

Université Polytechnique de Bobo-Dioulasso  
(UPB)

-----  
Ecole Supérieure d'Informatique  
(ESI)

-----  
Cycle des Ingénieurs de Conception en  
Informatique  
(CICI)

-----  
01 BP 1091 Bobo-Dioulasso 01  
Burkina Faso  
Tel.: (+226) 20 97 27 64



Délégation Générale à l'Informatique  
(DELGI)

-----  
01 BP 1332 Ouagadougou 01  
Burkina Faso  
Tél. : (+226) 50 32 48 88  
Fax. : (+226) 50 31 83 50

ANNEE ACADEMIQUE 2004-2005

## MEMOIRE DE FIN D'ETUDES

---

### Thème :

- **Inscription en ligne aux concours de la Fonction Publique ;**
- **Mise en ligne des situations, positions administratives et carrière des agents de la Fonction Publique.**

Juillet 2005 – décembre 2005

Présenté par :

**Hamidou GASSIMBE**

Elève ingénieur de conception en informatique

**Maître de stage : Dr Joachim TANKOANO, Délégué Général à l'Informatique**

**Superviseur : M. Loé SANOU, Doctorant à l'Université de Poitiers (France)**

## **REMERCIEMENTS**

Nous remercions :

- la Délégation Générale à l'Informatique, en particulier le Délégué Général à l'Informatique pour nous avoir accepté comme stagiaire au sein de sa structure, pour son soutien à l'Ecole Supérieure d'Informatique dans la formation des ingénieurs informaticiens et surtout pour la pertinence des sous-thèmes proposés ;
- l'Ecole Supérieure d'Informatique pour la qualité de la formation reçue durant ces deux années ;
- le personnel de la Direction des Services Informatiques, de la Direction du Recrutement et celui de la Direction des Ressources Humaines du Ministère de la Fonction Publique et de la Réforme de l'Etat pour toutes les informations qu'ils nous ont fournies durant ce stage ;
- M. KABORE Paul et M. DAVOU Moussa pour leur disponibilité et leurs conseils ;
- nos parents, tuteurs et ami(e)s qui nous ont apporté leurs soutiens morales et matériels durant ces six (6) mois de stage ;

# SOMMAIRE

<b>ABREVIATIONS.....</b>	<b>3</b>
<b>CONTEXTE .....</b>	<b>4</b>
<b>PARTIE 1 : GENERALITES .....</b>	<b>5</b>
I. STRUCTURES D'ACCUEIL	6
1. Délégation Générale à l'Informatique (DELGI).....	6
2. Ministère de la Fonction Publique et de la Réforme de l'Etat (MFPRE) .....	7
II. PROBLEMATIQUE	11
1. Problématique et objectifs .....	11
2. Contraintes à respecter .....	12
III. Méthodologies et concepts	13
1. UML (Unified Modeling Language).....	13
2. Processus de développement .....	16
3. XML (eXtensible Markup Language).....	18
4. Java.....	20
5. OpenCms .....	20
6. Modèle de développement en couche MVC.....	21
7. Architecture 3-tiers .....	22
<b>PARTIE 2 : INSCRIPTION EN LIGNE AUX CONCOURS DE LA FONCTION PUBLIQUE .....</b>	<b>24</b>
I. Découverte des besoins	26
1. Diagramme de cas d'utilisation.....	27
2. Planning .....	36
II. Analyse	37
1. Diagramme d'activité.....	37
2. Diagramme de classe.....	42
3. Modèle relationnel .....	43
4. Structure des fichiers XML .....	43
III. Conception	49
1. Architecture du système.....	49
2. Diagramme de classe d'implémentation.....	54
3. Diagramme de séquence .....	55
4. Diagramme d'enchaînement des pages Web.....	58
5. Diagramme de paquetage .....	59
6. Diagramme de composants.....	60
IV. Implémentation	61
1. Diagramme de déploiement .....	61

2. Procédure d'installation de l'application .....	62
<b>PARTIE 3 : MISE EN LIGNE DES SITUATIONS, POSITIONS ADMINISTRATIVES ET CARRIERE DES AGENTS DE LA FONCTION PUBLIQUE .....</b>	<b>64</b>
I. Découverte des besoins .....	66
1. Diagramme de cas d'utilisation.....	71
2. Planning .....	76
II. Analyse .....	77
1. Diagramme d'activité .....	77
2. Diagramme de classe.....	78
3. Modèle physique de données.....	79
4. Structure des fichiers XML .....	80
III. Conception .....	81
1. Architecture du système .....	81
2. Diagramme de classe d'implémentation.....	82
3. Diagramme de séquence .....	82
4. Diagramme d'enchaînement des pages Web.....	90
5. Diagramme de paquetage .....	91
6. Diagramme de composants.....	92
IV. Implémentation .....	93
1. Diagramme de déploiement .....	93
2. Procédure d'installation de l'application .....	94
<b>CONCLUSION.....</b>	<b>96</b>
<b>ANNEXES .....</b>	<b>97</b>
<b>ANNEXE-1 : QUELQUES ECRANS DES INSCRIPTIONS EN LIGNE.....</b>	<b>97</b>
<b>ANNEXE-2 : QUELQUES ECRANS DE CONSULTATION DES SITUATIONS ADMINISTRATIVES DES AGENTS DE LA FONCTION PUBLIQUE.....</b>	<b>108</b>
<b>GLOSSAIRE .....</b>	<b>115</b>
<b>BIBLIOGRAPHIE .....</b>	<b>117</b>

## ABREVIATIONS

<b>CENATRIN</b>	Centre National de Traitement de l'Information
<b>CSI</b>	Conseil Supérieur de l'Information
<b>DAAF</b>	Directeur des Affaires Administratives et Financières
<b>DELGI</b>	Délégation/Délégué Général à l'Informatique
<b>ESI</b>	Ecole Supérieure d'Informatique
<b>MFB</b>	Ministère des Finances et du Budget
<b>MFPRE</b>	Ministère de la Fonction Publique et de la Réforme de l'Etat
<b>RESINA</b>	Réseau Interurbain de l'Administration publique du Burkina Faso
<b>SIGASPE</b>	Système Intégré de Gestion Administrative et Salariale du Personnel de l'Etat
<b>SIGC</b>	Système Intégré de Gestion des Concours

## CONTEXTE

Le gouvernement du Burkina Faso a adopté le 29 janvier 1997 le deuxième plan directeur informatique national. L'objectif était de profiter des opportunités qu'offrent les Technologies de l'Information et de la Communication (TIC) pour contribuer au développement du pays, par la modernisation de l'administration publique, par la mise en place de solutions informatiques qui facilitent le travail et apportent plus de transparence au sein des institutions publiques.

La Délégation Générale à l'Informatique est l'institution publique chargée de la supervision et de la mise en œuvre de ces projets de modernisation des institutions publiques.

Dans ce cadre et dans le souci de contribuer à la formation des futurs cadres informaticiens, elle a accepté nous accueillir comme stagiaire afin de réfléchir sur les sous-thèmes suivants :

- inscription en ligne (via Internet) aux concours de la Fonction Publique ;
- consultation des carrières et des situations administratives des agents de la Fonction Publique (via Internet) ;
- proposition d'un modèle de mémorisation et de présentation uniforme des démarches administratives pour l'obtention d'un droit ou la satisfaction d'un devoir par le citoyen burkinabé ;
- proposition d'un système d'organisation de réunions et ateliers virtuels entre institutions publiques.

Ce stage obligatoire de six (6) mois, inscrit dans la formation des ingénieurs de conception de l'Ecole Supérieure d'Informatique a pour objectif de préparer les futurs ingénieurs aux réalités des entreprises.

Le présent rapport résume le travail effectué durant ces six (6) mois de stage. En premier lieu, nous présentons la méthodologie et les concepts utilisés pour la réalisation du projet. Ensuite, nous faisons un bilan des deux sous-thèmes traités.

**PARITE 1 :**  
**GENERALITES**

## **I. STRUCTURES D'ACCUEIL**

### **1. Délégation Générale à l'Informatique (DELGI)**

La DELGI a été créée en 1990 et faite institution publique sous l'autorité du Premier Ministre par les décrets N°97-159 PRES/PM du 17 avril 1997 et N°97-160 PRES/PM du 17 avril 1997. Elle a pour objectif la mise en œuvre de projets et schémas d'informatisation nationale.

#### **a. Missions de la DELGI**

Les technologies de l'information et de la communication peuvent incontestablement jouer un rôle déterminant dans :

- la réduction des effets négatifs de l'enclavement et de l'inexistence des infrastructures de base (écoles, structures sanitaires, routes, etc.) ;
- l'amélioration de l'existence de tout individu qui dépend en grande partie de la qualité et de la pertinence de l'information à laquelle il peut accéder ;
- le renforcement de la bonne gouvernance ;
- la création de nouvelles opportunités d'affaires et de services ;
- l'amélioration de la compétitivité des secteurs productifs et de l'économie dans le contexte actuel de la mondialisation.

La création de la Délégation générale à l'informatique traduit la volonté du Gouvernement du Burkina Faso de saisir ces opportunités pour le développement économique et social du pays.

La Délégation Générale à l'Informatique a pour mission :

- la planification, la réglementation et le contrôle de l'informatique ;
- la planification, le suivi de la formation et de la recherche en informatique ;
- la délivrance des agréments pour la fourniture d'équipements et de services informatiques ;
- l'assistance et le contrôle technique des services informatiques, des départements ministériels et institutions d'Etat, des centres de traitement informatique, des centres de formation en informatique et de toutes autres structures publiques dont l'activité entre dans le cadre de ses attributions ;
- la supervision et la validation des schémas directeurs informatiques des ministères, institutions et établissements publics ;
- la promotion et la vulgarisation de l'outil informatique ;

- de donner son avis sur tout marché informatique de l'Etat et de ses démembrements.

#### b. Organisation de la DELGI

Les politiques que met en oeuvre la Délégation Générale à l'Informatique sont proposées par le Conseil Supérieur à la Communication (CSC) et approuvées par le Conseil des Ministres.

La Délégation Générale à l'Informatique est dirigée par un Délégué Général à l'Informatique et comprend quatre départements :

- le département des études et de l'informatisation auquel je suis rattaché ;
- le département de la formation et de la promotion de l'informatique ;
- le département de l'administration générale et financière ;
- le département des infrastructures de communication informatique et de la maintenance ;

Elle avait sous sa tutelle le Centre National de Traitement de l'Information (CENATRIN) et entretient des rapports privilégiés avec l'Ecole Supérieure d'Informatique (ESI) de l'Université Polytechnique de Bobo-Dioulasso.

#### c. Environnement de travail de la DELGI

La DELGI dispose de onze (11) serveurs ; d'une soixantaine d'ordinateurs (IBM et HP) ; d'un routeur, d'un modem ; d'un pont radio ; de deux switchs ; de quatre (4) imprimantes ; des systèmes d'exploitation Windows 98/XP/NT/2000 professionnel et serveur, de la Redhat 8.0, des SGBD SQL Serveur, Oracle, MySQL.

## **2. Ministère de la Fonction Publique et de la Réforme de l'Etat (MFPRE)**

#### a. Historique du ministère

Le département actuel de la Fonction Publique et de la Réforme de l'Etat a comme ancêtre le Ministère créé en 1957, sous la dénomination de Ministère de la Fonction Publique. Cette nouvelle dénomination du Ministère témoignerait d'une volonté d'approfondissement et d'élargissement de la réforme globale de l'administration publique en cours depuis 1998 et, qui, sur bien d'aspects, avait laissé en marge les institutions républicaines et les établissements publics nationaux.

## b. Attributions

Aux termes du décret n° 2002-255/PRES/PM du 18 juillet 2002 portant attributions des membres du gouvernement, le Ministère de la Fonction Publique et de la Réforme de l'Etat assure la mise en œuvre et le suivi de la politique du Gouvernement en matière de Fonction Publique et de réforme de l'Etat.

A ce titre, et en relation avec tous les ministères, il est chargé :

### i. en matière de Fonction Publique

- du recrutement des agents de la Fonction Publique occupant les emplois interministériels des administrations de l'Etat ;
- de la formation des agents de la Fonction Publique ;
- de la réglementation relative à la gestion des agents de la Fonction Publique ;
- de la coordination des activités de toutes les structures centrales et déconcentrées de gestion des agents de la Fonction Publique ;
- du redéploiement des agents publics dans le cadre de la mise en œuvre des réformes institutionnelles ;
- de la protection et de la sécurité sociale des agents de la Fonction Publique ;

### ii. en matière de réforme de l'Etat

- de l'impulsion des actions et mesures de renforcement des capacités des administrations de l'Etat ;
- de la coordination, en relation avec tous les ministères concernés, des réformes institutionnelles entreprises par l'Etat au sein des administrations centrales, des administrations déconcentrées, des entreprises publiques et des collectivités locales ;
- de la promotion de l'évaluation externe des politiques publiques ;
- du suivi et de la mise en œuvre du plan national de bonne gouvernance ;
- de l'appui conseil aux départements ministériels et aux institutions publiques pour l'élaboration des instruments de programmation, d'évaluation et de suivi des politiques sectorielles ;
- de la définition des actions et mesures de déconcentration de la gestion des agents de la Fonction Publique ;
- de la réforme du système de gestion des structures de l'administration de l'Etat ;

- de la définition des normes de déconcentration des services de l'Etat dans le cadre du processus de décentralisation ;
- de la conduite des initiatives en matière de développement de la productivité des services publics ;
- de la définition et de l'actualisation des finalités de la réforme de l'Etat.

c. Organisation du Ministère

i. Cabinet du Ministre

L'organisation du Cabinet du Ministre est définie par le décret N° 2002 - 363/PRES/PM/MFPRE portant organisation du Ministère de la Fonction Publique et de la Réforme de L'Etat. Selon ce décret, le cabinet du Ministre se compose de Conseillers Techniques ; d'un Inspection Technique des Services ; d'un Secrétariat Particulier et d'un Protocole du Ministre.

ii. Secrétariat Général

Pour la mise en œuvre de la politique du gouvernement, le Ministre dispose d'un Secrétariat Général dont la composition et les attributions sont régies par les dispositions du décret N° 2002 - 363/PRES/PM/MFPRE portant organisation du Ministère de la Fonction Publique et de la Réforme de L'Etat. Le Secrétaire Général assiste le Ministre dans l'application de la politique du ministère. Il est chargé de la coordination administrative et technique des structures de l'administration centrale, des structures rattachées et de mission.

iii. Structures centrales

Les structures de l'administration centrale du Ministère de la Fonction Publique et de la Réforme de l'Etat sont constituées de la :

- Direction Générale de la Fonction Publique comprenant la :
  - Direction du Recrutement ;
  - Direction du Personnel de l'Etat ;
  - Direction des Sorties Assistées ;
  - Direction des Etudes et du Contentieux ;
- Direction Générale de la Réforme de l'Etat comprenant la :
  - Direction de la Réforme Globale ;
  - Direction des Réformes Sectorielles ;
- Direction de l'Administration et des Finances ;

- Direction des Etudes et de la Planification ;
- Direction des Ressources Humaines ;
- Direction des Services Informatiques ;
- Direction de la Communication et de la Presse Ministérielle.

## II. PROBLEMATIQUE

---

### 1. Problématique et objectifs

L'une des attributions du Ministère de la Fonction Publique et de la Réforme de l'Etat est le recrutement des agents de la Fonction Publique. A cet effet, il organise les concours d'entrée à la Fonction Publique. La participation à ces concours exige le dépôt d'un dossier de candidature. Les candidats doivent accomplir ces formalités avant la date fixée pour la clôture de la réception des dossiers et généralement à travers des files d'attentes de plus en plus longues.

Cette affluence devant les guichets de réceptions des dossiers ouverts à cet effet, a amené le Ministère de la Fonction Publique et de la Réforme de l'Etat à s'orienter vers une solution informatique afin de multiplier les canaux par lesquels les candidats peuvent remplir les formalités d'inscription et de leur éviter ainsi de longs déplacements et les bousculades devant les centres de réception des dossiers.

*Cette solution informatique doit permettre d'une part l'inscription en ligne (via Internet) aux concours de la Fonction Publique et d'autre part la possibilité de consulter en ligne les résultats de ces concours.*

Le Ministère de la Fonction Publique et de la Réforme de l'Etat et celui des Finances et du Budget ont choisi l'option de rationaliser et de moderniser la gestion du personnel de l'Etat. A cet effet, un Système Intégré de Gestion Administrative et Salariale du Personnel de l'Etat (SIGASPE) a été mis en place. L'un des nombreux avantages de ce système est qu'il offre la possibilité aux usagers de connaître leur situation administrative et l'historique de leurs dossiers en cours de traitement à travers des procédures de consultation prévues à cet effet. Cependant, cette consultation n'est possible qu'à partir des postes de travail connectés au système SIGASPE, ce qui oblige les agents désirant s'informer sur leur situation administrative ou sur l'évolution de leur dossier à se rendre à la Direction des Ressources Humaines de leur ministère afin d'avoir l'information. La future solution informatique a été retenue par le Ministère de la Fonction Publique et de la Réforme de l'Etat dans le but de réduire ces déplacements, principalement pour les agents qui travaillent hors de la capitale.

*Elle doit offrir la possibilité aux fonctionnaires de consulter en ligne leur situation administrative, leur carrière et l'historique de leurs dossiers en cours de traitement.*

La tenue d'une réunion ou d'un atelier nécessite le déplacement des participants, qui ont souvent des contraintes d'agenda ou d'emploi du temps. Pour atténuer ces contraintes, il a été retenu d'ajouter aux sites Web ministériels des fonctionnalités pouvant permettre l'organisation et la tenue de réunions et ateliers. A terme, ces fonctionnalités devraient permettre aux participants de se parler et de se voir tout en restant dans leur bureau.

*L'objectif est donc de permettre l'organisation et la tenue de réunions et ateliers virtuels entre les départements ministériels.*

Pour accéder à un droit, accomplir un devoir, remplir une formalité ou établir une pièce administrative, le citoyen burkinabé entreprend des démarches administratives auprès des ministères et institutions de la République. Un effort a été fait sur les sites Web des ministères et institutions de l'Etat pour décrire et expliquer ces démarches qui ne sont pas toujours bien connus par tous les citoyens. Toutefois, ces descriptions et explications ne respectent pas le même format de présentation et sont parfois incomplètes ou difficilement accessibles par le commun des mortels. Le but visé est de réaliser des fonctionnalités qui seront intégrées dans ces sites Web afin de corriger ces insuffisances. Ces fonctionnalités doivent permettre à terme de mettre en place un site portail des services publics.

*Le but visé est de permettre aux citoyens dans le cadre des démarches administratives, d'éviter les déplacements préliminaires visant à s'informer sur les services offerts par l'administration, sur leurs droits et devoirs ou sur les procédures réglementaires à l'obtention des documents désirés.*

## **2. Contraintes à respecter**

Le travail consiste à développer des fonctions qui satisfont aux sous-thèmes. Ces fonctions seront intégrées aux sites Web ministériels. Ces sites Web tournent actuellement sous le gestionnaire de contenu OpenCms (Content Management System) qui s'appuie sur les technologies J2EE (JSP, Servlets, JavaBean) et XML. Par conséquent, les fonctions à réaliser se feront avec les langages Java, XML et HTML, JavaScript.

Un soin particulier doit être apporté à la documentation (production de la javadoc) afin de faciliter la maintenance du système logiciel.

C'est également le cas pour la sécurité car le système est ouvert à Internet.

Le langage ou la méthode de modélisation n'est pas imposé. Cependant Java étant un langage orienté objet, nous utiliserons l'outil de modélisation orienté objet UML (Unified Modeling Language).

Enfin, dans le but de faciliter la maintenance des fonctions, nous devons choisir un modèle d'architecture logiciel clair, connu par un grand nombre de développeurs et applicable au développement Web. Le modèle de développement en couche « MVC » et la framework « Struts » ont été retenus. Ces concepts seront détaillés dans la suite du rapport.

### III. METHODOLOGIES ET CONCEPTS

---

#### 1. UML (Unified Modeling Language)

Java un langage purement objet proposé par la société américaine SUN Microsystems. Pour une adéquation avec cette caractéristique objet du langage java, il convient de choisir un langage ou méthode de modélisation de système orienté objet. D'où le choix d'UML. Certes, il existe d'autres méthodes objets de modélisation comme OMT, O\* mais UML est le plus utilisé pour la construction de système orienté objet. De plus, il existe autour de ce langage beaucoup d'outil de modélisation visuel (Rational Rose, ModelMaker, Power AMC ...) qui facilite la modélisation.

Par définition, UML est un langage pour spécifier, visualiser, construire et documenter les artefacts d'un système logiciel.

UML offre des éléments pour décrire les différents aspects d'un système :

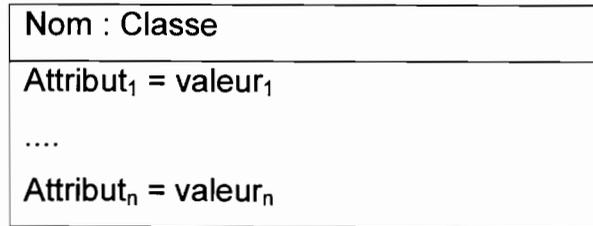
- les diagrammes relatifs à la structure statique
  - le diagramme de classe : structure les données du système définies comme un ensemble de relations entre classes ;

Notation d'une classe

Nom+propriété de la classe
Attributs de la classe
Méthodes de la classe

- le diagramme d'objet : illustre des objets et leur relations ;

Notation d'un objet



- le diagramme de cas d'utilisation

- intérêt des cas d'utilisation

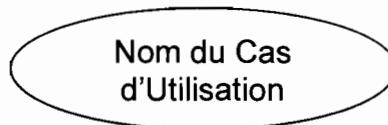
Les cas d'utilisation présentent les intérêts suivants :

- permettre une meilleure structuration des besoins des utilisateurs ;
- oblige ces derniers à définir clairement la manière dont ils voudraient interagir avec le système, les informations qu'ils entendent échanger et ce qui doit être fait pour obtenir les résultats escomptés ;

- définition du concept de cas d'utilisation

Un cas d'utilisation est une unité cohérente de fonctionnalités offert par un système, un sous-système ou une classe. Cette fonctionnalité se traduit d'une part par une séquence de messages échangés entre ce système et un ou plusieurs **agents externes (acteurs)** et d'autre part par un ensemble d'actions échangées par le système.

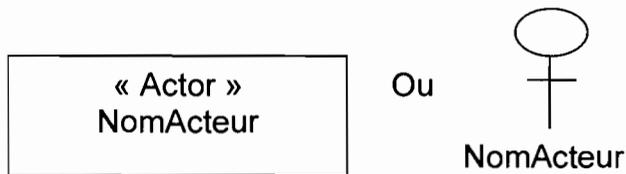
Un cas d'utilisation est symbolisé par une ellipse qui contient son nom.



- acteurs

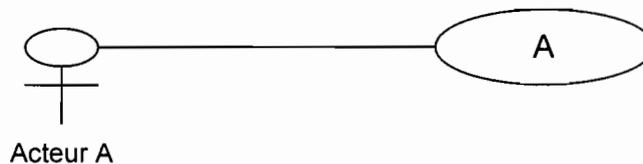
Un acteur est défini par un ensemble cohérent de rôles qu'un utilisateur ou une entité peut jouer en communiquant avec le système. Un acteur peut consulter et/ou modifier l'état du système en émettant ou en recevant des messages susceptibles d'être porteur de données.

Il existe deux notations possibles de l'acteur :

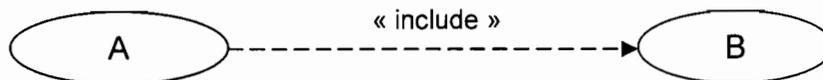


o relation des cas d'utilisation

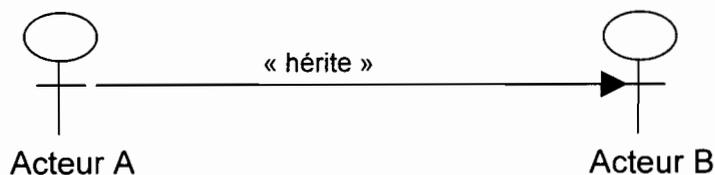
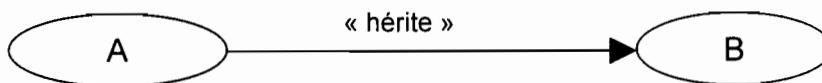
- **association** : représentation de la participation d'un acteur à un cas d'utilisation, des instances de l'acteur communique avec le cas d'utilisation ;



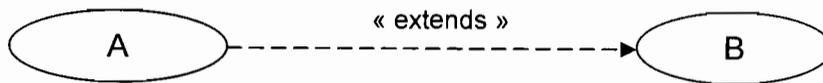
- **include** : permet d'indiquer qu'une instance d'un cas d'utilisation A contient également le comportement spécifié par le cas d'utilisation B ;



- **généralisation** : signifie qu'un acteur ou un cas d'utilisation est une spécialisation d'un autre ;



- *extends* : permet d'indiquer qu'une instance d'un cas d'utilisation B peut être augmentée (dans certaines conditions) par le comportement défini par un cas d'utilisation A ;



- le diagramme de séquence : représentation des interactions temporelles entre objets dans la réalisation d'une interface Homme - Système ou d'une opération ;
- le diagramme d'activité : structuration d'une opération en actions ;
- le diagramme d'états transitions : représentation du comportement des objets d'une classe en terme d'états et de transitions d'états ;
- le diagramme de collaboration : représentation des interactions entre objets ;
- le diagramme de composant : représentation de l'architecture des composants physiques d'une application ;
- le diagramme de déploiement : description du déploiement des composants sur le dispositif matériel.

