ➤ Le système lui demande de choisir l'établissement à modifier (2) ➤ L'utilisateur choisit l'établissement et valide (3) ➤ Le système lui présente le formulaire d'inscription de l'établissement avec les anciens paramètres (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5)

<ul style="list-style-type: none"> ➤ L'utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données. ➤ Le système met à jour les informations de l'établissement.

- *Supprimer un établissement*

<p><u>Cas d'utilisation</u> : Supprimer un établissement</p>
<p><u>Acteurs</u> : délégué d'arrondissement, Administrateur Système</p>
<p><u>Description</u> : ce cas d'utilisation permet de supprimer un établissement dans l'arrondissement</p>
<p><u>Pré-condition</u> : l'utilisateur doit être un délégué d'arrondissement ou l'administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à supprimer un établissement (1) ➤ Le système lui demande de choisir l'établissement à supprimer (2) ➤ L'utilisateur choisit l'établissement et valide (3) ➤ Le système demande confirmation de l'opération de suppression de cet établissement (4) ➤ L'utilisateur confirme la suppression (5) ➤ Le système valide l'opération (6)
<p><u>Post-condition</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'établissement ne fait plus partie de l'arrondissement

- *valider un établissement*

<p><u>Cas d'utilisation</u> : valider un établissement</p>
<p><u>Acteurs</u> : délégué d'arrondissement, Administrateur Système</p>
<p><u>Description</u> : ce cas d'utilisation permet l'inscription effective d'un établissement en lui associant un arrondissement.</p>
<p><u>Pré-conditions</u> :</p> <ul style="list-style-type: none"> ➤ l'utilisateur doit être un délégué d'arrondissement ou l'administrateur système qui est authentifié par un login et un mot de passe ➤ L'établissement doit être nouveau dans l'arrondissement.
<p><u>Description nominale</u> :</p>

<ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un établissement (1) ➤ Le système lui demande de choisir l'établissement (2) ➤ L'utilisateur choisit l'établissement puis appuis sur suivant (3) ➤ Le système valide l'opération (4)
<p><u>Post-conditions :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'établissement est ajouté dans la liste des établissements de cette classe ➤ L'établissement est de ce fait retiré de la liste d'attente

I. Cas d'utilisation gestion arrondissement

<p><u>Cas d'utilisation :</u> gestion d'arrondissement</p>
<p><u>Acteurs principaux :</u> délégué départemental, Administrateur système</p>
<p><u>Description :</u> le délégué départemental est responsable de la gestion d'arrondissement. Il procède à l'ajout des nouveaux arrondissements, la modification de certaines informations et la suppression des arrondissements. On peut afficher la liste des arrondissements inscrits dans une période, voir la fiche d'un arrondissement, les statistiques des arrondissements.</p>

DESCRIPTION DES SCENARIOS

- ***Ajouter un arrondissement***

<p><u>Cas d'utilisation :</u> Ajouter un arrondissement</p>
<p><u>Acteurs :</u> délégué départemental, Administrateur Système</p>
<p><u>Description :</u> ce cas d'utilisation permet d'ajouter un nouvel arrondissement dans un département.</p>
<p><u>Pré-condition :</u> l'utilisateur doit être un délégué départemental ou l'administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale :</u></p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à ajouter un arrondissement (1) ➤ Le système lui demande de fournir le code de l'arrondissement (2) ➤ L'utilisateur fournit le code de l'arrondissement et passe à suivant (3) ➤ Le système vérifie le code (4) ➤ Le système présente le formulaire d'inscription à l'utilisateur (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5) ➤ L'utilisateur remplit le formulaire et valide (6)

➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<u>Description alternative 1</u> : le code existe déjà
<ul style="list-style-type: none"> ➤ (1), (2), (3) ➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre code car celui-ci existe déjà (4) ➤ (5), (6), (7)
<u>Description alternative 2</u> : les champs obligatoires non remplis
<ul style="list-style-type: none"> ➤ (1), (2), (3), (4), (5), (6) ➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant de remplir tous les champs obligatoires (7') ➤ (7)
<u>Post-conditions</u> :
<ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met l'arrondissement dans la liste d'attente pour validation par le délégué département.

- **Modifier un arrondissement**

<u>Cas d'utilisation</u> : Modifier un arrondissement
<u>Acteurs</u> : délégué départemental, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier les informations d'un arrondissement
<u>Pré-condition</u> : l'utilisateur doit être un délégué département ou l'administrateur système qui est authentifié par un login et un mot de passe.
<u>Description nominale</u> :
<ul style="list-style-type: none"> ➤ L'utilisateur demande à modifier un arrondissement (1) ➤ Le système lui demande de choisir l'arrondissement à modifier (2) ➤ L'utilisateur choisit l'arrondissement et valide (3) ➤ Le système lui présente le formulaire d'inscription de l'arrondissement avec les anciens paramètres (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5) ➤ L'utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<u>Post-conditions</u> :
<ul style="list-style-type: none"> ➤ Le système met à jour la base de données. ➤ Le système met à jour les informations de l'arrondissement.

- *Supprimer un arrondissement*

<u>Cas d'utilisation</u> : Supprimer un arrondissement
<u>Acteurs</u> : délégué département, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de supprimer un arrondissement dans le département
<u>Pré-condition</u> : l'utilisateur doit être un délégué département ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à supprimer un arrondissement (1) ➤ Le système lui demande de choisir l'arrondissement à supprimer (2) ➤ L'utilisateur choisit l'arrondissement et valide (3) ➤ Le système demande confirmation de l'opération de suppression de cet établissement (4) ➤ L'utilisateur confirme la suppression (5) ➤ Le système valide l'opération (6)
<u>Post-condition</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'arrondissement ne fait plus partie du département

- *valider un arrondissement*

<u>Cas d'utilisation</u> : valider un arrondissement
<u>Acteurs</u> : délégué département, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet l'inscription effective d'un arrondissement en lui associant un département.
<u>Pré-conditions</u> : <ul style="list-style-type: none"> ➤ l'utilisateur doit être un délégué département ou l'administrateur système qui est authentifié par un login et un mot de passe ➤ L'arrondissement doit être nouveau dans le département.
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un arrondissement (1) ➤ Le système lui demande de choisir l'arrondissement (2) ➤ L'utilisateur choisit l'arrondissement puis appuis sur suivant (3) ➤ Le système valide l'opération (4)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'arrondissement est ajouté dans la liste des arrondissements de ce département

- L'arrondissement est de ce fait retiré de la liste d'attente

II. Cas d'utilisation gestion département

<u>Cas d'utilisation</u> : gestion département
<u>Acteurs principaux</u> : délégué régional, Administrateur système
<u>Description</u> : le délégué régional est responsable de la gestion du département. Il procède à l'ajout des nouveaux départements, la modification de certaines informations et la suppression des départements. On peut afficher la liste des départements inscrits dans une période, voir la fiche d'un département, les statistiques des départements.

DESCRIPTION DES SCENARIOS

- **Ajouter un département**

<u>Cas d'utilisation</u> : Ajouter un département
<u>Acteurs</u> : délégué régional, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'ajouter un nouvel département dans une région.
<u>Pré-condition</u> : l'utilisateur doit être un délégué régional ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à ajouter un département (1) ➤ Le système lui demande de fournir le code de département (2) ➤ L'utilisateur fournit le code de département et passe à suivant (3) ➤ Le système vérifie le code (4) ➤ Le système présente le formulaire d'inscription à l'utilisateur (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5) ➤ L'utilisateur remplit le formulaire et valide (6) ➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<u>Description alternative 1</u> : le code existe déjà <ul style="list-style-type: none"> ➤ (1), (2), (3) ➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre code car celui-ci existe déjà (4) ➤ (5), (6), (7)
<u>Description alternative 2</u> : les champs obligatoires non remplis <ul style="list-style-type: none"> ➤ (1), (2), (3), (4), (5), (6) ➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant de remplir tous les champs obligatoires (7')

➤ (7)
<u>Post-conditions</u> :
<ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met le département dans la liste d'attente pour validation par le délégué régional.

- ***Modifier un département***

<u>Cas d'utilisation</u> : Modifier un département
<u>Acteurs</u> : délégué régional, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier les informations d'un département
<u>Pré-condition</u> : l'utilisateur doit être un délégué régional ou l'administrateur système qui est authentifié par un login et un mot de passe.
<u>Description nominale</u> :
<ul style="list-style-type: none"> ➤ L'utilisateur demande à modifier un département (1) ➤ Le système lui demande de choisir le département à modifier (2) ➤ L'utilisateur choisit le département et valide (3) ➤ Le système lui présente le formulaire d'inscription de le département avec les anciens paramètres (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5) ➤ L'utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<u>Post-conditions</u> :
<ul style="list-style-type: none"> ➤ Le système met à jour la base de données. ➤ Le système met à jour les informations du département.

- ***Supprimer un département***

<u>Cas d'utilisation</u> : Supprimer un département
<u>Acteurs</u> : délégué département, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de supprimer un département dans la région
<u>Pré-condition</u> : l'utilisateur doit être un délégué département ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> :
<ul style="list-style-type: none"> ➤ L'utilisateur demande à supprimer un département (1) ➤ Le système lui demande de choisir le département à supprimer (2) ➤ L'utilisateur choisit le département et valide (3) ➤ Le système demande confirmation de l'opération de suppression de ce département

<p>(4)</p> <ul style="list-style-type: none"> ➤ L'utilisateur confirme la suppression (5) ➤ Le système valide l'opération (6)
<p><u>Post-condition</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le département ne fait plus partie du département

- *valider un département*

<p><u>Cas d'utilisation</u> : valider un département</p>
<p><u>Acteurs</u> : délégué régional, Administrateur Système</p>
<p><u>Description</u> : ce cas d'utilisation permet l'inscription effective d'un département en lui associant une région.</p>
<p><u>Pré-conditions</u> :</p> <ul style="list-style-type: none"> ➤ l'utilisateur doit être un délégué régional ou l'administrateur système qui est authentifié par un login et un mot de passe ➤ Le département doit être nouveau dans la région.
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un département (1) ➤ Le système lui demande de choisir le département (2) ➤ L'utilisateur choisit le département puis appuis sur suivant (3) ➤ Le système valide l'opération (4)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le département est ajouté dans la liste des départements de cette région ➤ Le département est de ce fait retiré de la liste d'attente

III. *Cas d'utilisation gestion de la région*

<p><u>Cas d'utilisation</u> : gestion région</p>
<p><u>Acteurs principaux</u> : Administrateur système</p>
<p><u>Description</u> : l'administrateur système est responsable de la gestion de la région. Il procède à l'ajout de nouvelles régions, la modification de certaines informations et la suppression des régions. On peut afficher la liste des régions inscrits dans une période, voir la fiche d'une région, les statistiques des régions.</p>

DESCRIPTION DES SCENARIOS

- **Ajouter une région**

<u>Cas d'utilisation</u> : Ajouter une région
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'ajouter une nouvelle région.
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à ajouter une région (1)➤ Le système lui demande de fournir le code d'une région (2)➤ L'utilisateur fournit le code de département et passe à suivant (3)➤ Le système vérifie le code (4)➤ Le système présente le formulaire d'inscription à l'utilisateur (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5)➤ L'utilisateur remplit le formulaire et valide (6)➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<u>Description alternative 1</u> : le code existe déjà <ul style="list-style-type: none">➤ (1), (2), (3)➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre code car celui-ci existe déjà (4)➤ (5), (6), (7)
<u>Description alternative 2</u> : les champs obligatoires non remplis <ul style="list-style-type: none">➤ (1), (2), (3), (4), (5), (6)➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant de remplir tous les champs obligatoires (7')➤ (7)
<u>Post-conditions</u> : <ul style="list-style-type: none">➤ Le système met à jour la base de données➤ Le système met la région dans la liste d'attente pour validation par le administrateur système.

- **Modifier une région**

<u>Cas d'utilisation</u> : Modifier une région
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier les informations d'une région
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par

un login et un mot de passe.

Description nominale :

- L'utilisateur demande à modifier une région (1)
- Le système lui demande de choisir la région à modifier (2)
- L'utilisateur choisit la région et valide (3)
- Le système lui présente le formulaire d'inscription de la région avec les anciens paramètres (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5)
- L'utilisateur modifie les paramètres souhaités et valide (6)
- Le système valide l'opération (7)

Post-conditions :

- Le système met à jour la base de données.
- Le système met à jour les informations de la région.

- ***Supprimer une région***

Cas d'utilisation : Supprimer une région

Acteurs : Administrateur Système

Description : ce cas d'utilisation permet de supprimer une région du système

Pré-condition : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe

Description nominale :

- L'utilisateur demande à supprimer une région (1)
- Le système lui demande de choisir la région à supprimer (2)
- L'utilisateur choisit la région et valide (3)
- Le système demande confirmation de l'opération de suppression de cette région (4)
- L'utilisateur confirme la suppression (5)
- Le système valide l'opération (6)

Post-condition :

- Le système met à jour la base de données
- Le département ne fait plus partie de la région

- ***valider une région***

Cas d'utilisation : valider une région

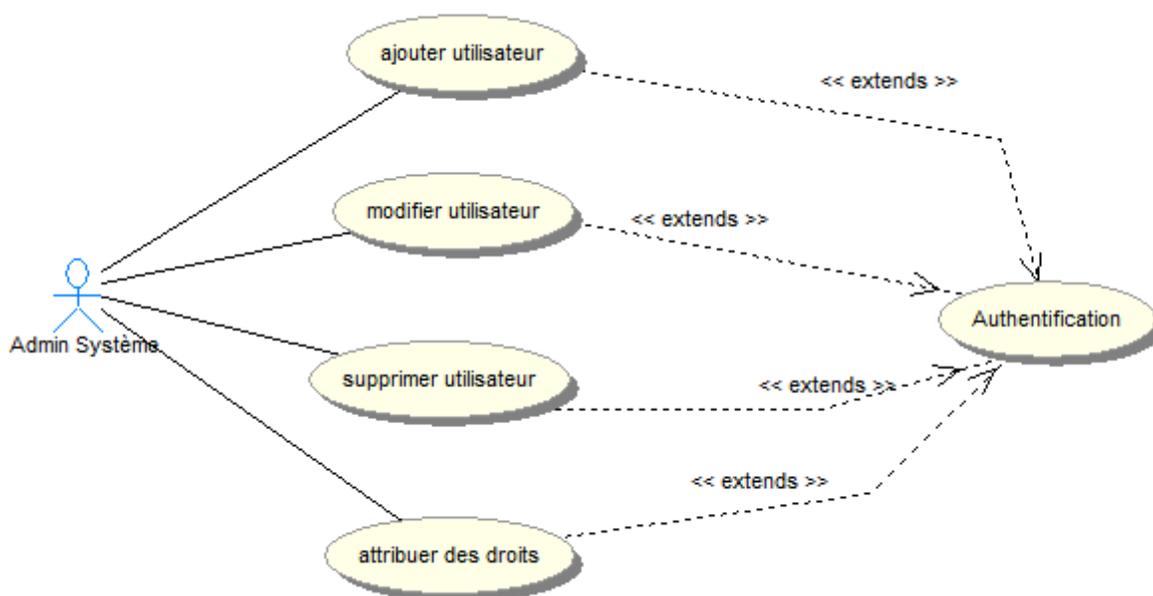
Acteurs : Administrateur Système

Description : ce cas d'utilisation permet l'inscription effective d'une région

<p><u>Pré-conditions :</u></p> <ul style="list-style-type: none"> ➤ l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe ➤ La région doit être nouvelle dans le système.
<p><u>Description nominale :</u></p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider une région (1) ➤ Le système lui demande de choisir la région (2) ➤ L'utilisateur choisit la région puis appuis sur suivant (3) ➤ Le système valide l'opération (4)
<p><u>Post-conditions :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ La région est ajoutée dans la liste des régions ➤ La région est de ce fait retiré de la liste d'attente

IV. Cas d'utilisation gestion des utilisateurs

<p><u>Cas d'utilisation :</u> gestion des utilisateurs</p>
<p><u>Acteurs principaux :</u> Administrateur système</p>
<p><u>Description :</u> l'administrateur système est responsable de la gestion des utilisateurs. Il procède à l'ajout d'un nouvel utilisateur en lui affectant des droits d'accès, la modification de certaines informations et la suppression des utilisateurs. On peut afficher la liste des utilisateurs.</p>



DESCRIPTION DES SCENARIOS

- **Ajouter un utilisateur**

<u>Cas d'utilisation</u> : Ajouter un utilisateur
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'ajouter un nouveau utilisateur dans la gestion du système.
<u>Pré-condition</u> : l'utilisateur doit l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'administrateur système demande à ajouter un utilisateur (1)➤ Le système lui demande de fournir le login de l'utilisateur (2)➤ L'administrateur système fournit le login de l'utilisateur et passe à suivant (3)➤ Le système vérifie le login (4)➤ Le système présente le formulaire d'ajout à l'administrateur système (mot de passe, statut ou grade) (5)➤ L'administrateur système remplit le formulaire et valide (6)➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<u>Description alternative 1</u> : le login existe déjà <ul style="list-style-type: none">➤ (1), (2), (3)➤ Le système envoie un message d'alerte à l'administrateur système lui demandant d'introduire un autre login car celui-ci existe déjà (4)➤ (5), (6), (7)
<u>Description alternative 2</u> : les champs obligatoires non remplis <ul style="list-style-type: none">➤ (1), (2), (3), (4), (5), (6)➤ Le système envoie un message d'alerte à l'administrateur système lui demandant de remplir tous les champs obligatoires (7')➤ (7)
<u>Post-conditions</u> : <ul style="list-style-type: none">➤ Le système met à jour la base de données➤ Le système ajoute l'utilisateur dans la liste des utilisateurs

- **Modifier un utilisateur**

<u>Cas d'utilisation</u> : Modifier un utilisateur
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier les coordonnées d'un utilisateur
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par

un login et un mot de passe
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L'administrateur système demande à modifier un utilisateur (1) ➤ Le système lui demande de choisir l'utilisateur à modifier (2) ➤ L'administrateur système choisit l'utilisateur et valide (3) ➤ Le système lui présente le formulaire d'ajout de l'utilisateur avec les anciens paramètres (login, mot de passe, statut ou grade) (5) ➤ L'administrateur système modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met à jour les informations de l'utilisateur

- ***Supprimer un utilisateur***

<p><u>Cas d'utilisation</u> : Supprimer un utilisateur</p>
<p><u>Acteurs</u> : Administrateur Système</p>
<p><u>Description</u> : ce cas d'utilisation permet de Supprimer un utilisateur de la gestion de l'établissement scolaire</p>
<p><u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L'administrateur système demande à supprimer un utilisateur (1) ➤ Le système lui présente la liste des utilisateurs à supprimer (2) ➤ L'administrateur système sélectionne l'utilisateur et appuis sur supprimer (3) ➤ Le système demande confirmation de l'opération de suppression de cet utilisateur (4) ➤ L'administrateur système confirme la suppression (5) ➤ Le système valide l'opération (6)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'utilisateur ne fait plus partie de la base de données

- ***Attribuer des droits à l'utilisateur***

<p><u>Cas d'utilisation</u> : Attribuer des taches à une catégorie d'utilisateur</p>
<p><u>Acteurs</u> : Administrateur Système</p>
<p><u>Description</u> : ce cas d'utilisation permet d'attribuer des taches à une catégorie</p>

d'utilisateur pour la gestion de l'établissement scolaire
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'administrateur système demande d'attribuer les droits à un utilisateur (1) ➤ Le système présente la liste des catégories d'utilisateur (2) ➤ L'administrateur système sélectionne la catégorie et appuis sur éditer le menu (3) ➤ Le système présente la liste des taches (4) ➤ L'administrateur système choisit les taches pouvant être exécuté par cette catégorie d'utilisateur et appuis sur valider (5) ➤ Le système demande à l'administrateur système s'il veut vraiment procéder à la modification des droits de cette catégorie d'utilisateur (6) ➤ L'administrateur système confirme la modification des droits (7) ➤ Le système valide l'opération (8)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Les droits sont octroyés à l'ensemble des utilisateurs ayant ce profil

ANNEXE B
LE CAHIER DE CHARGE

CAHIER DE CHARGE
AGREGATION EN LIGNE DE DONNEES
STATISTIQUES SUR LES ENSEIGNANTS DU
SECONDAIRE AU CAMEROUN.

