

Mise en place d'un système informatique de gestion de mutation des agents de l'enseignement primaire, secondaire et professionnel

[Implementation of a computerized transfer management system for primary, secondary and vocational education agents]

Elie MWEZ RUBUZ¹, Bazin NSHIMBA ILUNGA¹, Elam KYUNGU LUKOMBA², Patrice MUSUL NAWAJ³, and Dennis TSHIKUDI TSHIKUDI⁴

¹Département Informatique, Institut Supérieur Pédagogique, Kabongo, Haut-Lomami, RD Congo

²Département Informatique, Institut Supérieur de Statistique, Lubumbashi, Haut-Katanga, RD Congo

³Département Informatique, Université Méthodiste au Katanga, Mulungushi, Lualaba, RD Congo

⁴Département Informatique, Institut Supérieur de Techniques Appliquées, Kolwezi, Lualaba, RD Congo

Copyright © 2023 ISSR Journals. This is an open access article distributed under the ***Creative Commons Attribution License***, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT: The choice of this study theme is related to the object that attracted our attention as scientific researchers. It all started with an exchange and observation of the functioning of the control and payroll service of the teachers of the province of Haut-Katanga (SECOPE in the acronym), so we were driven by the desire to improve the process to allow good management and the smooth running of the operations related to the transfer of the agents as well as their pay. This study will serve to bring a computerized solution to the Ministry of Education and more precisely to the service of control and pay of the teachers by a management tool based on the functioning of the process of transfer of the agents and to better manage these last ones to maximize the receipts. This work has led to the following results (Know the number of all agents for a given period in relation to their task, Quickly exchange data between different positions involved, Make each agent's information accessible through a shared database that offers better data storage and access techniques while eliminating the loss of records of all transferred agents).

KEYWORDS: SECOPE, Agent, Transfer, Education, Ministry, Payment, Teachers.

RESUME: Le choix de ce thème d'étude est en rapport avec l'objet qui a attiré notre attention en tant que chercheur scientifique. Tout est parti par un échange et une observation de fonctionnement du service de contrôle et de la paie des enseignants de la province du Haut-Katanga (SECOPE en sigle), ainsi nous étions poussés par le goût de vouloir améliorer le processus pour permettre une bonne gestion et le bon déroulement des opérations en rapport avec la mutation des agents ainsi que leurs paies. Cette étude servira d'apporter une solution informatique au ministère de l'enseignement et plus précisément au service de contrôle et de la paie des enseignants par un outil de gestion basé sur le fonctionnement de processus de mutation des agents et de mieux gérer ces derniers pour maximiser les recettes. Ce travail a abouti aux résultats ci-après (Connaitre le nombre de tous les agents pour une période donnée par rapport à leur tâche, Echanger rapidement les données entre différents postes concernés, Rendre accessible les informations de chaque agent grâce à une base de données partagée qui offre des meilleures techniques de stockage et d'accès aux données tout en éliminant la perte de dossiers de tous les agents mutés).

MOTS-CLEFS: SECOPE, Agent, Mutation, Enseignement, Ministère, Paiement, Enseignants.

1 INTRODUCTION

Le besoin d'avoir une information sûre, avec un délai réduit constitue un véritable défi de toute entreprise et de toute personne. Il est ainsi important et nécessaire de concevoir des applications permettant d'accéder plus facilement aux informations tout en réduisant le temps d'accès [1]. Ayant fait plusieurs observations au service de contrôle et de la paie des enseignants (SECOPE) en tant que chercheur, nous avons constatés que la gestion de certaines tâches est encore manuelle et de fois avec de traitements informatiques limités.

Pour rendre facile l'accès aux informations, il existe plusieurs solutions informatiques; et pour ce qui nous concerne, le constat fait sur l'environnement, révèle que l'informatique est plus considéré et utilisé comme un outil et un support de certaines disciplines de l'enseignement, elle devient un objet d'étude en tant que discipline scientifique et administrative. Ce présent travail décrit toute l'analyse et la conception d'une application de gestion des mutations des agents. En effet les problèmes de gestion que connaît SCOPE sont les suivants:

- La perte des informations sur les agents mutés vu qu'ils sont enregistrés sur des supports papiers ainsi que la mauvaise conservation de ces dernières
- Mauvaise prise en charge des agents désirant d'être muté suite à un mauvais suivi de dossier en rapport avec la mutation
- Difficulté de produire des rapports statistiques des agents mutés pendant les périodes des contrôles physiques des enseignants (agents)
- Lors de la mise à jour, certaines informations des agents ne sont pas retrouvées

Ainsi compte tenu des problèmes cités ci-dessus, nous nous sommes posé la question ci-dessous: Comment arrivé à améliorer le processus de gestion des mutations des agents pour pallier aux différents problèmes ?

2 METHODOLOGIE

Pour arriver à atteindre l'objectif que nous nous sommes assignés, nous avons opté pour la méthode analytique pour la collecte de données et conceptuelle pour l'analyse du système à concevoir pour bien modéliser en découpant en itération les processus, et incrémenter par la suite pour pouvoir arriver à une solution meilleure [2].