## 2. Processus de développement

Le langage UML est intentionnellement indépendant de tout processus de développement. Le concepteur ou l'organisation peut choisir le processus de développement qui lui convient. Toutefois, les auteurs d'UML conseillent l'utilisation d'un processus dirigé par les cas d'utilisation, centré architecture, itératif et incrémental.

Le processus unifié UP (Unified Process) répond à cette recommandation des auteurs d'UML. Il est à ce jour le processus le plus utilisé. C'est un patron de processus pouvant être adapté à une large classe de systèmes logiciels, à différents domaines d'application, à différents types d'entreprise, à différents niveaux de compétences et à différentes tailles de projet.

Nous nous sommes inspirés de ce processus pour proposer une démarche pour notre projet. Nous suivons les étapes suivantes :

- découverte des besoins

Elle sera pilotée par les cas d'utilisation d'UML. Les diagrammes suivant seront réalisés :

- diagramme des cas d'utilisation ;
- diagramme de séquence (dans certains cas) ;
- nous finirons par l'élaboration d'un planning prévisionnel de réalisation.

- analyse

Diagramme UML à réaliser :

- diagramme d'état transition ou diagramme d'activité (si nécessaire) ;
- diagrammes relatifs à la structure statique (classes, objets).

Par ailleurs nous définirons :

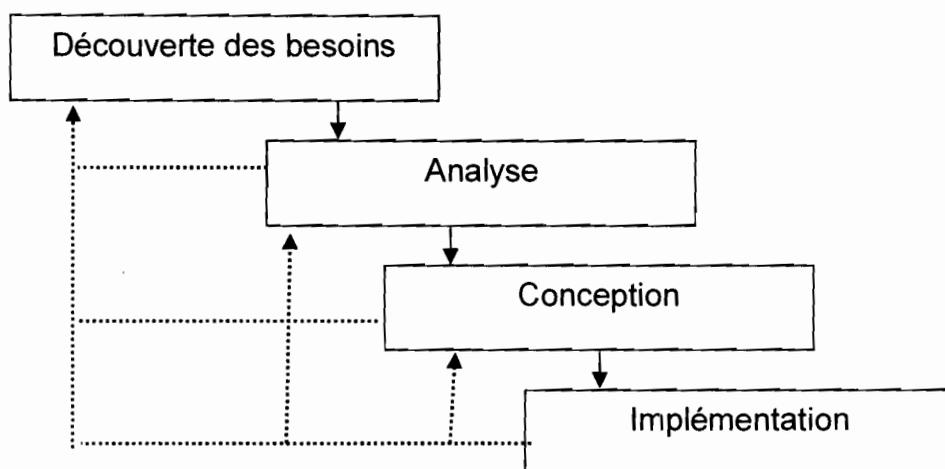
- le modèle relationnel, en transformant le modèle objet (diagramme de classe) en modèle relationnel grâce au concept de Mapping objet-relationnel ;
- les structures des fichiers XML (si nécessaire). En effet, nous allons dans certains cas devoir échanger des données avec d'autres systèmes externes grâce à XML. Il nous faudra donc définir les structures de ces fichiers.

- conception

- définition de l'architecture de l'application ;
- diagramme de classe d'implémentation : il s'agira de transformer le diagramme de classe en un diagramme plus adapté à la programmation en tenant compte des possibilités et des limites du langage de programmation. Par exemple, Java ne supportant pas l'héritage multiple, il faudra transformer le diagramme de classe de sorte à contourner cette contrainte. On peut également ajouter des classes qui faciliteront le développement.
- diagramme de séquence ou de collaboration : avec une architecture définie, un diagramme de classe plus proche de l'implémentation, nous pouvons décrire les interactions entre les composants du système dans la réalisation des opérations ;
- diagramme d'enchaînement des pages Web ;
- diagramme de paquetage ;
- diagramme de composants.

- implémentation
  - codage du logiciel ;
  - production de la javadoc ;
  - test unitaire et intégration des différentes couches ;
  - diagramme de déploiement ;
  - procédure d'installation.

Ce processus sera utilisé pour la réalisation des quatre sous-thèmes. Cependant, seuls les diagrammes indispensables à la construction du système seront réalisés.



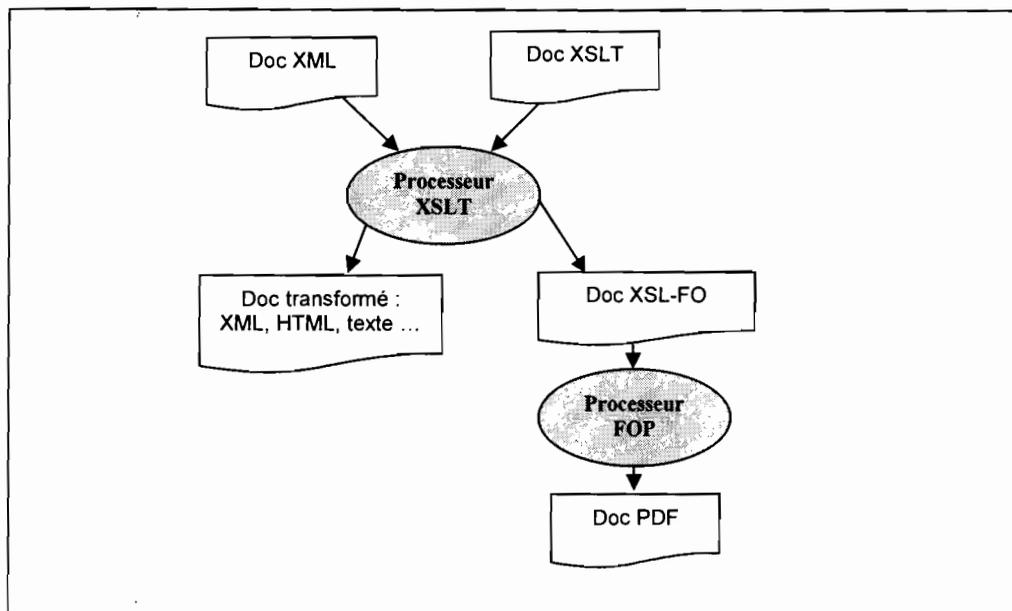
### 3. XML (eXtensible Markup Language)

*XML (eXtensible Markup Language)* est un standard universel proposé par la *W3C (World Wide Web Consortium)* pour la représentation des données. Un document XML contient à la fois des données et les indications sur le rôle que jouent ces données.

Nous l'utiliserons pour la réalisation des fichiers de configuration, pour l'échange de données avec d'autres applications et pour la publication des données sur le Web. Cette dernière possibilité n'est envisageable que si nous couplons XML au langage XSLT/XSL-FO.

*XSLT (Extensible Stylesheet Language Transformations)* est une partie du langage *XSL (Extensible Stylesheet Language)*. Il permet de transformer un document XML en un autre document XML, HTML, texte... Il nous permettra de publier directement des fichiers XML (pour les navigateurs supportant XML) et pour la transformation des fichiers XML en HTML (pour les navigateurs ne supportant pas XML).

*XSL-FO (XSL Formatting Objects)* est également une partie de XSL. Il permet d'exprimer de manière très précise le rendu d'un document. Nous l'utiliserons pour générer des documents PDF (format idéal pour éditer des états sur le Web) à partir de fichiers XML.



Processus de transformation d'un fichier XML en XML, HTML, texte, PDF... avec les processeurs XSLT et FOP (Formatting Objects Processor)

La manipulation d'un document XML requiert un parseur. A ce jour il existe deux types de parseur : DOM et SAX.

*SAX (Simple API for XML)* est une spécification JAVA. Il permet de manipuler des documents XML sans avoir besoin de construire entièrement la structure du document en mémoire. Son avantage est le gain en mémoire par rapport à DOM. Il est léger et rapide. Cependant, il ne permet d'accéder au document qu'en mode consultation.

*DOM (Document Object Model)* est une initiative de la W3C. Il représente un document XML sous forme d'arbre chargé en mémoire et fournit les interfaces permettant de le manipuler. Il a l'avantage d'être indépendant du langage de programmation et permet de modifier la structure d'un document chargé en mémoire. Cependant il est lent et gourmand en mémoire.

*JDOM est un JSR (Java Specification Request) pour la manipulation de document XML. Il est léger, rapide, facile d'utilisation et interopère avec DOM et SAX. Nous choisissons cette API pour manipuler nos documents XML car elle permet de travailler avec DOM et SAX (on profite ainsi des avantages de ces deux).*

#### **4. Java**

C'est le langage avec lequel nous allons travailler. Java est un langage purement objet développé par la société américaine Sun Microsystems. Il fonctionne sur le principe de machine virtuelle, ce qui rend les programmes java portables. Il nous servira à écrire nos classes métier, des Servlets (programmes java qui s'exécutent coté serveur) et à introduire du code dynamique dans nos pages Web.

#### **5. OpenCms**

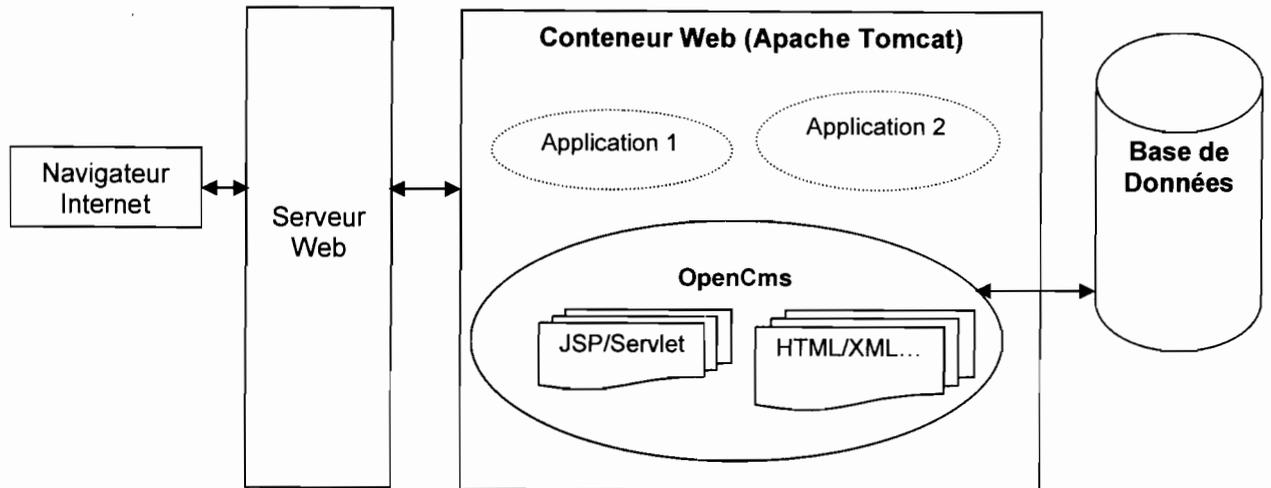
La DELGI, principal hébergeur des sites Web des départements ministériels du gouvernement du Burkina Faso, a orienté son choix sur le gestionnaire de contenu *OpenCmS* dans le but de faciliter la mise à jour de ces sites Web.

*OpenCmS* est l'acronyme de Open Content Management System. Il est basé sur les technologies J2EE/XML et requiert un conteneur Web (Tomcat ou autre), un SGBDR et un serveur Web (Apache ou autre) pour tourner. Il fonctionne à la fois sur un serveur Windows et Linux.

- Quelques fonctionnalités d'OpenCmS

- environnement de travail sous navigateur Web ;
- publication de contenu dynamique et statique ;
- workflow de gestion des tâches ;
- gestion intégrée des utilisateurs et des permissions ;
- mécanisme de modèle (template) en XML ou en JSP ;
- éditeur WYSIWYG, éditeur style MS Word pour la mise à jour des contenus ;
- utilisation des standards java et XML.

- Architectures d'OpenCms



## 6. Modèle de développement en couche MVC

En technologie de conception orientée objet, il est conseillé de ne pas confier trop d'actions à un seul objet, mais plutôt de répartir les différentes responsabilités d'actions entre plusieurs objets. MVC (Modèle-Vue-Contrôleur) est un patron de conception pour le développement d'interfaces utilisateurs (interfaces Web ou non). Il permet de séparer les données à afficher (Modèle) de la logique de présentation (Vue) et de la logique de navigation entre les écrans (Contrôleur).

MVC propose trois (3) modules :

- la Vue : c'est l'interface avec laquelle les utilisateurs communiquent avec l'application. Elle ne s'occupe pas de la gestion ou du stockage des données. Dans une application Web java, la Vue est représentée par les JSP (Java Server Page) ;
- le Modèle : c'est l'ensemble des objets qui représentent les données de l'application à afficher. Il est représenté par les JavaBeans dans une application Web java ;
- le Contrôleur : il réalise l'interface entre la « Vue » et le « Modèle ». Il reçoit les requêtes utilisateurs puis détermine quelles parties des objets « Modèle » sont requis et quelles « Vues » sont à utiliser. Dans une application Web java, le rôle du contrôleur est assuré par une Servlet.

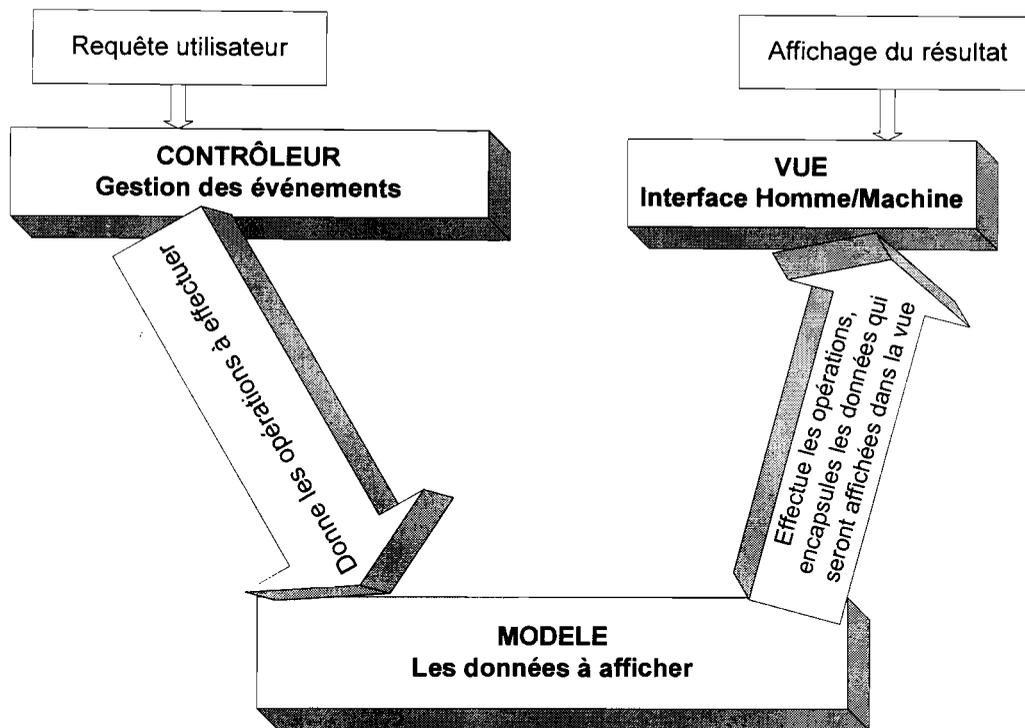
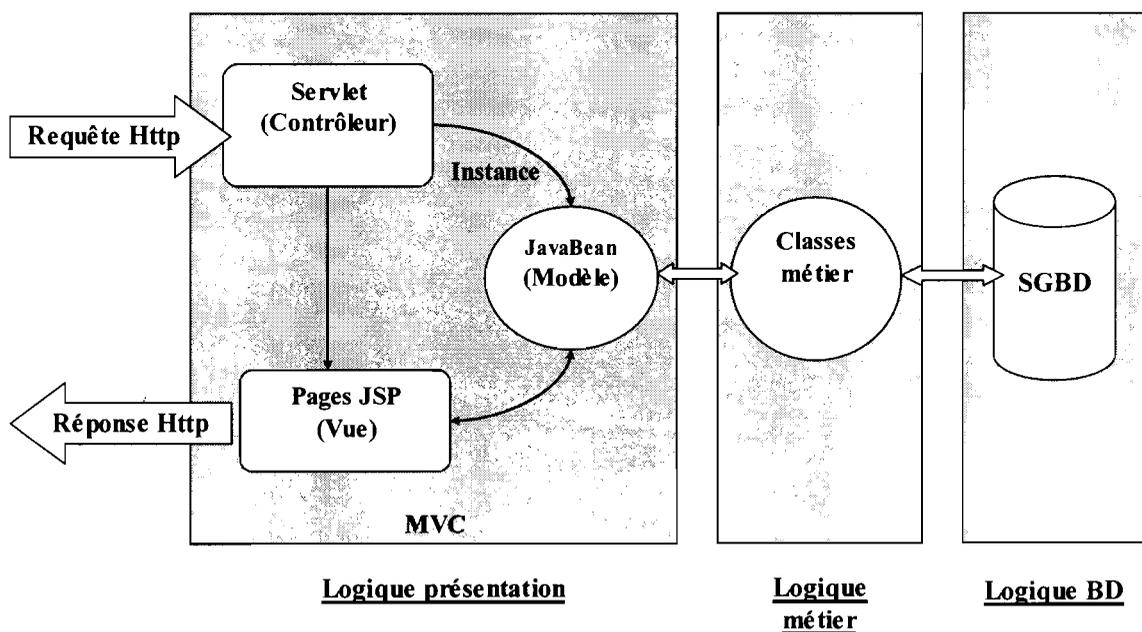


Schéma du Modèle-Vue-Contrôleur

## 7. Architecture 3-tiers

L'architecture 3-tiers est une structure logicielle à trois couches : une couche présentation qui peut être développée en utilisant MVC, une couche logique métier et une couche base de données.



Les couches d'une architecture 3-tiers

Principe de l'architecture 3-tiers (cas d'une application Web java) :

La *logique présentation* est contenue dans un conteneur Web (conteneur JSP/Servlet).

La *logique métier* peut également être dans le conteneur JSP/Servlet ou dans un conteneur JavaBean.

1-le client envoie une requête HTTP à destination d'une Servlet ;

2-la Servlet récupère les données transmises dans la requête HTTP et délègue les traitements avec/sur ces données à des composants JavaBean ou EJB (Entreprise JavaBean) ;

3-selon les traitements à effectuer, les composants EJB et/ou JavaBean peuvent accéder à des sources de données ;

4-une fois les traitements terminés, les composants rendent la main à la Servlet en lui retournant un résultat. La Servlet stocke ce résultat dans un contexte (session, requête) ;

5-la Servlet transmet la suite du traitement de la requête vers une JSP ;

6-la JSP récupère les données stockées par la Servlet dans un des contextes et génère la réponse HTTP.

**PARTIE 2:**  
**Inscription en ligne aux concours de la  
Fonction Publique**

- Le domaine d'étude

Le domaine d'étude est la Direction du Recrutement du Ministère de la Fonction Publique et de la Réforme de l'Etat.

- Les acteurs du projet

- Groupe de pilotage

Le groupe de pilotage prend les décisions relatives aux objectifs recherchés. Il fixe les orientations générales, les délais à respecter et définit les moyens à mettre en place pour la réalisation du projet. Il est constitué de :

M. Joachim TANKOANO : Délégué Général à l'Informatique ;

M. Youssouf OUATTARA : Secrétaire Général du MFPRE ;

M. Isidore OUEDRAOGO : Directeur des Services Informatiques ;

M. Edmond NIANDA : Directeur du recrutement ;

- Groupe de réalisation

Il est chargé de l'exécution du projet : l'étude, la conception et la réalisation de l'application. Il est composé de :

M. Moustapha BANDE : responsable du département des études et de l'informatisation de la DELGI ;

M. Hamidou GASSIMBE : étudiant stagiaire.

- Groupe des utilisateurs

Le groupe des utilisateurs a un rôle consultatif. Il est chargé de fournir toutes les informations nécessaires à la bonne conduite du projet. Il intervient aussi dans la validation des dossiers d'étude produits par le groupe de projet. Il se compose des informaticiens de la Direction des Services Informatiques du Ministère de la Fonction Publique, du personnel de la Direction du Recrutement et du Directeur de la Communication et de la Presse Ministérielle de la Fonction Publique.

## **I. DECOUVERTE DES BESOINS**

---

Le logiciel à développer est une fonctionnalité à intégrer au site Web du Ministère de la Fonction Publique afin de permettre l'inscription en ligne (via Internet) aux différents concours d'entrée à la Fonction Publique.

A travers nos entretiens avec les personnes ressources du domaine d'étude, il est ressorti les points suivants :

- le logiciel à développer devrait être accessible depuis une interface Web à partir d'un poste disposant d'une connexion Internet ;
- il doit offrir des interfaces permettant aux candidats de choisir le concours auquel ils souhaitent s'inscrire ;
- donner la possibilité aux inscrits, d'imprimer un récépissé d'inscription depuis un navigateur Web;
- permettre aux candidats de suivre leur dossier d'inscription en ligne, à l'aide d'un système d'identification (mot de passe par exemple),
- introduire un volet administration pour permettre l'ouverture/clôture des concours ;
- permettre de transférer les données relatives aux inscrits vers une base de données quelconque ou d'en importer ;

Par ailleurs, il a été demandé de réfléchir sur un procédé d'inscription sécurisée, un procédé qui évitera les inscriptions fictives par la mise en place d'un système d'authentification souple du candidat qui s'inscrit.

Nous avons proposé les deux solutions ci-dessous qui ont été retenues :

- les candidats s'inscriront au moyen d'un numéro de timbre spécial qu'ils devront acquérir auprès des institutions publiques ;
- les candidats pourront également s'inscrire auprès de tiers de confiance : des centres multimédias auxquels le Ministère de la Fonction Publique aura donné son agrément.