INTRODUCTION GENERALE

En entreprise, les données représentent un grand capital, à considérer au même niveau que le capital financier. Il convient donc les gérer convenablement car la maîtrise de ce facteur permet à toute entreprise de limiter les risques et les erreurs. Seulement, de l'âge de la pierre à nos jours, l'esprit perfectionniste de l'homme n'a cessé de lui permettre d'améliorer sa vie quotidienne. L'évolution perpétuelle des nouvelles technologies de l'information et de communication illustrent ce phénomène. Bien gérer a toute une autre définition aujourd'hui car le temps de réaction demandé pour la résolution d'un problème est de plus en plus court, il existe une envie croissante de gain en temps, de conservation des données, l'allègement de la charge de travail mais aussi l'amélioration des conditions de travail. L'informatique a été trouvée comme solution pour satisfaire les nouvelles demandes du monde moderne.

De nos jours le défi est lancé, la gestion automatique des données est en plein essor. Les produits répondants à ce domaine inondent le marché mondial et nous pouvons constater son impact sur le monde. Toutefois, dans notre pays ce domaine n'est pas encore autant en avance que dans les pays dits développés principalement dans le domaine de l'éducation secondaire. Bien que les systèmes de gestion manuelle et semi- automatique jusqu'ici utilisés ont bien fonctionné, ils répondent de moins en moins aux exigences de ce nouveau monde. En effet l'administration, pouvant être considérée comme une entreprise parce que fonctionnant de la même manière, la solution informatique dans sa gestion est un chemin inévitable si celle-ci veut être aussi efficace que les entreprises privées. Ainsi, voulant profiter de l'évolution continue en terme infrastructurel dans notre pays, nous nous proposons de développer un système informatique en ligne qui permettra d'optimiser la gestion du personnel enseignant du secondaire au Cameroun et qui nous permettra en même temps d'avoir de manière plus simple et rapide des statistiques sur ce secteur d'activité dans notre pays.

ORGANISATION DU PROJET

Il s'agit ici d'organiser fonctionnellement et structurellement le groupe du projet ainsi que ses relations avec l'extérieur.

NOM DES PARTICIPANTS AU PROJET	RESPONSABILITES
EBONG BEYEKE JUSTINE DIANE	Analyste et Concepteur (chef de projet)
BELIBI NDJIE ALBERT	Analyste, Concepteur et Développeur
ESSOMO MBOUTCHOUANG ERIC STEVE	Analyste, Concepteur et Développeur

Participants externes : Dr AZEGUE ONDOA (encadreur du projet)

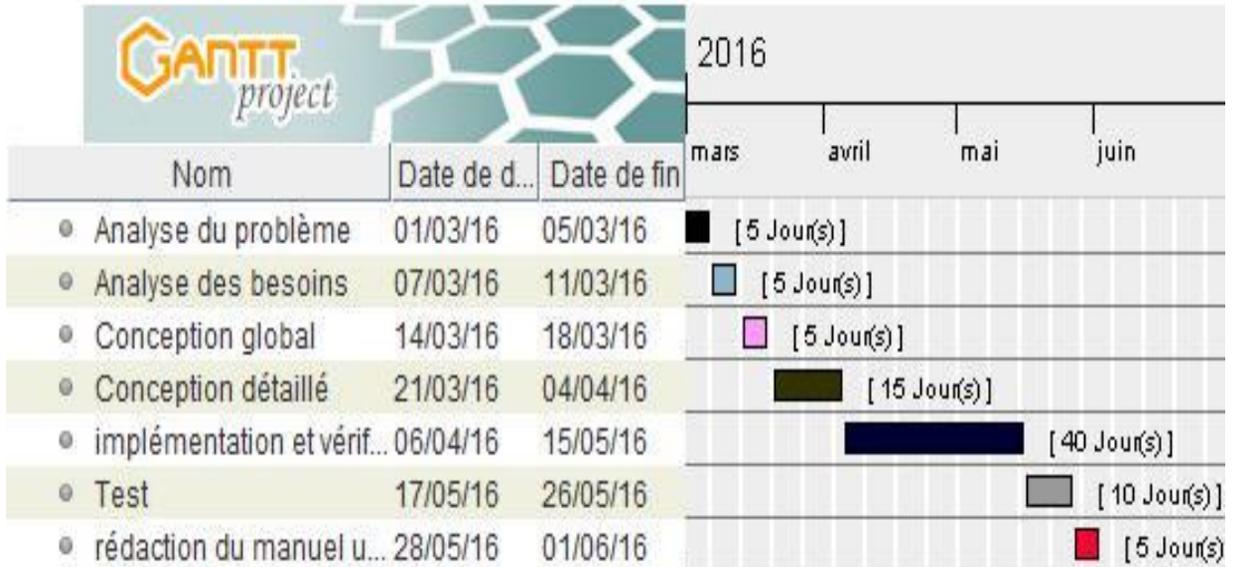
STRUCTURE DU PROJET

Il s'agit ici pour nous de décrire toutes les tâches du projet en découpant chaque phase en tâches principales puis en tâches détaillées.

TACHES				DUREE (EN JOURS)	CONTRAINTES
1	Analyse	a	Analyse du problème	5	
		b	Analyse des besoins	5	Après 1.a
2	Conception	a	Conception globale	5	Après 1
		b	Conception détaillée	15	Après 2.a
3	Implémentation et vérification			40	Après 2
4	Test			10	Après 2 et 3
5	Rédaction du manuel utilisateur			5	Après 5

PLAN DE DEROULEMENT DU PROJET

Nous présentons l'ordonnancement de nos diverses tâches à travers le diagramme de GANTT.



CHAPITRE 1 : PRESENTATION GENERALE DU PROJET

I. PRESENTATION DU PROJET

L'évolution considérable des nouvelles technologies de l'information et de la communication et leur accessibilité de plus en plus grandissante à un grand nombre dans notre pays renforce la volonté de l'administration de se moderniser afin de répondre aux exigences du monde aujourd'hui. Il s'avère par conséquent normal et même vital de profiter du réseau internet pour optimiser son rendement. En ce qui concerne les enseignants du secondaire du Cameroun, cette technologie pourrait permettre si elle est mise à contribution aux administrateurs de mieux les gérer sans que la distance ni les difficultés liés aux lenteurs, erreurs et limites humaines ne soient un problème. Dans ce cadre le présent projet consiste à réaliser une application web qui permettra de collecter des informations sur les enseignants, leur manipulation, leur stockage et leur restitution qui se fera sous un format exploitable par les tous les utilisateurs du système.

II. PRESENTATION DE L'ORGANISME D'ACCUEIL

Au Cameroun, l'organisme chargé de coordonner le fonctionnement de l'éducation secondaire est le ministère de l'enseignement secondaire (*MINESEC*). C'est lui qui surveille et définit la politique de l'éducation à l'échelle nationale. Le ministère a installé à la tête de chacune des régions du pays des délégations régionales qui sont chargées de coordonner les activités liées à l'éducation secondaire à l'échelle de leurs régions respectives. De même, nous avons au niveau de chacun des départements une délégation départementale dont les activités se limiteront à l'échelle de leurs départements respectifs. Enfin, nous avons les établissements scolaires. Et dans chacune de ces structures on trouve des enseignants qui y travaillent. Le flux d'information dans cette entreprise (*MINESEC*) peut être schématisé de la manière suivante :

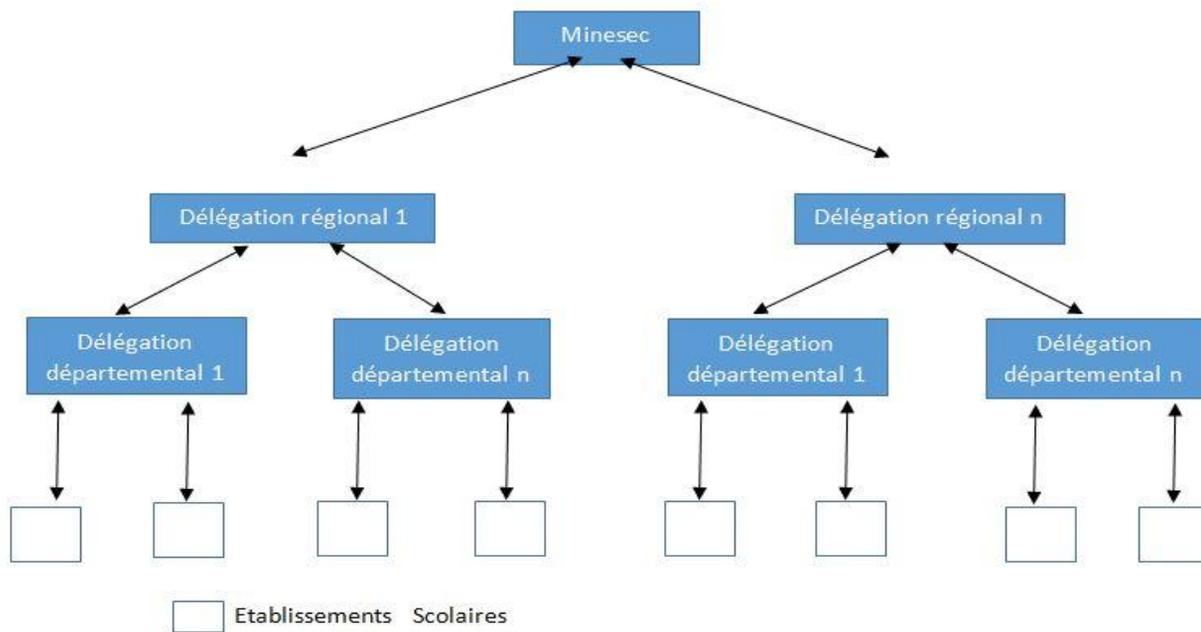


Schéma du flux d'information du MINESEC

CHAPITRE 2 : ETAT DE L'ART

Ce chapitre présente un état des lieux : il s'agit d'une étude de l'existant suivie de critique permettant au projet de présenter une amélioration résumant l'ensemble des solutions retenues.

I. ETUDE L'EXISTANT

Pour mener à bien sa mission, le MINESEC a mis en place un système manuel de collecte et stockage des données qui peut se résumer comme il suit :

- les enseignants lors du recensement du personnel remplissent un dossier (fiche de recensement) qui est déposé au niveau du chef d'établissement.
- le chef d'établissement collecte toutes les données relatives à son établissement et les envoie à sa délégation départementale.
- le délégué départemental rassemble toutes les données relatives à son département et les envoie à sa délégation régionale.
- le délégué régional rassemble toutes les données relatives à son département et les envoie au ministère où elles seront exploitées et stockées dans des archives papier.

1. CRITIQUE DE L'EXISTANT

L'étude de l'existant nous a permis de dégager un certain nombre de lacunes.

➤ *Le processus manuel*

- L'abondance des documents peut ralentir les services.
- Risque de mélanger les documents : ce qui peut être fatal.
- La réponse aux besoins réels de personnel sur le terrain n'est pas aisée
- La rapidité dans le traitement des opérations nécessaires pour un meilleur fonctionnement n'est pas assurée
- La rapidité dans la collecte des informations sur le terrain nécessaire pour un meilleur fonctionnement n'est pas assurée

➤ *Les solutions informatiques*

Le SIGIPES : Le Système informatique de gestion intégré des personnels de l'Etat et de la solde gère la totalité du personnel de la fonction publique donc peut nous fournir des statistiques en ce qui concerne les enseignants du minesec. Mais il ne peut pas aider les responsables de l'éducation à la prise décision. Car ne nous renseigne pas sur les besoins du système éducatif pour mieux être plus performant. Et le SIGIPES ne nous donne aucune information sur la situation du secteur privé de l'éducation secondaire

2. PRESENTATION DU TRAVAIL DEMANDE

Au regard des problèmes relevés, nous proposons de concevoir une application qui résoudra la majorité sinon tous les problèmes cités ci-dessus. Ainsi, elle devra apporter les solutions suivantes :

- Performance
- Respect des principes des Interfaces Homme/Machine (IHM) tels que l'ergonomie et la fiabilité.
- Réduction les tâches manuelles qui nous permettraient de gagner en spatio-temporel
- Archivage des informations
- Evolutivité et paramétrable

Elle devra en plus être dotée d'une interface conviviale et être facile à utiliser tout en étant plus efficace.

3. MODELE ET METHODOLOGIE ADOPTES

a. Le modèle de développement

Un modèle de développement logiciel présente toutes les étapes du développement, de sa conception à sa disparition. L'objectif d'un tel découpage est de permettre de définir des jalons intermédiaires permettant la validation du développement logiciel, c'est-à-dire la conformité du logiciel avec les besoins exprimés, et la vérification du processus de développement. L'origine de ce découpage provient du constat que les erreurs ont un coût d'autant plus élevé qu'elles sont détectées tardivement dans le processus de réalisation. Le cycle de vie permet de détecter les erreurs au plus tôt et ainsi de maîtriser la qualité du logiciel, les délais de sa réalisation et les coûts associés. Pour la réalisation de ce système, nous avons opté pour le modèle de cycle de vie en V qui part du principe que les procédures de vérification de la conformité du logiciel aux spécifications doivent être élaborées dès les phases de conception. L'énorme intérêt du cycle en V est qu'il est un excellent support à la formalisation de notre relation avec le futur-utilisateur, Il nous oblige à réfléchir aux différents aspects de sa demande. La phase de spécification nous permet de vérifier qu'on a bien compris la demande des utilisateurs, en effet, c'est l'équipe de gestion des services Internet de l'INBMI qui valide la spécification.

b. Méthodologie

Le plus grand avantage d'une méthode orientée objet est qu'elle permet de structurer un système sans centrer l'analyse uniquement sur les données ou uniquement sur les traitements mais sur les deux à la fois. Une telle approche a pour but de modéliser les propriétés statiques et dynamiques de l'environnement du système. Elle met en correspondance le problème et la solution, en préservant la structure et le comportement du système analysé. Ceci, nous a conduit à adopter l'approche orientée objet pour modéliser notre système en se basant sur les diagrammes UML.

c. Outils de développement

Pour la réalisation du projet nous avons choisi de travailler avec :

- ArgoUML pour la modélisation des diagrammes
- NotePad++ comme éditeur de texte
- WampServer (qui contient le serveur Apache et MySQL pour la gestion des bases de données)
- PHP, HTML et JavaScript comme langage de programmation
- Fpdf pour imprimer en pdf
- JpGraph pour dessiner les graphiques

➤ *ArgoUML*

ArgoUML est un logiciel libre de création de diagrammes UML. Programmé en Java, il est édité sous licence EPL 1.0. Il est multilingue, supporte la génération de code et l'ingénierie inverse. ArgoUML supporte sept types de diagramme : cas d'utilisation, classes, séquence, état, collaboration et activité et déploiement. Il permet aussi la génération de code à partir de diagrammes de classes dans les langages : Java, C++, PHP, C# et SQL.

➤ *Notepad++*

Notepad++ est un éditeur de texte générique codé en C++, qui intègre la coloration syntaxique de code source pour les langages et fichiers C, C++, Java, C#, XML, HTML, PHP, JavaScript, Python etc. ainsi que pour tout autre langage informatique, car ce logiciel propose la possibilité de créer ses propres colorations syntaxiques pour un langage quelconque. Ce logiciel, basé sur la composante Scintilla, a pour but de fournir un éditeur léger (aussi bien au niveau de la taille du code compilé que des ressources

occupées durant l'exécution) et efficace. Il est également une alternative au bloc-notes de Windows (d'où le nom). Le projet est sous licence GPL.

Il ne bloque pas le fichier en cours d'édition et détecte toute modification apportée à celui-ci par un autre programme (il propose de le recharger).

➤ **WampServer**

WampServer est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL.

Il dispose d'une interface d'administration permettant de gérer et d'administrer ses serveurs au travers d'une icône près de l'horloge de Windows.

➤ **MySQL**

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.

➤ **HTML**

L'Hypertext Markup Language, généralement abrégé HTML, est le format de données conçu pour représenter les pages web. C'est un langage de balisage permettant d'écrire de l'hypertexte, d'où son nom. HTML permet également de structurer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie, et des programmes informatiques. Il permet de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. Il est souvent utilisé conjointement avec des langages de programmation (*JavaScript*) et des formats de présentation (*feuilles de style en cascade*).

➤ **PHP**

PHP: Hypertext Preprocessor, plus connu sous son sigle PHP (*acronyme récursif*), est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale. PHP est un langage impératif orienté objet qui a permis de créer un grand nombre de sites web célèbres, comme Facebook, Wikipédia, etc. Il est considéré comme la base de la création des sites Internet dits dynamiques.

➤ ***JavaScript***

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs. C'est un langage orienté objet à prototype, c'est-à-dire que les bases du langage et ses principales interfaces sont fournies par des objets qui ne sont pas des instances de classes, mais qui sont chacun équipés de constructeurs permettant de créer leurs propriétés, et notamment une propriété de prototypage qui permet d'en créer des objets héritiers personnalisés. En outre, les fonctions sont des objets de première classe.

➤ ***Fpdf***

FPDF est une classe PHP libre qui permet de créer des fichiers PDF directement depuis PHP sans dépendre d'une librairie extérieure.

➤ ***JpGraph***

JpGraph est une librairie PHP dédiée à la représentation graphique de données. Il produit des images. C'est au programmeur de fournir l'ensemble des données nécessaires à la réalisation du graphique voulu.

CHAPITRE 3 : ANALYSE DES BESOINS ET SPECIFICATIONS

Dans ce chapitre, nous présenterons les objectifs de notre application, ce qui nous amène à identifier les possibilités du système et les besoins des utilisateurs que nous essayerons de projeter dans des diagrammes de cas d'utilisations globales et détaillés.

I. ANALYSE DES BESOINS

L'application envisagée devra satisfaire des besoins fonctionnels qui seront exécutés par le système et des besoins non fonctionnels qui vont perfectionner la qualité logicielle du système.

1. LES BESOINS FONCTIONNELS

Les besoins fonctionnels ou besoins métiers représentent les actions que le système doit exécuter, il ne devient opérationnel que s'il les satisfait. Ces besoins se résument dans le tableau suivant :

NOM RUBRIQUE	BESOINS FONCTIONNELS	DESCRIPTIF
Gestion des utilisateurs (enseignant, censeur, surveillant général, personnel de délégation, etc)	<ul style="list-style-type: none">- Créer un compte utilisateur- Supprimer un compte- Valider un formulaire (profil)- rejeter un formulaire (profil)- Transférer un utilisateur à un autre service- Consulter la liste des transferts reçus- Valider un transfert- Afficher la liste des utilisateurs sous son administration- Consulter les statistiques	Chaque administrateur est chargé de la gestion des utilisateurs qui se trouvent dans sa structure. Et ne peuvent pas gérer ceux qui ne se trouvent pas sous ses ordres

Gestion des formulaires (profils)	<ul style="list-style-type: none"> - Remplir le formulaire - Modifier son formulaire - Afficher son formulaire - Consulter les statistiques 	Chaque détenteur de compte obtient un formulaire lorsque son compte est créé. il le remplit et l'envoie à son administrateur pour validation. Ce formulaire pourra le modifier après
Gestion des structures (délégations régional, délégation départemental, établissement)	<ul style="list-style-type: none"> - Créer une structure - Supprimer une structure - Transférer un administrateur à une structure - Changer l'administrateur d'une Structure - Consulter les statistiques 	Les structures sont gérées par l'administrateur principal.
Gestion des besoins d'un établissement	<ul style="list-style-type: none"> - Remplir le formulaire d'un établissement - Modifier le formulaire d'un établissement - Consulter les statistiques 	Un établissement est géré par son administrateur qui est le chef d'établissement.
Gestion de département	<ul style="list-style-type: none"> - Valider le formulaire d'un établissement - Rejeter le formulaire d'un établissement - Afficher la liste des établissements - Afficher la liste des utilisateurs - Afficher la liste des chefs d'établissement - Consulter les statistiques 	Un département est géré par son l'administrateur qui est le délégué départemental.