3 PRESENTATION DU SECOPE / HAUT-KATANGA

Les bureaux de la division provinciale du service de contrôle et paie des agents (SECOPE) est une institution publique dont la direction provinciale de Lubumbashi se situé au numéro 105, de l'avenue des chutes au quartier MAKUTANO, dans la commune de Lubumbashi, province du Haut-Katanga, ville de Lubumbashi en République Démocratique du Congo. Avant 1984, la paie des enseignants était caractérisée par:

- L'existence des écoles fictives;
- L'ignorance par le pouvoir des effectifs réels des écoles et des enseignants;
- L'existence des enveloppes salariales fantaisistes déclarées faussement déficitaires par des bureaux gestionnaire d'écoles;
- Pléthore des enseignants faussaires dans les rangs des enseignants

Face à cette situation, le gouvernement Zaïrois en partenariat avec la coopération technique belge mis sur pied un projet appelé services des maîtrises des effectifs des enseignants qui aurait pour objectif de chercher à présenter au gouvernement un nombre plus au moins exacte des enseignants dans la ville province de Kinshasa. La maîtrise des effectifs et la fiscalisation des enveloppes salariales à Kinshasa ont poussés le gouvernement par le biais du ministère de l'enseignement primaire, secondaire et professionnel à élargir l'expérience dans les provinces intérieures du pays. A partir de ce moment, le projet connaît une structuration et une organisation efficaces, des projets datés d'une direction centrale basée à Kinshasa ayant à sa tête un directeur chef de service secondé par deux directeurs suivi de la présence du personnel dans les écoles et dans les bureaux gestionnaires, mise à jour régulier de fichier de paie vers les années 2011-2013, la bancarisation des salaires intervient. Le rôle du SECOPE se trouve réduit, il continue à présenter les effectifs destinés à être pris en compte par le ministre du budget, désormais émet les états liquidatifs pour la paie des enseignants dans les banques.

3.1 FONCTIONNEMENT DU SYSTEME

3.1.1 PROCESSUS DE MUTATION VOLONTAIRE

Le processus de mutation volontaire des agents commence lorsqu'un agent effectue une demande de mutation auprès du gestionnaire. Le gestionnaire analyse la demande de l'agent qui peut être refusée ou acceptée. Le gestionnaire transmet ladite demande auprès de la cellule informatique. Cette dernière lorsqu'elle reçoit la demande de l'agent, elle effectue la mise à jour des informations de l'agent et du poste sollicité.

Après la mise à jour, la cellule informatique établit la fiche de modification de l'agent et le listing de paie des agents puis les transmettent au secrétaire général pour la validation (approbation). Après validation de la fiche et du listing de paie, le secrétaire général retourne la fiche au gestionnaire avec tous les détails de l'affectation. Ainsi l'agent est muté au poste demandé et son enveloppe salariale quitte de l'ancien poste vers le nouveau poste dont le chef d'établissement devra réceptionner pour réorganiser la paie.

3.1.2 MUTATION DISCIPLINAIRE

Le processus de mutation disciplinaire commence n'importe quand c'est-à-dire à chaque fois qu'une peine est infligée à l'agent. Le chef d'établissement ouvre l'action disciplinaire à l'agent c'est-à-dire on lui sanctionne par rapport à l'acte posé et informe ou appelle le gestionnaire pour interroger l'agent concerné.

L'agent explique ou exprime l'acte posé au gestionnaire qui, après explication de l'agent, il clôture l'action disciplinaire qui a été ouverte par le chef d'établissement ; si l'acte posé est lourd, le gestionnaire demande l'attestation de services rendus au chef direct de l'agent ; sinon la mutation disciplinaire sera annulée ou n'aura pas lieu.

Le chef d'établissement considéré comme le chef direct de l'agent édite le document demandé et le présente au gestionnaire. Le gestionnaire rédige la demande de commission d'affectation de l'agent à un nouveau poste et le transmet à la cellule informatique pour mettre à jour les informations de l'agent incriminé. Après la mise à jour des informations, la cellule informatique envoie la fiche de mise à jour au gestionnaire pour muter l'agent au nouveau poste. Fin processus

3.2 DIAGRAMME D'ACTIVITES

Les diagrammes d'activités permettent de mettre l'accent sur les traitements.