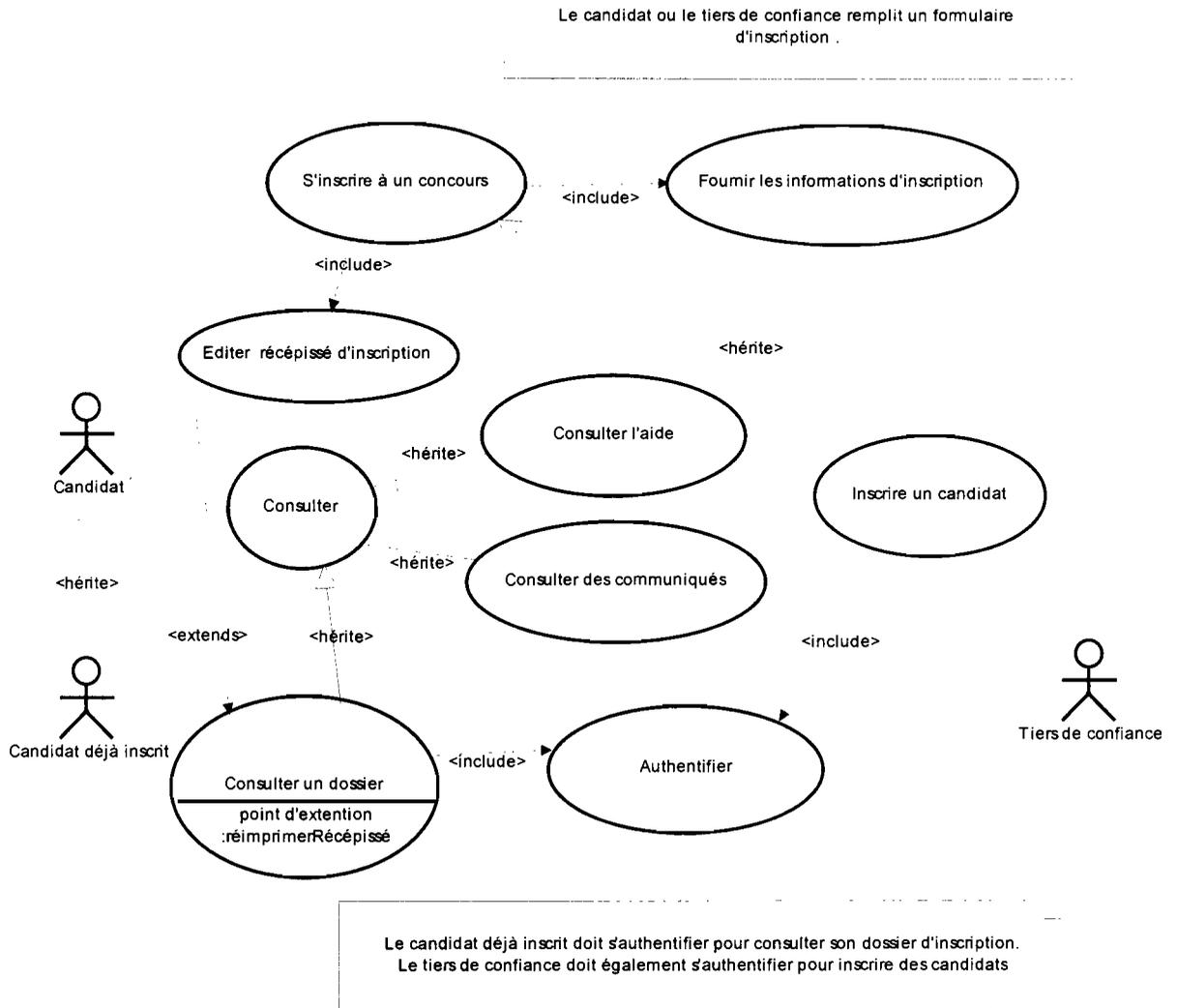
## 1. Diagramme de cas d'utilisation

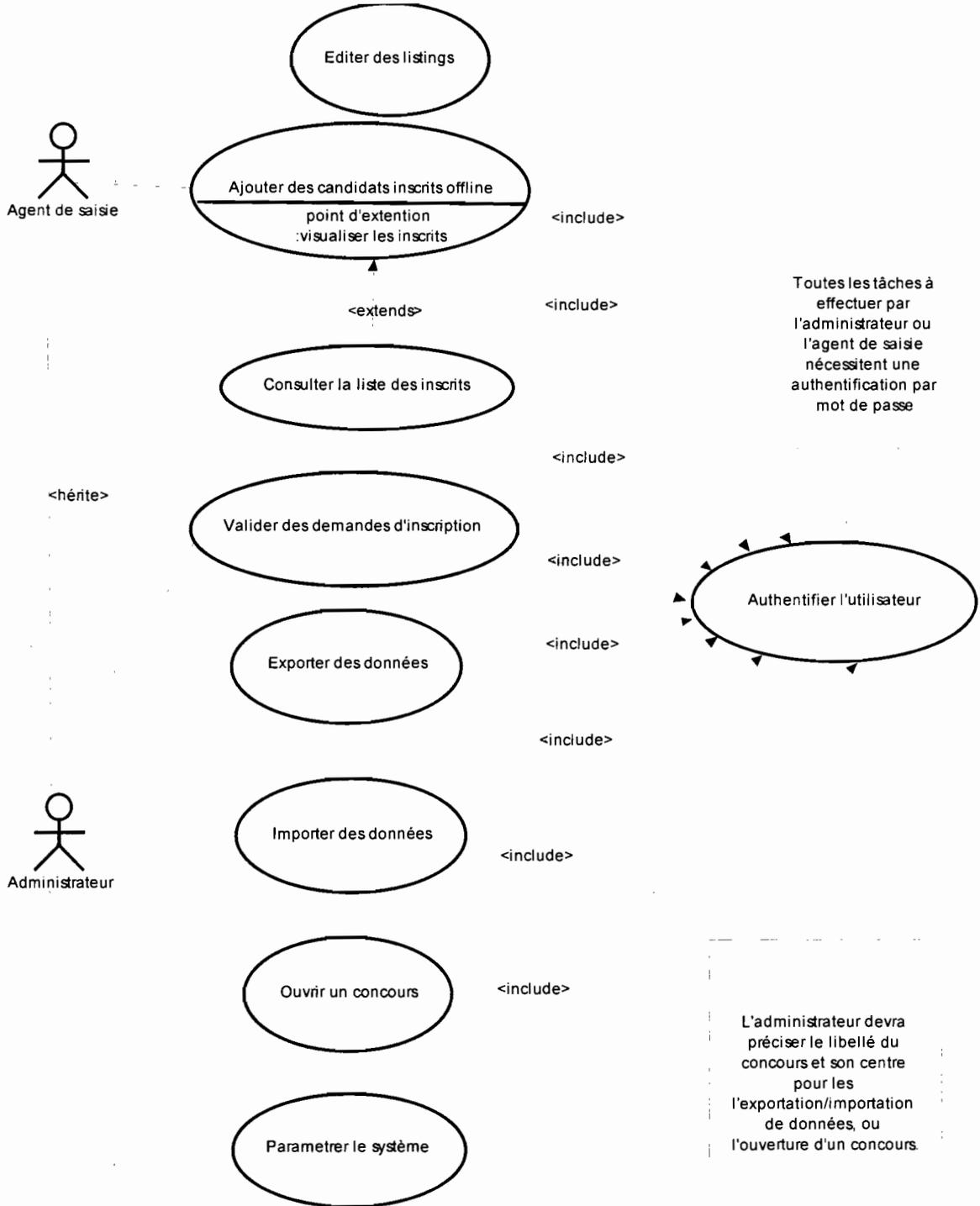
### a. Les acteurs de notre système et leur rôle

- les acteurs principaux (ceux qui exploitent directement les fonctionnalités principales du système) ;
  - Candidat :
    - il peut consulter la liste des concours ouverts ;
    - il peut s'inscrire à un concours ouvert au moyen d'un code d'inscription (figurant sur un timbre spécial à gratter) ;
    - demander des informations sur la procédure d'inscription ;
    - demander des informations sur un concours ;
    - demande à consulter son dossier s'il est déjà inscrit ;
    - demander à réimprimer un récépissé d'inscription s'il est déjà inscrit ;
  - Tiers de confiance :
    - il peut inscrire un candidat sans code d'inscription mais en s'authentifiant ;
    - imprimer un récépissé d'inscription pour un candidat qu'il a inscrit ;
    - consulter les concours ouverts.
- les acteurs secondaires (ceux qui sont chargés de l'administration ou de la maintenance du système) ;
  - l'administrateur du système
    - initialise/paramètre le système ;
    - ouvre/clôture un concours ;
    - valide/refuse les demandes d'inscription ;
    - exporte/importe des données ;

Il peut également effectuer les mêmes tâches que l'agent de saisie.
  - l'agent de saisie (autre que l'administrateur)
    - édite des listings de la répartition des secrétaires de séance et des présidents de jury ;
    - édite les listes des candidats (liste de la répartition des candidats par sous-centre, listing des noms et prénoms des candidats par sous-centre) ;
    - ajout de candidats inscrits offline au système.

b. Le diagramme





Paramétrer le système : c'est définir des concours, leurs types, les ministères concernés par ces dits concours; définir les utilisateurs et leurs droits; définir les paramètres d'accès au serveur Mail pour l'envoi de courrier de confirmation des inscriptions (courrier envoyé lors de validation des demandes)

c. Description textuelle des cas d'utilisation

Le cas d'utilisation est détaillé en un ou plusieurs scénarii. Un scénario est une suite d'échanges entre des acteurs et le système pour décrire un cas d'utilisation dans un contexte particulier. C'est un enchaînement précis et ordonné d'opérations pour réaliser le cas d'utilisation.

La spécification de tous les scénarii est difficile, voire impossible. Nous avons donc sélectionné les plus fréquents (scénario optimal) et les plus pertinents.

Titre	S'inscrire à un concours
Résumé	Ce cas d'utilisation permet à un candidat d'adresser un dossier d'inscription électronique au MFPRE
Acteurs	Candidat, Système Informatique (SI)
Date de création	05 août 2005
Version	V 1.0
Scénario nominal	<p>1-le client demande la liste des concours ouverts  2-le SI affiche la liste (A<sub>1</sub>)  3-le candidat choisit un concours  4-le SI demande au candidat de remplir le formulaire d'inscription  5-le candidat remplit le formulaire et le soumet au système  6-le SI vérifie les informations du formulaire  7-le SI demande au candidat de confirmer la demande d'inscription (A<sub>2</sub>)  8-le candidat confirme la demande d'inscription (A<sub>3</sub>)  9-le SI enregistre les données du formulaire puis édite un récépissé d'inscription pour le candidat  FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : Aucun concours ouvert  Ce scénario commence au point 1 du scénario nominal  2-le SI indique qu'aucun concours n'est ouvert  FIN</p> <p>(A<sub>2</sub>) : données du formulaire non valide  Ce scénario commence au point 6 du scénario nominal  7-le SI informe que des données du formulaire sont erronées  Ce scénario reprend au point 4 du scénario nominal</p> <p>(A<sub>3</sub>) : infirmation du candidat  Ce scénario commence au point 7 du scénario nominal  8-le candidat demande l'annulation de l'inscription  9-le SI annule la procédure d'inscription  FIN</p>

Titre	Inscrire un candidat
Résumé	Ce cas d'utilisation permet à un tiers de confiance de s'authentifier pour inscrire des candidats
Acteurs	Tiers de confiance (TC), Système Informatique (SI)
Date de création	05 août 2005
Version	V 1.0
Scénario nominal	<p>1-le TC demande à s'authentifier  2-le SI système affiche l'interface d'authentification  3-le TC saisit ses références  4-le SI vérifie la validité des références saisies  5-le SI affiche la liste des concours ouverts (A<sub>1</sub>)  6-le TC choisit un concours  7-le SI demande au TC de remplir le formulaire d'inscription  8-le TC remplit le formulaire et le soumet au système  9-le SI vérifie les informations du formulaire  10-le SI demande au TC de confirmer la demande d'inscription (A<sub>2</sub>)  11-le TC confirme la demande d'inscription (A<sub>3</sub>)  12-le SI enregistre les données du formulaire puis édite un récépissé d'inscription  FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : référence non valide  Ce scénario commence au point 4 du scénario nominal  5-le SI indique que les références saisies ne sont pas valides (A<sub>12</sub>)  Ce scénario reprend au point 2 du scénario nominal</p> <p>(A<sub>12</sub>) : Aucun concours ouvert  Ce scénario commence au point 5 du scénario alternatif (A<sub>1</sub>)  5-le SI indique qu'aucun concours n'est ouvert  FIN</p> <p>(A<sub>2</sub>) : données du formulaire non valide  Ce scénario commence au point 9 du scénario nominal  10-le SI informe que des données du formulaire sont erronées  Ce scénario reprend au point 8 du scénario nominal</p> <p>(A<sub>3</sub>) : infirmation du TC  Ce scénario commence au point 10 du scénario nominal  11-le TC demande à annuler l'inscription  12-le SI annule la procédure d'inscription  FIN</p>

Titre	Consulter un dossier
Résumé	Ce cas d'utilisation permet à un candidat déjà inscrit de consulter son dossier
Acteurs	Candidat déjà inscrit, Système Informatique (SI)
Date de création	05 août 2005
Version	V 1.0
Scénario nominal	1-le candidat demande à consulter son dossier 2-le SI demande ses références 3-le candidat saisit ses références 4-le SI vérifie la validité des références saisies 5-le SI affiche le dossier du candidat (A <sub>1</sub> ) FIN
Scénario alternatif	(A <sub>1</sub> ) : aucun dossier trouvé ou références non valides Ce scénario commence au point 4 du scénario nominal 5-le SI indique qu'aucun dossier ne satisfait la demande Ce scénario reprend au point 2 du scénario nominal

Titre	Editer des listings
Résumé	Ce cas d'utilisation permet d'éditer des états : état sur la répartition des candidats par centre, état sur la répartition des secrétaires de séance et président de jury, la liste des candidats par concours et par centre
Acteurs	Agent de saisit, Système Informatique (SI)
Date de création	05 août 2005
Version	V 1.0
Scénario nominal	1-l'agent de saisie demande à s'authentifier 2-le SI demande ses références 3-l'agent saisit ses références 4-le SI vérifie la validité des références saisies 5-le SI affiche les opérations autorisées pour l'agent (A <sub>1</sub> ) 6- l'agent choisit un état à éditer 7-le SI demande de choisir un centre 8-l'agent choisit un centre 9-le SI lance l'édition de l'état FIN
Scénario alternatif	(A <sub>1</sub> ) : références incorrectes Ce scénario commence au point 4 du scénario nominal 5-le SI informe que les références sont incorrectes Ce scénario reprend au point 2 du scénario nominal

Titre	Ajouter des candidats inscrits « offline »
Résumé	Ce cas d'utilisation permet aux agents du MFPRE d'enregistrement des candidats dans le système. Il s'agit des candidats qui ont choisi de s'inscrire selon le mode traditionnel.
Acteurs	Agent de saisie, Système Informatique (SI)
Date de création	05 août 2005
Version	V 1.0
Scénario nominal	<p>1-l'agent de saisie demande à s'authentifier  2-le SI demande ces références  3-l'agent saisit ses références  4-le SI vérifie la validité des références saisies  5-le SI affiche les opérations autorisées pour l'agent (A<sub>1</sub>)  6-l'agent choisit l'opérations d'ajout de candidat  7-le SI demande de choisir le concours et le centre  8-l'agent de saisie fait son choix  9-le SI demande de remplir le formulaire d'ajout  10-l'agent s'exécute  11-le SI vérifie les informations saisies  12-le SI enregistrement les données du formulaire (A<sub>2</sub>)  FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : références incorrectes  Ce scénario commence au point 4 du scénario nominal  5-le SI informe que les références sont incorrectes  Ce scénario reprend au point 2 du scénario nominal</p> <p>(A<sub>2</sub>) : données du formulaire non valide  Ce scénario commence au point 11 du scénario nominal  12-le SI informe que des données du formulaire sont erronées  Ce scénario reprend au point 9 du scénario nominal</p>

Titre	Valider les demandes d'inscription
Résumé	Ce cas d'utilisation permet à l'administrateur de valider ou de refuser les demandes d'inscription des candidats. A l'issue de cette opération les candidats sont informés de l'état de leur dossier (accepté ou rejeté)
Acteurs	Administrateur, Système Informatique (SI)
Date de création	06 août 2005
Version	V 1.0
Scénario nominal	<p>1- l'administrateur demande à s'authentifier  2-le SI demande ces références  3-l'administrateur saisit ses références  4-le SI vérifie la validité des références saisies  5-le SI affiche les opérations autorisées pour l'administrateur (A<sub>1</sub>)  6-l'administrateur choisit l'opérations de validation des demandes d'inscription  7-le SI demande de choisir le concours et le centre  8- l'administrateur fait son choix  9-le SI affiche la liste des candidats correspondant au choix de l'administrateur (candidats inscrits dans le centre et pour le concours choisi) (A<sub>2</sub>)  10-l'administrateur choisit les dossiers à valider et/ou à supprimer  11-le SI valide/supprimer les dossiers en question puis informe les candidats par courrier électronique  FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : références incorrectes  Ce scénario commence au point 4 du scénario nominal  5-le SI informe que les références sont incorrectes  Ce scénario reprend au point 2 du scénario nominal</p> <p>(A<sub>2</sub>) : aucun candidat trouvé  Ce scénario commence au point 8 du scénario nominal  9-le SI informe qu'aucun candidat ne répond aux critères  Ce scénario reprend au point 7 du scénario nominal</p>

Titre	Authentification de l'utilisateur
Résumé	Ce cas d'utilisation permet de s'authentifier (mot de passe et nom utilisateur) pour effectuer des opérations d'administration réservées à un groupe d'utilisateur.
Acteurs	Agent de saisie ou l'administrateur, Système Informatique (SI)
Date de création	06 août 2005
Version	V 1.0
Scénario nominal	<p>1- l'agent de saisie ou l'administrateur demande à s'authentifier  2-le SI demande ces références  3- l'agent de saisie ou l'administrateur saisit ses références  4-le SI vérifie la validité des références saisies  5-le SI affiche les opérations autorisées pour l'utilisateur (A<sub>1</sub>)  FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : références incorrectes  Ce scénario commence au point 4 du scénario nominal  5-le SI informe que les références sont incorrectes  Ce scénario reprend au point 2 du scénario nominal</p>

Titre	Exportation des données
Résumé	Ce cas d'utilisation permet à l'administrateur du système d'exporter les données relatives aux candidats.
Acteurs	L'administrateur, Système Informatique (SI)
Date de création	06 août 2005
Version	V 1.0
Scénario nominal	<p>1- l'administrateur demande à s'authentifier  2-le SI demande ces références  3- l'administrateur saisit ses références  4-le SI vérifie la validité des références saisies  5-le SI affiche les opérations autorisées pour l'administrateur (A<sub>1</sub>)  6-l'administrateur choisit d'exporter des données  7-le SI demande de choisir le concours et son centre  8-l'administrateur fait son choix  9-le SI génère un fichier qu'il envoie à l'utilisateur  FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : références incorrectes  Ce scénario commence au point 4 du scénario nominal  5-le SI informe que les références sont incorrectes  Ce scénario reprend au point 2 du scénario nominal</p>

Titre	Importation des données
Résumé	Ce cas d'utilisation permet à l'administrateur d'importation des données relatives à la répartition des candidats ou des secrétaires de séance.
Acteurs	L'administrateur, Système Informatique (SI)
Date de création	06 août 2005
Version	V 1.0
Scénario nominal	<p>1- l'administrateur demande à s'authentifier  2-le SI demande ces références  3- l'administrateur saisit ses références  4-le SI vérifie la validité des références saisies  5-le SI affiche les opérations autorisées pour l'administrateur (A<sub>1</sub>)  6-l'administrateur choisit d'importer des données  7-le SI demande de choisir un centre, la nature des données à importer (répartition des candidats ou celle des secrétaires) et de spécifier le chemin d'accès du fichier XML contenant les données  8-l'administrateur s'exécute  9-le SI essaie de télécharger le fichier sur le serveur  10-Le SI informe que le téléchargement s'est effectué avec succès (A<sub>2</sub>)  FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : références incorrectes  Ce scénario commence au point 4 du scénario nominal  5-le SI informe que les références sont incorrectes  Ce scénario reprend au point 2 du scénario nominal</p> <p>(A<sub>2</sub>) : échec de téléchargement  Ce scénario commence au point 9 du scénario nominal  10- le SI informe que le fichier n'est pas accessible, est trop volumineux ou qu'il y a un problème de connexion.  FIN</p>

## 2. Planning

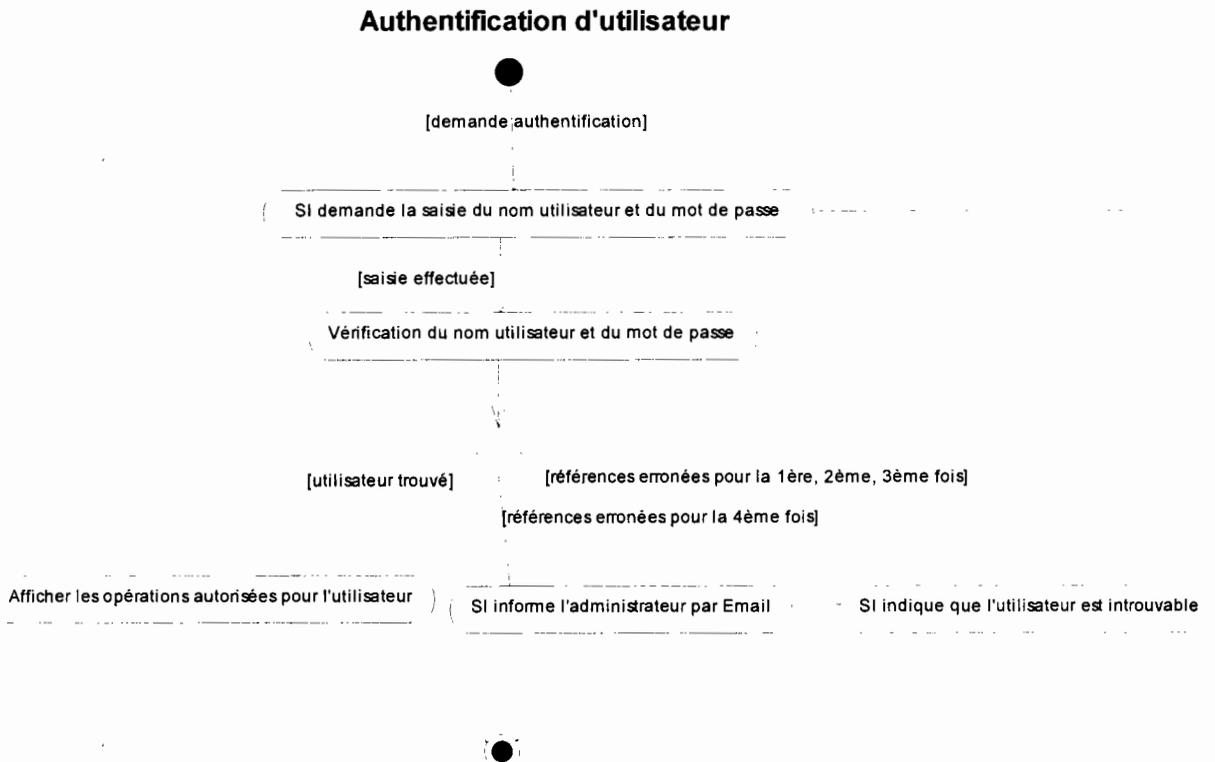
Les cas d'utilisation ont permis de dégager les fonctionnalités du futur système, et d'évaluer le travail à réaliser, ce qui nous permet d'élaborer un planning de réalisation.