Gestion de région	<ul style="list-style-type: none"> - Afficher la liste des établissements - Afficher la liste des utilisateurs - Afficher la liste des chefs d'établissement - Afficher la liste des délégués départementaux - Consulter les statistiques 	Une région est gérés par son l'administrateur qui est le délégué régional.
Gestion des administrateurs	<ul style="list-style-type: none"> - Créer compte administrateur - Transformer un utilisateur en administrateur - Supprimer un administrateur - Consulter les statistiques 	les administrateurs secondaires sont gérés par l'administrateur principal
Gestion des visiteurs	<ul style="list-style-type: none"> - Consulter les statistiques 	Ce sont tous ceux qui peuvent avoir accès

2. LES BESOINS NON FONCTIONNELS

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du système mais plutôt identifient des contraintes internes et externes du système. Les principaux besoins non fonctionnels de notre application ce résumé dans les points suivants :

- Le code doit être clair pour permettre des futures évolutions ou améliorations
- L'ergonomie : l'application offre une interface conviviale et facile à utiliser ;
- La sécurité : l'application doit respecter la confidentialité des données ;
- Garantir l'intégrité et la cohérence des données à chaque mise à jour et à chaque insertion.

3. LES DIAGRAMMES DE CAS D'UTILISATIONS

Ils montrent les interactions fonctionnelles entre les acteurs et le système

Acteur : rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système étudié. Un acteur participe à au moins un cas d'utilisation.

Cas d'utilisation (use case) : ensemble de séquences d'actions réalisées par le système produisant un résultat observable intéressant pour un acteur particulier. Collection de scénarios reliés par un objectif utilisateur commun.

Association : utilisée dans ce type de diagramme pour relier les acteurs et les cas d'utilisation par une relation qui signifie simplement « participe à ».

Inclusion : le cas d'utilisation de base en incorpore explicitement un autre, de façon obligatoire, à un endroit spécifié dans ses enchaînements.

Extension : le cas d'utilisation de base en incorpore implicitement un autre, de façon optionnelle, à un endroit spécifié indirectement dans celui qui procède à l'extension

Généralisation : les cas d'utilisation descendants héritent de la description de leur parent commun. Chacun d'entre eux peut néanmoins comprendre des relations spécifiques supplémentaires avec d'autres acteurs ou cas d'utilisation.

a. Identification des acteurs

Un acteur représente un rôle joué par une personne qui interagit avec le système. Par définition, les acteurs sont à l'extérieur du système. Les acteurs se recrutent parmi les utilisateurs du système et aussi parmi les responsables de sa configuration et de sa maintenance. D'où, les acteurs potentiels qui risquent d'interagir avec l'application sont:

- ***L'administrateur principal*** : c'est le gestionnaire suprême du système. Il crée et supprime des comptes aux autres utilisateurs du système. Il est le seul à pouvoir changer les statuts de ces derniers (*les faire passer du statut de simple utilisateur à administrateur intermédiaire*). Il crée et supprime les établissements, affecte à chaque structure un chef de structure, définit les droits des administrateurs intermédiaires.
- ***L'administrateur régional*** : il peut remplir et modifier ses informations. il valide le formulaire rempli par les enseignants travaillant dans sa structure et les administrateurs départementaux de sa région.
- ***L'administrateur départemental*** : il peut remplir et modifier ses informations. Il valide le formulaire rempli par les enseignants travaillant dans sa structure et les chefs d'établissement se trouvant dans son département.
- ***L'administrateur d'établissement*** : il peut remplir et modifier ses informations. Il valide le formulaire rempli par les enseignants travaillant dans sa structure, définit les besoins en personnel enseignant de son établissement, transfère les enseignants de sa structure vers une autre et valide le transfert des enseignants qui lui sont envoyés.

Nb : tous les administrateurs peuvent créer des comptes aux utilisateurs mais seul l'administrateur principal peut les supprimer.

- **Les enseignants** : ils peuvent remplir et modifier des informations dans leur compte.
- **Les visiteurs** : au même titre que les utilisateurs principaux du système, ils peuvent consulter les informations mises à leur disposition.

Nb : un utilisateur est tout acteur qui est détenteur d'un compte.

b. Les différents cas d'utilisation

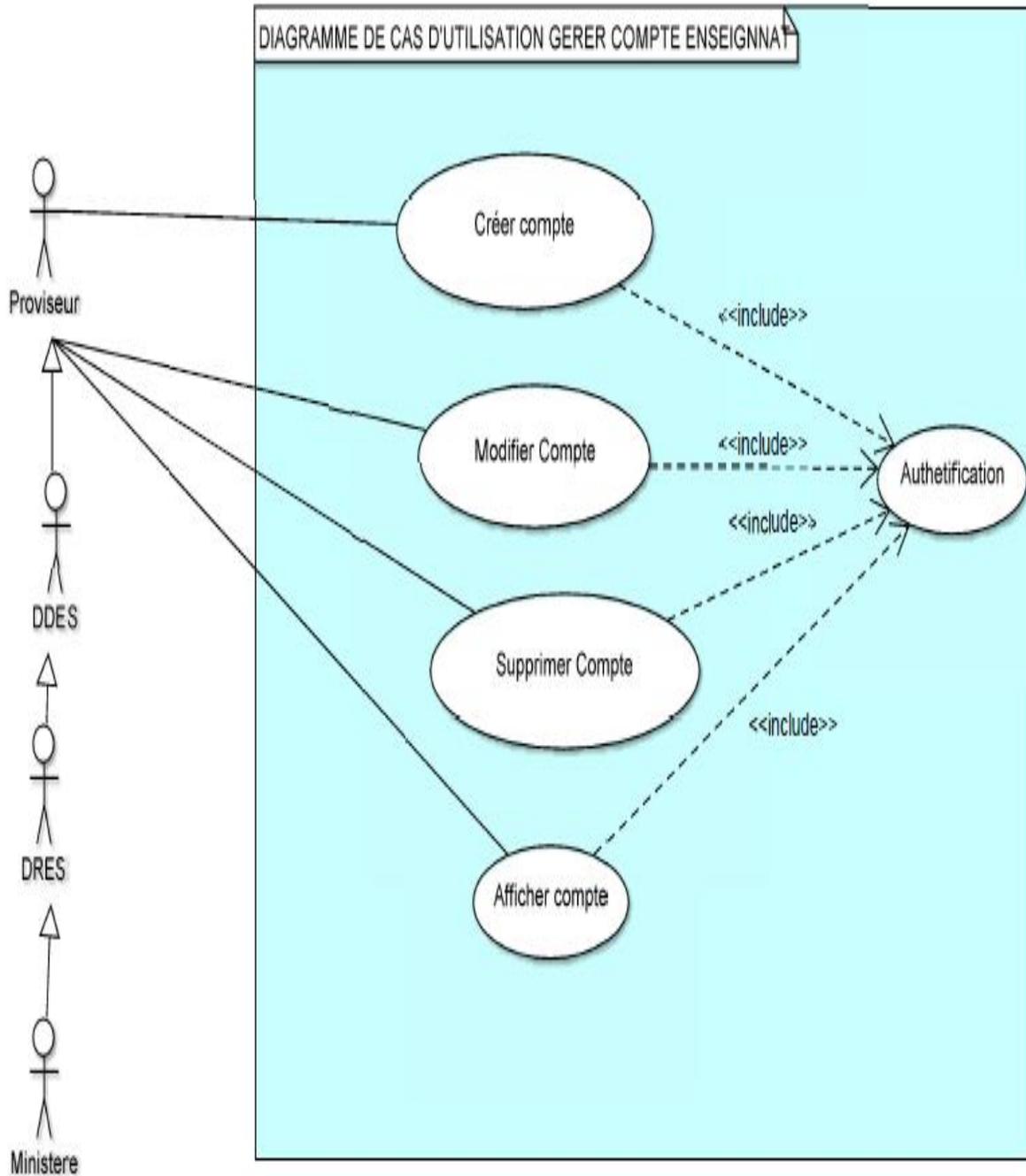
L'étude de cas d'utilisation a pour objectif de déterminer ce que chaque utilisateur attend du système. La détermination du besoin est basée sur la représentation de l'interaction entre l'acteur et le système.

▪ Cas d'utilisation préliminaire

- **Authentification** : permet d'identifier chaque utilisateur, et de lui donner l'accès aux fonctionnalités propices.
- **Gestion du compte de l'enseignant** : Créer compte, modifier compte, supprimer compte
- **Cas d'utilisation gestion des informations d'un compte/ gestion du formulaire** : remplir formulaire, modifier formulaire, afficher la liste des formulaires en attente,
- **Cas d'utilisation valider un formulaire** : afficher un formulaire, valider un formulaire, rejeter un formulaire
- **Cas d'utilisation gestion du transfert des enseignant** : transférer un enseignant, afficher la liste des transferts, valider un transfert
- **Cas d'utilisation gestion des besoins d'un établissement** : remplir les besoin, modifier les besoin, afficher la liste des besoins
- **Cas d'utilisation valider les besoins d'un établissement** : afficher les besoin, valider les besoin, rejeter le besoin
- **Cas d'utilisation gestion des comptes administrateur** : créer un compte, modifier un compte, supprimer un compte, attribuer des droits
- **Cas d'utilisation gestion des régions** : créer région, modifier région, supprimer région
- **Cas d'utilisation gestion des départements** : créer département, modifier département, supprimer département
- **Cas d'utilisation gestion des arrondissements** : créer arrondissement, modifier arrondissement, supprimer arrondissement



- Cas d'utilisation gestion du compte de l'enseignant



Cas d'utilisation : gestion du compte de l'enseignant

Acteurs principaux : Provisieur, Administrateur système

Description : le Provisieur est responsable de la gestion des enseignants. Il procède à la création des comptes des enseignants, la modification des comptes et la suppression de ceux-ci. On peut afficher la liste des comptes inscrits dans une période.

DESCRIPTION DES SCENARIOS

- **Créer compte enseignant**

<u>Cas d'utilisation</u> : créer compte enseignant
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'ajouter un nouvel enseignant dans un établissement
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à créer un compte enseignant (1)➤ Le système lui demande de fournir le matricule de l'enseignant (2)➤ L'utilisateur fournit le matricule de l'enseignant et passe à suivant (3)➤ Le système vérifie le matricule (4)➤ valide l'opération (5)
<u>Description alternative 1</u> : le matricule existe déjà <ul style="list-style-type: none">➤ (1), (2), (3)➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre matricule car celui-ci existe déjà (4)➤ (5)
<u>Post-conditions</u> : <ul style="list-style-type: none">➤ Le système met à jour la base de données➤ Le système met le compte créé dans la liste des comptes

- **Modifier un compte**

<u>Cas d'utilisation</u> : Modifier un compte
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier un compte dans l'établissement
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe

Description nominale :

- L'utilisateur demande à modifier un compte (1)
- Le système lui demande de choisir le compte à modifier (2)
- L'utilisateur choisit le compte et valide (3)
- Le système lui présente le formulaire de création du compte avec les anciens paramètres (5)
- L'utilisateur modifie les paramètres souhaités et valide (6)
- Le système valide l'opération (7)

Post-conditions :

- Le système met à jour la base de données
- Le système met à jour les informations du compte.

• **Supprimer un compte**

Cas d'utilisation : Supprimer un compte

Acteurs : Proviseur, Administrateur Système

Description : ce cas d'utilisation permet de supprimer un compte dans l'établissement

Pré-condition : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe

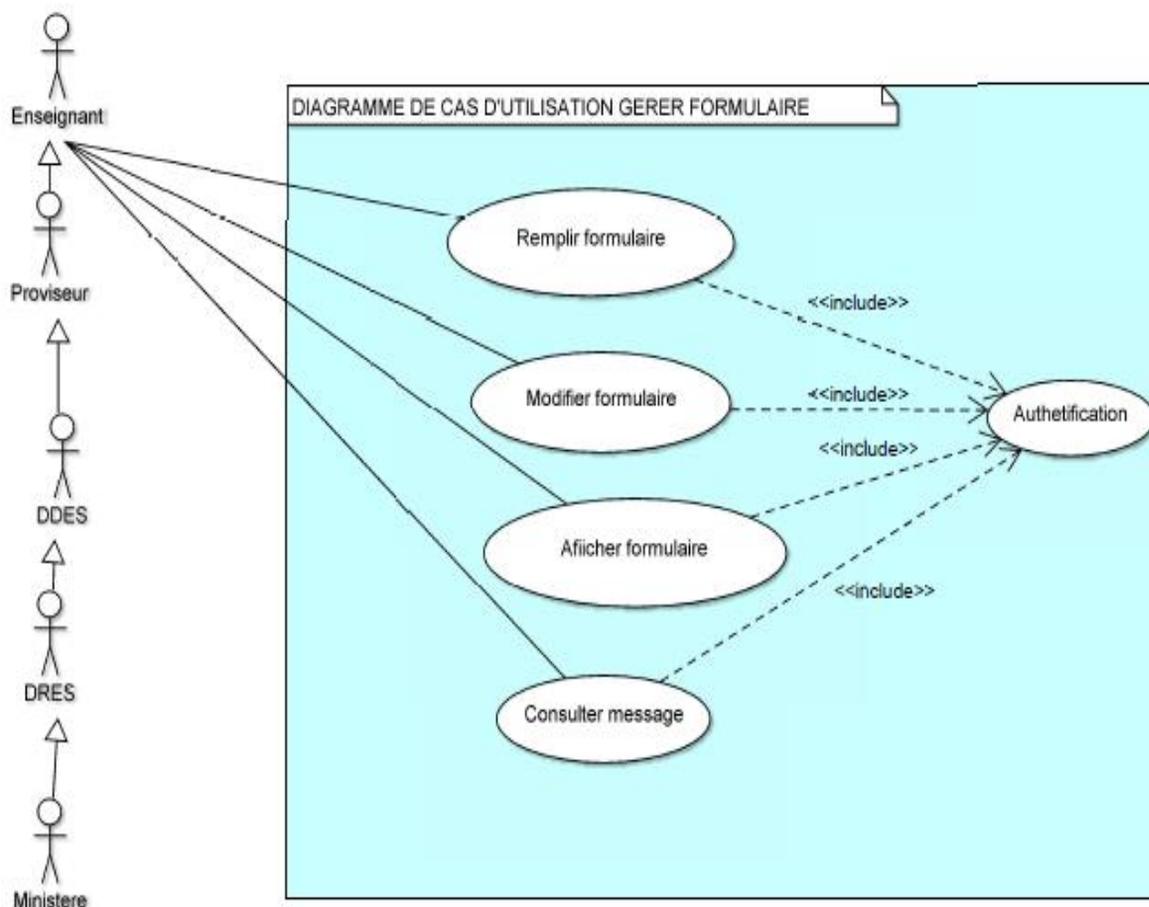
Description nominale :

- L'utilisateur demande à supprimer un compte (1)
- Le système lui demande de choisir le compte à supprimer (2)
- L'utilisateur choisit le compte et valide (3)
- Le système demande confirmation de l'opération de suppression de ce compte (4)
- L'utilisateur confirme la suppression (5)
- Le système valide l'opération (6)

Post-condition :

- Le système met à jour la base de données
- Le compte ne fait plus partie des comptes de l'établissement

▪ CAS D'UTILISATION GESTION DES INFORMATIONS D'UN COMPTE/ GESTION DU FORMULAIRE



Cas d'utilisation : gestion du formulaire
Acteurs principaux : proviseur, délégué départemental, délégué régional Administrateur système
Description chaque acteur est responsable de son formulaire qu'il peut remplir, modifier, afficher. Il peut aussi consulter les différents messages de notification qu'il reçoit sur son compte.

DESCRIPTION DES SCENARIOS

• **Remplir le formulaire**

Cas d'utilisation : remplir le formulaire
Acteurs : Proviseur, Administrateur Système
Description : ce cas d'utilisation permet de remplir les informations relatives à un enseignant

<p><u>Pré-condition</u> : l'utilisateur doit être un enseignant, proviseur, délégué départemental, délégué régional ou l'administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale</u> :</p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à remplir le formulaire (1) ➤ Le système lui demande de fournir son login et son mot de passe (2) ➤ L'utilisateur fournit son login et son mot de passe et passe à suivant (3) ➤ Le système vérifie le matricule (4) ➤ Le système présente le formulaire d'inscription à l'utilisateur (nom et prénom, date de naissance, nom du père, nom de la mère, adresse, téléphone, email, lieu de naissance,...) (5) ➤ L'utilisateur remplit le formulaire et valide (6) ➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<p><u>Description alternative 1</u> : le login et son mot de passe ne sont pas les siens</p> <ul style="list-style-type: none"> ➤ (1), (2), (3) ➤ Le système envoie un message d'alerte à l'utilisateur lui demandant d'introduire un autre matricule car celui-ci est incorrect (4) ➤ (5), (6), (7)
<p><u>Description alternative 2</u> : les champs obligatoires non remplis</p> <ul style="list-style-type: none"> ➤ (1), (2), (3), (4), (5), (6) ➤ Le système envoie un message d'alerte à l'utilisateur lui demandant de remplir tous les champs obligatoires (7') ➤ (7)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met le formulaire dans la liste d'attente pour validation

- **Modifier le formulaire**

<p><u>Cas d'utilisation</u> : Modifier un formulaire</p>
<p><u>Acteurs</u> : Proviseur, Administrateur Système</p>
<p><u>Description</u> : ce cas d'utilisation permet de modifier les informations d'un enseignant</p>

dans le système.
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à modifier un formulaire (1) ➤ Le système lui demande de choisir le formulaire à modifier (2) ➤ L'utilisateur choisit le formulaire et valide (3) ➤ Le système lui présente le formulaire d'inscription avec les anciens paramètres (nom et prénom, date de naissance, nom du père, nom de la mère, adresse, téléphone, email, lieu de naissance...) (5) ➤ L'utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met à jour les informations de l'utilisateur