3.2.1 PROCESSUS DE MUTATION VOLONTAIRE

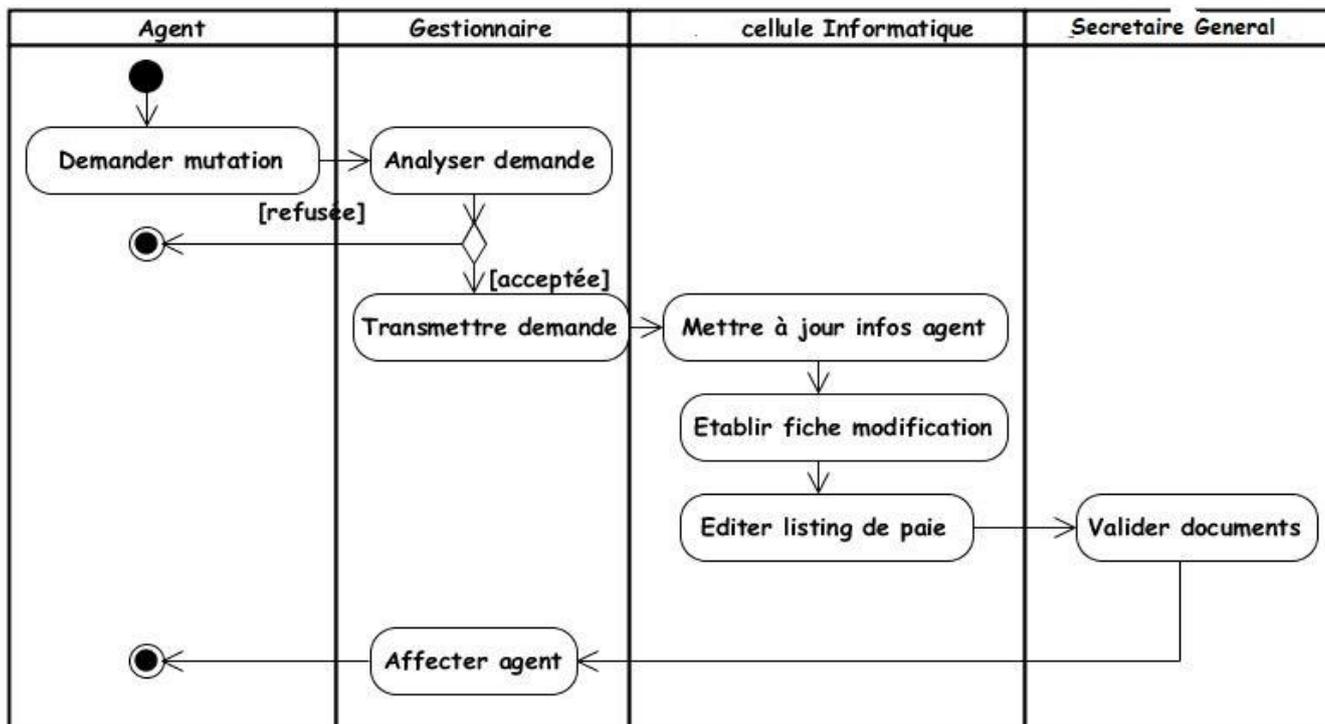


Fig. 1. Diagramme d'activités du processus métier de mutation

3.2.2 PROCESSUS DE MUTATION DISCIPLINAIRE

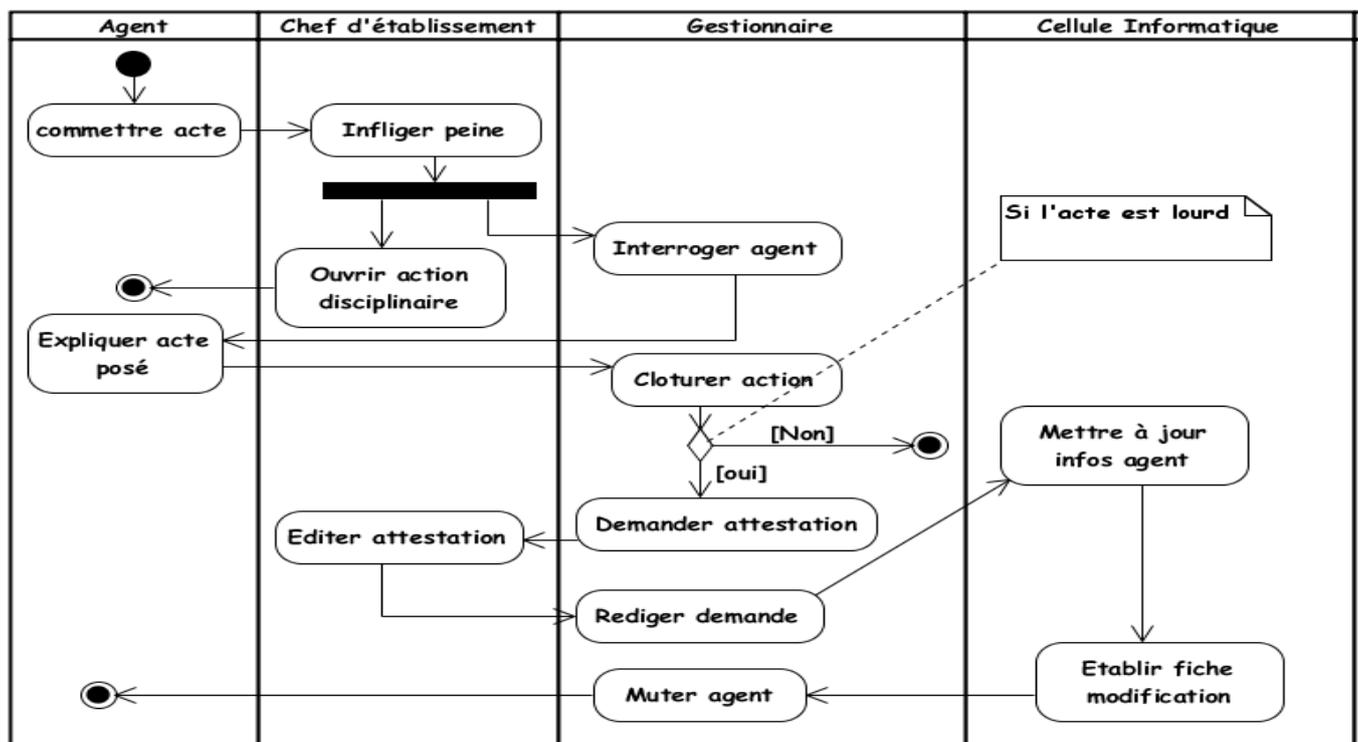


Fig. 2. Diagramme d'activités du processus métier

4 CONCEPTION DE LA SOLUTION

Notre projet s'inscrit dans la sphère organisationnelle et technique du service de contrôle et de la paie des enseignants (SECOPE). Celui-ci englobe toutes les informations relatives à la gestion de mutation des agents. Les structures du système d'information existant, les plus directement impliquées dans la gestion des informations sur les mutations des agents effectuées sont l'inspection (gestionnaire), la cellule informatique, le secrétariat provincial de direction et les établissements.

Notre projet aura pour aboutissement d'abord la formalisation d'un nouveau système informatique qui permettra une gestion efficace des informations sur toutes les mutations effectuées par les services impliqués; ensuite l'automatisation de certaines tâches programmables de ce système. Parmi ces différentes tâches, nous relèverons la gestion des informations relatives aux agents mutés, lesquelles informations relèvent d'un processus à deux ordres:

- Les informations relatives à la demande de mutation: l'enregistrement des demandes traitées pour chaque agent pour une meilleure identification
- Les informations relatives à la mise à jour des fichiers des agents: la mise à jour des infos sur des agents, l'établissement et l'impression des fiches d'affectation et des listings de paie des agents

4.1 IDENTIFICATION DES ACTEURS

Les utilisateurs humains directs: identifiez tous les profils possibles, sans oublier l'administrateur, l'opérateur de maintenance, etc. Les autres systèmes connexes qui interagissent aussi directement avec le système. Vérifiez que les acteurs communiquent bien directement avec le système, par émission et/ou réception de messages avec le système, mais uniquement par le biais d'un des véritables acteurs [3].