<b>Etape</b>	<b>Période</b>	<b>Durée (semaine)</b>
Découverte des besoins	01 août – 07 août 2005	1
Analyse	08 août – 21 août 2005	2
Conception	22 août – 04 septembre 2005	2
implémentation	05 septembre – 05 octobre 2005	4

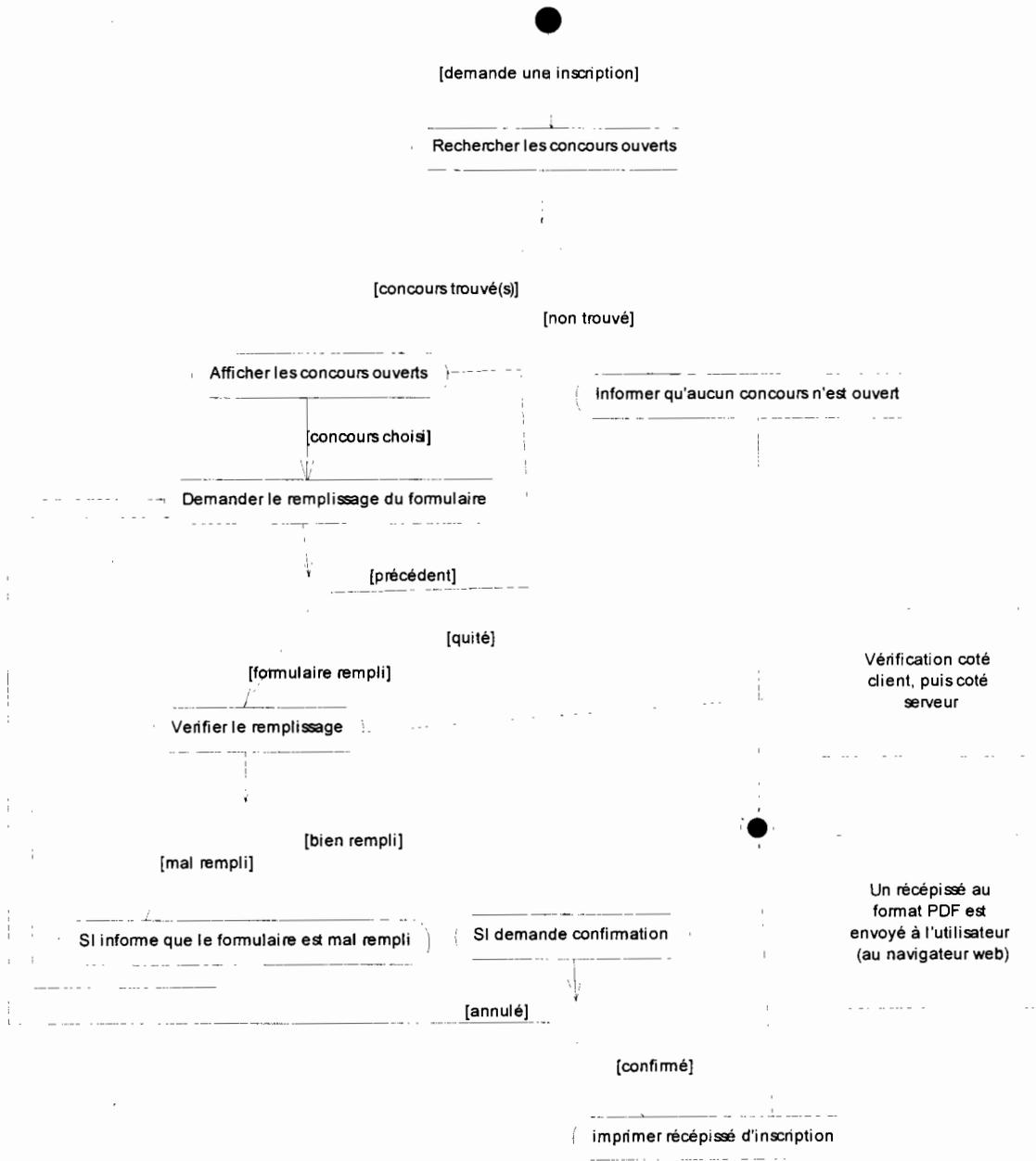
## II. ANALYSE

### 1. Diagramme d'activité

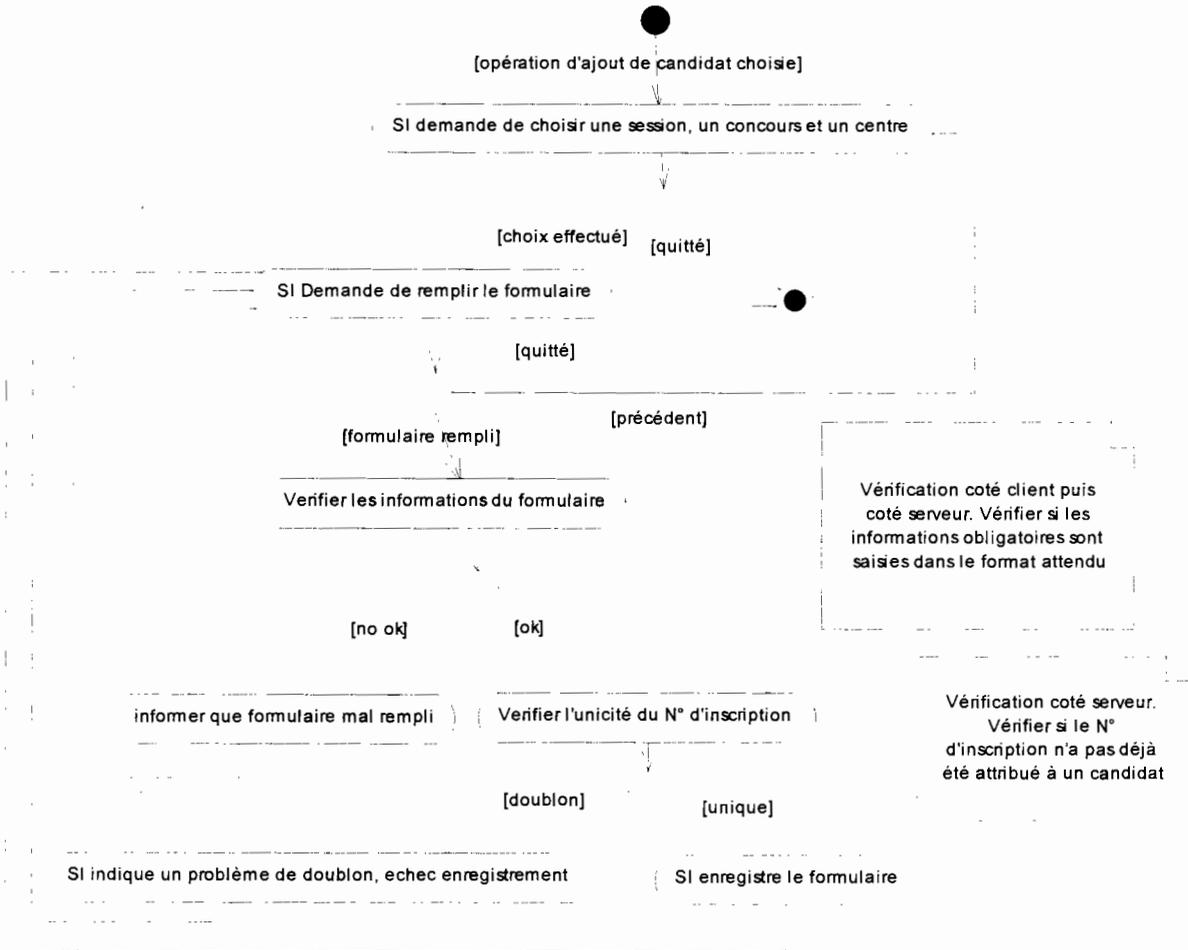
Le but de ce diagramme est d'aider à mieux décrire le comportement d'une méthode ou d'un cas d'utilisation. Il permettra d'avoir une meilleure représentation des enchaînements des tâches réalisées pour remplir une fonction.



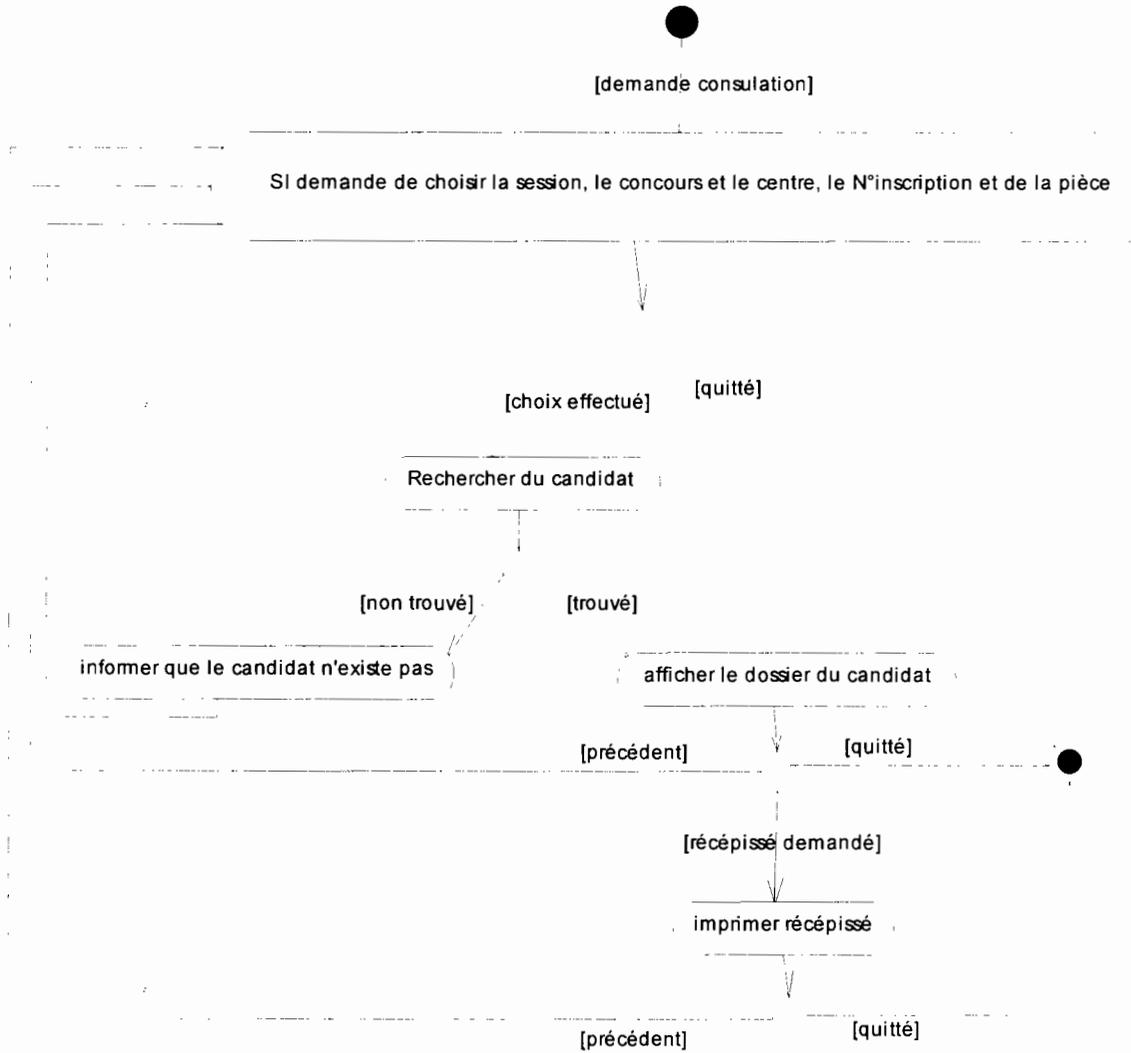
## S'inscrire à un concours



### Ajouter des candidats inscrits « offline »



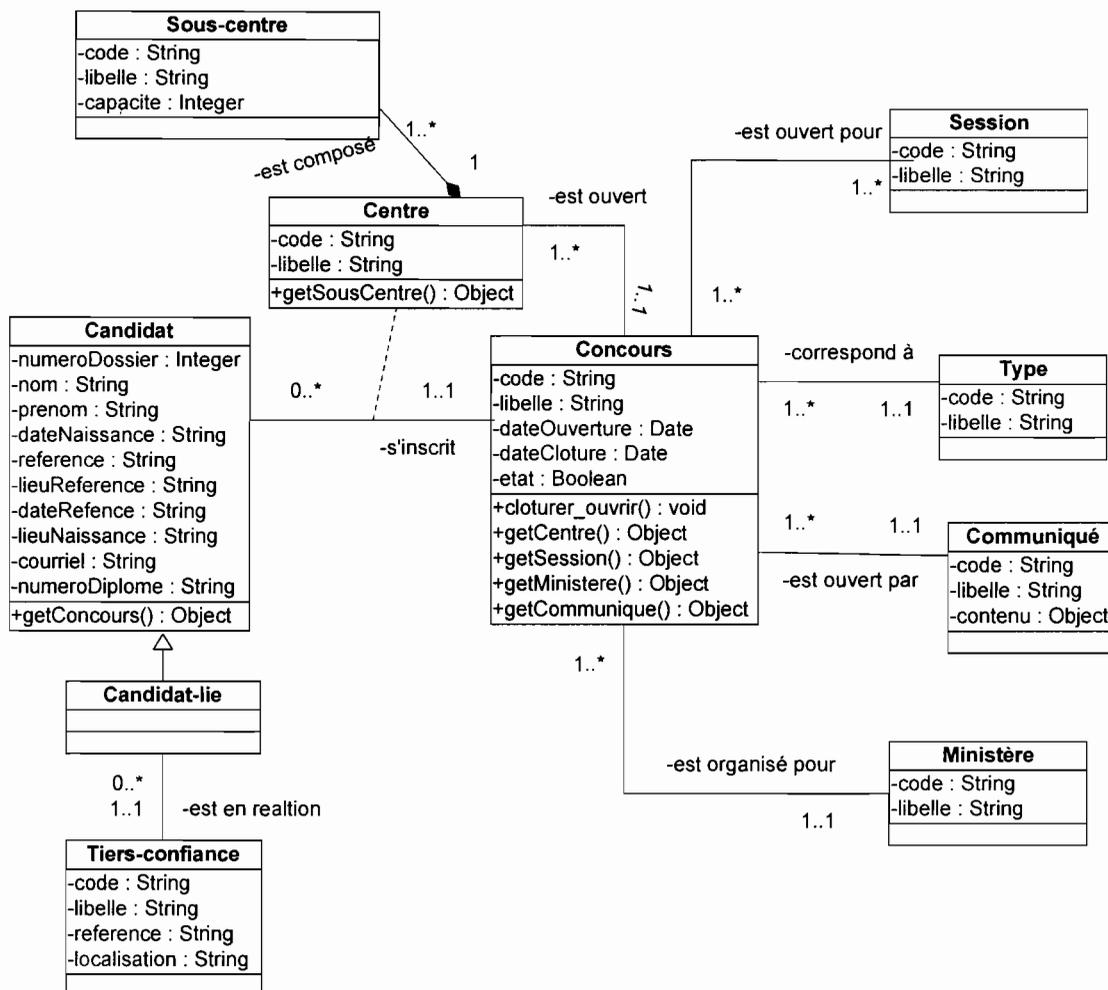
## Consulter un dossier





## 2. Diagramme de classe

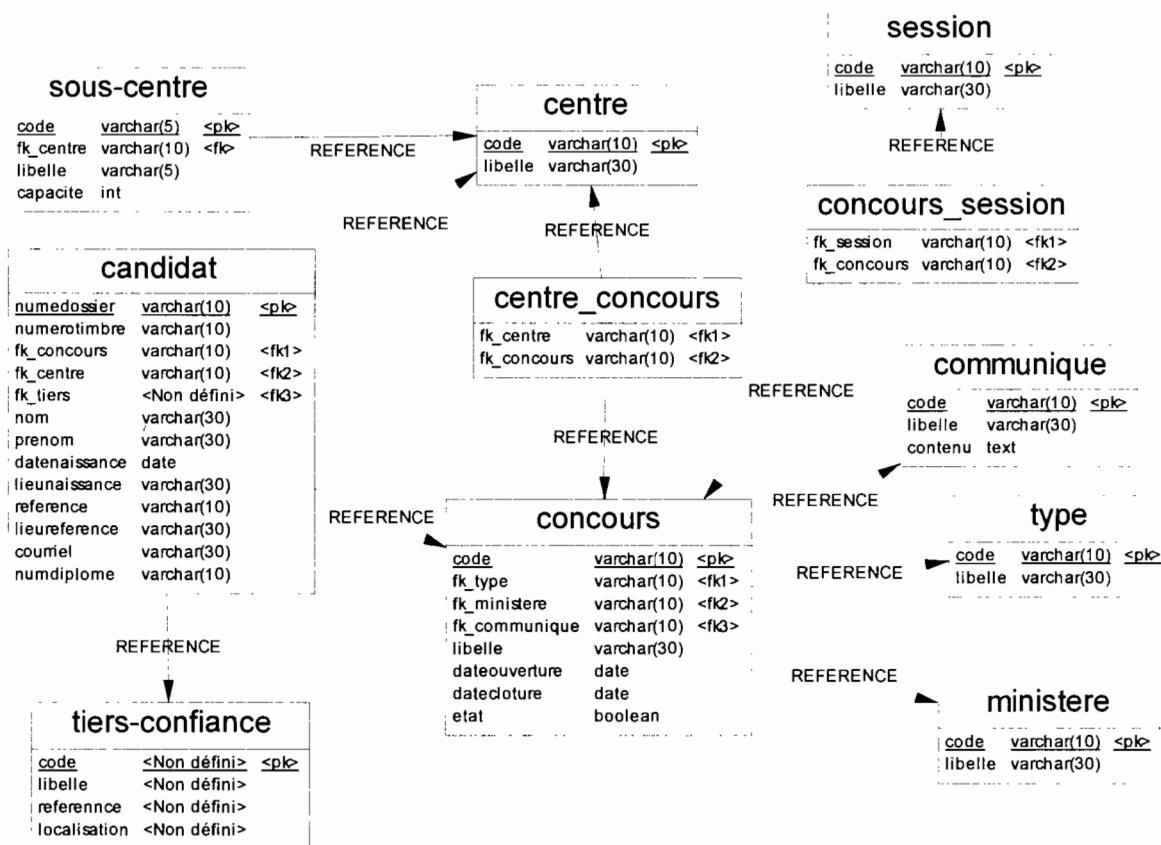
### a. Diagramme



### b. Description des classes

Les rôles de ces classes, la description de leurs attributs et méthodes seront dans la « javadoc » que nous produirons lors de la phase d'implémentation.

### 3. Modèle relationnel



### 4. Structure des fichiers XML

Notre système échangera des données avec un autre en cours de développement. Il s'agit du Système Intégré de Gestion des Concours (SIGC).

Il offrira une interface d'extraction des données sur les candidats inscrits en ligne pour le SIGC.

Le SIGC disposera d'un module d'aide à la décision pour répartir les candidats par sous-centre (à l'aide des données importées de notre base données).

Le SIGC devra par la suite nous envoyer cette répartition qui servira à produire des états sur le Web au format PDF, l'objectif étant de permettre aux centres abritant les concours, disposant de connexion Internet, de pouvoir éditer des états. En effet, certains centres hors de la capitale ne sont pas sur le réseau informatique de la Fonction Publique. Par conséquent, ils ne pourront pas accéder au SIGC, d'où la délégation de l'édition des états à notre système ouvert à Internet, à tous les centres hors la capitale.

Outre la répartition des candidats par centre, notre système offrira la possibilité d'éditer d'autres états tels que la répartition des secrétaires et des présidents de jury dans les différents centres. Pour ce dernier, notre système recevra des données en provenance de SIGC qui lui permettra de produire les états, et cela dans l'objectif de permettre un accès plus large aux états produits.

Pour ce faire, nous avons choisi d'utiliser XML pour l'échange des données. Chaque fichier respecte un schéma connu des deux systèmes. Pour produire les états au format PDF, nous utilisons des fichiers de présentation XSL-FO que nous ne présenterons pas dans ce rapport de synthèse.

a. Grammaire et schéma des fichiers XML pour l'exportation de données contenues dans les tables

Comme HTML, XML est un langage à balises. Nous allons décrire la grammaire des balises des documents XML en utilisant les DTD (*Document Type Definition*). La DTD permet de définir les éléments et les attributs avec leurs imbrications éventuelles. Elle précise aussi le fait que les éléments sont facultatifs, obligatoires ou répétés ; pour cela, les opérateurs + (signifiant 1 à N) et \* (signifiant 0 à N) sont utilisés dans la DTD. Il est possible de rendre obligatoire certains attributs en choisissant l'option #REQUIRED. Nous utilisons dans nos DTD, les deux types de base : #PCDATA (élément de texte sans descendant ni attribut contenant des caractères à analyser) et CDATA (données brutes entre guillemets qui seront prises en compte comme un texte sans autre analyse).

Une table de notre base de données a un titre, des attributs et éventuellement des données. La DTD ci-dessous décrit une table comme un ensemble composé d'un titre, d'un attribut et de plusieurs lignes qui sont des ensembles. Le titre est une chaîne de caractères quelconques (PCDATA). L'attribut et la ligne sont composés de cellules. Une cellule est une chaîne de caractères.

- Grammaire

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<!ELEMENT table (titre, attribut, ligne+)>
```

```
<!ELEMENT titre (#PCDATA)>
```

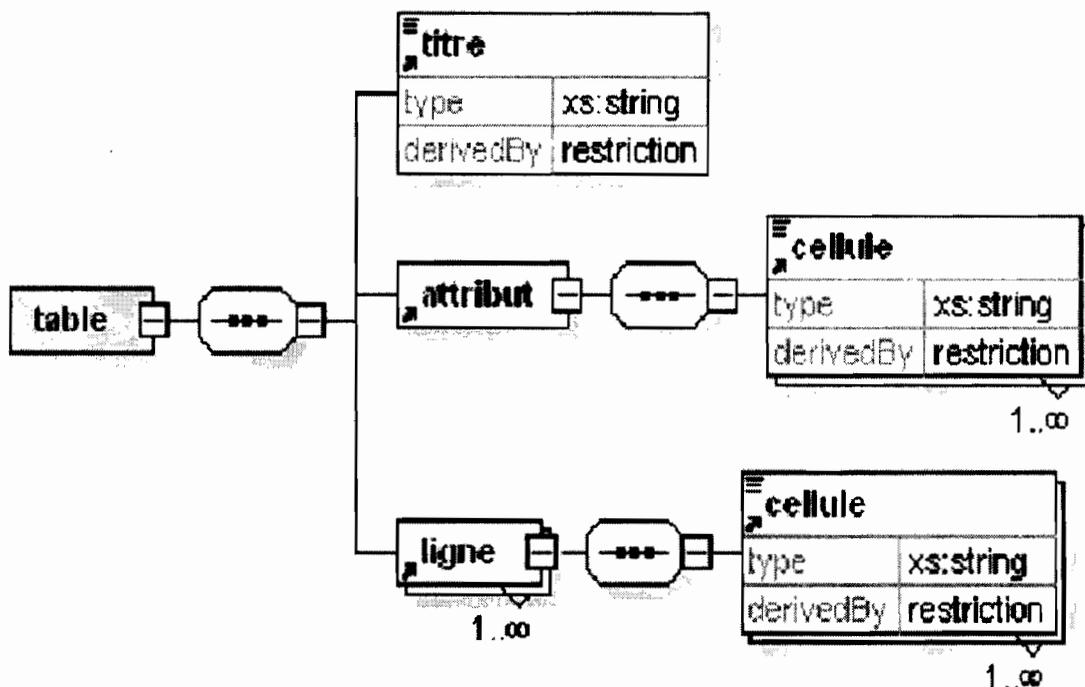
```
<!ELEMENT attribut (cellule+)>
```

```
<!ELEMENT ligne (cellule+)>
```

```
<!ELEMENT cellule (#PCDATA)>
```

- Schéma

Le schéma est plus précis que la grammaire. Il donne plus de détails sur le type des données (entier, chaîne de caractère ...). Nous l'avons réalisé avec l'outil Altova XMLSpy.



b. Grammaire et schéma des fichiers XML contenant des données sur la répartition des candidats par centre et sous centre

En fonction du nombre de candidats inscrits, les organisateurs du concours évaluent le nombre de jours nécessaires pour le déroulement des épreuves écrites et/ou orales. Chaque journée fait l'objet de plusieurs programmations. La programmation consiste à répartir les candidats (par tranche de numéro) qui prendront part à un concours dans un sous centre. Le nombre de surveillants et de salles est également défini.

- Grammaire

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<!ELEMENT repartition (commentaire, journee+)>
```

```
<!ELEMENT commentaire (#PCDATA)>
```

```
<!ELEMENT journee (programmation+)>
```

```
<!ATTLIST journee CDATA #REQUIRED >
```

```
<!ELEMENT programmation (sous-centre, concours, tranche-numero, nombre-surveillant, nombre-salle)>
```

```
<!ELEMENT sous-centre (#PCDATA)>
```

```
<!ELEMENT concours (#PCDATA)>
```

```
<!ELEMENT tranche-numero (debut, fin)>
```

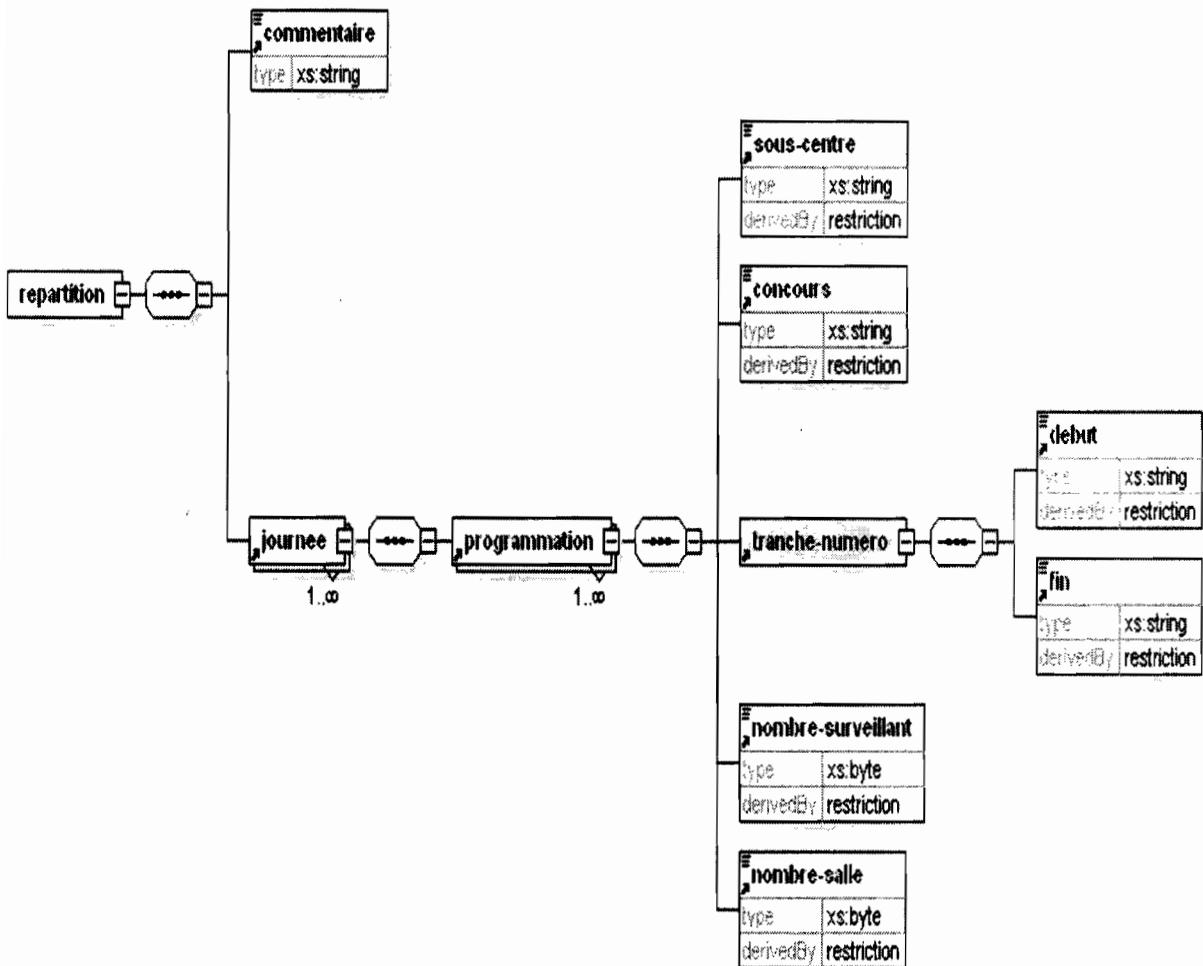
```
<!ELEMENT debut (#PCDATA)>
```

```
<!ELEMENT fin (#PCDATA)>
```

```
<!ELEMENT nombre-salle (#PCDATA)>
```

```
<!ELEMENT nombre-surveillant (#PCDATA)>
```

- Schéma



c. Grammaire et schéma des fichiers XML contenant des données sur la répartition des secrétaires dans les centres

Cette répartition des secrétaires de séances et des présidents de jury est similaire à celle des candidats.