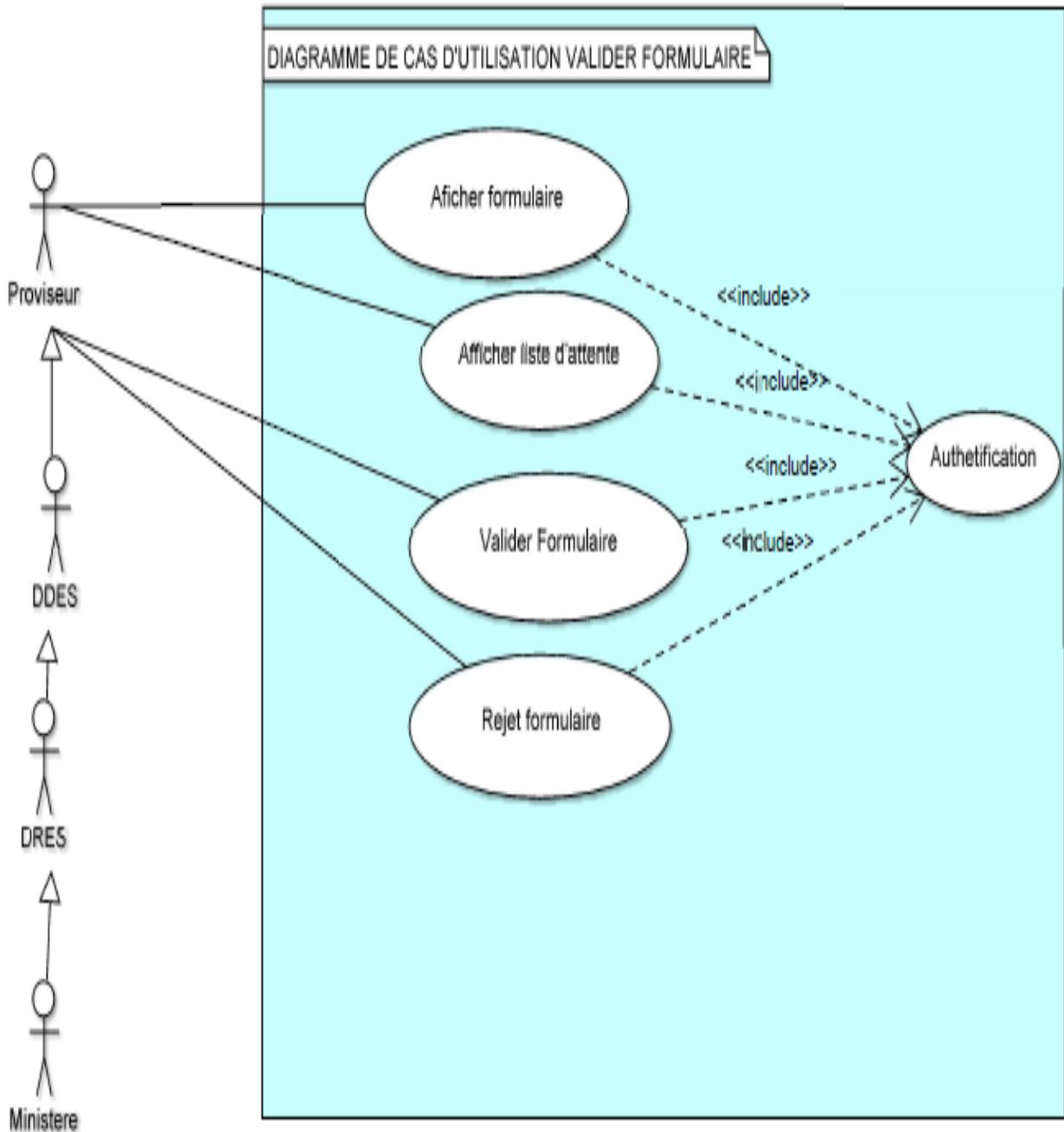
- **Afficher la liste des formulaires**

<u>Cas d'utilisation</u> : afficher la liste des formulaires
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'afficher la liste des formulaires inscrits dans le système.
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à consulter la liste des formulaires (1) ➤ Le système lui demande de choisir le niveau à consulter (établissement, département, région, pays) (2) ➤ L'utilisateur choisit le niveau en précisant son nom et valide (<u>exemple</u> : niveau départemental, département du moungo) (3) ➤ Le système lui présente la liste des formulaires inscrits dans ce niveau (5)

- **Consulter les messages de notification**

<u>Cas d'utilisation</u> : consulter les messages de notification
<u>Acteurs</u> : Enseignant, les délégués département et régionaux, proviseur, administrateur système
<u>Description</u> : ce cas d'utilisation permet de consulter les messages de notification d'un compte
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe.
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à consulter son compte (1)➤ Le système lui ouvre son compte (2)➤ L'utilisateur demande à consulter ses messages (3)➤ Le système lui présente ses messages de notifications (5)

▪ CAS D'UTILISATION VALIDER FORMULAIRE ENSEIGNANT / USER



Cas d'utilisation : valider formulaire enseignant

Acteurs principaux : proviseur, Administrateur système

Description : un administrateur est responsable des enseignants de sa structure. De ce fait il est habilité à valider leur compte de même que leur formulaire. Il est possible d'afficher un formulaire, afficher la liste d'attente, valider un formulaire ou rejeter un formulaire.

DESCRIPTION DES SCENARIOS

- **Afficher un formulaire**

<u>Cas d'utilisation</u> : afficher un formulaire
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'afficher un formulaire inscrit dans le système.
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à consulter un formulaire (1)➤ Le système lui demande de choisir le formulaire à consulter (2)➤ L'utilisateur choisit le formulaire et valide (<u>exemple</u> : niveau départemental, département du moungo) (3)➤ Le système lui présente le formulaire sollicité (5)

- **Afficher la liste d'attente**

<u>Cas d'utilisation</u> : afficher la liste d'attente des formulaires
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'afficher la liste des formulaires en attente de validation.
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à consulter la liste des formulaires en attente de validation(1)➤ Le système lui demande de choisir le niveau à consulter (établissement, département, région, pays) (2)➤ L'utilisateur choisit le niveau en précisant son nom et valide (<u>exemple</u> : niveau départemental, département du moungo) (3)➤ Le système lui présente la liste des formulaires en attente de validation dans ce niveau (5)

- **Valider un formulaire**

<u>Cas d'utilisation</u> : valider un formulaire
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet valider un formulaire inscrit dans la liste d'attente.
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un formulaire (1) ➤ Le système lui présente la liste des formulaires inscrits dans la liste d'attente (2) ➤ L'utilisateur choisi le formulaire à valider(3) ➤ Le système lui affiche le formulaire (4) ➤ L'utilisateur vérifie les informations et appuis sur OK (5)
<u>Post conditions</u> : <ul style="list-style-type: none"> ➤ Le système inscrit le formulaire dans la base de données et le nombre de formulaire augmente. ➤ Le système envoie un message pour notifier l'enseignant de la validation de son formulaire

- **Rejeter un formulaire**

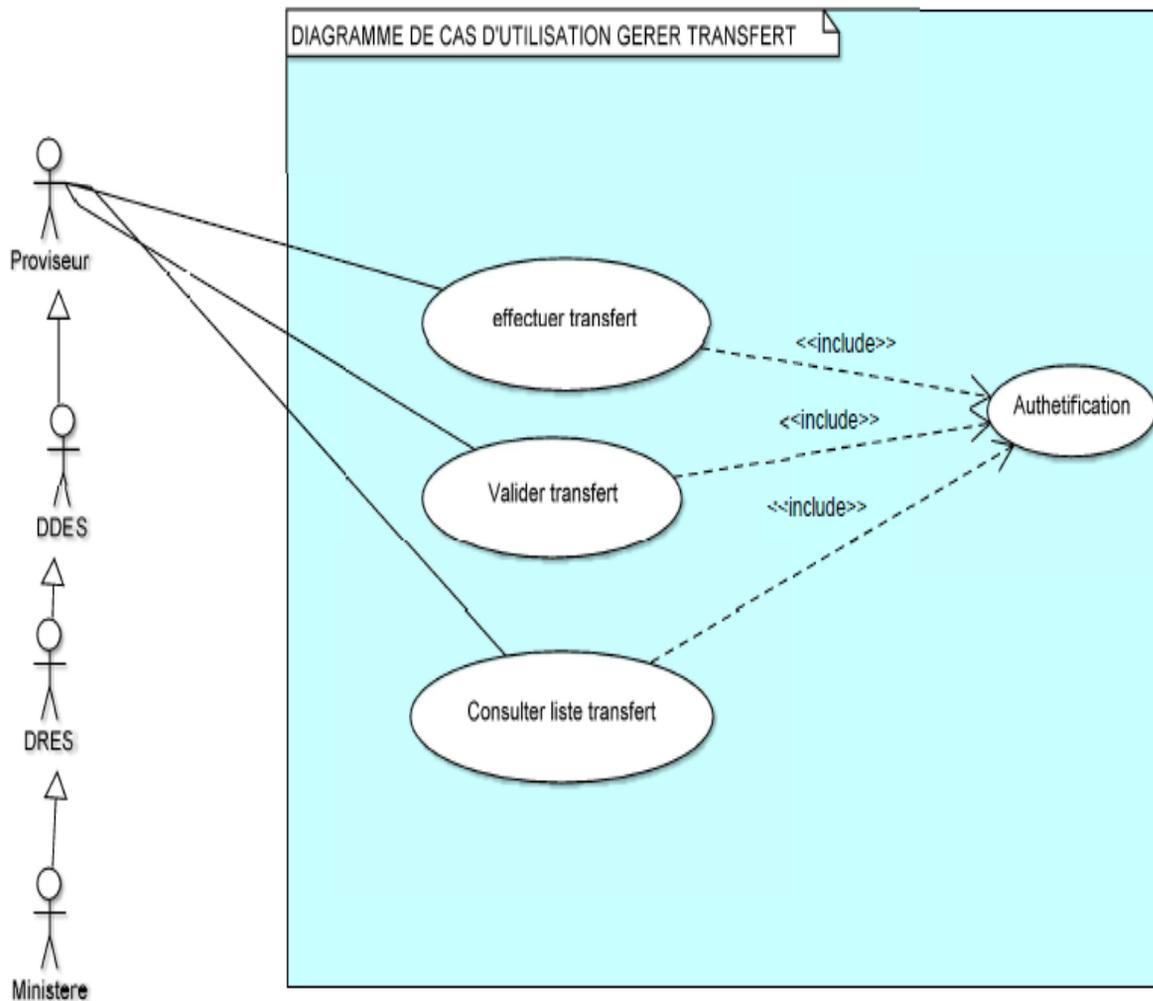
<u>Cas d'utilisation</u> : rejeter un formulaire
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de rejeter un formulaire inscrit dans la liste d'attente.
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un formulaire (1) ➤ Le système lui présente la liste des formulaires inscrits dans la liste d'attente (2) ➤ L'utilisateur choisi le formulaire à valider(3) ➤ Le système lui affiche le formulaire (4)

- L'utilisateur vérifie les informations et si elles ne sont pas conformes il annule la validation et rejette le formulaire en appuyant sur rejet. (5)

Post conditions :

- Le système inscrit le formulaire rejeté dans la liste des formulaires rejetés
- Le système envoie un message pour notifier l'enseignant du rejet de son formulaire

▪ Gestion du transfert des enseignants



DESCRIPTION DES SCENARIOS

- **Transférer un enseignant**

<ul style="list-style-type: none">• <u>Cas d'utilisation</u> : gestion du transfert des enseignants
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'envoyer un enseignant d'un établissement à un autre.
<u>Pré-conditions</u> : <ul style="list-style-type: none">➤ l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe.➤ L'enseignant doit être inscrit dans un établissement
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à envoyer un enseignant (1)➤ Le système lui demande de choisir l'enseignant à envoyer (2)➤ L'utilisateur choisit l'enseignant et valide (3)➤ Le système lui présente le formulaire de transfert (4)➤ L'utilisateur remplit le formulaire et valide (5)
<u>Post-conditions</u> : <ul style="list-style-type: none">➤ Le système met à jour la base de données➤ L'enseignant est supprimé dans la liste des enseignants de cet établissement➤ L'enseignant est désormais dans la liste d'attente de l'établissement d'accueil

- **Consulter la liste de transfert**

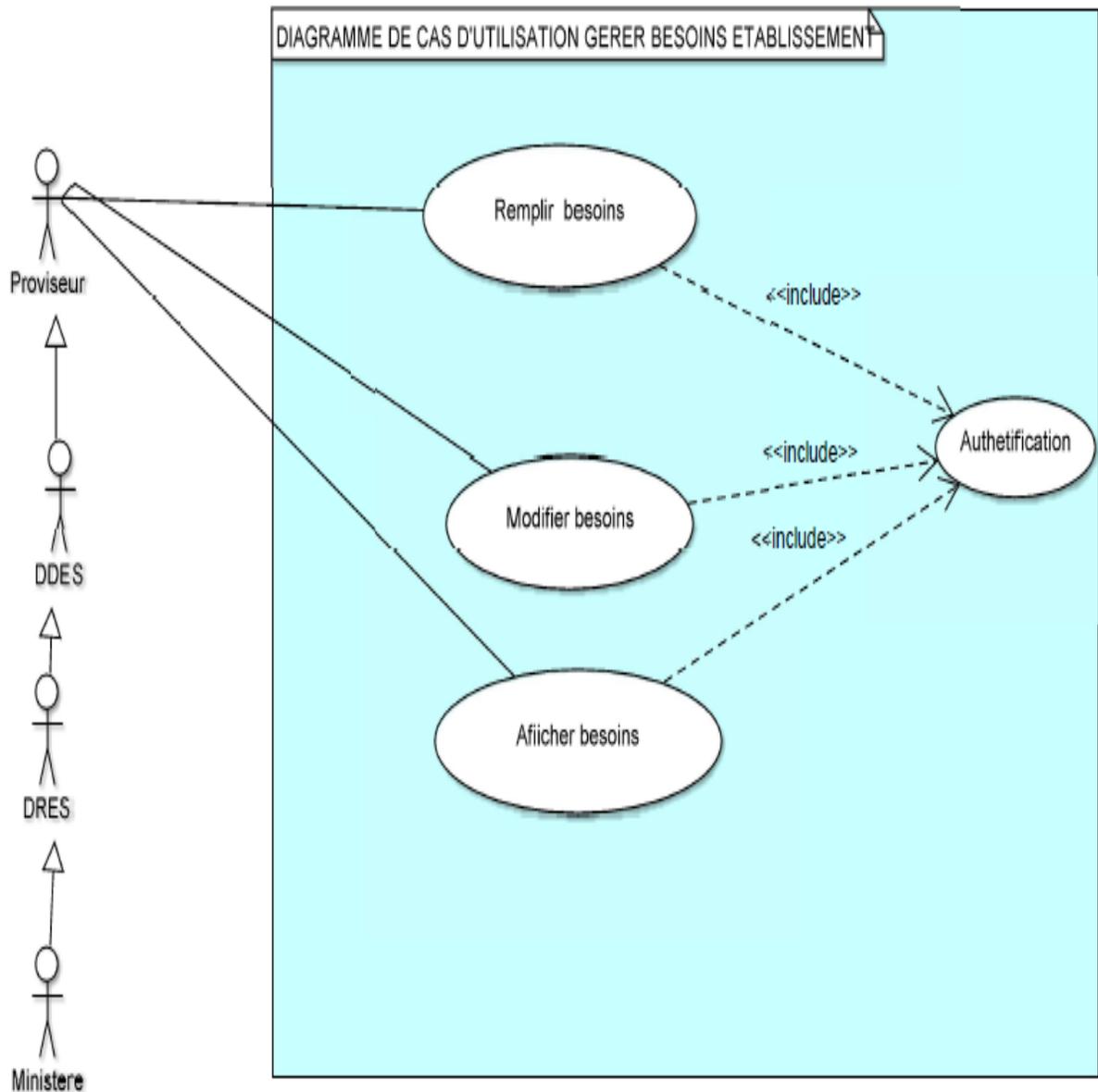
<u>Cas d'utilisation</u> : gestion du transfert des enseignants
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de consulter la liste des transferts en attente de validation
<u>Pré-condition</u> : l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à consulter la liste des transferts en attente de validation(1)

- Le système lui demande de choisir le niveau à consulter (établissement, département, région, pays) (2)
- L'utilisateur choisit le niveau en précisant son nom et valide (exemple : niveau départemental, département du moungo) (3)
- Le système lui présente la liste des transferts en attente de validation dans ce niveau (5)

- **Valider un transfert**

<u>Cas d'utilisation</u> : gestion du transfert des enseignants
<u>Acteurs</u> : Proviseur, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de valider la réception et l'intégration d'un enseignant dans un établissement suite à un envoi dans le processus de transfert.
<u>Pré-conditions</u> : <ul style="list-style-type: none"> ➤ l'utilisateur doit être un proviseur ou l'administrateur système qui est authentifié par un login et un mot de passe. ➤ L'enseignant doit être inscrit dans la liste d'attente des transferts de l'établissement d'accueil
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider le transfert d'un personnel (1) ➤ Le système lui présente la liste d'attente des transferts et lui demande de choisir l'enseignant à valider (2) ➤ L'utilisateur choisit l'enseignant et valide (3) ➤ Le système lui présente le formulaire de transfert (4) ➤ L'utilisateur remplit le formulaire et valide la réception du personnel (5)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'enseignant est supprimé de la liste d'attente des enseignants de cet établissement ➤ L'enseignant est désormais dans la liste du personnel de l'établissement d'accueil ➤ L'enseignant reçoit un message de notification pour la validation de son transfert

- Gestion besoins de l'établissement



Cas d'utilisation : gestion d'établissement

Acteurs principaux : proviseur, délégué départemental, Administrateur système

Description : le délégué départemental est responsable de la gestion d'établissement.

On peut remplir les besoins, modifier et afficher les besoins des établissements.

DESCRIPTION DES SCENARIOS

- **Remplir un besoin**

<u>Cas d'utilisation</u> : remplir un besoin
<u>Acteurs</u> : proviseur, délégué départemental, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet formuler les besoins d'un établissement
<u>Pré-condition</u> : l'utilisateur doit être un proviseur, délégué départemental ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à remplir les besoins d'un établissement (1)➤ Le système lui demande de fournir le code de l'établissement (2)➤ L'utilisateur fournit le code de l'établissement et passe à suivant (3)➤ Le système vérifie le code (4)➤ Le système présente le formulaire d'inscription des besoins à l'utilisateur (code, nom, date de création, catégorie, statut,...) (5)➤ L'utilisateur remplit le formulaire et valide (6)➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<u>Description alternative 1</u> : le code existe déjà <ul style="list-style-type: none">➤ (1), (2), (3)➤ Le système envoie un message d'alerte à l'utilisateur lui demandant d'introduire un autre code car celui-ci ne lui est pas attribué (4)➤ (5), (6), (7)
<u>Description alternative 2</u> : les champs obligatoires non remplis <ul style="list-style-type: none">➤ (1), (2), (3), (4), (5), (6)➤ Le système envoie un message d'alerte à l'utilisateur lui demandant de remplir tous les champs obligatoires (7')➤ (7)
<u>Post-conditions</u> : <ul style="list-style-type: none">➤ Le système met à jour la base de données➤ Le système met les besoins de l'établissement dans la liste d'attente pour validation par le délégué départemental.

- **Modifier les besoins d'un établissement**

<u>Cas d'utilisation</u> : Modifier les besoins d'un établissement
<u>Acteurs</u> : proviseur, délégué départemental, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier les besoins d'un établissement
<u>Pré-condition</u> : l'utilisateur doit être un proviseur, délégué départemental ou l'administrateur système qui est authentifié par un login et un mot de passe.
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à modifier les besoins d'un établissement (1) ➤ Le système lui demande de choisir l'établissement à modifier (2) ➤ L'utilisateur choisit l'établissement et valide (3) ➤ Le système lui présente le formulaire d'inscription des besoins de l'établissement avec les anciens paramètres (code, nom, date de création, adresse, catégorie, statut,...) (5) ➤ L'utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données. ➤ Le système met à jour les informations sur les besoins de l'établissement.