Tableau 1. Recensement des utilisateurs

ACTEUR	ROLE
Agent	Tout enseignant demandant une mutation
Gestionnaire	Le chef chargé de traiter les demandes de mutation des agents
Cellule informatique	Elle est chargée de toutes les mises à jour des informations des agents et l'édition du listing de paie des agents
Administrateur système	Il est chargé de gérer les utilisateurs du système informatique.
Secrétaire	Il est chargé d'approuver les dossiers de mutations des agents

4.2 DIAGRAMME DE CONTEXTE DU METIER

La modélisation du processus métier est une activité dont l'objectif est de construire un modèle qui décrit les aspects statiques et dynamiques du cas d'utilisation en étude en ignorant les détails de l'implémentation technique et informatique [4]. Il est indiqué de placer le système en examen c'est-à-dire sous étude dans son environnement. Pour cela on met en place un diagramme de contexte:

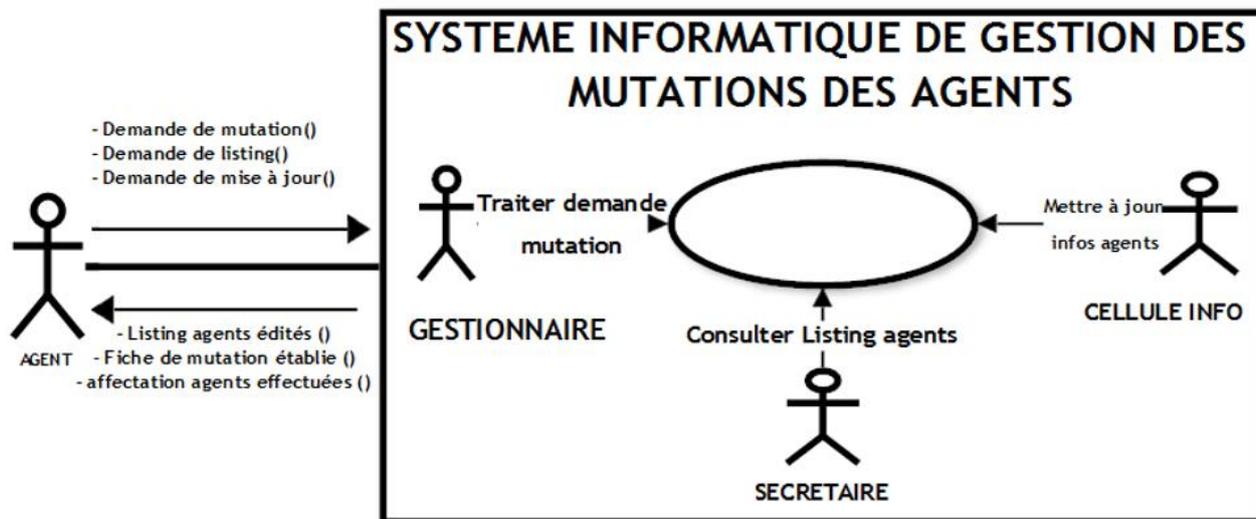


Fig. 3. Diagramme de contexte du système informatique

4.3 ANALYSE ET SPECIFICATIONS DES FONCTIONNALITES

Cette phase représente un point de vue « fonctionnel » de l'architecture système. Par le biais des cas d'utilisation, nous serons en contact permanent avec les acteurs du système en vue de définir les limites de celui-ci, et ainsi éviter de trop s'éloigner des besoins réels de l'utilisateur final.

4.3.1 DIAGRAMME DES CAS D'UTILISATION

Le diagramme de cas d'utilisation suivant illustre le fonctionnement du processus de gestion de mutation au sein du service de contrôle et de la paie des enseignants. Un cas d'utilisation spécifie, une fonction offerte par l'application à son environnement, il est spécifié par un intitulé [5].