- Grammaire

```
<?xml version="1.0" encoding=" ISO-8859-1"?>
```

```
<! ELEMENT repartition (commentaire, journee+)>
```

```
<! ELEMENT commentaire (#PCDATA)>
```

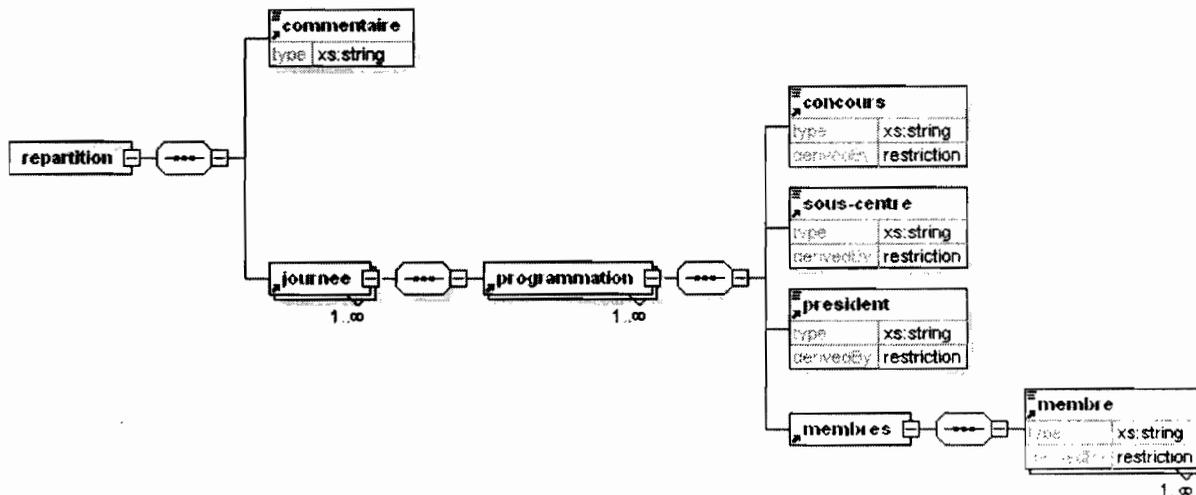
```
<! ELEMENT journee (programmation+)>
```

```
<! ATTLIST journee CDATA #REQUIRED>
```

```
<! ELEMENT programmation (concours, sous-centre, president, membres)>
```

- <! ELEMENT concours (#PCDATA)>
- <! ELEMENT sous-centre (#PCDATA)>
- <! ELEMENT president (#PCDATA)>
- <! ELEMENT membres (membre+)>
- <! ELEMENT membre (#PCDATA)>

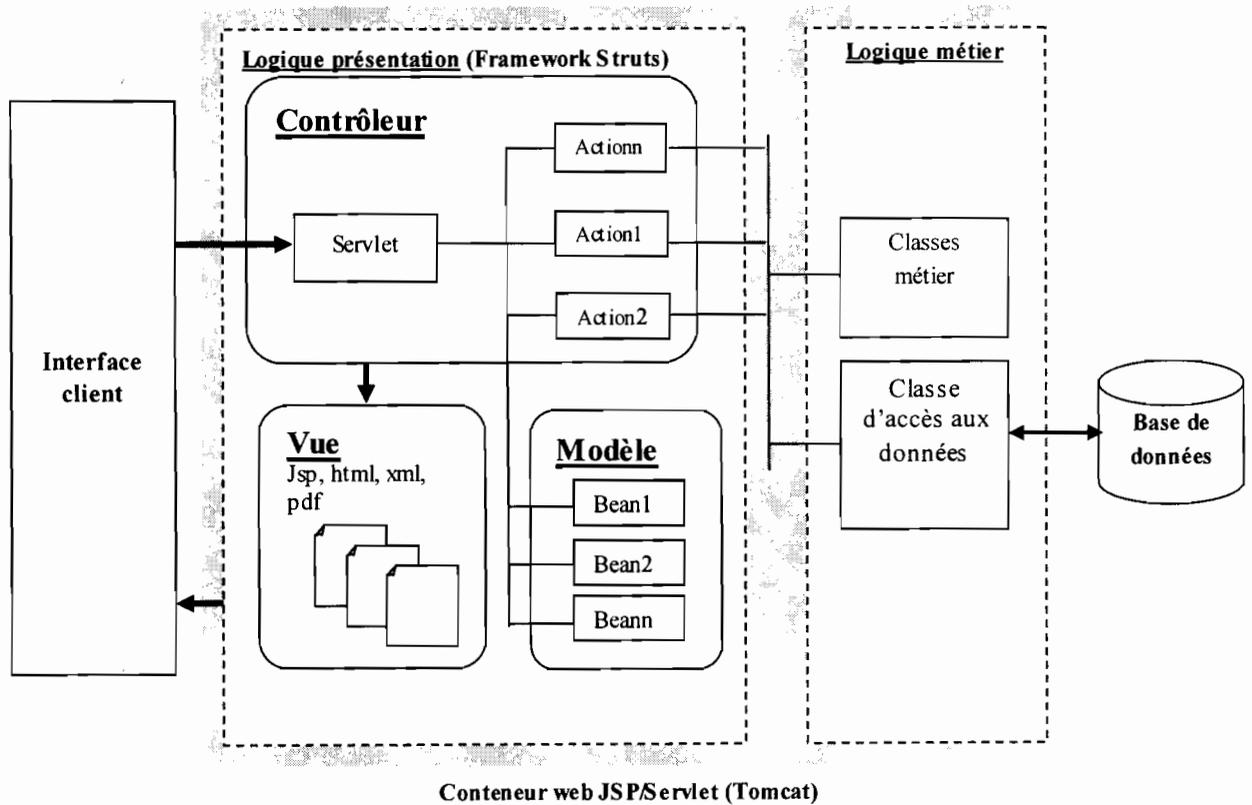
• Schéma



### III. CONCEPTION

#### 1. Architecture du système

Nous allons utiliser la « framework Apache/Jakarta/Struts » qui fournit un cadre standard de développement d'applications Web en Java respectant le modèle MVC2. Ce framework présente également d'autres avantages. En effet, il fournit une Servlet générique et paramétrable (le développeur n'a pas besoin d'écrire un contrôleur), il permet de réaliser des formulaires dynamiques grâce à des balises spécifiques, de construire l'application par module...



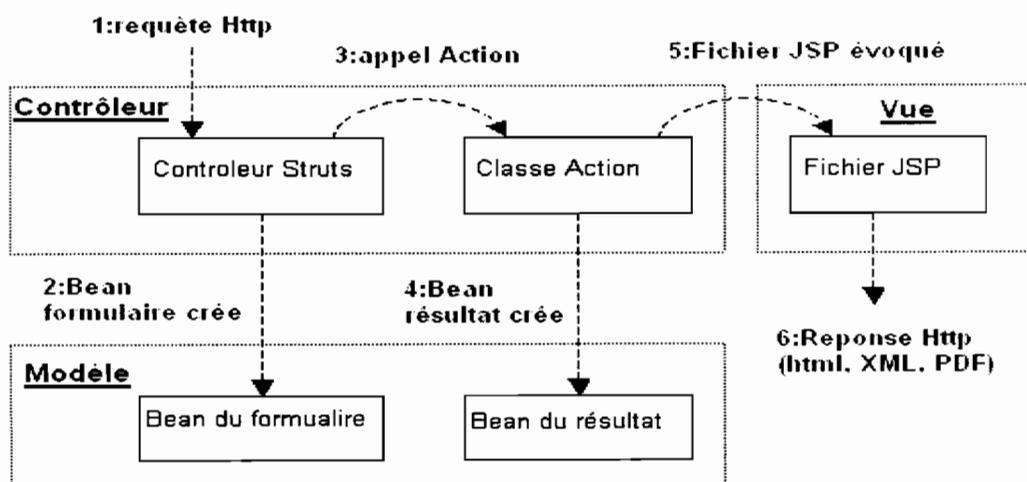
Architecture du système (3-tiers)

#### Description de l'architecture

- la *Servlet* prend les informations dont elle a besoin dans un fichier de configuration appelé *struts-config.xml* ;
- si la requête du client contient des paramètres de formulaire, ceux-ci sont mis par le contrôleur dans un objet *Bean*.

- dans le fichier de configuration struts-config.xml, à chaque URL devant être traitée on associe : le nom de la classe de type *Action* chargée de traiter la requête, le nom du *Bean* chargé de mémoriser les informations du formulaire (cas de l'envoi d'un formulaire au contrôleur).
- muni de ces informations fournies par son fichier de configuration, à la réception d'une demande d'URL par un client, le contrôleur est capable de déterminer s'il y a un *Bean* à créer et lequel. Il crée une instance du *Bean* et y stocke les informations provenant du formulaire et contenu dans la requête HTTP. Une fois créé, le *Bean* peut vérifier que les données qu'il a stockées et qui proviennent du formulaire sont valides ou non. Une méthode du *Bean* appelée « *validate* » est appelée par le contrôleur. Si les données se révèlent invalides, le contrôleur passera la main à une *vue* dont il trouvera le nom dans son fichier de configuration. L'échange est alors terminé.
- si les données du *Bean* sont correctes, ou s'il n'y a pas de vérification ou d'objet *Bean*, le contrôleur passe la main à l'objet de type *Action* associé à l'URL. Il le fait en demandant l'exécution de la méthode « *execute* » de cet objet à laquelle il transmet la référence du *Bean* qu'il a éventuellement construit. C'est dans cette méthode « *execute* » que le développeur fait éventuellement appel à des *classes métier* ou à des *classes d'accès aux données*. A la fin du traitement, l'objet *Action* rend au contrôleur le nom de la *vue* qu'il doit envoyer en réponse au client.
- le contrôleur envoie cette réponse. L'échange avec le client est terminé.

Description plus technique de la logique présentation



1. le client envoie une requête à l'application. La requête est prise en charge par la Servlet d'entrée (Contrôleur Struts) ;
2. la Servlet stocke les données de la requête dans un objet javaBean (si la requête contient des données) ;
3. la Servlet d'entrée analyse la requête et réoriente celle-ci vers un contrôleur adapté (objet Action) ;
4. le contrôleur sélectionné par la Servlet d'entrée est responsable de l'exécution des traitements nécessaires à la satisfaction de la requête. Il sollicite les objets métier lorsque nécessaire. Ces objets métier lui fournissent des données qu'il encapsule dans des javaBeans ou dans des variables sessions ;
5. le contrôleur sélectionne la JSP qui sera en charge de la construction de la réponse et lui transmet éventuellement les JavaBean contenant les données métiers ;
6. la JSP construit la réponse en faisant appel aux javaBeans ou aux variables sessions qui lui ont été transmis et l'envoie au navigateur Web ;

### Apport de la framework Struts

L'apport le plus significatif de Struts est la structuration de la couche de présentation des applications suivant le pattern MVC2. Le développeur est bien mieux guidé dans sa tâche. En général, il lui suffit de personnaliser les actions pour les traitements adéquats des requêtes. Il en découle un développement plus efficace mais également une meilleure homogénéisation du code des applications.

On aboutit facilement à une application robuste, évolutive et maintenable avec un niveau d'expertise minime.

### La méthodologie de développement Struts :

- la définition des « Vues ». On distingue les « Vues » qui sont des formulaires et les autres.
  - chaque « Vue » formulaire donne naissance à une définition dans le fichier struts-config.xml. On y définit les informations suivantes :
    - le nom de la classe Bean qui contiendra les données du formulaire ainsi que l'indication sur le fait que les données doivent être vérifiées ou non. Si elles doivent être vérifiées et qu'elles s'avèrent invalides, on doit indiquer la vue à envoyer en réponse au client ;

- le nom de la classe Action chargée de traiter le formulaire ;
- le nom de toutes les vues susceptibles d'être envoyées en réponse au client une fois que la requête a été traitée. La classe Action choisira l'une d'elles selon le résultat du traitement ;
  - o chaque vue fait l'objet d'une page JSP ;
- l'écriture des classes JavaBean correspondant aux « Vues » formulaires ;
- l'écriture des classes Action chargées de traiter les formulaires ;
- l'écriture des éventuelles classes métier ou d'accès aux données.

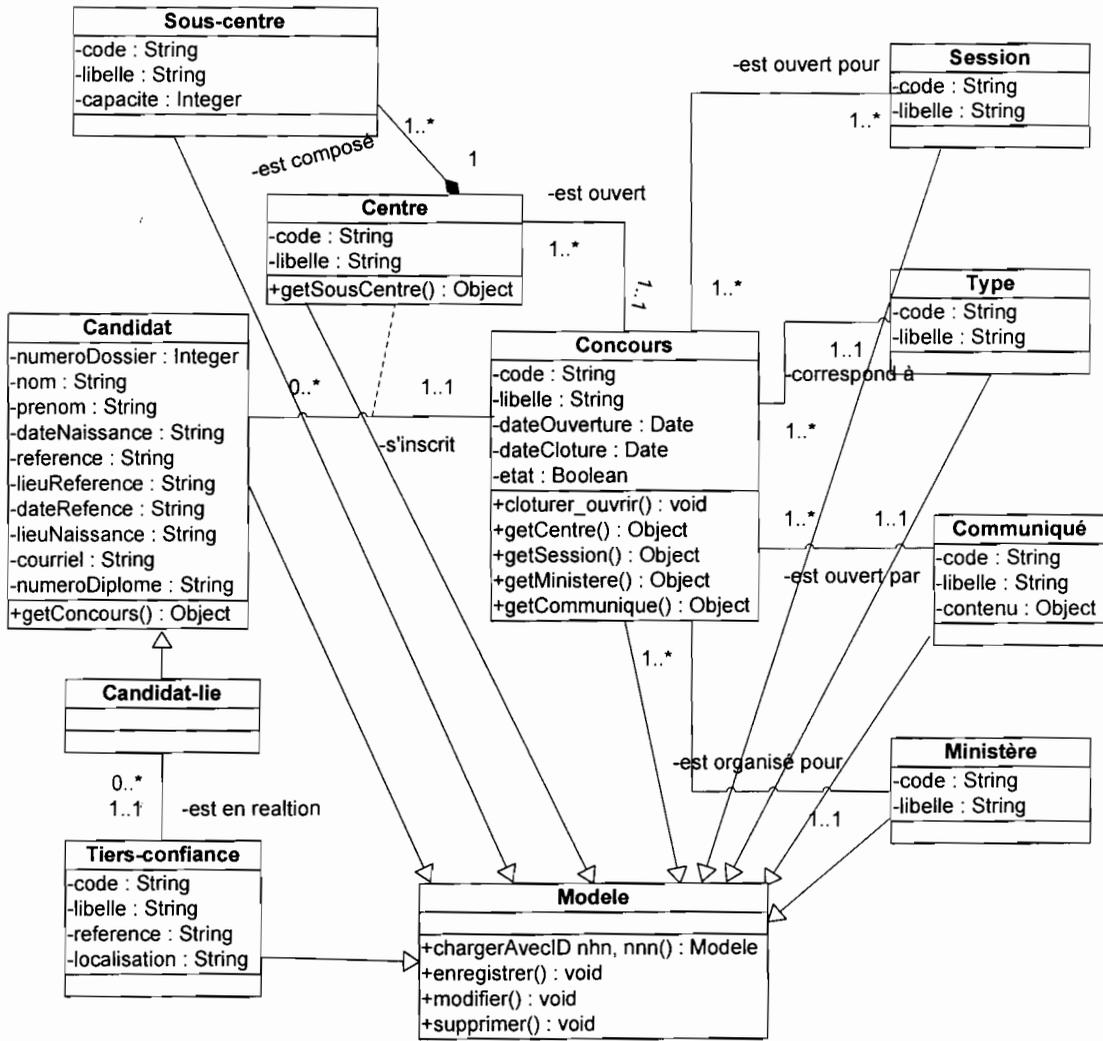
### Les composants (JSP, JavaBean, Servlet, classes métier) de l'application

Vue (JSP)	Bean	Action	Classes métier
concours.jsp	x	ConcoursAction	Concours , Ministere , Type
fiche_inscription.jsp	fiche_inscriptionBean	fiche_inscriptionAction	Concours , Centre
confirmation.jsp	x	confirmationAction	
recepisser.jsp	recepisserBean	recepisserAction	Candidat
info_concours.jsp	info_concoursBean	info_concoursAction	Concours , Type, Ministere , Sous- centre, Centre, Session
demande_consultation.jsp	x	demande_consultationAction	Concours , Centre
fiche_consultation.jsp	fiche_consultationBean	fiche_consultationAction	Candidat, Centre, Session
tele_inscription.htm	x	x	x
condition-generale.htm	x	x	x
	x	x	x
p_administration.jsp		p_administrationAction	

p_parametrage.jsp	x	x	x
p_centre.jsp	p_centreBean	p_centreAction	Centre
p_ministere.jsp	p_ministereBean	p_ministereAction	Ministere
p_concours.jsp	p_concoursBean	p_concoursAction	Concours
p_serveur_stmp.jsp	p_serveur_stmpBean	p_serveur_stmpAction	x
p_lancement_concours.jsp	p_lancement_concours Bean	Action	
p_type.jsp	p_typeBean	p_typeAction	
p_choix_session_concours_centre.jsp	x	p_choix_session_concoursAction	Concours , Centre
p_ajout_candidat.jsp	p_ajout_candidatBean	p_ajout_candidatAction	Candidat
p_choix_validation.jsp	x	p_choix_validationAction	Concours
p_validation.jsp	x	p_validationAction	Concours , Centre
p_choix_listing_candidat.jsp		p_choix_listing_candidatAction	Centre
p_listing_candidat.jsp	p_listing_candidatBean	p_listing_candidatAction	Candidat, Centre, Sous- Centre
p_repartition-candidat.jsp	x	p_repartition-candidatAction	Centre, Session, Sous- Centre
p_repartition-secretaire.jsp	x	p_repartition-secretaireAction	Centre, Session, Sous- Centre
p_choix_exportation.jsp	x	p_choix_exportationAction	Centre, Concours
p_exportation.jsp	p_exportationBean	p_exportationAction	Candidat
upload.jsp	uploadBean	uploadAction	Candidat

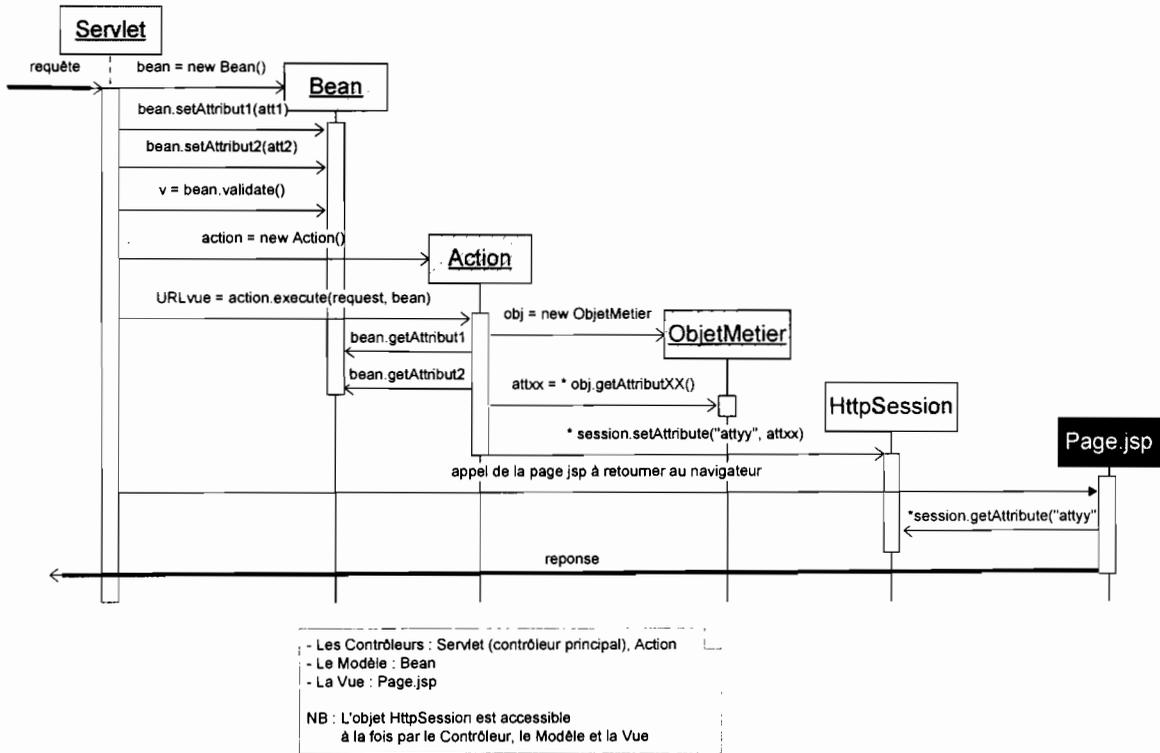
Ce tableau contient toutes les vues JSP de notre application, il fait la correspondance entre l'objet Vue (JSP), l'objet Bean qui stocke les données (cas de l'envoi d'un formulaire), l'objet Action qui traite les données et interagit avec les objets métier pour extraire ou enregistrer des données dans la Base de données.