- **Afficher les besoins**

<u>Cas d'utilisation</u> : afficher la liste des besoins des établissements
<u>Acteurs</u> : Proviseur, délégué départemental, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'afficher la liste des besoins des établissements inscrits dans le système.
<u>Pré-condition</u> : l'utilisateur doit être un proviseur, délégué départemental ou l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à consulter la liste des besoins des établissements (1) ➤ Le système lui demande de choisir le niveau à consulter (établissement, département, région, pays) (2) ➤ L'utilisateur choisit le niveau en précisant son nom et valide (<u>exemple</u> : niveau

départemental, département du moungo) (3)

- Le système lui présente la liste des besoins des établissements inscrits dans ce niveau (5)

- Valider les besoins d'un établissement

Cas d'utilisation : valider les besoins d'un établissement

Acteurs principaux : proviseur, délégué département, Administrateur système

Description : le délégué départemental est responsable des établissements de son département. De ce fait il est habilité à valider leurs besoins. Il est possible d'afficher un des besoins, valider des besoins.

DESCRIPTION DES SCENARIOS

- **Afficher les besoins d'un établissement**

Cas d'utilisation : afficher la liste des besoins des établissements

Acteurs : Proviseur, délégué départemental, Administrateur Système

Description : ce cas d'utilisation permet d'afficher la liste des besoins des établissements inscrits dans le système.

Pré-condition : l'utilisateur doit être un proviseur, délégué départemental ou l'administrateur système qui est authentifié par un login et un mot de passe

Description nominale :

- L'utilisateur demande à consulter la liste des besoins des établissements (1)
- Le système lui demande de choisir le niveau à consulter (établissement, département, région, pays) (2)
- L'utilisateur choisit le niveau en précisant son nom et valide (exemple : niveau départemental, département du moungo) (3)
- Le système lui présente la liste des besoins des établissements inscrits dans ce niveau (5)

- **valider les besoins d'un établissement**

<u>Cas d'utilisation</u> : valider les besoins d'un établissement
<u>Acteurs</u> : délégué d'arrondissement, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet la validation des besoins d'un établissement
<u>Pré-conditions</u> : <ul style="list-style-type: none"> ➤ l'utilisateur doit être un délégué départemental ou l'administrateur système qui est authentifié par un login et un mot de passe ➤ les besoins doivent avoir été soumis.
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un besoin (1) ➤ Le système lui demande de choisir l'établissement (2) ➤ L'utilisateur choisit l'établissement et vérifie les besoins puis appuis sur suivant (3) ➤ Le système valide l'opération (4)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Les besoins de l'établissement sont ajoutés dans la liste des besoins des établissements de ce département

▪ Gestion du compte des administrateurs

<u>Cas d'utilisation</u> : gestion du compte de l'administrateur
<u>Acteurs principaux</u> : tous les administrateurs, Administrateur système
<u>Description</u> : l'administrateur système est responsable de la gestion des administrateurs. Il procède à la création des comptes des administrateurs, la modification des comptes et la suppression de ceux-ci. On peut afficher la liste des comptes inscrits dans une période.

DESCRIPTION DES SCENARIOS

- **Créer un compte administrateur**

<u>Cas d'utilisation</u> : créer un compte administrateur
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'ajouter un nouvel administrateur dans un système
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à créer un compte administrateur (1)➤ Le système lui demande de fournir le matricule de l'administrateur (2)➤ L'utilisateur fournit le matricule de l'administrateur et passe à suivant (3)➤ Le système vérifie le matricule (4)➤ valide l'opération (5)
<u>Description alternative 1</u> : le matricule existe déjà <ul style="list-style-type: none">➤ (1), (2), (3)➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre matricule car celui-ci existe déjà (4)➤ (5)
<u>Post-conditions</u> : <ul style="list-style-type: none">➤ Le système met à jour la base de données➤ Le système met le compte crée dans la liste des comptes administrateurs

- **Modifier un compte administrateur**

<u>Cas d'utilisation</u> : Modifier un compte
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier un compte dans le système
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à modifier un compte (1)

<ul style="list-style-type: none"> ➤ Le système lui demande de choisir le compte à modifier (2) ➤ L'utilisateur choisit le compte et valide (3) ➤ Le système lui présente le formulaire de création du compte avec les anciens paramètres (5) ➤ L'utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<p><u>Post-conditions :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met à jour les informations du compte.

- **Supprimer un compte administrateur**

<u>Cas d'utilisation :</u> Supprimer un compte
<u>Acteurs :</u> Administrateur Système
<u>Description :</u> ce cas d'utilisation permet de supprimer un compte dans le système
<u>Pré-condition :</u> l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<p><u>Description nominale :</u></p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à supprimer un compte (1) ➤ Le système lui demande de choisir le compte à supprimer (2) ➤ L'utilisateur choisit le compte et valide (3) ➤ Le système demande confirmation de l'opération de suppression de ce compte (4) ➤ L'utilisateur confirme la suppression (5) ➤ Le système valide l'opération (6)
<p><u>Post-condition :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le compte ne fait plus partie des comptes du système

- **Valider un compte administrateur**

<u>Cas d'utilisation :</u> valider un compte
<u>Acteurs :</u> Administrateur Système
<u>Description :</u> ce cas d'utilisation permet l'inscription effective d'un compte

<p><u>Pré-conditions :</u></p> <ul style="list-style-type: none"> ➤ l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe ➤ le compte doit être nouveau dans le système.
<p><u>Description nominale :</u></p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un compte (1) ➤ Le système lui demande de choisir le compte et vérifie ses informations (2) ➤ L'utilisateur choisit le compte puis appuis sur suivant (3) ➤ Le système valide l'opération (4)
<p><u>Post-conditions :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le compte est ajouté dans la liste des comptes du système ➤ Le compte est de ce fait retiré de la liste d'attente

Cas d'utilisation gestion département

<p><u>Cas d'utilisation :</u> gestion département</p>
<p><u>Acteurs principaux :</u> délégué régional, Administrateur système</p>
<p><u>Description :</u> le délégué régional est responsable de la gestion du département. Il procède à l'ajout des nouveaux départements, la modification de certaines informations et la suppression des départements. On peut afficher la liste des départements inscrits dans une période, voir la fiche d'un département, les statistiques des départements.</p>

DESCRIPTION DES SCENARIOS

- **Ajouter un département**

<p><u>Cas d'utilisation :</u> Ajouter un département</p>
<p><u>Acteurs :</u> délégué régional, Administrateur Système</p>
<p><u>Description :</u> ce cas d'utilisation permet d'ajouter un nouvel département dans une région.</p>
<p><u>Pré-condition :</u> l'utilisateur doit être un délégué régional ou l'administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale :</u></p>

<ul style="list-style-type: none"> ➤ L'utilisateur demande à ajouter un département (1) ➤ Le système lui demande de fournir le code de département (2) ➤ L'utilisateur fournit le code de département et passe à suivant (3) ➤ Le système vérifie le code (4) ➤ Le système présente le formulaire d'inscription à l'utilisateur (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5) ➤ L'utilisateur remplit le formulaire et valide (6) ➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<p><u>Description alternative 1</u> : le code existe déjà</p> <ul style="list-style-type: none"> ➤ (1), (2), (3) ➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre code car celui-ci existe déjà (4) ➤ (5), (6), (7)
<p><u>Description alternative 2</u> : les champs obligatoires non remplis</p> <ul style="list-style-type: none"> ➤ (1), (2), (3), (4), (5), (6) ➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant de remplir tous les champs obligatoires (7') ➤ (7)
<p><u>Post-conditions</u> :</p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met le département dans la liste d'attente pour validation par le délégué régional.

- **Modifier un département**

<u>Cas d'utilisation</u> : Modifier un département
<u>Acteurs</u> : délégué régional, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier les informations d'un département
<u>Pré-condition</u> : l'utilisateur doit être un délégué régional ou l'administrateur système qui est authentifié par un login et un mot de passe.
<u>Description nominale</u> :

<ul style="list-style-type: none"> ➤ L'utilisateur demande à modifier un département (1) ➤ Le système lui demande de choisir le département à modifier (2) ➤ L'utilisateur choisit le département et valide (3) ➤ Le système lui présente le formulaire d'inscription de le département avec les anciens paramètres (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5) ➤ L'utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<p><u>Post-conditions :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données. ➤ Le système met à jour les informations du département.

- **Supprimer un département**

<p><u>Cas d'utilisation :</u> Supprimer un département</p>
<p><u>Acteurs :</u> délégué département, Administrateur Système</p>
<p><u>Description :</u> ce cas d'utilisation permet de supprimer un département dans la région</p>
<p><u>Pré-condition :</u> l'utilisateur doit être un délégué département ou l'administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale :</u></p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à supprimer un département (1) ➤ Le système lui demande de choisir le département à supprimer (2) ➤ L'utilisateur choisit le département et valide (3) ➤ Le système demande confirmation de l'opération de suppression de ce département (4) ➤ L'utilisateur confirme la suppression (5) ➤ Le système valide l'opération (6)
<p><u>Post-condition :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le département ne fait plus partie du département

- **valider un département**

<u>Cas d'utilisation</u> : valider un département
<u>Acteurs</u> : délégué régional, Administrateur Système
<u>Description</u> : ce cas d'utilisation permet l'inscription effective d'un département en lui associant une région.
<u>Pré-conditions</u> : <ul style="list-style-type: none"> ➤ l'utilisateur doit être un délégué régional ou l'administrateur système qui est authentifié par un login et un mot de passe ➤ Le département doit être nouveau dans la région.
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider un département (1) ➤ Le système lui demande de choisir le département (2) ➤ L'utilisateur choisit le département puis appuis sur suivant (3) ➤ Le système valide l'opération (4)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le département est ajouté dans la liste des départements de cette région ➤ Le département est de ce fait retiré de la liste d'attente

V. **CAS D'UTILISATION GESTION DE LA REGION**

<u>Cas d'utilisation</u> : gestion région
<u>Acteurs principaux</u> : Administrateur système
<u>Description</u> : l'administrateur système est responsable de la gestion de la région. Il procède à l'ajout de nouvelles régions, la modification de certaines informations et la suppression des régions. On peut afficher la liste des régions inscrits dans une période, voir la fiche d'une région, les statistiques des régions.

DESCRIPTION DES SCENARIOS

- **Ajouter une région**

<u>Cas d'utilisation</u> : Ajouter une région
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'ajouter une nouvelle région.
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'utilisateur demande à ajouter une région (1)➤ Le système lui demande de fournir le code d'une région (2)➤ L'utilisateur fournit le code de département et passe à suivant (3)➤ Le système vérifie le code (4)➤ Le système présente le formulaire d'inscription à l'utilisateur (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5)➤ L'utilisateur remplit le formulaire et valide (6)➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<u>Description alternative 1</u> : le code existe déjà <ul style="list-style-type: none">➤ (1), (2), (3)➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant d'introduire un autre code car celui-ci existe déjà (4)➤ (5), (6), (7)
<u>Description alternative 2</u> : les champs obligatoires non remplis <ul style="list-style-type: none">➤ (1), (2), (3), (4), (5), (6)➤ Le système renvoi un message d'alerte à l'utilisateur lui demandant de remplir tous les champs obligatoires (7')➤ (7)
<u>Post-conditions</u> : <ul style="list-style-type: none">➤ Le système met à jour la base de données➤ Le système met la région dans la liste d'attente pour validation par l'administrateur système.

- **Modifier une région**

<u>Cas d'utilisation</u> : Modifier une région
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier les informations d'une région
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe.
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à modifier une région (1) ➤ Le système lui demande de choisir la région à modifier (2) ➤ L'utilisateur choisit la région et valide (3) ➤ Le système lui présente le formulaire d'inscription de la région avec les anciens paramètres (code, nom, date de création, adresse, téléphone, email, logo, catégorie, statut,...) (5) ➤ L'utilisateur modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données. ➤ Le système met à jour les informations de la région.

- **Supprimer une région**

<u>Cas d'utilisation</u> : Supprimer une région
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de supprimer une région du système
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'utilisateur demande à supprimer une région (1) ➤ Le système lui demande de choisir la région à supprimer (2) ➤ L'utilisateur choisit la région et valide (3) ➤ Le système demande confirmation de l'opération de suppression de cette région (4)

<ul style="list-style-type: none"> ➤ L'utilisateur confirme la suppression (5) ➤ Le système valide l'opération (6)
<p><u>Post-condition :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le département ne fait plus partie de la région

- **valider une région**

<p><u>Cas d'utilisation :</u> valider une région</p>
<p><u>Acteurs :</u> Administrateur Système</p>
<p><u>Description :</u> ce cas d'utilisation permet l'inscription effective d'une région</p>
<p><u>Pré-conditions :</u></p> <ul style="list-style-type: none"> ➤ l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe ➤ La région doit être nouvelle dans le système.
<p><u>Description nominale :</u></p> <ul style="list-style-type: none"> ➤ L'utilisateur demande à valider une région (1) ➤ Le système lui demande de choisir la région (2) ➤ L'utilisateur choisit la région puis appuis sur suivant (3) ➤ Le système valide l'opération (4)
<p><u>Post-conditions :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ La région est ajoutée dans la liste des régions ➤ La région est de ce fait retiré de la liste d'attente

VI. CAS D'UTILISATION GESTION DES UTILISATEURS

<p><u>Cas d'utilisation :</u> gestion des utilisateurs</p>
<p><u>Acteurs principaux :</u> Administrateur système</p>
<p><u>Description :</u> l'administrateur système est responsable de la gestion des utilisateurs. Il procède à l'ajout d'un nouvel utilisateur en lui affectant des droits d'accès, la modification de certaines informations et la suppression des utilisateurs. On peut afficher la liste des</p>

utilisateurs.

DESCRIPTION DES SCENARIOS

- **Ajouter un utilisateur**

<u>Cas d'utilisation</u> : Ajouter un utilisateur
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet d'ajouter un nouvel utilisateur dans la gestion du système.
<u>Pré-condition</u> : l'utilisateur doit l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none">➤ L'administrateur système demande à ajouter un utilisateur (1)➤ Le système lui demande de fournir le login de l'utilisateur (2)➤ L'administrateur système fournit le login de l'utilisateur et passe à suivant (3)➤ Le système vérifie le login (4)➤ Le système présente le formulaire d'ajout à l'administrateur système (mot de passe, statut ou grade) (5)➤ L'administrateur système remplit le formulaire et valide (6)➤ Le système vérifie les champs obligatoires et il valide l'opération (7)
<u>Description alternative 1</u> : le login existe déjà <ul style="list-style-type: none">➤ (1), (2), (3)➤ Le système envoie un message d'alerte à l'administrateur système lui demandant d'introduire un autre login car celui-ci existe déjà (4)➤ (5), (6), (7)
<u>Description alternative 2</u> : les champs obligatoires non remplis <ul style="list-style-type: none">➤ (1), (2), (3), (4), (5), (6)➤ Le système envoie un message d'alerte à l'administrateur système lui demandant de remplir tous les champs obligatoires (7')➤ (7)
<u>Post-conditions</u> :

- Le système met à jour la base de données
- Le système ajoute l'utilisateur dans la liste des utilisateurs

- **Modifier un utilisateur**

<u>Cas d'utilisation</u> : Modifier un utilisateur
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de modifier les coordonnées d'un utilisateur
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'administrateur système demande à modifier un utilisateur (1) ➤ Le système lui demande de choisir l'utilisateur à modifier (2) ➤ L'administrateur système choisit l'utilisateur et valide (3) ➤ Le système lui présente le formulaire d'ajout de l'utilisateur avec les anciens paramètres (login, mot de passe, statut ou grade) (5) ➤ L'administrateur système modifie les paramètres souhaités et valide (6) ➤ Le système valide l'opération (7)
<u>Post-conditions</u> : <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Le système met à jour les informations de l'utilisateur

- **Supprimer un utilisateur**

<u>Cas d'utilisation</u> : Supprimer un utilisateur
<u>Acteurs</u> : Administrateur Système
<u>Description</u> : ce cas d'utilisation permet de Supprimer un utilisateur de la gestion de l'établissement scolaire
<u>Pré-condition</u> : l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe
<u>Description nominale</u> : <ul style="list-style-type: none"> ➤ L'administrateur système demande à supprimer un utilisateur (1)

<ul style="list-style-type: none"> ➤ Le système lui présente la liste des utilisateurs à supprimer (2) ➤ L'administrateur système sélectionne l'utilisateur et appuis sur supprimer (3) ➤ Le système demande confirmation de l'opération de suppression de cet utilisateur (4) ➤ L'administrateur système confirme la suppression (5) ➤ Le système valide l'opération (6)
<p><u>Post-conditions :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ L'utilisateur ne fait plus partie de la base de données

- **Attribuer des droits à l'utilisateur**

<p><u>Cas d'utilisation :</u> Attribuer des taches à une catégorie d'utilisateur</p>
<p><u>Acteurs :</u> Administrateur Système</p>
<p><u>Description :</u> ce cas d'utilisation permet d'attribuer des taches à une catégorie d'utilisateur pour la gestion de l'établissement scolaire</p>
<p><u>Pré-condition :</u> l'utilisateur doit être l'administrateur système qui est authentifié par un login et un mot de passe</p>
<p><u>Description nominale :</u></p> <ul style="list-style-type: none"> ➤ L'administrateur système demande d'attribuer les droits à un utilisateur (1) ➤ Le système présente la liste des catégories d'utilisateur (2) ➤ L'administrateur système sélectionne la catégorie et appuis sur éditer le menu (3) ➤ Le système présente la liste des taches (4) ➤ L'administrateur système choisit les taches pouvant être exécuté par cette catégorie d'utilisateur et appuis sur valider (5) ➤ Le système demande à l'administrateur système s'il veut vraiment procéder à la modification des droits de cette catégorie d'utilisateur (6) ➤ L'administrateur système confirme la modification des droits (7) ➤ Le système valide l'opération (8)
<p><u>Post-conditions :</u></p> <ul style="list-style-type: none"> ➤ Le système met à jour la base de données ➤ Les droits sont octroyés à l'ensemble des utilisateurs ayant ce profil

CHAPITRE 4 : CONCEPTION

Nous aborderons dans ce chapitre la deuxième phase (Elaboration) qui se concentre essentiellement sur la définition de l'architecture du système, l'analyse et la conception des besoins et des exigences des utilisateurs. L'activité d'analyse et de conception permet de traduire les besoins fonctionnels et les contraintes issues du cahier des charges et de la spécification des exigences dans un langage plus professionnel et compréhensible par tous les individus intervenants dans la réalisation et l'utilisation de l'application.

I. ARCHITECTURE DU SYSTEME

Dans les phases préliminaires du développement d'une application ou de la refonte d'un système d'information, la définition de l'architecture technique consiste à faire les choix de technologies et d'organisation de composants logiciels les plus adaptés aux besoins et aux contraintes de l'organisation d'accueil. Ces choix sont ensuite relayés au sein de notre projet, guidant la conception et permettant la transformation d'un modèle fonctionnel en application performante et robuste.

➤ Présentation de l'architecture à 2 niveaux

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

➤ Présentation de l'architecture à 3 niveaux

Dans l'architecture à 3 niveaux (appelées architecture 3-tiers), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre:

- Le client: le demandeur de ressources
- Le serveur d'application (appelé aussi **middleware**): le serveur chargé de fournir la ressource mais faisant appel à un autre serveur
- Le serveur secondaire (généralement un serveur de base de données), fournissant un service au premier serveur.

➤ **Architecture adoptée**

Nous avons choisi pour notre système l'architecture à 2 niveaux. Car son fonctionnement est celui qui se rapproche le plus du fonctionnement que nous avons pensé pour notre système.

II. CONCEPTION DETAILLEE

UML est une approche orientée objet de modélisation qui permet de modéliser un problème d'une manière standard. UML évite de se définir comme une méthodologie, comme son nom l'indique, c'est un langage « visuel » qui permet d'exprimer la compréhension d'un système : il comporte 9 principaux diagrammes regroupés dans deux vues différentes:

Vue Statique (cinq diagrammes structurels)

- Diagramme de Cas d'utilisation.
- Diagramme de Classes.
- Diagramme d'Objets.
- Diagramme de Composants.
- Diagramme de Déploiement.