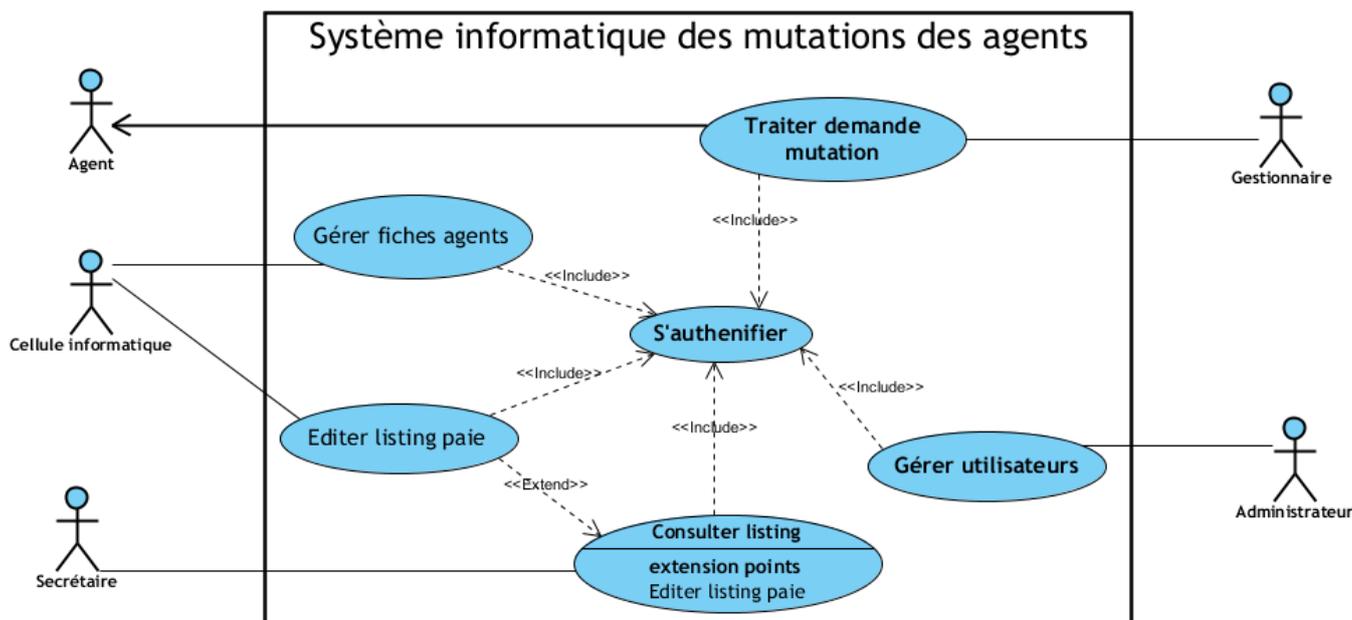


Fig. 4. Diagramme de cas d'utilisation du système informatique

4.4 DIAGRAMME DE CLASSES DE CONCEPTION

Dans ces diagrammes, nous allons produire des classes qui seront à l'implémentation du système informatique. Pour une classe, le couplage est la mesure de la quantité d'autres classes avec lesquelles elle est connectée par des associations, des relations de dépendances, etc., durant toute l'élaboration du diagramme de conception, il faut veiller à conserver un couplage faible pour obtenir une application plus évolutive et plus facile à maintenir [6].

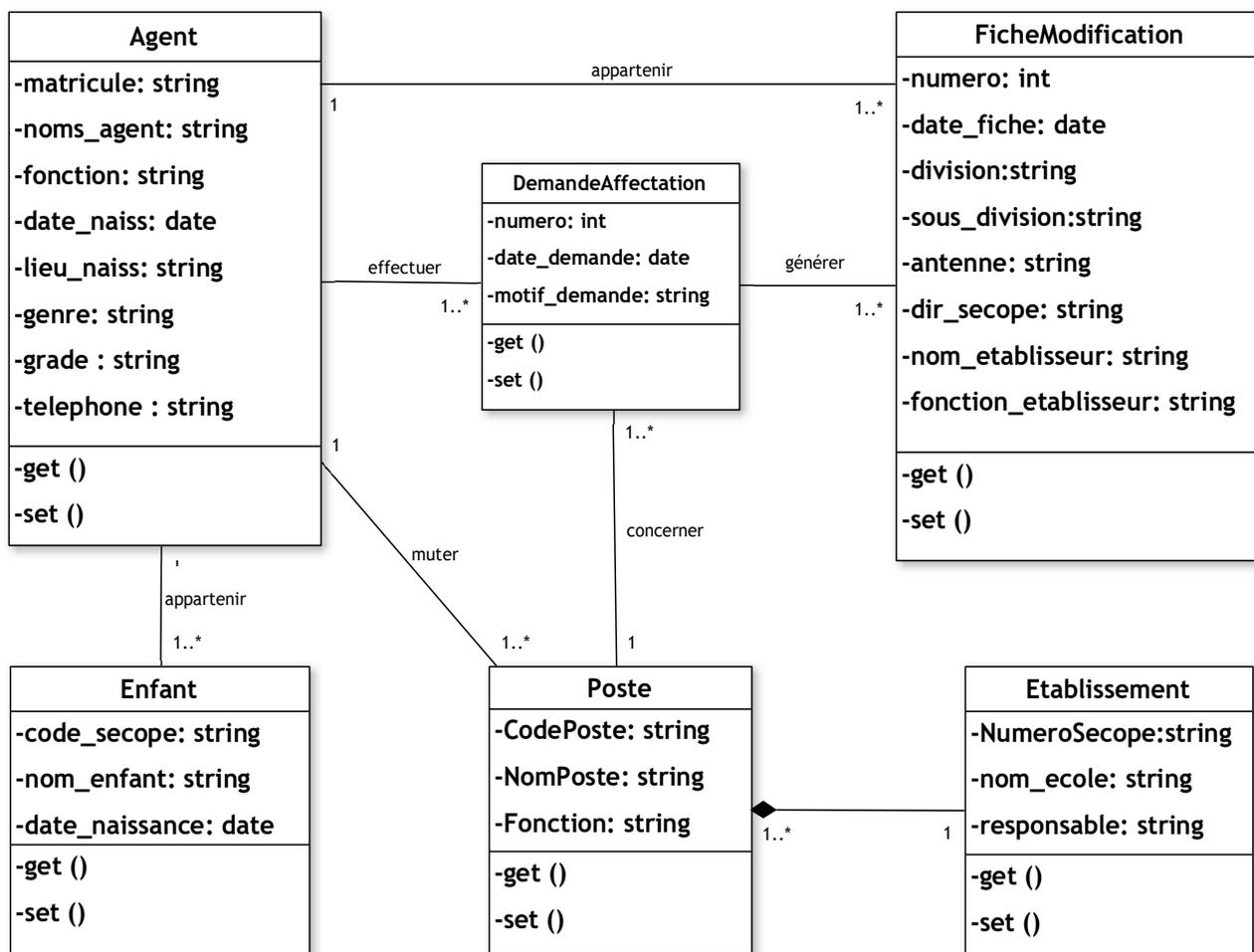


Fig. 5. Diagramme de classes de conception

5 PRESENTATION DE LA SOLUTION INFORMATIQUE

5.1 CONCEPTION DES ARCHITECTURES DU LOGICIEL

L'architecture logiciel décrit d'une manière symbolique et schématique les différents éléments d'un ou de plusieurs systèmes informatiques, leurs interrelations et leurs interactions. Contrairement aux spécifications produites par l'analyse fonctionnelle, le modèle d'architecture, produit lors de la phase de conception, ne décrit pas ce que doit réaliser un système informatique mais plutôt comment il doit être conçu de manière à répondre aux spécifications. L'analyse décrit le « quoi faire » alors que l'architecture décrit le « comment le faire ? » [7].

Une structure un ensemble d'éléments du logiciel liés par une relation spécifique alors qu'une vue est la représentation de la structure. Un pattern d'architecture = la description d'une architecture-type ayant été testée sur plusieurs logiciels et éprouvée par le temps. Le style capture les concepts et principes fondamentaux d'une certaine manière d'organiser les composants de son logiciel pour atteindre un objectif spécifique et satisfaire les exigences d'un cas spécifique de logiciel. Il existe plusieurs patterns d'architecture:

5.2 CHOIX DES ARCHITECTURES LOGICIELLES

5.2.1 CHOIX DE STYLE DE SYSTÈME DISTRIBUÉ

Le style de système distribué que nous avons choisi pour notre système est celui d'architecture en niveaux et le style d'architecture en tiers.