## 2. Diagramme de classe d'implémentation

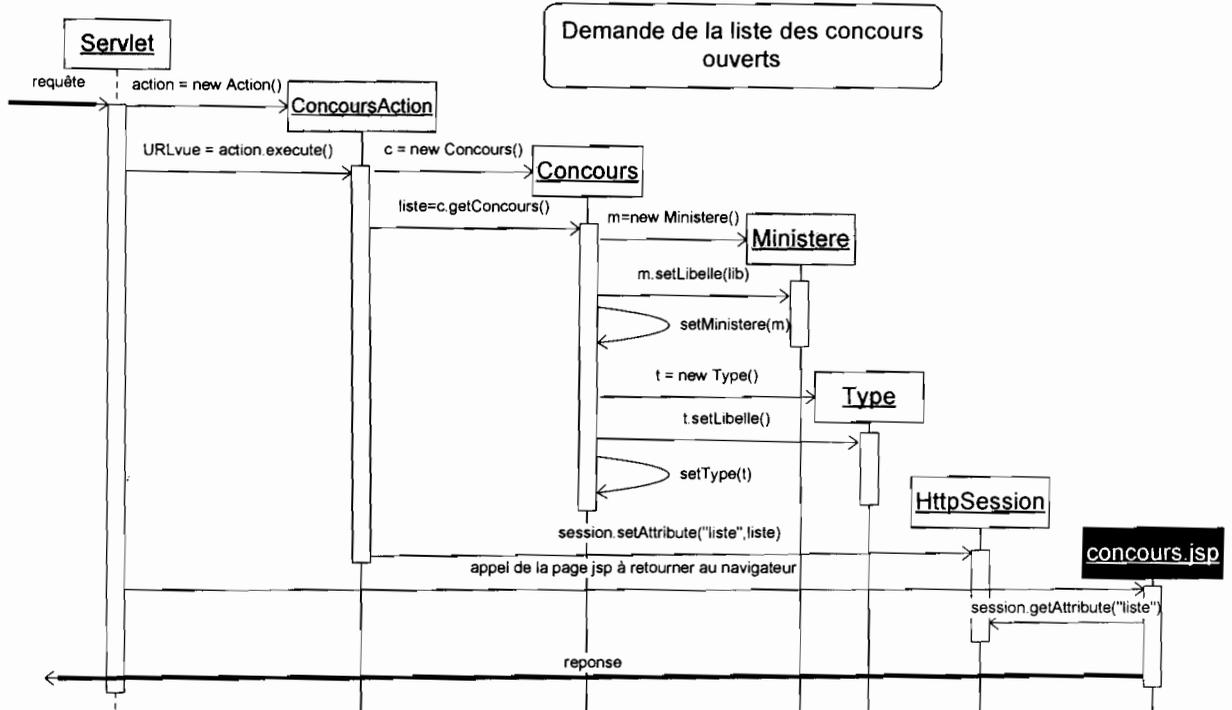


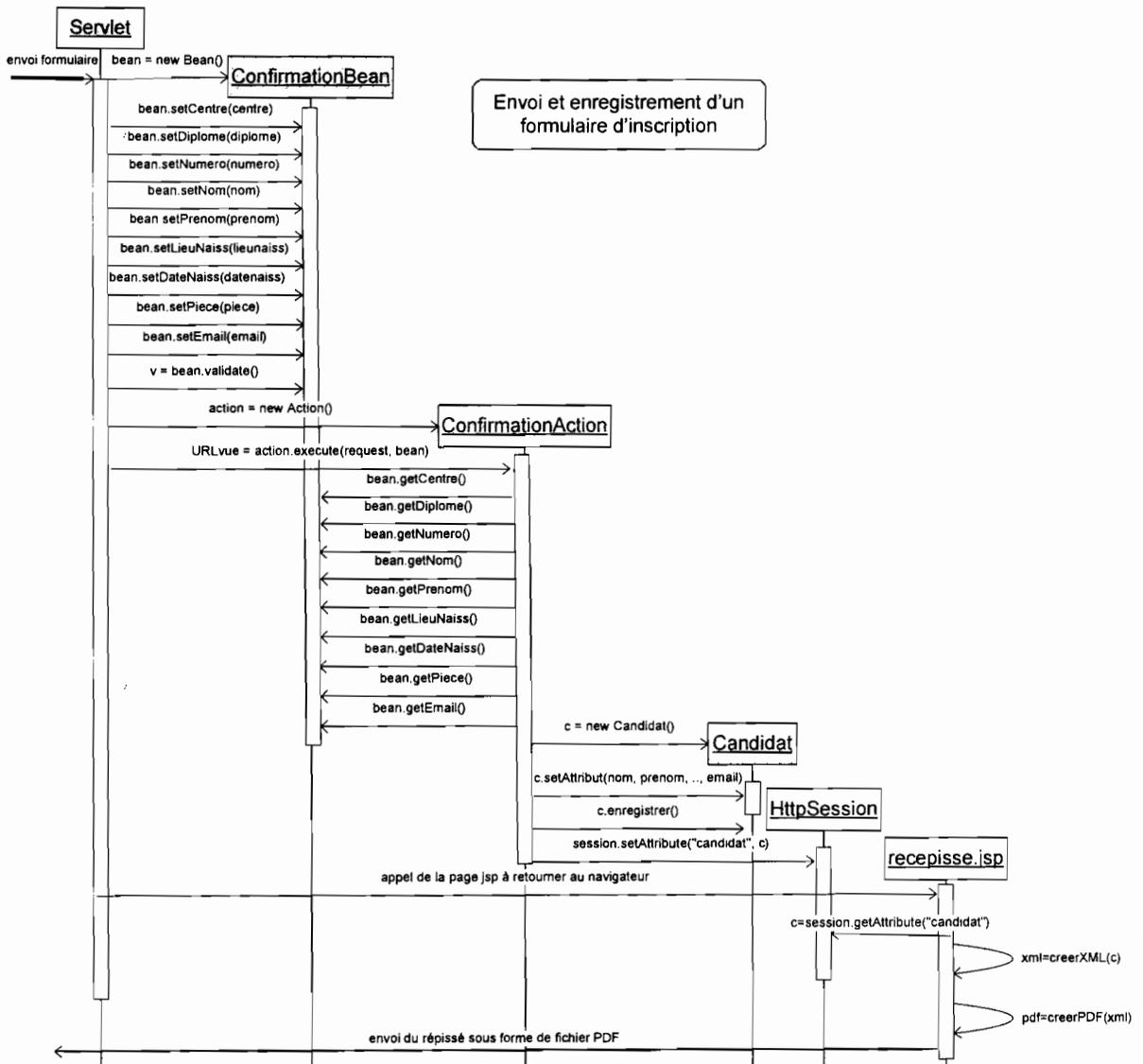
### 3. Diagramme de séquence

#### a. Diagramme de séquence générale

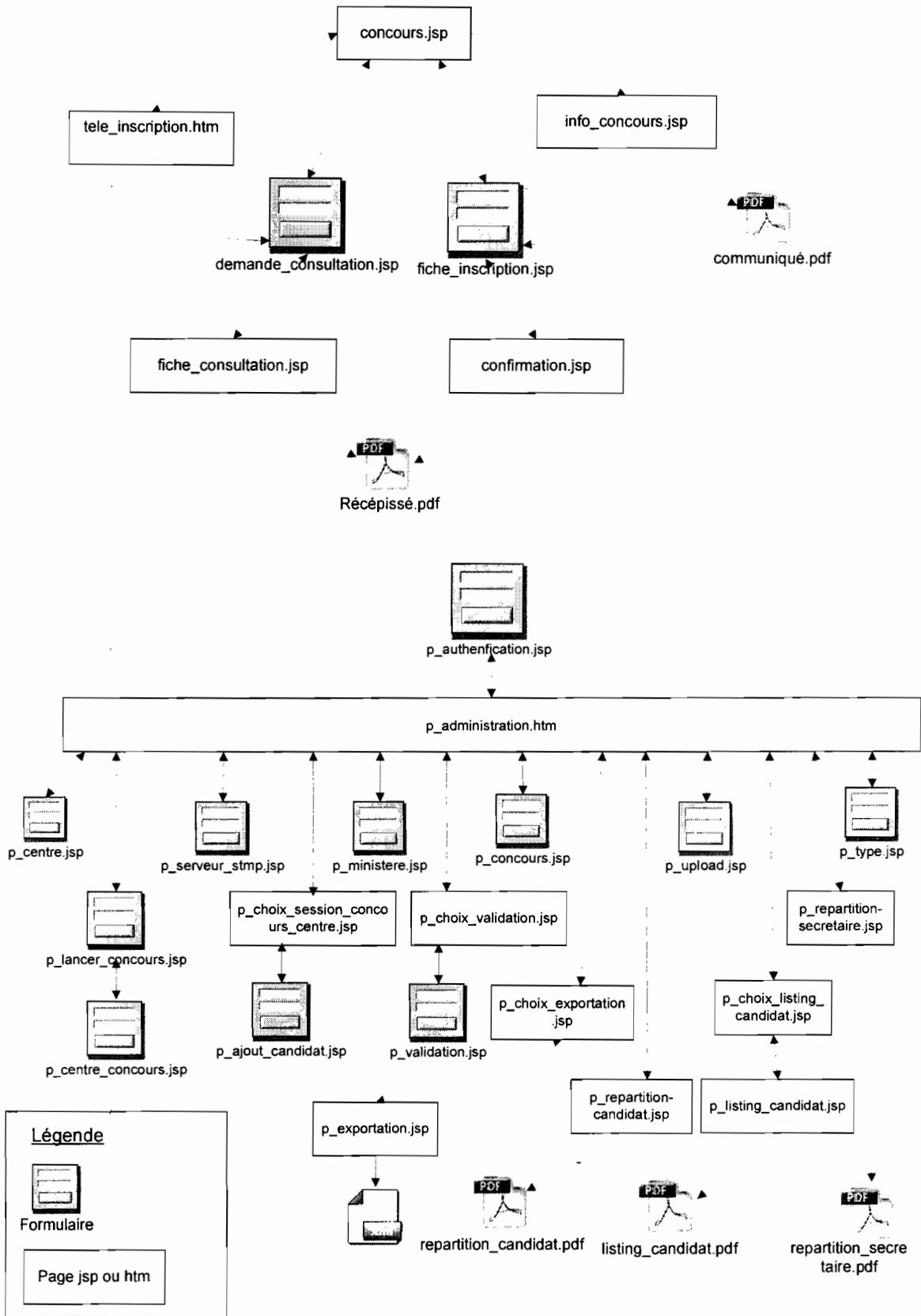


b. Quelques exemples

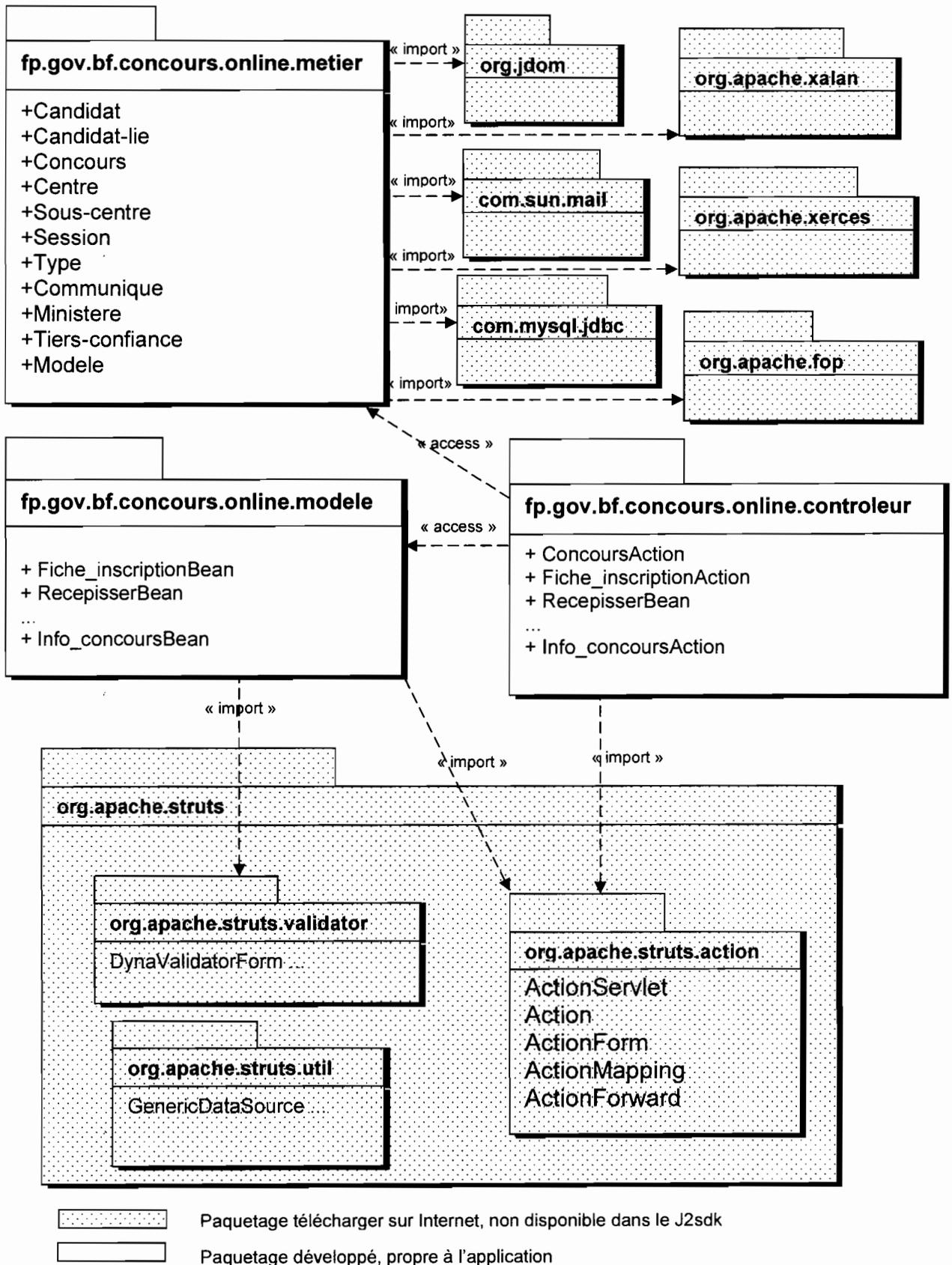




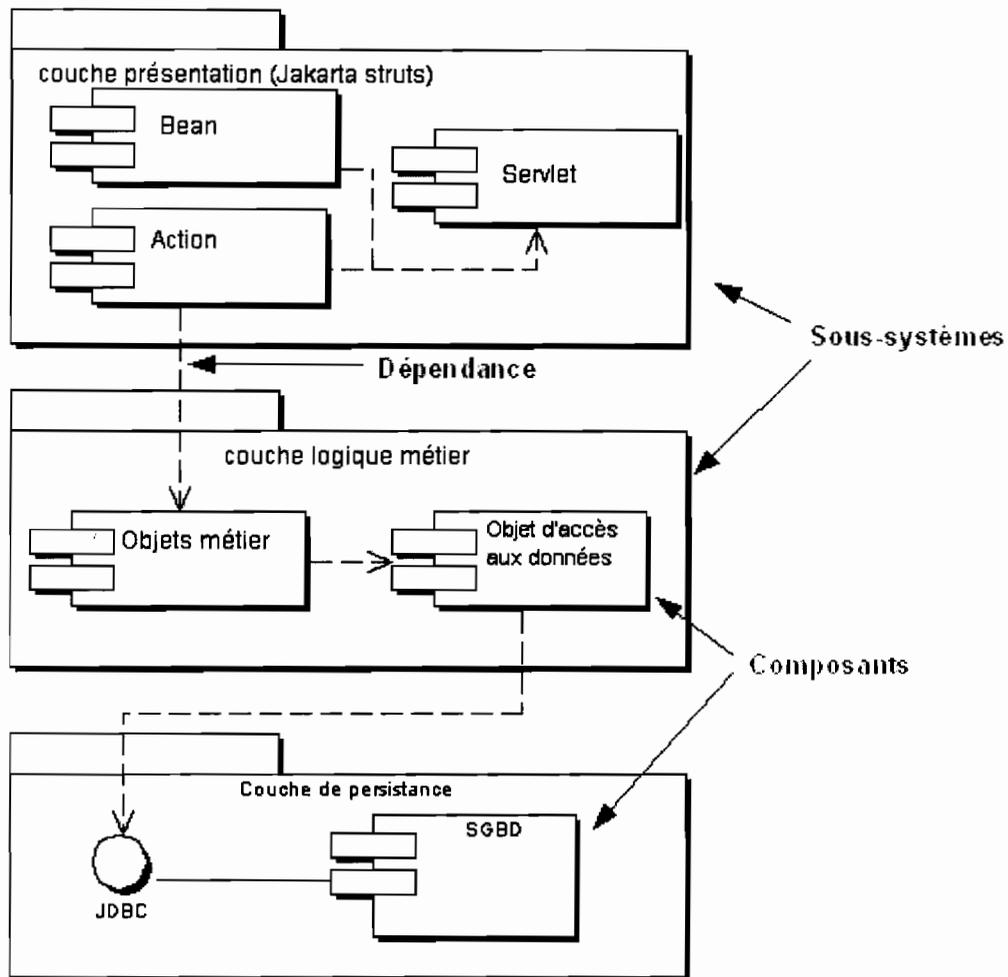
### 4. Diagramme d'enchaînement des pages Web



### 5. Diagramme de paquetage



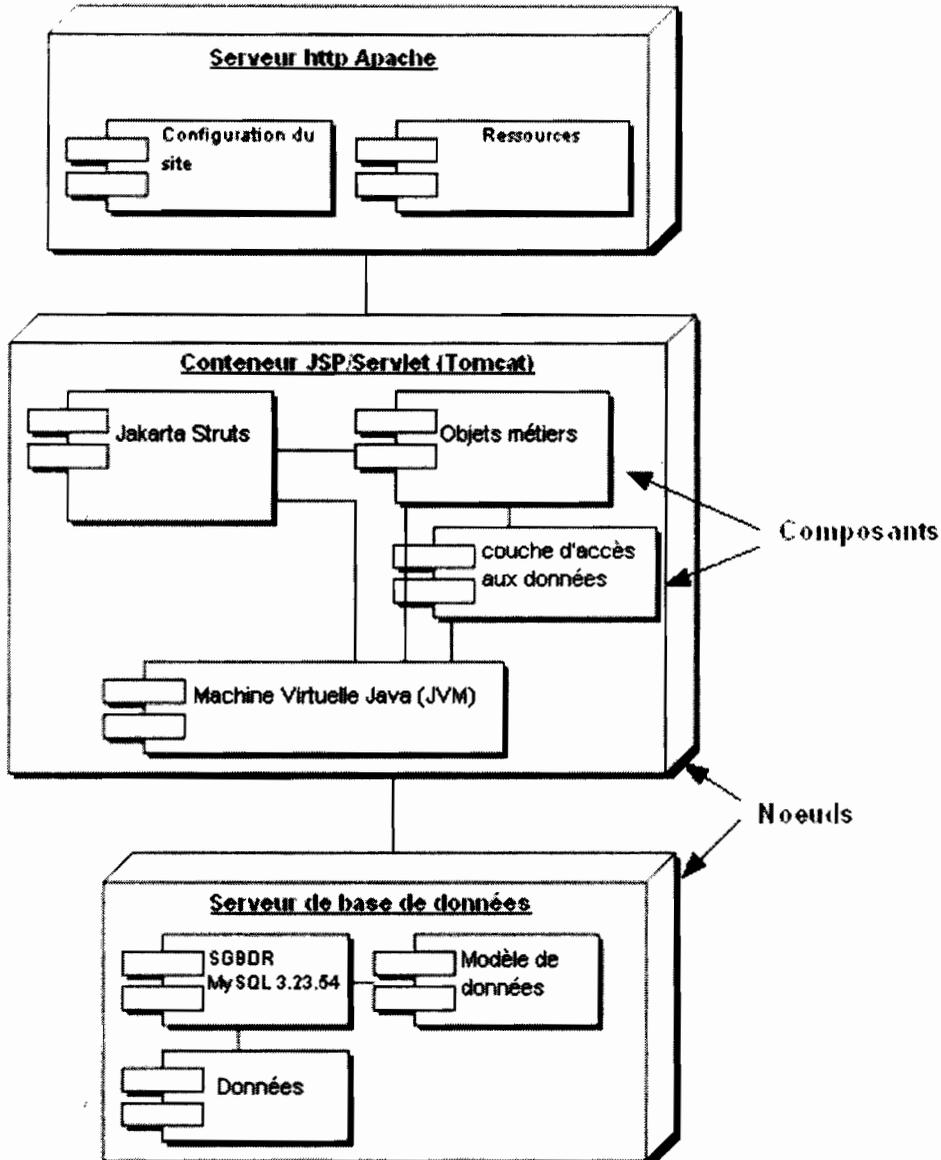
## 6. Diagramme de composants



## IV. IMPLEMENTATION

Cette phase a consisté à la réalisation (codage, intégration et test) de l'application Web. Nous expliquons comment déployer, installer et configurer le logiciel.

### 1. Diagramme de déploiement



## 2. Procédure d'installation de l'application

L'application a été testée sous Windows XP service pack 2 et sous la redhat Entreprise WS avec les logiciels suivant :

- Apache Tomcat 5.0.27 : téléchargeable gratuitement à l'url <http://archive.apache.org/dist/tomcat/tomcat-5/archive/v5.0.27/>
- j2sdk 1.4.2 : téléchargeable gratuitement sur <http://java.sun.com/j2se/1.4.2/download.html>
- MySQL 3.23.54 et 4.1.9 téléchargeable gratuitement sur <http://dev.mysql.com/downloads/mysql/4.0.html>
- OpenCms 5.0.1 également téléchargeable gratuitement sur <http://www.opencms.org/opencms/en/download/archive.html>

Pour faciliter l'installation de l'application, les fichiers d'installation ont été regroupés dans un répertoire nommé *concours-fonction-publique* dont le contenu est le suivant :

- *concours.zip* : contient les pages dynamiques jsp, les images, les pages statiques html et les fichiers de configuration de l'application ;
- *bibliotheque* : c'est le répertoire qui contient toutes les bibliothèques (.jar) nécessaires au fonctionnement de l'application ;
- *script-sql.sql* : ce fichier texte contient les instructions SQL pour la création de la base de données.

Après avoir installé les logiciels ci-dessus (Apache Tomcat, j2sdk, MySQL, OpenCms) :

- lancer OpenCms ;
- importer le fichier compressé *concours.zip* en demandant sa décompression (l'application est ainsi installée, il ne reste plus qu'à le configurer) ;
- après la décompression, OpenCms crée un répertoire *concours* contenant les fichiers décompressés. Ouvrez le fichier *concours\config.xml* pour modifier le chemin d'accès à la base de données, définir le nom et le mot de passe de l'utilisateur MySQL qui a droit à la base de données de l'application ;
- copier les fichiers du répertoire *bibliotheque* dans le répertoire *lib* d'OpenCms ;

- créer la base de données en exécute les instructions SQL contenues dans *script-sql.sql* ;

Redémarrer votre serveur Apache Tomcat et vérifier que l'application fonctionne en invoquant sur votre serveur l'url *http://127.0.0.1:8080/concours/concours.jsp* (Attention le port 8080 est celui d'écoute de votre serveur, il peut être différent).

**PARTIE 3 :**  
**Mise en ligne des situations, positions  
administratives et carrière des agents de la  
Fonction Publique**

- Le domaine d'étude

Le domaine d'étude est la Direction des Ressources Humaines du Ministère de la Fonction Publique et de la Réforme de l'Etat.

- Les acteurs du projet

- Groupe de pilotage

Il est constitué de :

M. Joachim TANKOANO : Délégué Général à l'Informatique ;

M. Youssouf OUATTARA : Secrétaire Général du MFPRE ;

M. Isidore OUEDRAOGO : Directeur des Services Informatiques ;

M. Prosper KAMBIRE : Directeur Général de la Fonction Publique ;

- Groupe de réalisation

Il est composé de :

M. Moustapha BANDE : responsable du département des études et de l'informatisation de la DELGI ;

M. Hamidou GASSIMBE : étudiant stagiaire.

- Groupe des utilisateurs

Il se compose des informaticiens de la Direction des Services Informatiques du Ministère de la Fonction Publique, du personnel de la Direction des Ressources Humaines et du Directeur de la Communication et de la Presse Ministérielle de la Fonction Publique.

## **I. DECOUVERTE DES BESOINS**

---

- Rappel des objectifs

L'objectif est de développer une application Web accessible depuis Internet, qui permettra aux fonctionnaires de consulter en ligne leur situation administrative, leur carrière et l'historique de leurs dossiers en cours de traitement. Il s'agit de mettre à la disposition des usagers la partie consultation de la base de données SIGASPE.

- Besoins exprimés

Les personnes ressources du domaine d'étude ont exprimé les besoins suivants :

- permettre aux fonctionnaires de consulter en ligne :
  - les informations relatives à leur carrière et à leur situation administrative ;
  - l'historique de leur dossier (pour ceux disposant d'un dossier en cours de traitement) ;
  - d'envoyer des courriers au MFPRE pour signaler d'éventuelles erreurs constatées dans la base de données ou faire des suggestions ;
- trouver un système d'authentification des usagers qui souhaitent accéder au système dans le but de préserver la confidentialité des informations.

- Présentation de l'existant

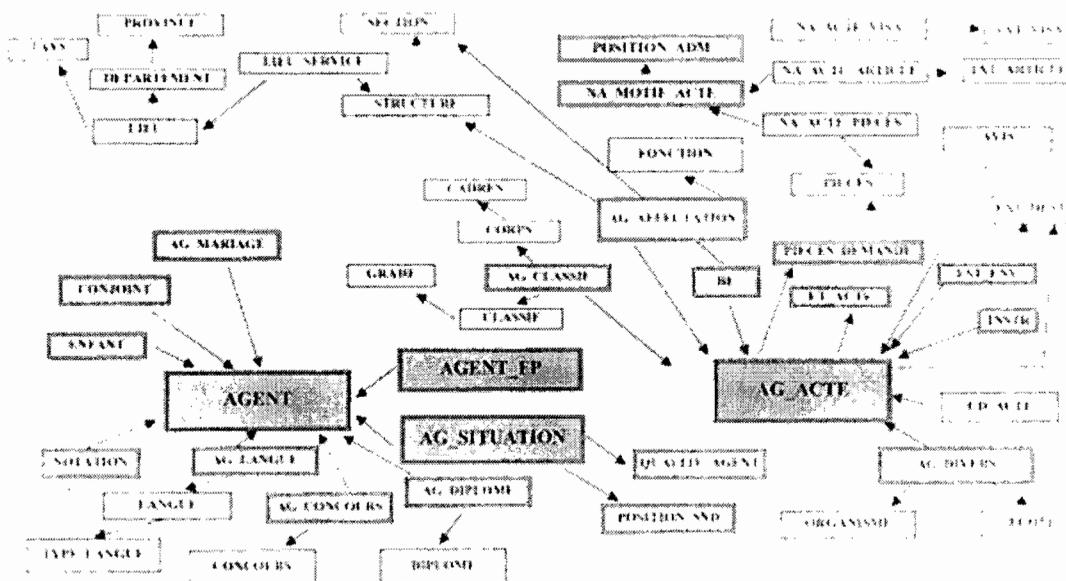
- SIGASPE

Le Ministère de la Fonction Publique et de la Réforme de l'Etat et celui des Finances et du Budget (MFB) ont mis en place un Système Intégré de Gestion Administrative et Salariale du Personnel de l'Etat (SIGASPE). L'un des nombreux avantages de ce système est qu'il offre la possibilité aux usagers de connaître leur situation administrative et l'historique de leurs dossiers en cours de traitement à travers des procédures de consultation prévues à cet effet.