Vue Dynamique (quatre diagrammes comportementaux)

- Diagramme de Séquence.
- Diagramme d'activités.
- Diagramme d'états transitions.
- Diagramme de Collaboration.

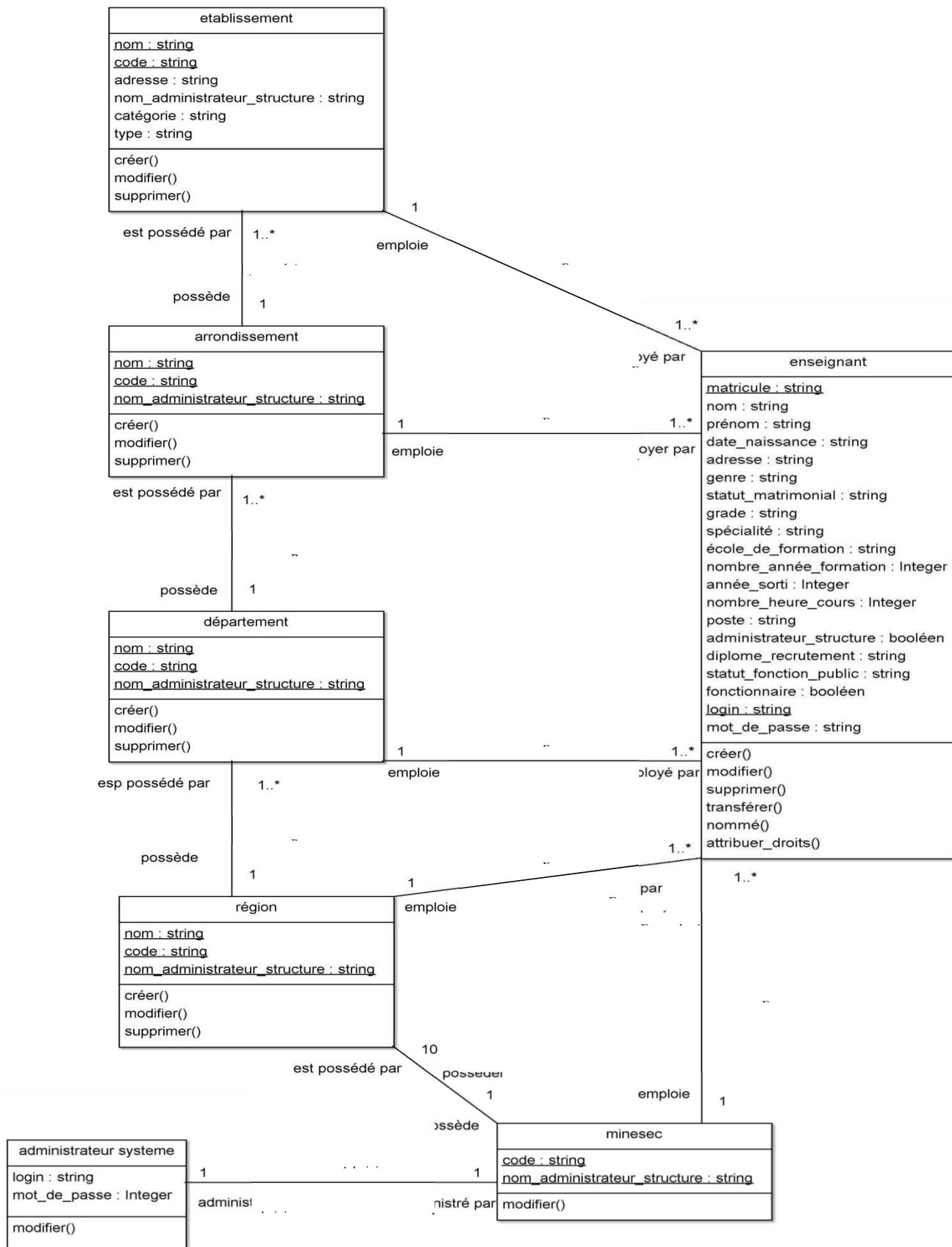
A cet effet on présente quelques diagrammes de modélisation, qu'on a jugé les plus importants pour la compréhension du fonctionnement du système.

1. DIAGRAMME DES CAS D'UTILISATION

Les diagrammes de cas d'utilisation représentent toutes les interactions des utilisateurs avec le système, comme décrit dans le chapitre précédent « *Analyse des besoins et spécification* »

2. DIAGRAMME DE CLASSES

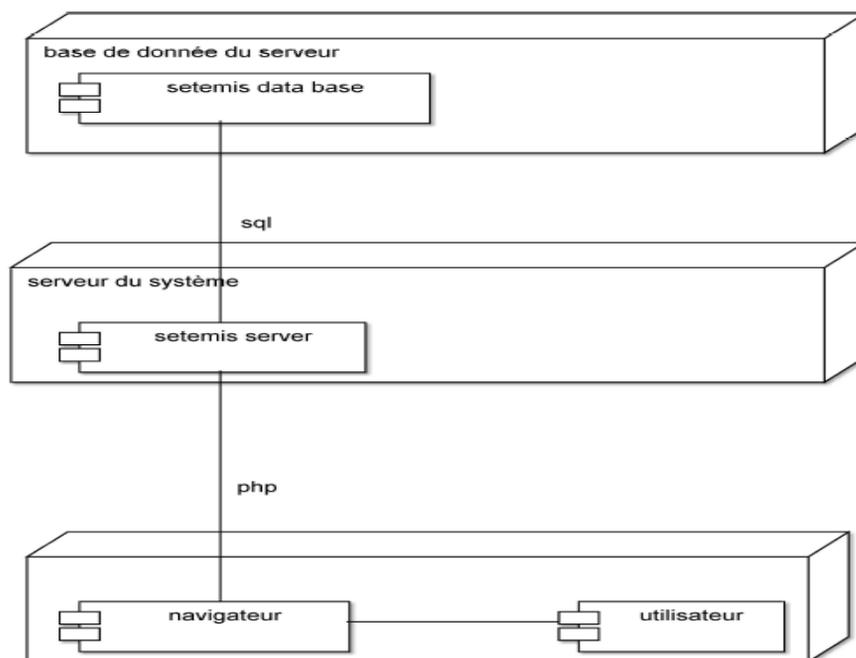
Le diagramme de classes représente les classes constituant le système et les associations entre elles. Les diagrammes de classes expriment de manière générale la structure statique d'un système, en termes de classe et de relations entre ces classes. De même qu'une classe décrit un ensemble d'objets, une association décrit un ensemble de liens ; les objets sont des instances de classes et les liens sont des instances de relations. Le diagramme de classe général du système se présente ainsi :



D'après l'étude du système existant et des différents diagrammes de cas d'utilisation, nous avons pu dégager les principales classes illustrées dans la figure ci-dessus pour avoir une vue plus claire du système étudié. A partir de ce diagramme, on dégage les entités de la base de données de l'application à développer.

3. DIAGRAMME DE DEPLOIEMENT

Le diagramme de déploiement est une vue statique qui sert à représenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que les relations entre eux.



4. DIAGRAMME DE SEQUENCE

Il montre la séquence verticale des messages passés entre objets au sein d'une interaction. La ligne de vie est la représentation de l'existence d'un élément participant dans un diagramme de séquence. Cela peut être un acteur ou le système en modélisation d'exigences, des objets logiciels en conception préliminaire ou conception détaillée. Après l'étude des cas d'utilisation, nous avons pu dégager les diagrammes de séquences correspondants dont voici les plus importants :

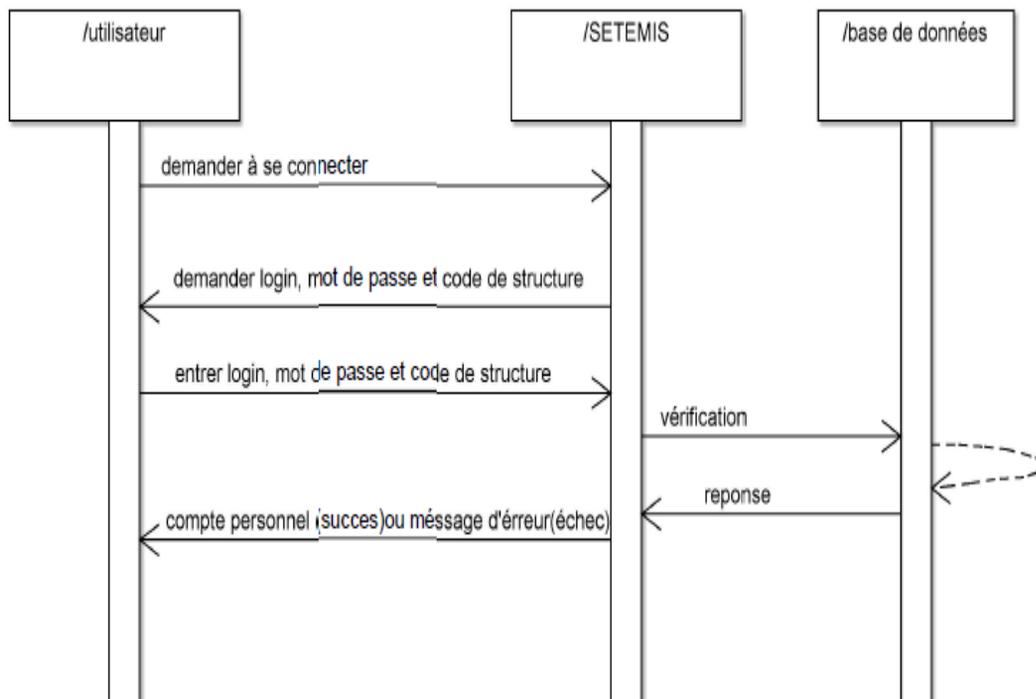
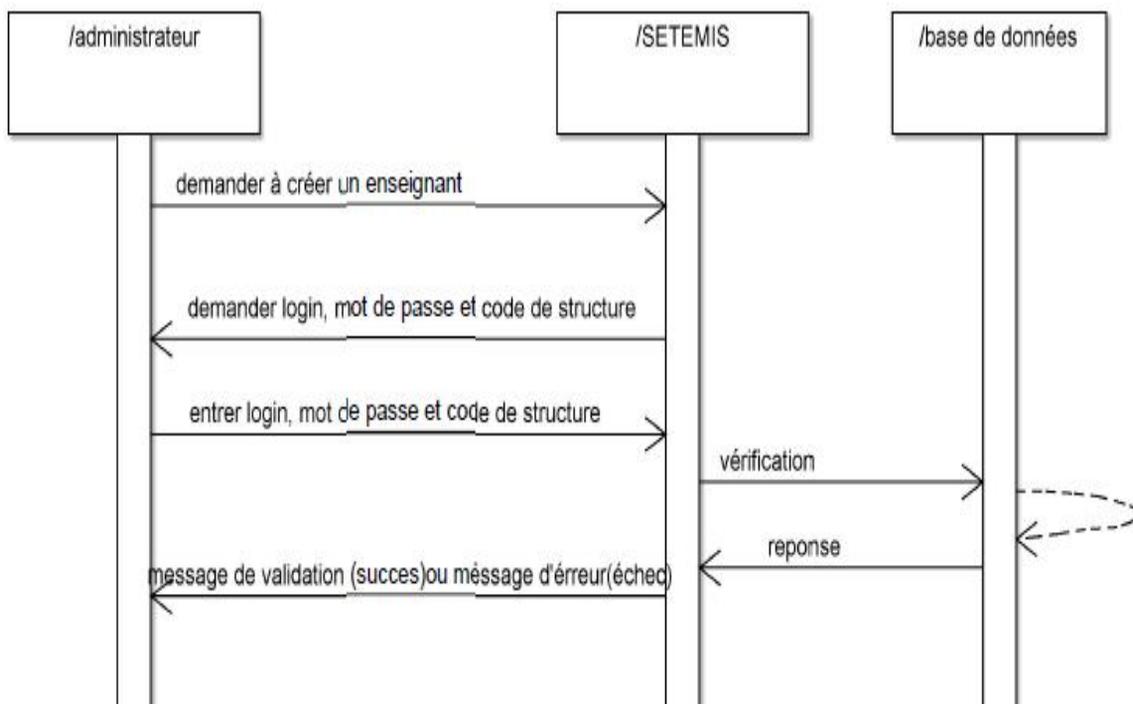


Diagramme de séquence de connexion

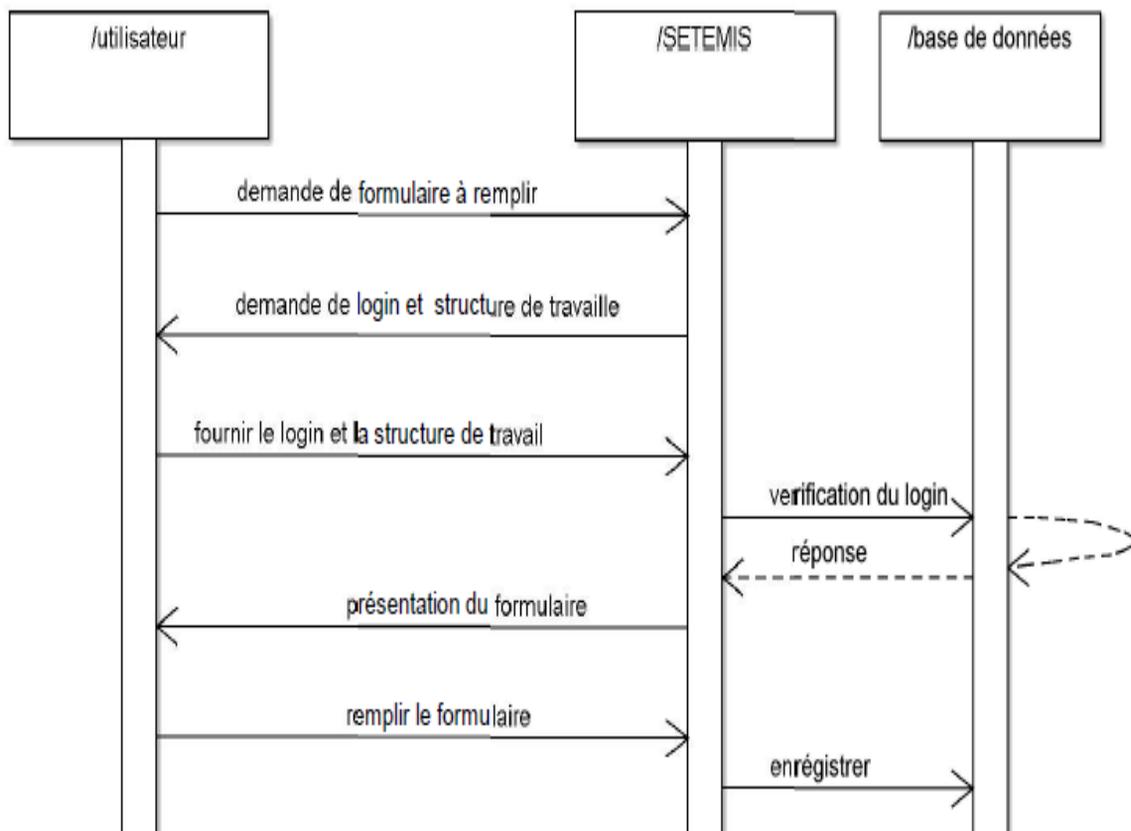
L'utilisateur demande à se connecter, le système lui demande d'entrer ses login, mot de passe et structure de travaille, le système vérifie les informations et si elles sont bonne, l'utilisateur est connecté, sinon il reçoit un message d'erreur.

a. Créer des comptes enseignants



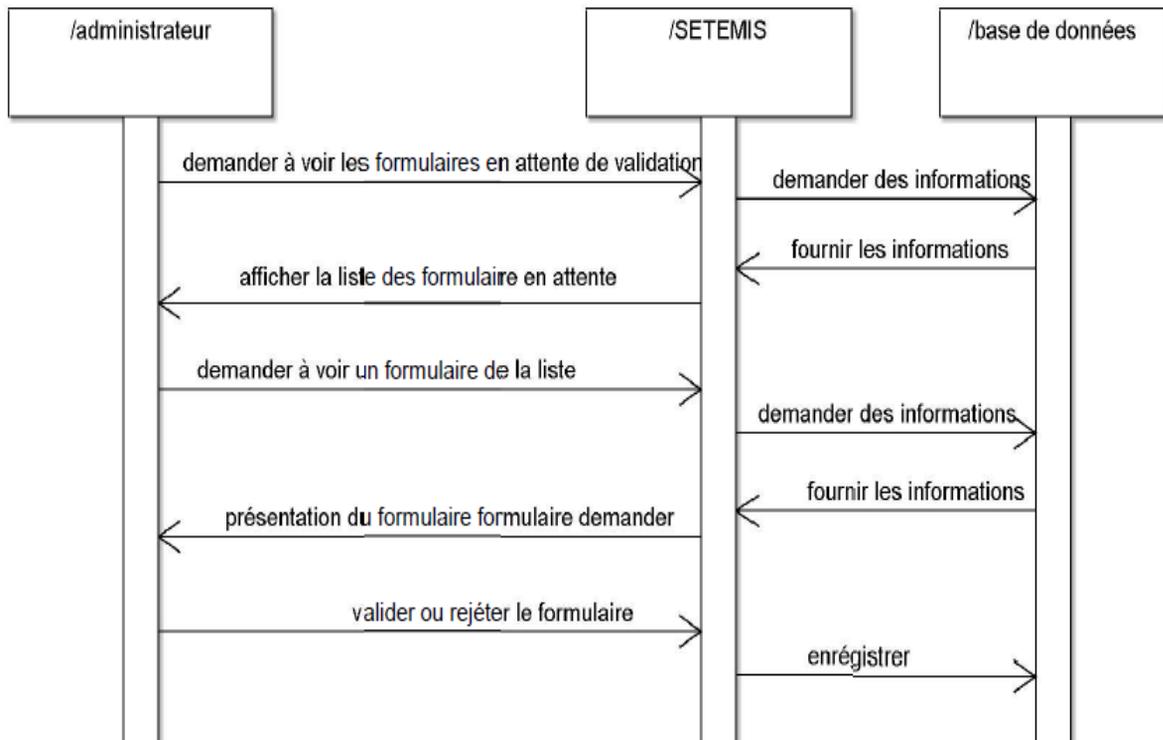
L'administrateur demande à créer un enseignant, le système lui demande d'entrer un login, un mot de passe et le code de la structure dans laquelle il va être enregistré. Après avoir fourni toutes ces informations le système les vérifie et valide l'opération si le login n'existe pas déjà dans la base de données.

b. Remplir formulaire



L'utilisateur demande à remplir son formulaire, le système lui demande d'entrer son login et le type de structure dans laquelle il se trouve. Après avoir fourni toutes ces informations le système les vérifie. Si le login n'est pas correct le système renvoie un message d'erreur. Sinon il fournit un formulaire à remplir correspondant au type de structure choisie. L'utilisateur remplit le formulaire et clique sur valider Si tous les champs obligatoire son remplie le formulaire est enregistré sinon on reçoit un message d'erreur.

c. Valider un formulaire



L'administrateur demande à voir la listes des formulaires en attente de validation, le système la lui donne, l'administrateur choisit un formulaire, le système lui présente ce formulaire avec les informations correspondante. L'administrateur peut donc valider ou rejeter le dit formulaire.

CHAPITRE 5 : REALISATION

Après la conception, il est question de la réalisation et l'implémentation. Dans ce chapitre nous allons nous assurer que le système est prêt pour à être exploité par les utilisateurs finaux.