5.2.1.1 LE STYLE D'ARCHITECTURE EN NIVEAUX

Ce style spécifie le nombre de niveaux géographiques et organisationnels où vont se situer les environnements d'exécution du système [8]. Nous aurons le niveau de départemental qui représente le métier de gestion de mutation des agents au sein de SECOPE et le niveau central qui représente l'hébergeur distant.

5.2.1.2 LE STYLE D'ARCHITECTURE EN TIERS (*TIERS* SIGNIFIE « PARTIE » EN ANGLAIS)

Ce style spécifie l'organisation des composants d'exploitation mis en œuvre pour réaliser le système. Chaque partie indique une responsabilité technique à laquelle souscrivent les différents composants d'exploitation d'un système.

Le style d'architecture **3-tiers** correspond à la configuration la plus simple d'un système client/serveur. Dans ce cas, *il incombe aux clients de gérer l'interface utilisateur et les processus d'exploitation. Les serveurs ont pour responsabilité de traiter le stockage des données ainsi que la logique applicative et l'interception des requêtes utilisateur.* Ce type d'architecture est parfaitement bien adapté aux systèmes départementaux, dans la mesure où les concepts et les processus manipulés n'existent qu'une seule fois au sein d'un département de l'entreprise.

5.2.2 CHOIX DE STYLE DE SYSTÈME INTERACTIF

Le choix a porté sur le modèle MVC en rapport avec le système interactif vu l'importance et l'avantage de ce style qui permet la séparation des couches logicielles. Le modèle qui est devenu aujourd'hui le standard dans la conception des logiciels, permet de séparer la couche de présentation, de la couche de persistance ou d'accès aux données ainsi que celle de la logique applicative.

5.2.3 CHOIX DE L'ARCHITECTURE DE RÉFÉRENCE

Se basant sur l'environnement dans lequel nous allons instaurer notre système informatique, le choix de l'architecture de référence tombe sur celle dite application web client léger.

Application web client léger: c'est un pattern architectural le plus classique aujourd'hui et correspond donc aux applications Internet/intranet pour lesquelles la configuration du client n'est pas maîtrisable, à ceci près que l'on requiert côté client un navigateur web assez récent, comprenant le langage JavaScript. Le client navigue sur des pages dotées d'intelligence (programmée en JavaScript).

Les composants majeurs du pattern architectural client web léger se trouvent sur le serveur. Dans bien des sens, cette architecture est effectivement celle d'une application web minimale [9].

5.3 CHOIX DES ARCHITECTURES MATÉRIELLES (PHYSIQUES)

5.3.1 CHOIX DES COMPOSANTS (LOGICIELS)

Les composants sont des briques de base importantes dans la modélisation des aspects physiques d'un système. Un composant est une partie physique remplaçable d'un système qui fournit la réalisation d'un ensemble d'interfaces et s'y conforme.

On utilise les composants pour modifier les éléments physiques qui peuvent se trouver sur un nœud, comme les exécutables, les bibliothèques, les tables, les fichiers et les documents. Il regroupe aussi physiquement les éléments logiques (classes, interfaces, collaborations).

5.3.2 CHOIX DES MATERIELS

Il représente un processus ou un périphérique sur lequel les composants peuvent être déployés. Un processus est un nœud qui a des capacités de traitement c'est-à-dire il peut exécuter un composant. Un périphérique est un nœud qui n'a pas de capacité de traitement [10]. Les matériels utilisés sont les suivants:

- Les postes de travail: PC ou téléphones intelligents (Smart phones);
- Les différents éléments pour la connectivité sans fil comme le routeur, modem, câbles réseaux,