SIGASPE a été développé sous oracle 8i (« *Developer 2000* » d'oracle et le SGBDR oracle 8i) selon le modèle client/serveur. Il s'utilise sur le Réseau Interurbain de l'Administration publique du Burkina Faso (RESINA), par le MFPRE, le MFB et les DAAF (Directeur des Affaires

Administratives et Financières) des ministères via des micro-ordinateurs individuels connectés au réseau RESINA.

Le modèle physique de la base de données



- **RESINA** (Réseau Interurbain de l'Administration publique du Burkina Faso)  
 C'est un réseau fibre optique qui reliera toutes les administrations publiques. A l'état actuellement, RESINA est une interconnexion par fibre optique de trente (30) bâtiments administratifs abritant les départements ministériels et quelques institutions (dont la DELGI, le Ministère de la Fonction Publique et de la Réforme de l'Etat, le Ministère des Finances et du Budget) et une interconnexion de douze (12) villes par liaisons spécialisées.

La gestion prévisionnelle des effectifs du personnel de l'administration publique est améliorée grâce à ce Réseau qui favorisera la promotion du Système Intégré de Gestion Administrative et Salariale du Personnel de l'Etat (SIGASPE).

Pour des raisons de sécurité, RESINA n'a aucune ouverture sur le réseau Internet. Cependant, son extension doit permettre à toute l'administration de se connecter gratuitement à l'Internet et permettre aux fonctionnaires de se connecter depuis leur domicile s'ils sont équipés.

La connaissance de SIGASPE et du RESINA nous a semblé primordiale dans la proposition d'un système acceptable qui réponde aux attentes des décideurs et des usagers du système.

- Diagnostic

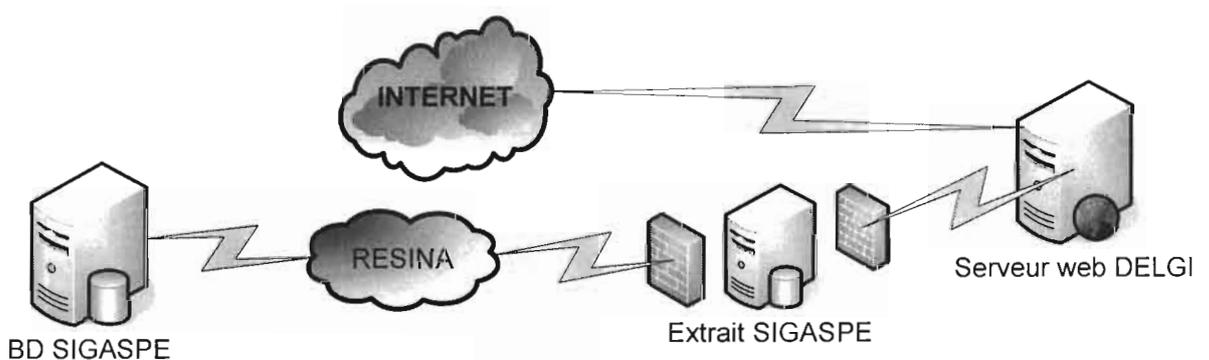
Les utilisateurs souhaitent mettre en ligne des informations contenues dans la base de données SIGASPE. Cependant, ils ne désirent pas une ouverture de cette base de données sur Internet.

Deux solutions qui pourraient satisfaire leurs attentes ont été proposées.

- Première solution

Cette solution consiste à mettre en place, une base de données (un extrait de la base de donnée SIGASPE) ne contenant que les informations à publier. Elle sera localisée sur le serveur de base de données MySQL de la DELGI. L'application Web s'y connectera pour mettre à la disposition des internautes les informations recherchées.

Cette base de données sera automatiquement mise à jour à chaque fois que des ajouts ou des modifications seront apportés à la base de données SIGASPE.



Légende		
Symbole	Total	Description
	1	Serveur Web
	2	Serveur de base de données
	4	Liaison
	1	RESINA ou INTERNET
	2	Pare-feu

### Mise en œuvre

Cette solution ne nécessite l'acquisition d'aucun matériel. En effet, la DELGI dispose d'un serveur Web où sera déployée l'application et d'un serveur de base de données qui hébergera l'extrait de la base de données SIGASPE.

L'extrait de la base de données sera sous MySQL 3.23.54 (Système de Gestion de Base de Données utilisé par la DELGI).

Nous allons également mettre en place une procédure de mise en jour de l'extrait de Base Données, déclenché par un triggers à chaque mise à jour de la base de données SIGASPE.

Ajout de règles au pare-feu logiciel (ajout de linge à la table **lptable**) sur le serveur de Base de données de la DELGI. Par exemple :

- interdire toute sortie vers le serveur de Base de données du SIGASPE ou vers toute autre système de réseau RESINA ;
- aucune connexion initiée de l'extérieur (Internet) ne devra traverser le pare-feu.

L'application Web sera réalisée sous OpenCms en utilisant les technologies J2EE (JSP, Servlet) avec MySQL 3.23.54 comme système de gestion de base de données. Le site Web de la Fonction Publique à travers laquelle l'application sera accessible est sous OpenCms et utilise MySQL 3.23.54 comme SGBDR. D'où le choix de ces outils de développement.

### Avantages :

- les données renvoyées par l'application aux utilisateurs sont à jour ;
- facilité de mise en œuvre de la solution ;
- coût nul (aucun matériel, ni logiciel n'est à acquérir).

### Inconvénient :

- cette solution établit une liaison physique entre SIGASPE et le réseau Internet, ce qui n'est pas souhaitable dans le cas actuel.

– Deuxième solution (solution retenue)

Cette solution a été proposée suite au rejet de la première solution par une partie des utilisateurs, en l'occurrence la DELGI et les cadres du MFPRE (ayant en charge la gestion du système), qui ne souhaitent pas l'ouverture du RESINA vers le réseau Internet (pour l'instant).

Le principe est le suivant :

Il n'aura aucune relation physique entre le serveur de base de données de la DELGI et celui de base de données oracle de SIGASPE.

Un extrait de la base de données sera sur le serveur de base de données MySQL de la DELGI comme dans la précédente solution.

Les mises à jour de l'extrait de la base de données se feront à travers une interface Web que l'application à développer devra offrir.

Nous proposerons un script écrit en PL/SQL pour l'extraction des données de la base de données du SIGASPE vers un fichier XML ; ces fichiers seront transportés au moyen de supports magnétiques (Clé USB, CD-ROM) sur un poste connecté à Internet ; à travers une interface que l'application offrira, les utilisateurs se connecteront pour mettre à jour l'extrait de la base de données.

Avantages :

- satisfaction des deux parties décideurs (DELGI et MFPRE) qui trouvent la solution assez sécurisée ;
- coût nul (aucun matériel ni logiciel à acquérir).

Mise en œuvre

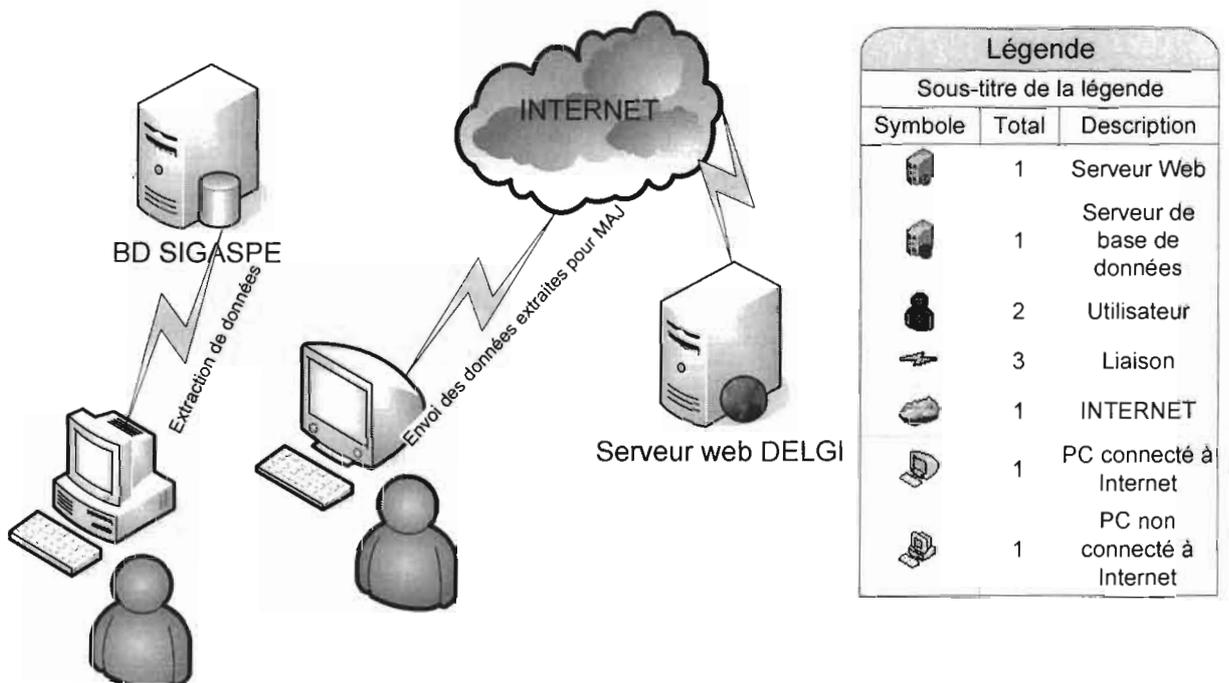
Cette solution ne nécessite l'acquisition d'aucun matériel.

L'extrait de la base de données sera sous MySQL 3.23.54 (Système de Gestion de Base de Données utilisé par la DELGI).

En plus de l'application, nous proposerons un script PL/SQL pour l'extraction de données de la base de données du SIGASPE vers un fichier XML.

L'application Web sera réalisée sous OpenCms en utilisant les technologies J2EE (JSP, Servlet) avec MySQL 3.23.54 comme système de gestion de base

de données. Les informaticiens du domaine d'étude ont demandé une application évolutive vers la solution une.

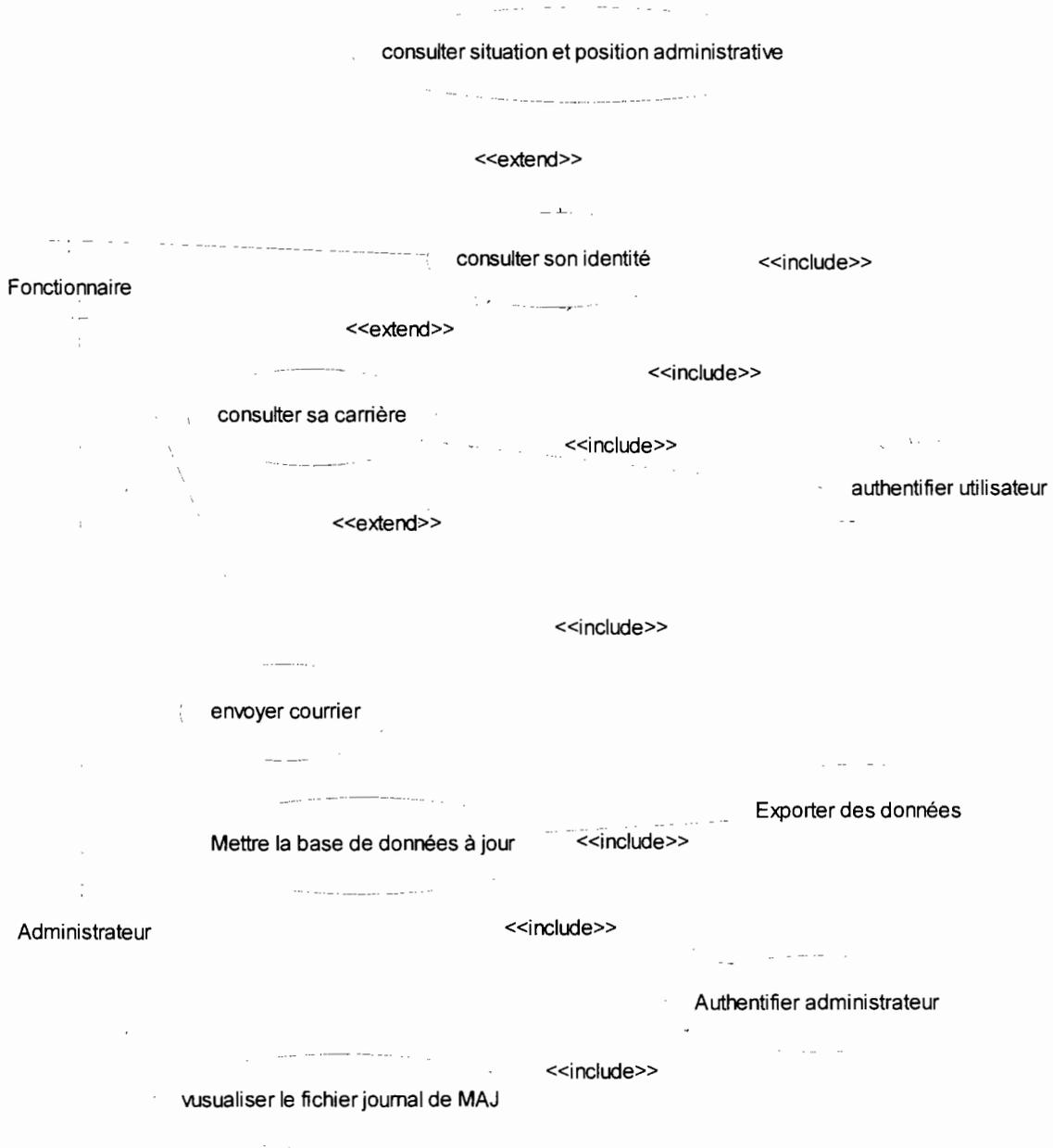


## 1. Diagramme de cas d'utilisation

### a. Les acteurs de notre système et leur rôle

- les acteurs principaux
  - fonctionnaire :
    - il peut consulter son identité ;
    - il peut consulter sa situation et sa position administrative ;
    - il peut consulter sa carrière (engagement, avancement, dossiers en cours de traitement ...) ;
    - il peut envoyer un courrier pour demander des explications ou signaler une anomalie constatée ;
- les acteurs secondaires
  - l'administrateur du système
    - met à jour la base de données ;
    - visualise le fichier journal de mise à jour de la base de données.

b. Le diagramme



**c. La description textuelle des cas d'utilisation**

Titre	Consulter situation et position administrative
Résumé	Ce cas d'utilisation permet aux fonctionnaires de consulter leur situation et position administrative.
Acteurs	Fonctionnaire, Système Informatique (SI)
Date de création	17 octobre 2005
Version	V 1.0
Scénario nominal	1-le fonctionnaire demande à consulter sa situation et sa position administrative; 2-le système demande de s'authentifier 3-l'utilisateur saisit ces références (n° matricule et mot de passe) 4-le système vérifie les références saisies 5-le système affiche les informations relatives à la situation et la position administrative de l'agent (A <sub>1</sub> ) FIN
Scénario alternatif	(A <sub>1</sub> ) : référence non valide Ce scénario commence au point 4 du scénario nominal 5-le SI indique que les références saisies ne sont pas valides Ce scénario reprend au point 2 du scénario nominal

Titre	Consulter sa carrière
Résumé	Ce cas d'utilisation permet aux fonctionnaires de consulter leur carrière (dossiers d'avancement, dossier d'engagement...)
Acteurs	Fonctionnaire, Système Informatique (SI)
Date de création	17 octobre 2005
Version	V 1.0
Scénario nominal	1-le fonctionnaire demande à consulter sa carrière; 2-le système demande de s'authentifier 3-l'utilisateur saisit ces références (n° matricule et mot de passe) 4-le système vérifie les références saisies 5-le système affiche les informations relatives à la carrière de l'agent (A <sub>1</sub> ) FIN
Scénario alternatif	(A <sub>1</sub> ) : référence non valide Ce scénario commence au point 4 du scénario nominal 5-le SI indique que les références saisies ne sont pas valides ; Ce scénario reprend au point 2 du scénario nominal

Titre	Consulter son identité
Résumé	Ce cas d'utilisation permet aux fonctionnaires de consulter leurs identités, leur carrière, leur situation et position administrative.
Acteurs	Fonctionnaire, Système Informatique (SI)
Date de création	17 octobre 2005
Version	V 1.0
Scénario nominal	<p>1-le fonctionnaire demande à consulter ses identités;                  2-le système demande de s'authentifier ;                  3-l'utilisateur saisit ces références (n° matricule et mot de passe) ;                  4-le système vérifie les références saisies ;                  5-le système affiche les informations demandées ;(A<sub>1</sub>)                  6-le fonctionnaire demande à consulter sa carrière ; (A<sub>2</sub>)</p> <p>7- le système affiche les informations relatives à la carrière de l'agent ;                  8-le fonctionnaire demande à quitter l'application ;                  9-le système met fin à la session</p> <p>FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : référence non valide                  Ce scénario commence au point 4 du scénario nominal                  5-le SI indique que les références saisies ne sont pas valides ;                  Ce scénario reprend au point 2 du scénario nominal</p> <p>(A<sub>1</sub>) :l'utilisateur fait un autre choix                  6-le fonctionnaire demande à consulter sa situation et sa position administrative;                  7- le système affiche les informations relatives à la carrière de l'agent ;</p> <p>(A<sub>2</sub>) :l'utilisateur fait un autre choix                  6-le fonctionnaire demande à envoyer un mail à l'administrateur du système;                  7-le système affiche l'interface de saisie de pour courrier électronique ;                  8-le fonctionnaire saisit sous courrier et valide ;                  9-le système envoie le courrier saisi ;                  Ce scénario reprend au point 6 du scénario nominal</p> <p>(A<sub>1</sub>) :l'utilisateur décide de mettre fin à sa session                  6-le fonctionnaire demande à mettre fin à sa session de consultation;                  7-le système met fin à la session ;                  FIN</p>

Titre	Mettre à jour la base de données
Résumé	Ce cas d'utilisation permet à l'administrateur de mettre à jour la base de données. En effet, il s'agit de l'extrait de la base de données SIGASPE qui doit être périodiquement mise en jour avec des données en provenance de la base SIGASPE (sous forme de fichier xml)
Acteurs	Fonctionnaire, Système Informatique (SI)
Date de création	17 octobre 2005
Version	V 1.0
Scénario nominal	<p>1-l'administrateur saisit ses références (mot de passe et nom utilisateur) pour se connecté au système ;</p> <p>2-le système vérifie la validité des références saisies ;</p> <p>3-le système affiche les tâches d'administration que peut effectuer l'administrateur ; (A<sub>1</sub>)</p> <p>4-l'administrateur saisit le chemin d'accès aux fichiers qui serviront à la mise de la base de données ;</p> <p>5-le système vérifie l'existence des fichiers ;</p> <p>6-le système télécharge les fichiers sur le serveur ; (A<sub>2</sub>)</p> <p>7-le système vérifie la validité des fichiers ; (A<sub>3</sub>)</p> <p>8-le système informe que les fichiers ont été téléchargés et sont valides, il rend la main à l'administrateur ; (A<sub>4</sub>)</p> <p>9-l'administrateur demande d'effectuer la mise à jour de la base de données ;</p> <p>10-le système met à jour les données et crée un fichier journal de mise à jour (ce journal contient les informations telles que : la durée de l'opération, les éventuellement problèmes survenus lors de l'opération);</p> <p>11-l'administrateur demande à visualiser le journal de mise à jour ;</p> <p>12-le système affiche le journal de mise à jour ;</p> <p>13-l'administrateur demande à mettre fin à sa session de travail ;</p> <p>14-le système ferme sa session ;</p> <p>FIN</p>
Scénario alternatif	<p>(A<sub>1</sub>) : référence non valide Ce scénario commence au point 4 du scénario nominal 3-le SI indique que les références saisies ne sont pas valides ; Ce scénario reprend au point 2 du scénario nominal</p> <p>(A<sub>2</sub>) : fichier(s) introuvable(s) ou inaccessibles 6-le SI informe que les fichiers à télécharger sont inaccessibles ou introuvables ; Ce scénario reprend au point 4 du scénario nominal</p> <p>(A<sub>3</sub>) : problème de téléchargement 7-Le SI informe que les fichiers ne peuvent pas être téléchargés pour des raisons de connexion ou de taille (fichiers trop volumineux) Ce scénario reprend au point 3 du scénario nominal</p> <p>(A<sub>4</sub>) : format de fichiers incorrects ou fichiers mal formés 8-le SI informe que des fichiers sont au format non attendu (format non XML) ou sont mal formés (fichiers XML ne respectant la grammaire prédéfinie) Ce scénario reprend au point 3 du scénario nominal</p> <p>FIN</p>

## 2. Planning

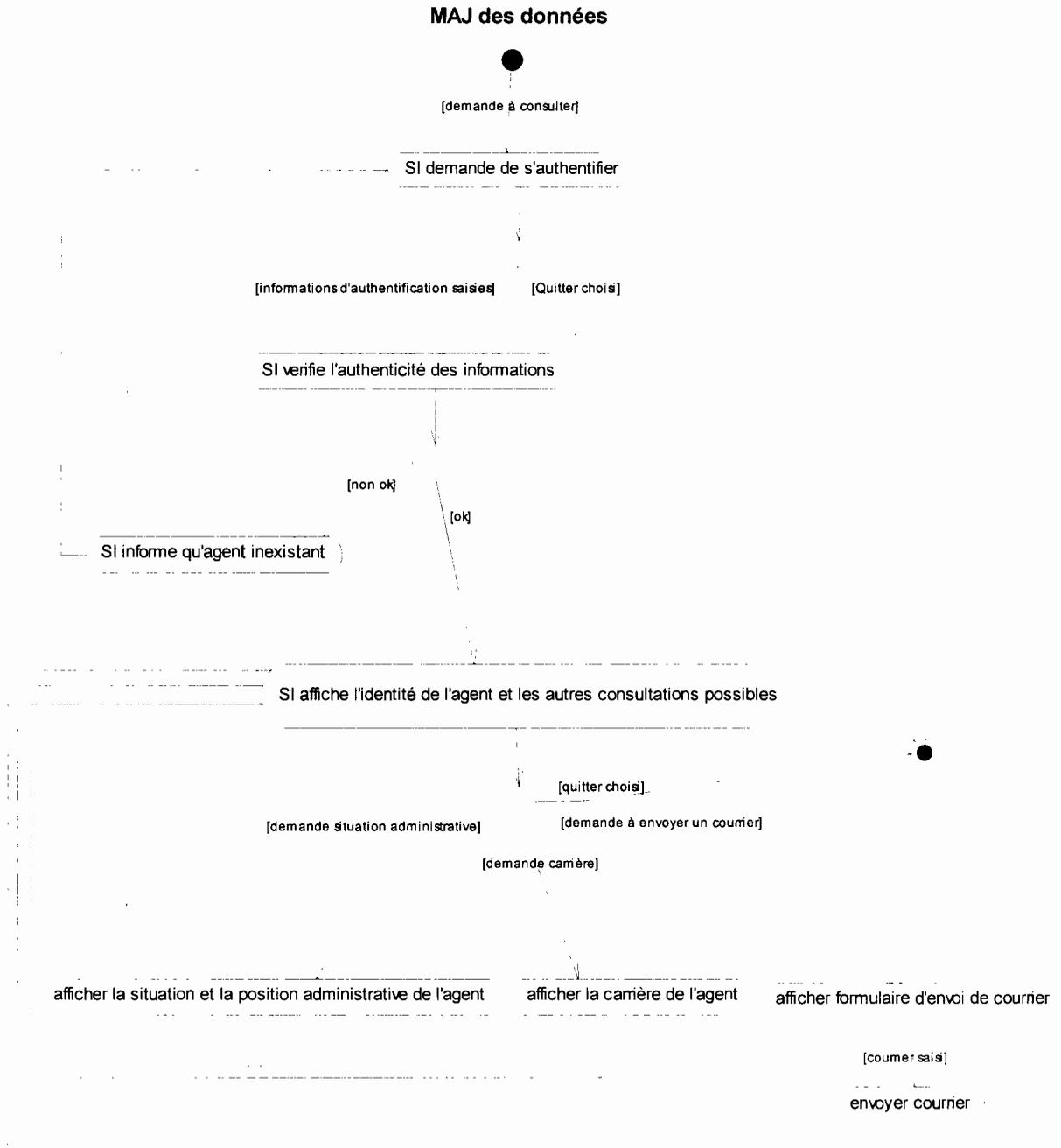
Les cas d'utilisation ont permis de dégager les fonctionnalités du futur système et d'évaluer le travail à réaliser. Un planning de réalisation peut donc être élaboré.