I. ENVIRONNEMENT DE DEVELOPPEMENT

Pour la réalisation de ce travail, nous avons eu recours aux environnements suivants:

➤ Environnement Matériel

Pour développer l'application, nous avons utilisé comme environnement matériel trois ordinateurs HP compact qui possèdent comme caractéristiques :

Machine 1 :

Un processeur Intel Pentium® Core2 Duo, 2.49 GHz.

- Une mémoire vive de 4Go.
- Un disque dur 250 Go.
- Un écran 17 pouces.

Machine 2 :

Un processeur Intel Pentium® Core2 Duo, 2.49 GHz.

- Une mémoire vive de 4Go.
- Un disque dur 500 Go.
- Un écran 17 pouces.

Machine 3 :

Un processeur Intel Pentium® Core2 Duo, 2.49 GHz.

- Une mémoire vive de 4Go.
- Un disque dur 300 Go.
- Un écran 17 pouces.

➤ Environnement Logiciel :

- Windows 10
- ArgoUML pour la modélisation des diagrammes
- NotePad++ comme éditeur de texte
- WampServer (qui contient le serveur Apache et MySQL pour la gestion des bases de données)
- PHP, HTML et JavaScript comme langage de programmation
- Fpdf pour imprimer en pdf
- JpGraph pour dessiner les graphiques

II. PRINCIPALES INTERFACES GRAPHIQUES

La conception des interfaces de l'application est une étape très importante puisque toutes les interactions avec le cœur de l'application passent à travers ces interfaces, on doit alors guider l'utilisateur avec les messages d'erreurs et de notification si besoin. Dans cette partie, nous allons présenter quelques interfaces de l'application, répondant aux recommandations ergonomiques de compatibilité, de guidage, de clarté, d'homogénéité et de souplesse. Nous avons choisi l'administrateur principal comme utilisateur vu qu'il présente à travers ces interactions la majeure partie des principales fonctionnalités de l'application.

1. LA PAGE D'ACCUEIL

C'est la première interface du système, elle nous permet d'avoir accès à plusieurs tableaux et graphiques de statistiques sur les enseignants. C'est l'interface qui est accessible à tous.

Pour consulter les statistiques, on choisit sur les propositions à gauche les statistiques que l'on veut et la forme sur laquelle on les veut.

Exemple : consultons le tableau des statistiques à l'échelle national.

1. En survolant « national » à gauche de l'écran on a :



2. on choisit « Statistiques »:



3. en cliquant on obtient le tableau suivant :

Connexion :

Cette interface permet à l'utilisateur de s'authentifier et de se connecter au système. L'utilisateur doit entrer son login, son mot de passe et le code de la structure dans laquelle il veut se connecter pour accéder au système. En cas d'erreur un message d'alerte s'affiche :

The image shows the login interface for the SETEMIS system. It features a central form titled "ADMISTRATION SETEMIS" (note the typo). The form contains three input fields: "NOM UTILISATEUR:", "MOT DE PASSE:", and "CODE ADMINISTRATION:". The "CODE ADMINISTRATION:" field is a dropdown menu with the text "-Choisir un code administration-". Below the form is a "Connexion" button. To the right of the form is a "Nouvelle capture" button. The background is light blue with a header bar that says "Connexion au système SETEMIS." and a footer with copyright information: "©Copyright Administration SETEMIS 2016" and "Système Informatique de Gestion des Enseignants du Secondaire SETEMIS".

1. connectons-nous en tant que administrateur.

Nom utilisateur : admin

Mot de passe : admin

Code administration : MINESEC

Connexion au système SETEMIS.

ADMISTRATION SETEMIS

NOM UTILISATEUR : admin

MOT DE PASSE : ●●●●●

CODE ADMINISTRATION : MINESEC

Connexion

©Copyright Administration SETEMIS 2016

Système Informatique de Gestion des Enseignants du Secondaire SETEMIS

2. après connexion on à la page suivante :

Gestion Formulaire

-  Remplir Formulaire
-  Formulaires en attente
-  Valider | Rejeter
-  Modifier | Supprimer

Affectation/Nomination

-  Effectuer Nomination
-  Effectuer Affectation
-  Valider Affectation

Gestion Besoins

-  Ajouter Besoins
-  Liste Besoins
-  Modifier | Supprimer

Gestion Utilisateurs

-  Imprimer la liste
-  Liste Utilisateurs
-  Ajouter Utilisateur
-  Modifier | Supprimer
-  Editer les droits

Gestion Etablissement

-  Liste Etablissement
-  Ajouter Etablissement
-  Modifier | Supprimer

Gestion Departement

-  Liste Departement
-  Ajouter departement
-  Modifier | Supprimer

Gestion Administrateurs

-  Liste Admin
-  Ajouter Admin
-  Modifier | Supprimer
-  Editer droits

C'EST ICI LA PAGE D'ACCUEIL.

CONCLUSION GENERALE

L'objectif de notre projet de fin d'étude était de concevoir et implémenter un système de gestion et d'agrégation en ligne des statistiques des enseignants du secondaire au Cameroun. Le point de départ de la réalisation de ce projet était la récolte des informations nécessaires pour dresser un état de l'existant, présenter un aperçu sur la problématique ainsi que les avantages que pourrai avoir un tel système s'il est déployé. Par la suite, nous nous sommes intéressés à l'analyse et la spécification des besoins qui nous a permis de distinguer les différents acteurs interagissant avec le système. L'objectif de la partie suivante était la conception détaillée, dans laquelle nous avons fixé la structure globale de l'application. Le dernier volet de notre projet était la partie réalisation qui a été consacrée à la présentation des outils du travail et les interfaces les plus significatives de notre application. L'apport de ce travail a été d'une importance très considérable, en effet, il nous a permis : de suivre une méthodologie de travail bien étudié et d'approfondir nos connaissances dans le monde du développement des applications. La réalisation de ce projet, nous a permis d'apprendre et de toucher du doigt une partie des divers aspects du métier de développeur et de celui du concepteur. Et surtout ce projet nous a permis d'expérimenter les difficultés et les nombreux avantages d'un travail de groupe.

ANNEXE A : UML

DIAGRAMME DE CAS D'UTILISATION

Montre les interactions fonctionnelles entre les acteurs et le système.

Acteur : rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système étudié. Un acteur participe à au moins un cas d'utilisation.

Cas d'utilisation (use case) : ensemble de séquences d'actions réalisées par le système produisant un résultat observable intéressant pour un acteur particulier. Collection de scénarios reliés par un objectif utilisateur commun.

Association : utilisée dans ce type de diagramme pour relier les acteurs et les cas d'utilisation par une relation qui signifie simplement « participe à ».

Inclusion : le cas d'utilisation de base en incorpore explicitement un autre, de façon obligatoire, à un endroit spécifié dans ses enchaînements.

Extension : le cas d'utilisation de base en incorpore implicitement un autre, de façon optionnelle, à un endroit spécifié indirectement dans celui qui procède à l'extension

Généralisation : les cas d'utilisation descendants héritent de la description de leur parent commun. Chacun d'entre eux peut néanmoins comprendre des relations spécifiques supplémentaires avec d'autres acteurs ou cas d'utilisation.

DIAGRAMME DE SEQUENCE

Montre la séquence verticale des messages passés entre objets au sein d'une interaction.

Ligne de vie : représentation de l'existence d'un élément participant dans un diagramme de séquence. Cela peut être un acteur ou le système en modélisation d'exigences, des objets logiciels en conception préliminaire ou conception détaillée.

Message : élément de communication unidirectionnel entre objets qui déclenche une activité dans l'objet destinataire. La réception d'un message provoque un événement dans l'objet récepteur.

La flèche pointillée représente un retour au sens UML. Cela signifie que le message en question est le résultat direct du message précédent.

Spécification d'activation : bande blanche qui représente une période d'activité sur une ligne de vie.

Message synchrone : envoi de message pour lequel l'émetteur se bloque en attente du retour qui est représenté par une flèche pleine.

Un message asynchrone, au contraire, est représenté par une flèche ouverte.

Occurrence d'interaction : une interaction peut faire référence explicitement à une autre interaction grâce à un cadre avec le mot-clé « ref » et indiquant le nom de l'autre interaction.

UML 2 a ajouté une nouvelle notation très utile : les cadres d'interaction. Chaque cadre possède un opérateur et peut être divisé en fragments. Les principaux opérateurs sont :

loop: boucle. Le fragment peut s'exécuter plusieurs fois, et la condition de garde explicite l'itération.

opt: optionnel. Le fragment ne s'exécute que si la condition fournie est vraie.

alt: fragments alternatifs. Seul le fragment possédant la condition vraie s'exécutera.

DIAGRAMME DE CLASSES

Classe: description abstraite d'un ensemble d'objets qui partagent les mêmes propriétés (attributs et associations) et comportements (opérations et états).

Attribut : donnée déclarée au niveau d'une classe, éventuellement typée, à laquelle chacun des objets de cette classe donne une valeur. Un attribut peut posséder une multiplicité et une valeur initiale. Un attribut dérivé (« / ») est un attribut dont la valeur peut être déduite d'autres informations disponibles dans le modèle.

Opération : élément de comportement des objets, défini de manière globale dans leur classe. Une opération peut déclarer des paramètres (eux-mêmes typés) ainsi qu'un type de retour.

Association : relation sémantique durable entre deux classes, qui décrit un ensemble de liens entre instances. Une association est bidirectionnelle par défaut, sauf si l'on restreint sa navigabilité en ajoutant une flèche.

Rôle : nom donné à une extrémité d'une association ; par extension, manière dont les instances d'une classe voient les instances d'une autre classe au travers d'une association.

Multiplicité : le nombre d'objets (min...max) qui peuvent participer à une relation avec un autre objet dans le cadre d'une association. Multiplicités fréquentes :

0..1 = optionnel (mais pas multiple)

1 = exactement 1

0..* = * = quelconque

1..* = au moins 1

ANNEXE C
LE GUIDE D'UTILISATION

GUIDE D'UTILISATION SETEMIS

RESUME

SETEMIS est une application conçue pour fournir des statistiques sur les enseignants du secondaire au Cameroun. Elle permet de collecter en ligne des informations sur les enseignants, les stocker et de fournir sous forme de tableaux, graphiques et liste, la répartition en temps réel de ces enseignants sur l'étendue du territoire national en fonction de plusieurs paramètres (sexe, grade, situation, âge etc.).

1. PAGE D'ACCUEIL

SETEMIS (SEcondary TEaching Management Information System)

Accueil **5** Connexion

statistiques du personnel enseignant

NATIONAL **1**

REGION **2**

DEPARTEMENT

ETABLISSEMENT

Informations sur le formulaire

FORMULAIRE **3**

Informations générales

tous les professeurs sortent de l'ENS de yaounde sont informés que les revele de note ... Lire la suite **4**

tous les professeurs sortent de l'ENS de yaounde sont informés que les revele de note ... Lire la suite

Aujourd'hui, nous sommes le 02-06-2016

SETEMIS (SEcondary TEaching Management Information System) **1**

SETEMIS compte actuellement 21 personnel enseignant dont 8 Femme(s) et 13 Homme(s).

Avec ce systeme d'agregation statistique du personnel enseignant (**Camerounais**), vous avez la possibilite de voir a temps reel, les statistiques d'ordre national, regional, departemental et meme pour un etablissement sous forme de chiffre (tableau), sous forme graphique (pourcentage) et sous forme de liste pour savoir quel enseignant est dans tel ou tel region, departement ou etablissement. ces statistiques sont par Genre, Grade, Corps de metier, ecole de formation, et toutes les combinaisons possible par exemple Grade/Genre, Statut/Genre, Statut/ecole de formation etc.

©Copyright Administration SETEMIS 2016 SETEMIS (SEcondary TEaching Management Information System)

C'est la première page du site. Elle est répartie en 5 zones.

- **La zone 1 :**

SETEMIS (SEcondary TEaching Management Information System)

SETEMIS compte actuellement 21 personnel enseignant dont 8 Femme(s) et 13 Homme(s).

Avec ce systeme d'agregation statistique du personnel enseignant (**Camerounais**), vous avez la possibilite de voir a temps reel, les statistiques d'ordre national, regional, departemental et meme pour un etablissement sous forme de chiffre (tableau), sous forme graphique (pourcentage) et sous forme de liste pour savoir quel enseignant est dans tel ou tel region, departement ou etablissement. ces statistiques sont par Genre, Grade, Corps de metier, ecole de formation, et toutes les combinaisons possible par exemple Grade/Genre, Statut/Genre, Statut/ecole de formation etc.

C'est celle où vont s'afficher les résultats des requêtes émises par les visiteurs.

- **La zone 2.**

C'est celle où les visiteurs choisissent les statistiques et la forme sous laquelle ils

statistiques du personnel enseignant

- NATIONAL
- REGION
- DEPARTEMENT
- ETABLISSEMENT

veulent les consulter.

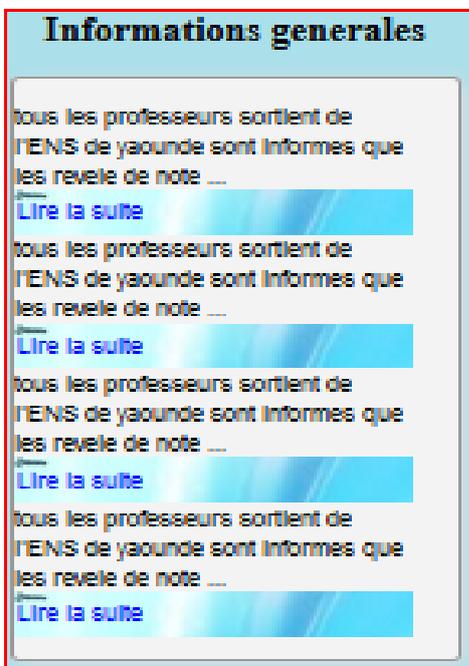
- « NATIONAL » : nous donne les statistiques à l'échelle nationale
- « REGION » : nous donne les statistiques à l'échelle d'une région
- « DEPARTEMENT » : nous donne les statistiques à l'échelle d'un département
- « ETABLISSEMENT » : nous donne les statistiques à l'échelle d'un établissement

- La zone 3



C'est celle où l'on peut consulter la liste des formulaires en attente et imprimé en exemple de formulaire.

- La zone 4



C'est celle où s'affiche les informations mise à la disposition de tous par le système.

- **La zone 5**



C'est celle qui permet aux utilisateurs du système de se connecter à leurs comptes personnels.

2. Consulter les statistiques :

Pour consulter les statistiques, on va faire le choix des informations à trier et du type de représentation voulue dans la zone 2 de la page d'accueil.

Exemple : nous voulons consulter la répartition à l'échelle nationale des enseignants du secondaire en fonction de leur sexe, leur secteur d'activité et leur grade sous forme de tableau, graphique et liste.

d. Posons le curseur de la souris sur « national »



En survolant « national », le système nous permet de choisir la représentation sous laquelle nous voulons nos statistiques.

- « statistiques » : nous donne les statistiques sous forme de tableau

- « graphiques » : nous donne les statistiques sous forme graphique

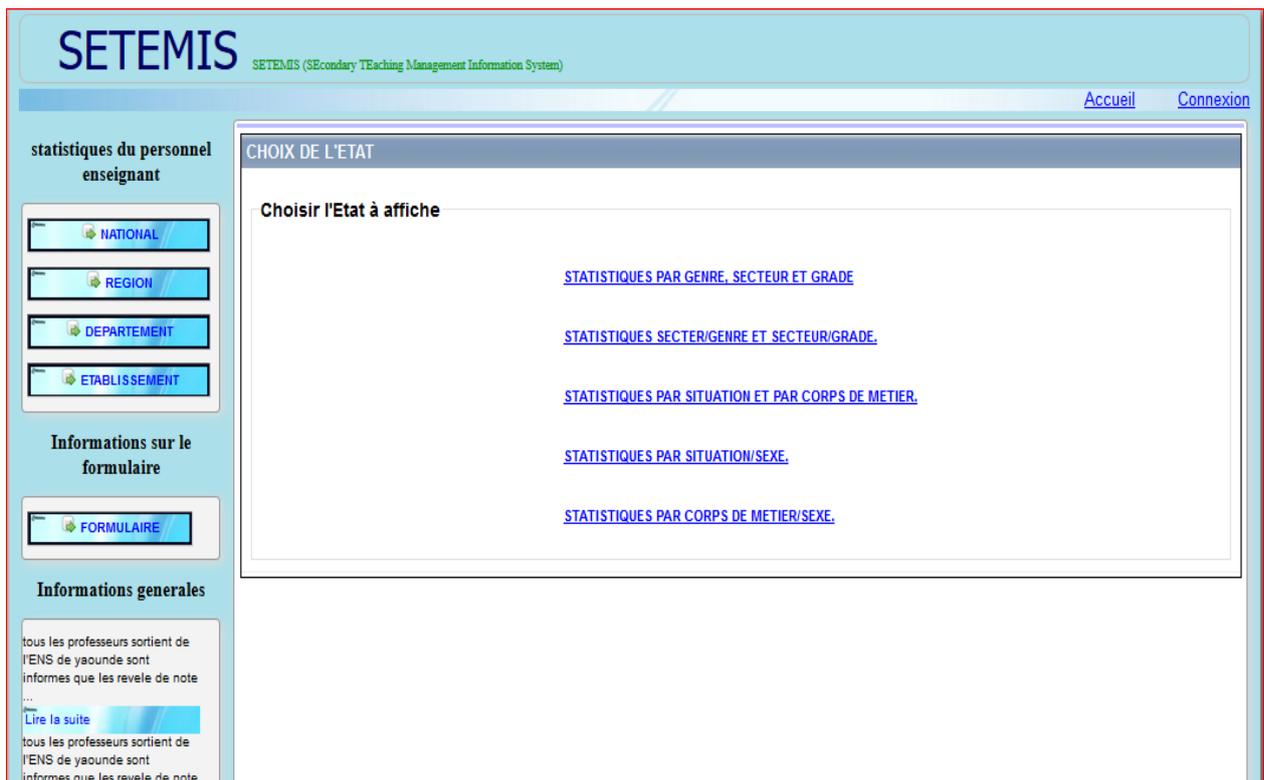
- « listes » : nous donne une liste d'enseignant en fonction des paramètres préalablement choisis.

e. Pour les besoins de notre exemple, nous allons choisir « statistiques »



En posant le curseur sur notre choix celui-ci s'éclaircit.

f. Après avoir cliqué sur « Statistiques » on obtient la page ci-dessous :



Cette page nous présente une liste de statistique.il ne nous reste plus qu'à cliquer sur ce qui nous intéresse. Pour les besoins de notre exemple nous allons cliquer sur : « STATISTIQUES PAR GENRE, SECTEUR ET GRADE ».

On obtient la page suivante :

Statistique National par Genre, Secteur et Grade [Imprimer](#)

REGION	GENRE		SECTEUR			GRADE								
	FEMMES	HOMMES	Publique	Privé	TOTAL	PLEG	PCEG	PLET	PCET	PENI	PAENI	PEP'S	PAEP'S	TOTAL
ADAMAOUA	1	0	1	0	1	1	0	0	0	0	0	0	0	1
CENTRE	3	10	5	8	13	6	5	0	0	0	0	0	0	13
EST	0	1	1	0	1	0	0	1	0	0	0	0	0	1
EXTREME-NORD	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LITTORAL	1	0	1	0	1	1	0	0	0	0	0	0	0	1
NORD	1	0	1	0	1	1	0	0	0	0	0	0	0	1
NORD-OUEST	0	1	1	0	1	0	1	0	0	0	0	0	0	1
OUEST	1	0	1	0	1	0	0	1	0	0	0	0	0	1
SUD	0	1	1	0	1	0	0	0	0	1	0	0	0	1
SUD-OUEST	1	0	1	0	1	1	0	0	0	0	0	0	0	1
TOTAL	8	13	13	8	21	10	6	2	0	1	0	0	0	21

g. la représentation graphique :

Pour représenter graphiquement, il suffit de choisir « Graphiques » et puis de sélectionné le graphique que l'on veut afficher.