DIAGRAMME DE COMPOSANTS

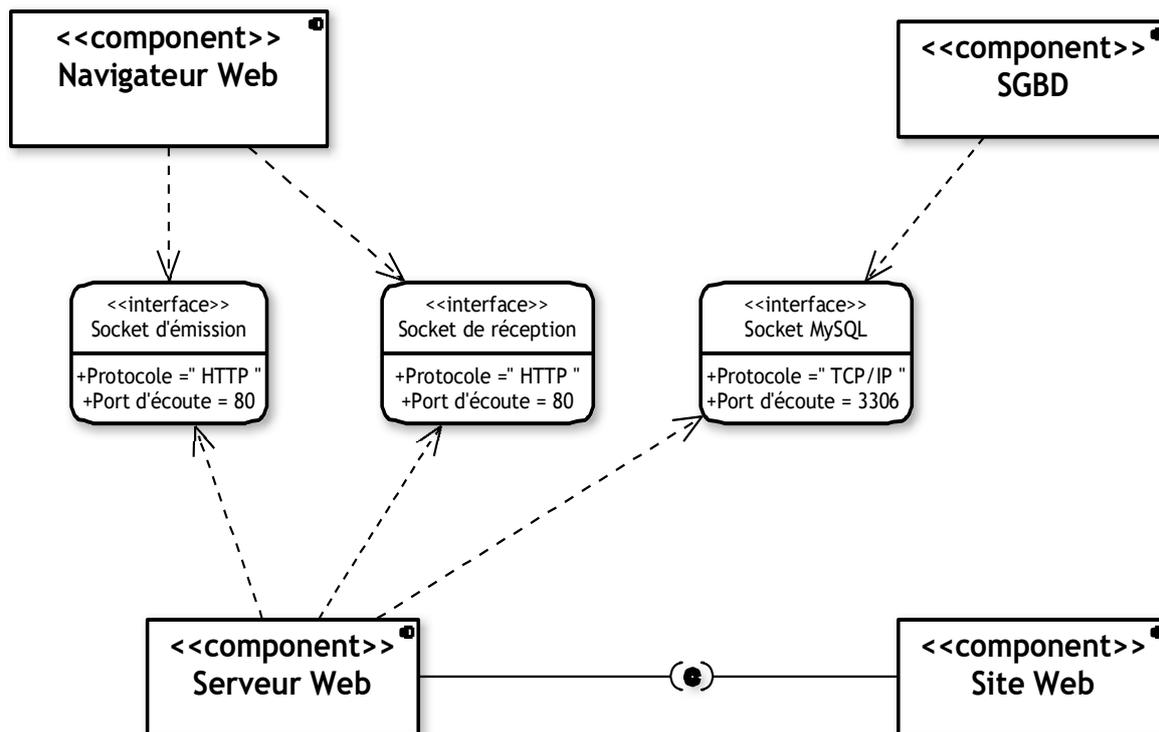


Fig. 6. Diagramme de composants systèmes

DIAGRAMME DE DEPLOIEMENT

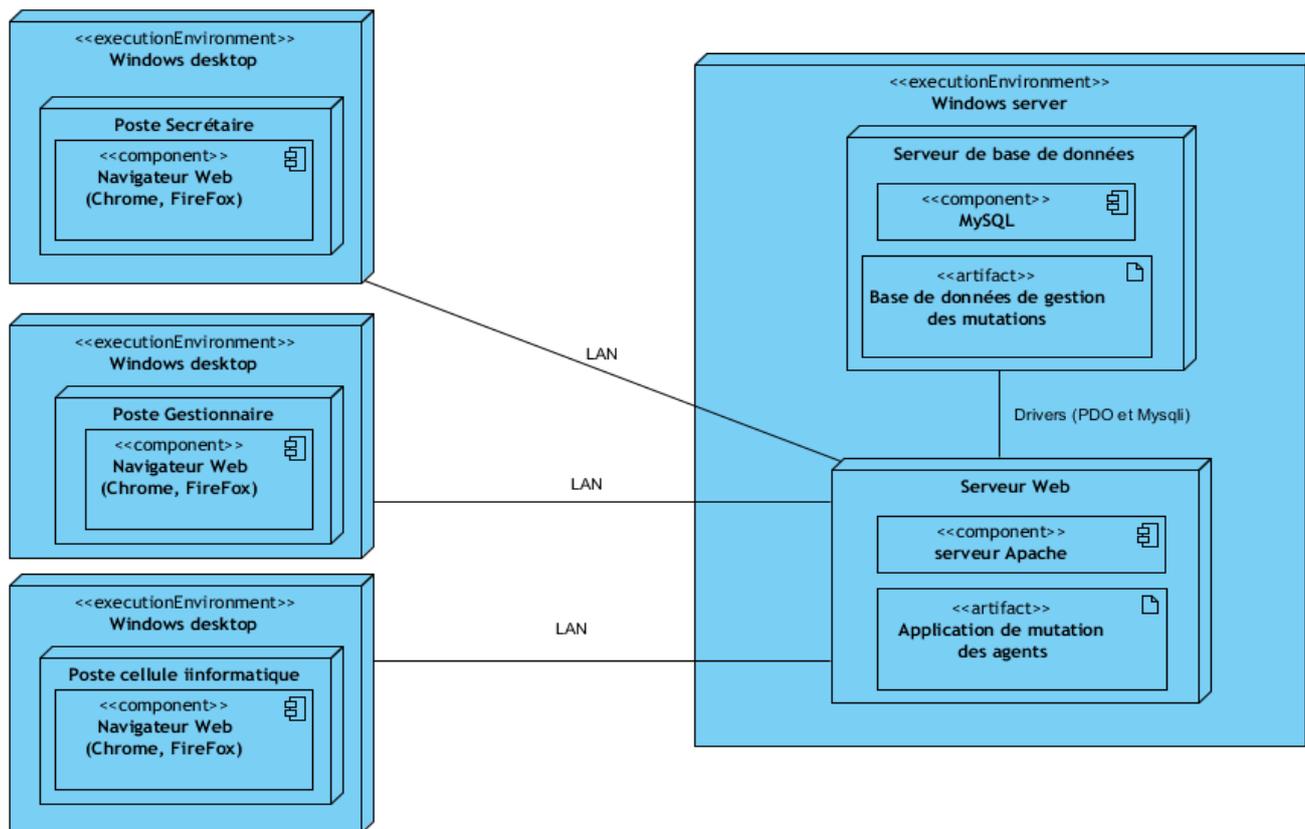


Fig. 7. Diagramme de déploiement du système

6 ENVIRONNEMENT DE TRAVAIL ET TECHNOLOGIES

Les technologies, logiciels, Framework mis en œuvre pour développer le logiciel sont les suivants: Les éditeurs Web sont des applications qui facilitent le formatage des données pour la conception des pages Web et ils ont une importance capitale à être présentés dans ce travail en vue de montrer à nos lecteurs ceux que nous avons utilisés pour développer de ce site web. Notre matériel tourne sur un système d'exploitation Windows de Microsoft [11].

6.1 FRAMEWORK

Un Framework informatique est un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel [Maurice CHAVELLI, 2016]. Autrement dit une base cohérente avec des briques toutes prêtes à disposition. Il existe des Framework pour tous les langages de programmation et en particulier pour PHP. L'utilité d'un Framework est d'éviter de passer du temps à développer ce qui a déjà été fait par d'autres souvent plus compétents et qui a en plus été utilisé et validé par de nombreux utilisateurs.

L'idée des Framework est de vous permettre de développer plus vite et à moindre coût. En effet, une partie des opérations répétitives sont fournies par le Framework. Par exemple, l'architecture MVC est très largement utilisée actuellement dans le monde du développement web. Alors que le concept est assez « simple », si vous voulez l'implémenter vous-même, cela va vous prendre un certain temps (et même un temps certain). L'usage du Framework est donc une plus-value.