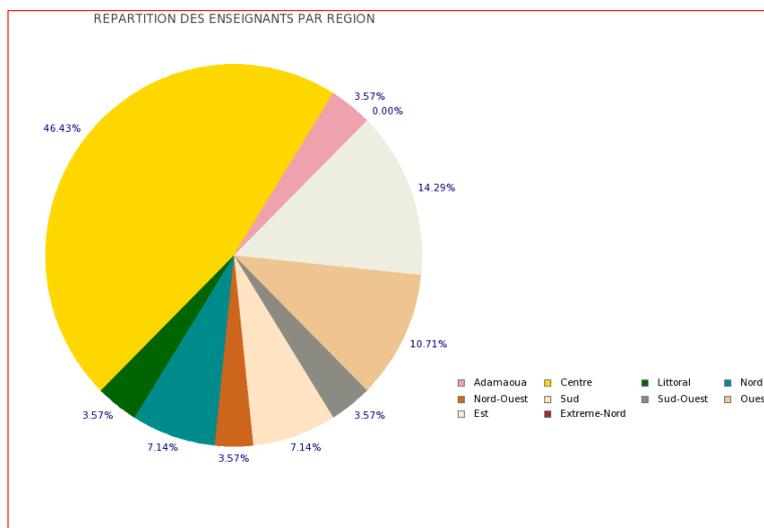
Exemple :

a.



En cliquant sur région on obtient le graphique suivant :

b.



h. Les listes :

Comme pour les autres représentations il suffit de choisir liste et de choisir la liste que l'on veut afficher :

Exemple :



CHOIX DE L'ETAT A FOURNIR LA LISTE DU PERSONNEL ENSEIGNANT

Choisir l'Etat à afficher la liste du personnel enseignant

PAR SEXE PAR SECTEUR PAR GRADE

DETACHEMENT EN STAGE GENERALE

E FORMATION POSTE PAR AGE

En faisant un clic sur « genre » on obtient :

a.

LISTES DU PERSONNEL ENSEIGNANT NATIONAL.

Renseignement sur la sexe.

Genre : - Choisir le genre - ▾

Annuler Valider

Ici on va choisir le genre et cliquer sur valider. Lorsqu'on choisit « Féminin » on sera redirigé vers la page suivante :

SETEMIS (SEcondary TEaching Management Information System)

Accueil Connexion

statistiques du personnel enseignant

NATIONAL REGION DEPARTEMENT ETABLISSEMENT

Informations sur le formulaire

FORMULAIRE

Informations generales

LISTE DES ENSEIGNANTS NATIONAL.

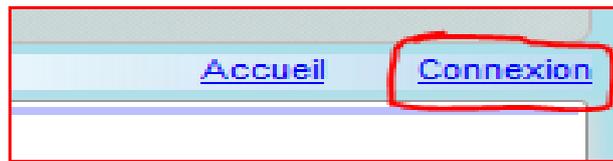
Rechercher Imprimer Graphique

Liste des enseignants de : Feminin

MATRICULE	NOM	PRENOM	MATRIMONIALE	REGION_ORIGINE	STRUCTURE	POSTE	SITUATION
04W167	CHEWO	STEPHANIE JOSIANE	ETABLISSEMENT	CENCEUR	INTEGRE	Feminin	INTEGRE
08E420B	DIBEND	FRANCIS	ETABLISSEMENT	ENSEIGNANT		Feminin	
16AD0002	ALANG	FRANCE	DRES	DELEGUE REGIONAL	INTEGRE	Feminin	INTEGRE
16AD0001	AMPELE	NADEGE	DRES	DELEGUE REGIONAL	INTEGRE	Feminin	INTEGRE
16AD0003	BILONG	BERTILLE	DRES	DELEGUE REGIONAL	INTEGRE	Feminin	INTEGRE
16AD0004	BISSOU	DOMINIQUE	DRES	DELEGUE REGIONAL	INTEGRE	Feminin	INTEGRE
16AD0006	CHOUMKEU	VANESSA	DRES	DELEGUE REGIONAL	INTEGRE	Feminin	INTEGRE
16AD0008	DJITEU	DIANE	DRES	DELEGUE REGIONAL	INTEGRE	Feminin	INTEGRE

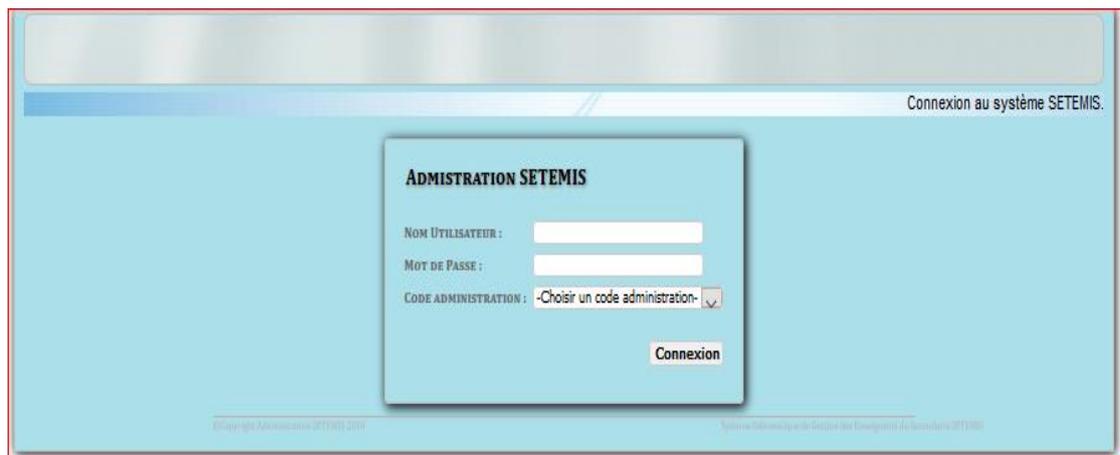
Elle nous présente la liste des enseignants de sexe féminin à l'échelle nationale.

IV. Se connecter.



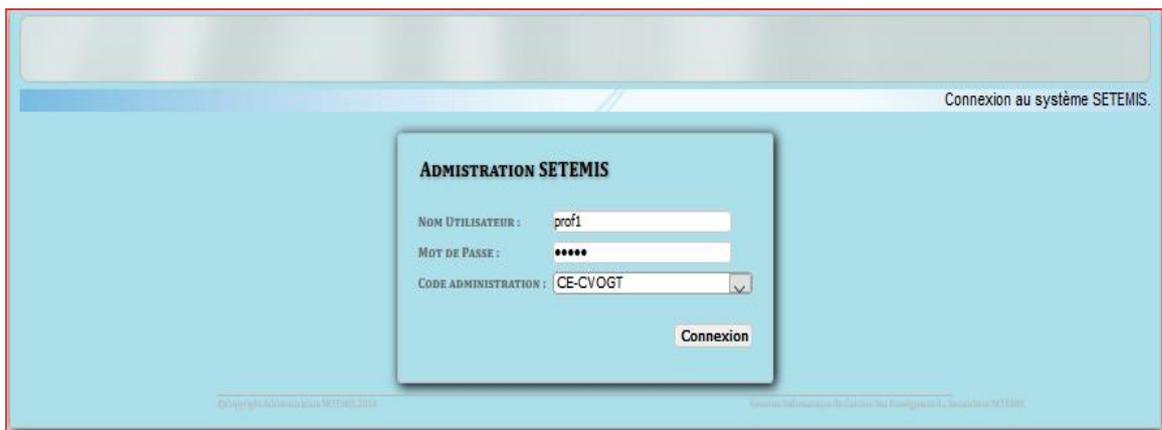
Pour se connecter à un compte, on va cliquer sur « connexion » dans la zone 4 de la page.

Après avoir choisi connexion, on accède à la page de connexion.

A screenshot of the SETEMIS login page. The page has a light blue background. At the top right, it says 'Connexion au système SETEMIS.'. In the center, there is a white box with a grey border titled 'ADMISTRATION SETEMIS'. Inside this box, there are three input fields: 'NOM UTILISATEUR:' with an empty text box, 'MOT DE PASSE:' with an empty text box, and 'CODE ADMINISTRATION:' with a dropdown menu showing '- Choisir un code administration-'. Below these fields is a 'Connexion' button. At the bottom of the page, there is small text: 'Copyright Administration SETEMIS 2014' on the left and 'Système d'Information des Services des Enseignants de Secondaire SETEMIS' on the right.

Cette page permet de se connecter à son compte personnel en rentrant son login, son mot de passe et en choisissant le code d'administration de la structure dans laquelle on travaille.

Exemple :

A screenshot of the SETEMIS login page, similar to the previous one, but with example data entered. The 'NOM UTILISATEUR:' field contains 'prof1', the 'MOT DE PASSE:' field contains '*****', and the 'CODE ADMINISTRATION:' dropdown menu is set to 'CE-CVOGT'. The 'Connexion' button is still present. The footer text is the same as in the previous screenshot.

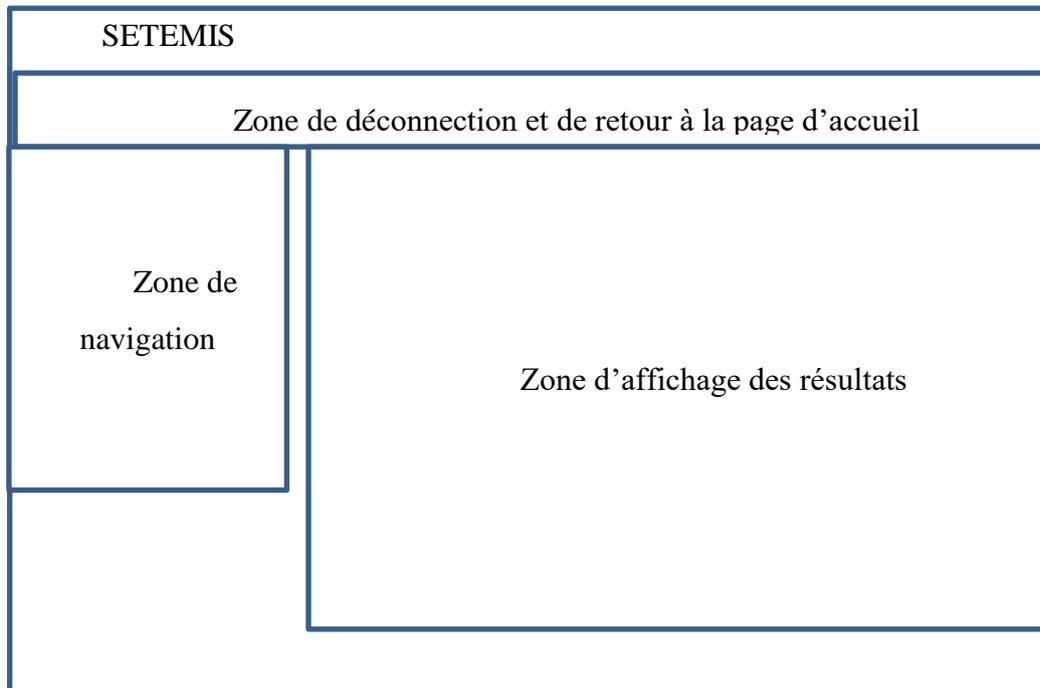
On a un enseignant qui a pour :

- nom d'utilisateur : prof1
- Mot de passe : prof1

Et qui enseigne au collège Vogt de Yaoundé qui a pour code d'administration : CE-CVOGT

V. Comptes utilisateurs

Les interfaces des comptes utilisateurs sont divisées en 3 zones :



NB : d'un type de compte à un autre la zone de navigation n'est pas la même. Elle nous présente les fonctionnalités que l'on est en droit de manipuler.

Ainsi en fonction des types d'utilisateur on aura 5 interfaces de comptes utilisateurs :

- Administrateur système :

SETEMIS

[Page visiteur](#) | [Bienvenue admin](#) | [Accueil](#) | [Déconnexion](#)

Accueil

C'EST ICI LA PAGE D'ACCUEIL.

Gestion Formulaire

- Remplir Formulaire
- Formulaires en attente
- Valider | Rejeter
- Modifier | Supprimer

Affectation/Nomination

- Effectuer Nomination
- Effectuer Affectation
- Valider Affectation

Gestion Besoins

- Ajouter Besoins
- Liste Besoins
- Modifier | Supprimer

Gestion Utilisateurs

- Imprimer la liste
- Liste Utilisateurs
- Ajouter Utilisateur
- Modifier | Supprimer
- Editer les droits

Gestion Etablissement

- Liste Etablissement
- Ajouter Etablissement
- Modifier | Supprimer

Gestion Departement

- Liste Departement
- Ajouter departement
- Modifier | Supprimer

Gestion Administrateurs

- Liste Admin
- Ajouter Admin
- Modifier | Supprimer
- Editer droits

©Copyright Administration SIGES,ENS 2015

Systeme Informatique de Gestion des Enseignants du Secondaire, SIGES,ENS

- Délégué régional :

SETEMIS SETEMIS (Secondary Teaching Management Information System)

[Page visiteur](#) | [Bienvenue ALANG](#) | [Accueil](#) | [Déconnexion](#)

Accueil

C'EST ICI LA PAGE D'ACCUEIL

Gestion Formulaire

- Remplir Formulaire
- Formulaires en attente
- Valider | Rejeter
- Modifier | Supprimer

Affectation/Nomination

- Effectuer Affectation
- Valider Affectation

Gestion Besoins

- Liste Besoins

Gestion Utilisateurs

- Imprimer la liste
- Liste Utilisateurs
- Ajouter Utilisateur
- Modifier | Supprimer

Gestion Etablissement

- Liste Etablissement

Gestion Departement

- Liste Departement

©Copyright Administration SETEMIS 2016 SETEMIS (Secondary Teaching Management Information System)

- Enseignant :

SETEMIS SETEMIS (Secondary Teaching Management Information System)

[Page visiteur](#) | [Bienvenue 04W167](#) | [Accueil](#) | [Déconnexion](#)

Accueil

C'EST ICI LA PAGE D'ACCUEIL

Gestion Formulaire

- Remplir Formulaire
- Formulaires en attente
- Modifier | Supprimer

©Copyright Administration SETEMIS 2016 SETEMIS (Secondary Teaching Management Information System)

- Délégué départemental

SETEMIS SETEMIS (Secondary Teaching Management Information System)

[Page visiteur](#) | [Bienvenue DONG](#) | [Accueil](#) | [Déconnexion](#)

Accueil

C'EST ICI LA PAGE D'ACCUEIL

Gestion Formulaire

- Remplir Formulaire
- Formulaires en attente
- Valider | Rejeter
- Modifier | Supprimer

Affectation/Nomination

- Effectuer Affectation
- Valider Affectation

Gestion Besoins

- Liste Besoins

Gestion Utilisateurs

- Imprimer la liste
- Liste Utilisateurs
- Ajouter Utilisateur
- Modifier | Supprimer

Gestion Etablissement

- Liste Etablissement

Gestion Departement

- Liste Departement

©Copyright Administration SETEMIS 2016 SETEMIS (Secondary Teaching Management Information System)

- Chef d'établissement

SETEMIS SETEMIS (Secondary Teaching Management Information System)

[Page visiteur](#) | [Bienvenue NJOUMBA](#) | [Accueil](#) | [Déconnexion](#)

Accueil

C'EST ICI LA PAGE D'ACCUEIL

Gestion Formulaire

- Remplir Formulaire
- Formulaires en attente
- Valider | Rejeter
- Modifier | Supprimer

Affectation/Nomination

- Effectuer Affectation
- Valider Affectation

Gestion Besoins

- Ajouter Besoins
- Liste Besoins
- Modifier | Supprimer

Gestion Utilisateurs

- Imprimer la liste
- Liste Utilisateurs
- Ajouter Utilisateur
- Modifier | Supprimer

Gestion Etablissement

- Liste Etablissement

Gestion Departement

- Liste Departement

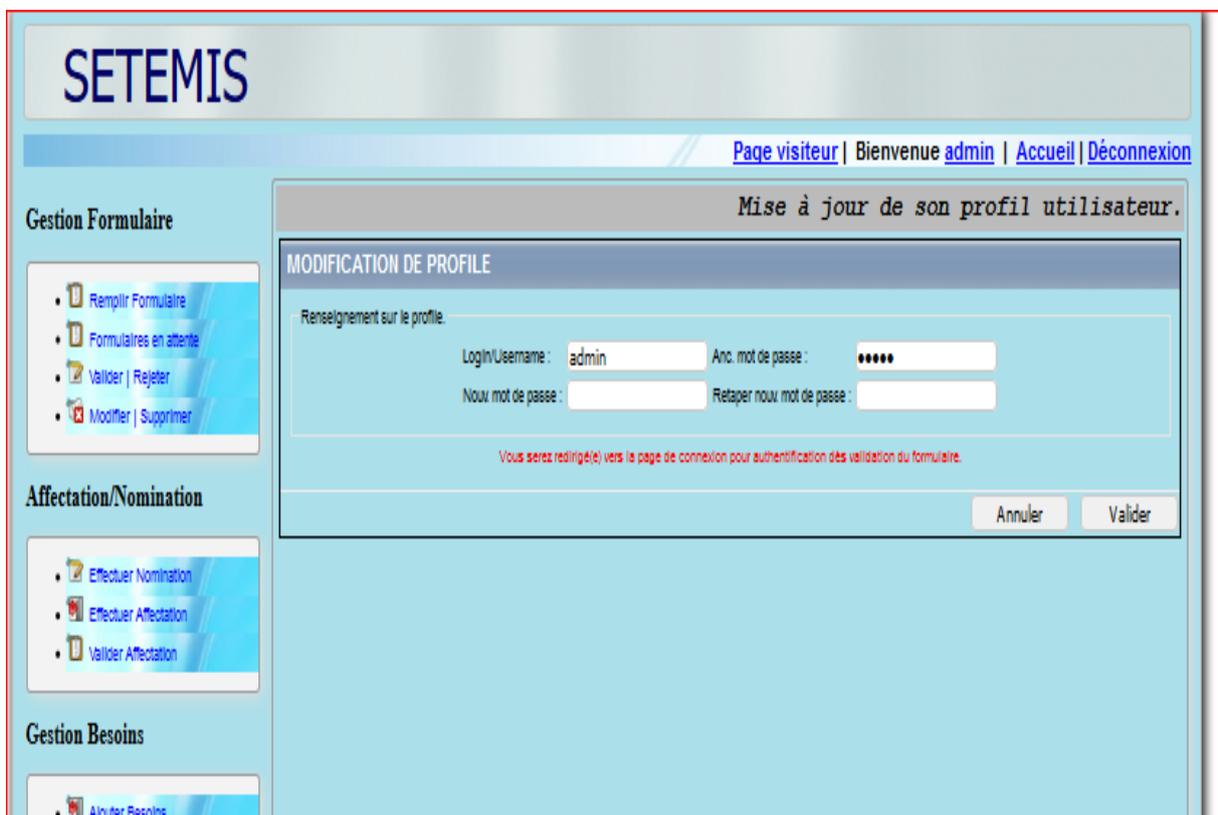
©Copyright Administration SETEMIS 2016 SETEMIS (Secondary Teaching Management Information System)

VI. Modifier son login et son mot de passe

Pour modifier son login et son mot de passe, on clique sur son nom d'utilisateur en haut dans la page :



On est redirigé vers cette page :



VII. Se déconnecter

Pour se déconnecter on clique sur Déconnexion.