6.2 PRÉSENTATION DU FRAMEWORK CODEIGNITER

Ainsi, CodeIgniter est un Framework PHP qui va nous permettre de développer plus rapidement des applications (sites) web. Il fournit un ensemble varié d'outils qui nous permettra de réaliser les tâches les plus communes, tout en restant très

simple. Il a en effet été développé avec pour objectif la simplicité et la rapidité. CodeIgniter est un Framework MVC. Il implémente les modèles, les vues et les contrôleurs. Toutefois, il est très souple quant au concept de MVC. Il ne nécessite par exemple pas l'usage systématique d'un modèle comme d'autres Framework. Aussi, CodeIgniter est facilement extensible.

6.3 CHOIX DU LANGAGE DE PROGRAMMATION

WAMP: est un acronyme informatique signifiant: « Windows, Apache, MySQL, PHP » dans la majorité des cas mais aussi parfois, Il joint PHPMyAdmin à MySQL, un outil écrit en PHP permettant de gérer des bases de données MySQL. En utilisant WAMP, nous pouvons installer un serveur web complet, qui permettra de faire tous les tests de pages PHP en toute facilité.

Le développement d'une application web dynamique demande l'inclusion de script et de code pour garantir le dynamisme des pages et la liaison avec la base de données. Pour le développement de notre site, nous avons utilisé HTM (Hypertexte Markup Language) comme langage principal, car c'est le langage de programmation des pages web « de base ». Et pour cause: c'est ce langage qui va structurer les données de pages, indiquer comment elles sont faites et quels en sont les éléments.

C'est lui qui indiquera quels sont les titres, comment le texte est divisé en paragraphe, etc. Etant limité, nous y avons ajouté les autres langages pour apporter le dynamisme à notre site. C'est entre autre:

- Le JavaScript: (à ne pas confondre avec Java) est un langage très largement employé sur Internet côté client, même s'il peut aussi fonctionner côté serveur. Il a été mis au point par Netscape Communications. Ses instructions sont incluses dans le code HTML des pages envoyées sur le poste client et elles sont traitées directement par le navigateur
- Le PHP (Hypertext Preprocessor): Initialement appelé Personal Home Page, il a été développé à l'origine par Rasmus Lerdorf en 1994 pour enregistrer le nombre de visiteurs sur son site. Il a vite été perfectionné par la communauté Internet pour devenir un langage de script côté serveur, à la fois très simple et très performant. Il s'interface parfaitement avec des bases de données MySQL, mais il peut aussi exploiter d'autres bases de données (Informix, dBase, Oracle, SyBase, PostgreSQL...)

6.4 CHOIX DU SYSTEME DE GESTION DES BASES DE DONNEES

Pour la réalisation de cette application web, nous avons opté pour le SGBD MySQL comme gestionnaire de base de données.

MySQL: C'est un produit gratuit et Open Source constitué d'un serveur SQL qui supporte différents systèmes de stockage, plusieurs logiciels clients et bibliothèques, outils d'administration, ainsi que de nombreuses interfaces de programmation. MySQL est l'un des Systèmes de Gestion de base de données open source les plus populaires au monde, il est distribué sous une double licence GPL et propriétaire. MySQL peut vous aider à concevoir des applications de base de données évolutives et hautement performantes.

6.5 CHOIX DU SERVEUR WEB (APACHE)

Le serveur HTTP Apache est un serveur HTTP open-source pour les systèmes d'exploitation modernes. Le but de ce projet est de fournir un serveur sécurisé, efficace et extensible qui fournit des services HTTP respectant les standards HTTP actuels. Nous l'avons utilisé pour nous aider à configurer les Virtual hosts afin de nous permettre de faire des tests sur le déploiement en réseau local.

7 CONCLUSION

Pour bien assurer la gestion de mutation des agents, il vaut mieux avoir des bons outils matériels ou humains pouvant aider à bien l'assurer pour la bonne évolution des activités y afférentes.

Dans ce travail, il a été question de développer une application web de gestion de mutation des agents au sein de SECOPE, qui est une solution efficace qui résout les problèmes de gestion évoqués ci - haut. Il est mieux de signaler qu'il existe un très grand nombre des logiciels pouvant assurer cette gestion ou ce suivi, mais il vaut mieux adapter l'outil aux spécifications de la société. C'est ainsi que nous avons été amenés à développer le logiciel selon les fonctionnalités que les utilisateurs ont formulées.

Signalons par ailleurs que ce projet a été conçu selon les principes des méthodes agiles en utilisant la démarche **2TUP** qui donne des bonnes pratiques de développement logiciel appuyée par le langage de modélisation informatique UML (de l'anglais Unified Modeling Language).

REFERENCES

- [1] John S. Mayo, Matériaux de l'information et communication, Science Américaine, 1986.
- [2] Madeleine GRAWITZ, Méthodes des sciences sociales, Paris Cedex, 2006.
- [3] John Wiley et Sons, Inc, le management-des-processus-métier-BPMN-pour les nuls, Edition Limitée, Etats-Unis.
- [4] LAURENT AUDIBERT, de l'apprentissage à la pratique, cours et exercice, Edition, Eyrolles.
- [5] P. Roques et F. Vallée, UML 2 en action de l'analyse des besoins à la conception, 4^e édition, Eyrolles, février 2007.
- [6] P. Roques, Les cahiers du programmeur UM2 modéliser une application web, Eyrolles 4^e édition 2008.
- [7] Guilbert Olivier, le langage de modélisation objet UML, Université Bordeaux I, 2010.
- [8] Pascal Roques et Franck Vallée, UML2 en action de l'analyse des besoins à la conception, Edition Eyrolles, Paris 2004.
- [9] Xavier Blanc et Isabelle Mounier: UML 2 pour les développeurs, Ed Eyrolles, 2005.
- [10] Di Gallo Frederick: cours génie logiciel, Edition Cnam Bordeaux, 1999-2000.
- [11] Maurice CHAVELLI, Découvrez le framework Laravel, 2^e édition, openclassrooms.com, Avril 2016.