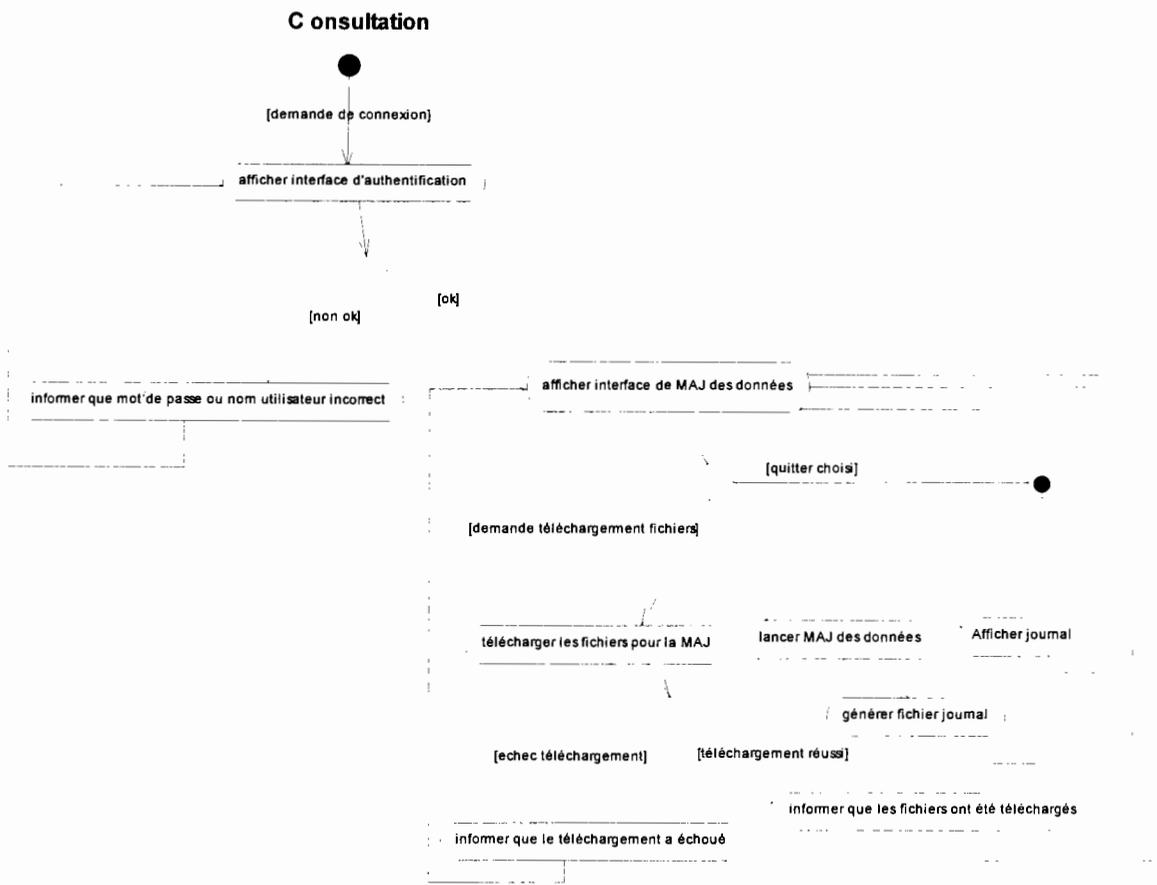
<b>Etape</b>	<b>Période</b>	<b>Durée (semaine)</b>
Découverte des besoins	15– 21 octobre 2005	1
Analyse	22 – 04 novembre 2005	2
Conception	05 – 18 novembre 2005	2
implémentation	18 novembre -11 décembre 2005	3

## II. ANALYSE

### 1. Diagramme d'activité

Ce diagramme permettra de mieux décrire le comportement d'une méthode ou d'un cas d'utilisation. Il nous permettra d'avoir une meilleure représentation des enchaînements des tâches réalisées pour remplir une fonction.





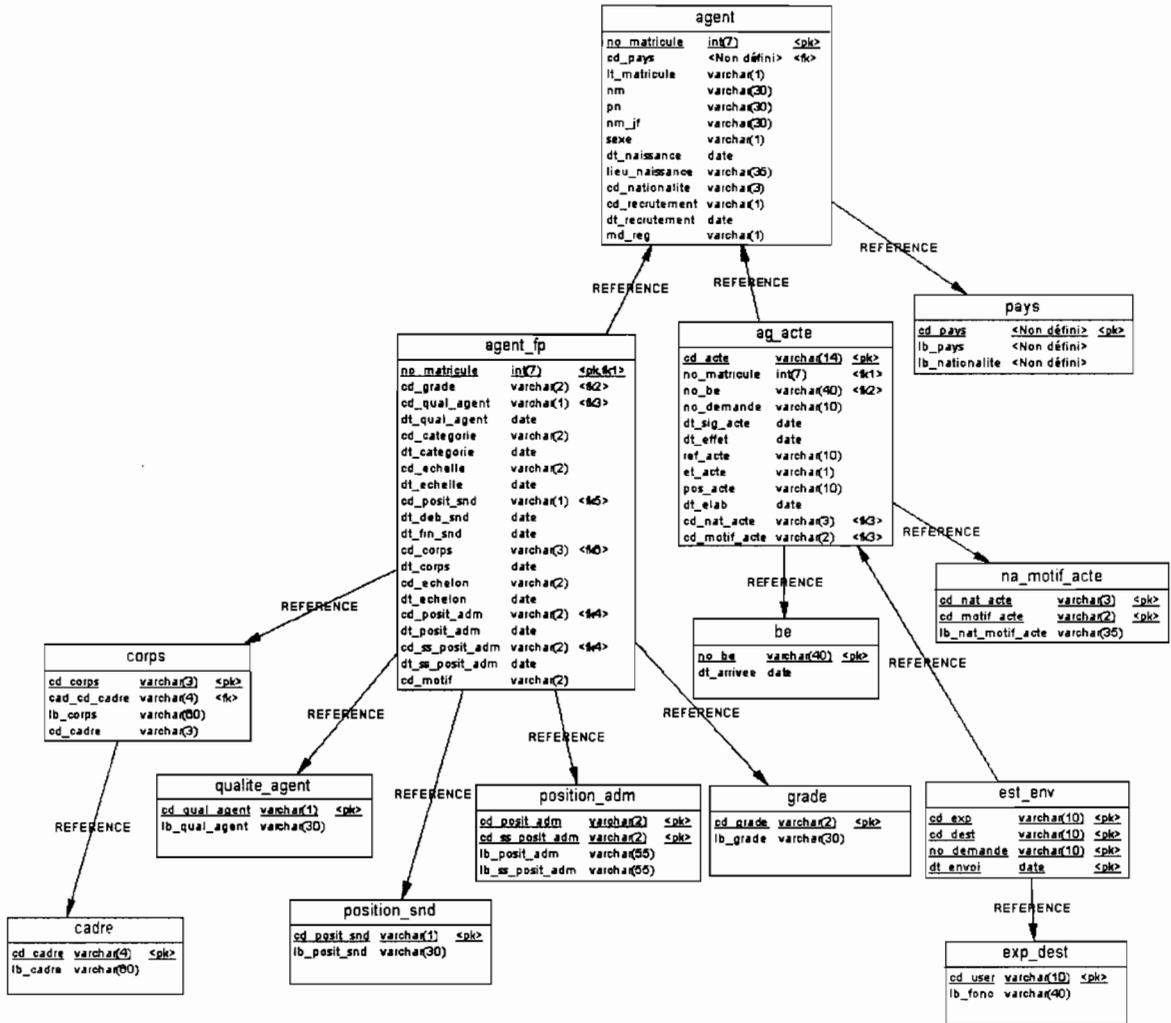
## 2. Diagramme de classe

Un diagramme de classe ne sera pas utile dans cette application. En effet, l'application se limitera à envoyer des requêtes SQL à la base de données pour pouvoir afficher les données dans les formulaires.

Les classes standard du J2SDK (Java 2 Standard Kit Development) et en particulier les classes de l'API (Application Programming Interface) JDBC (Java Database Connectivity) et les classes du J2EE (Java 2 Entreprise Edition) suffiront largement pour la réalisation de l'application. Néanmoins, nous aurons à développer quelques classes d'implémentations décrites dans la javadoc. Ce sont essentiellement les classes qui serviront à envoyer un courrier électronique et à mettre à jour la base de données.

### 3. Modèle physique de données

C'est un extrait du modèle physique de données de SIGASPE.



#### 4. Structure des fichiers XML

La mise à jour de la base de données sous MySQL se fera à partir de fichiers XML. Ces fichiers contiendront les nouveaux enregistrements (ou enregistrements modifiés) dans la base de données SIGASPE. Ces fichiers XML respectent la grammaire ci-après.

##### o Grammaire

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<!ELEMENT attribut (cellule+)>
```

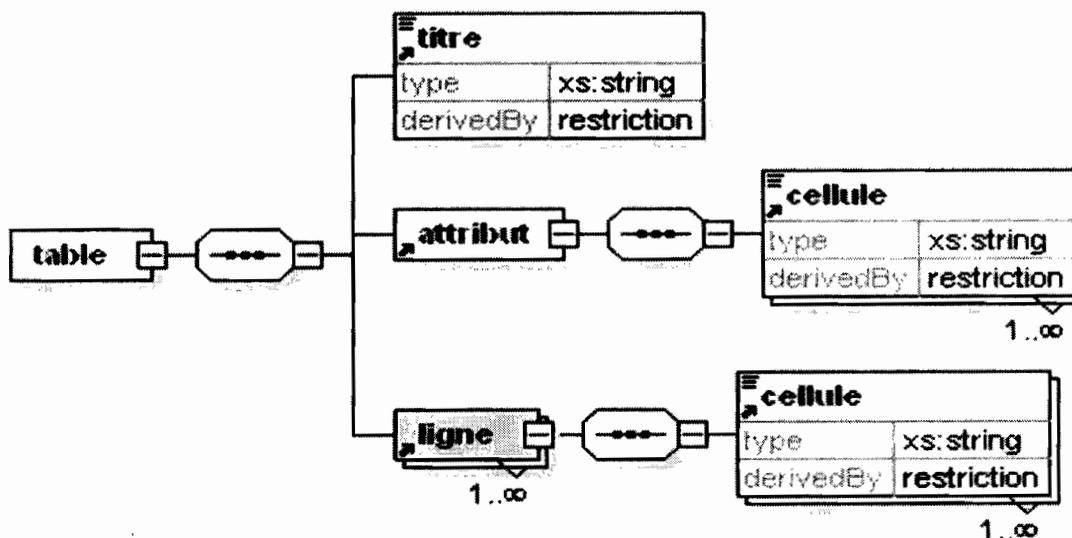
```
<!ELEMENT cellule (#PCDATA)>
```

```
<!ELEMENT ligne (cellule+)>
```

```
<!ELEMENT table (titre, attribut, ligne+)>
```

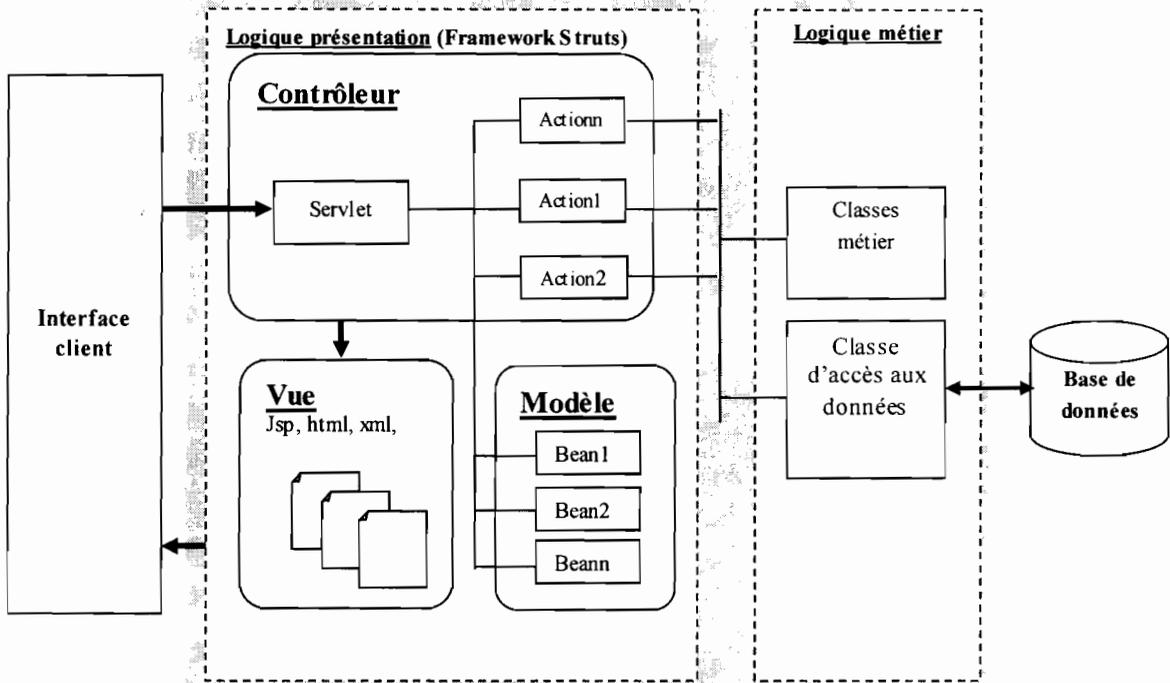
```
<!ELEMENT titre (#PCDATA)>
```

##### o Schéma



### III. CONCEPTION

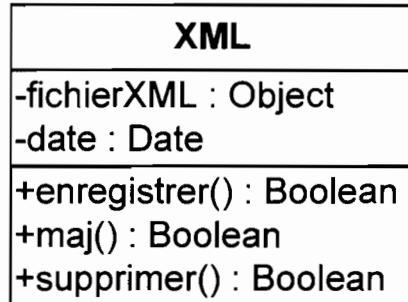
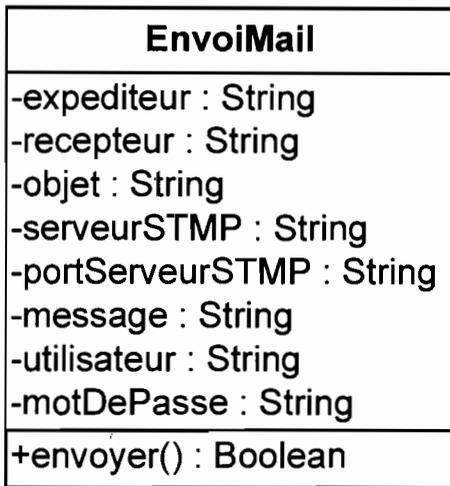
#### 1. Architecture du système



#### Les composants MVC de l'application

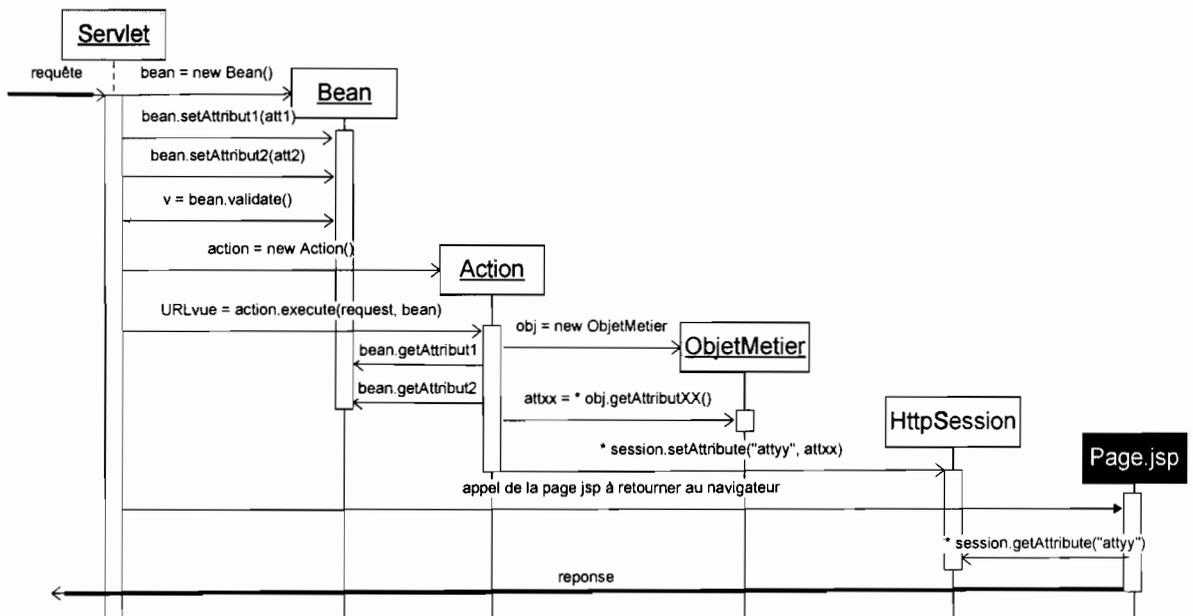
Vue (JSP)	Bean	Action
connexion.jsp	x	connexionAction
identite_agent.jsp	identite_agentBean	identite_agentAction
situation.jsp	situationBean	situationAction
carriere.jsp	carriereBean	carriereAction
consultation.jsp	consultationBean	consultationAction
courrier.jsp	courrierBean	courrierAction
p_administration.jsp	p_administrationBean	p_administrationAction
p_majbd.jsp	p_majbdBean	p_majbdAction

## 2. Diagramme de classe d'implémentation

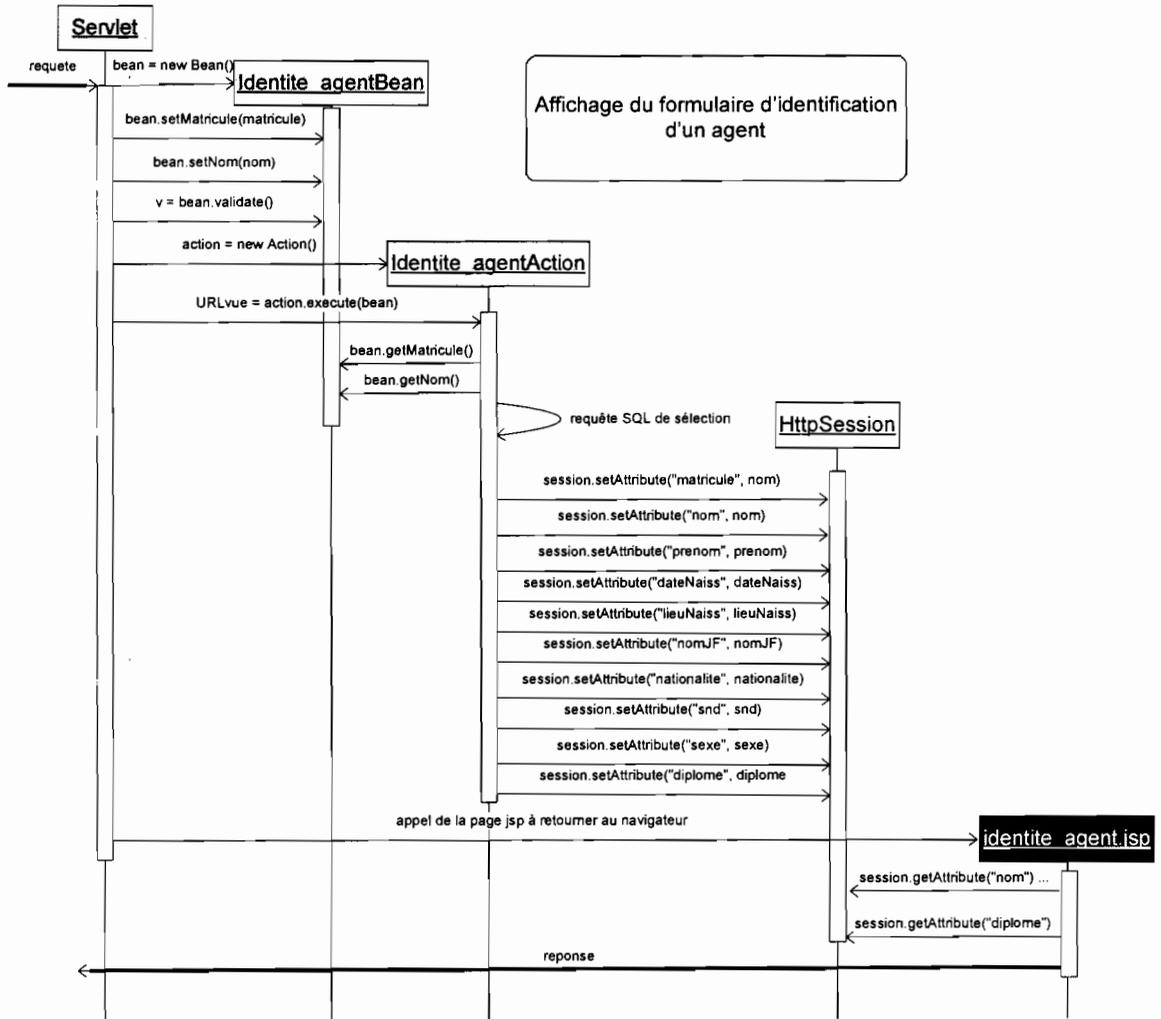


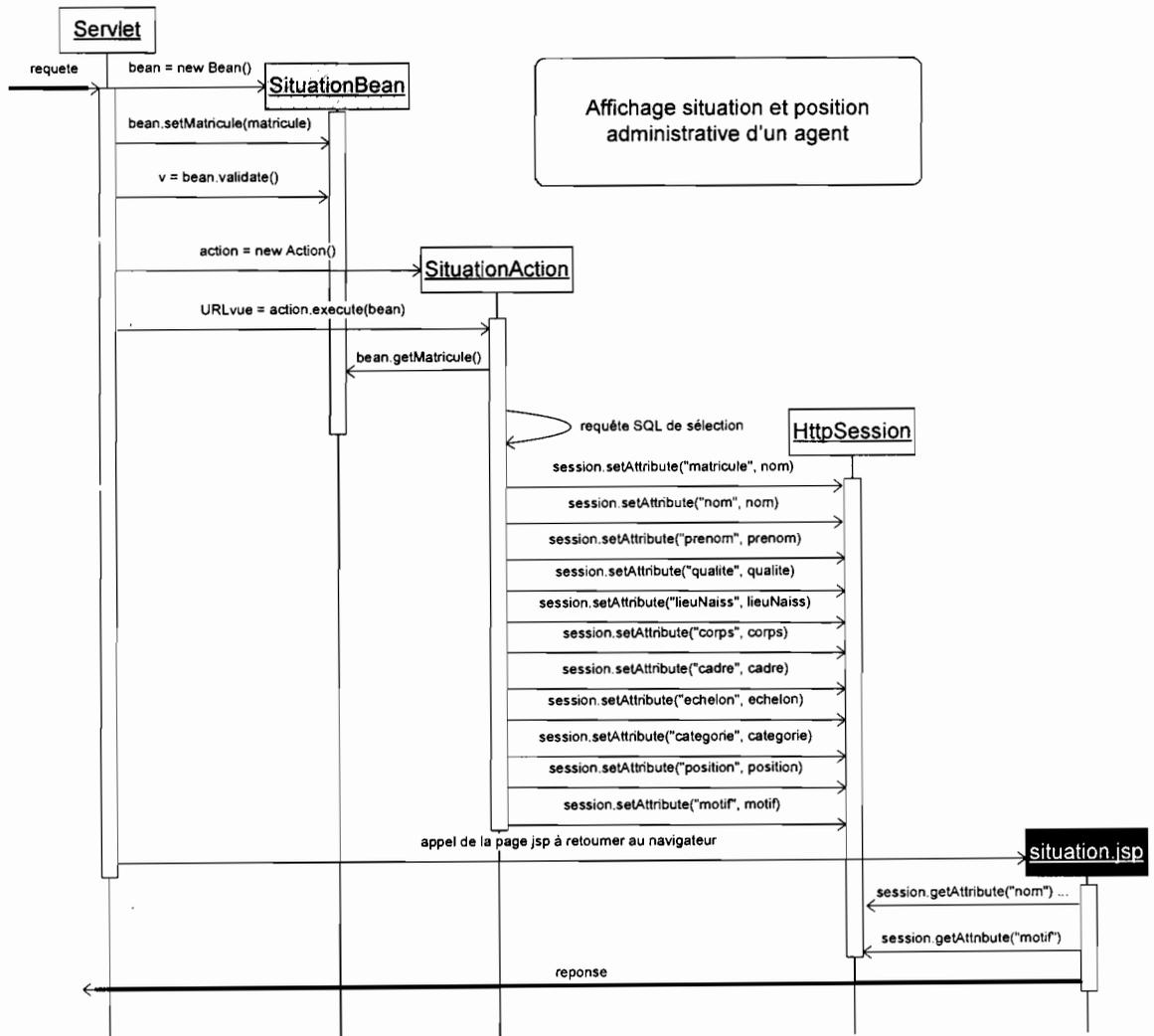
## 3. Diagramme de séquence

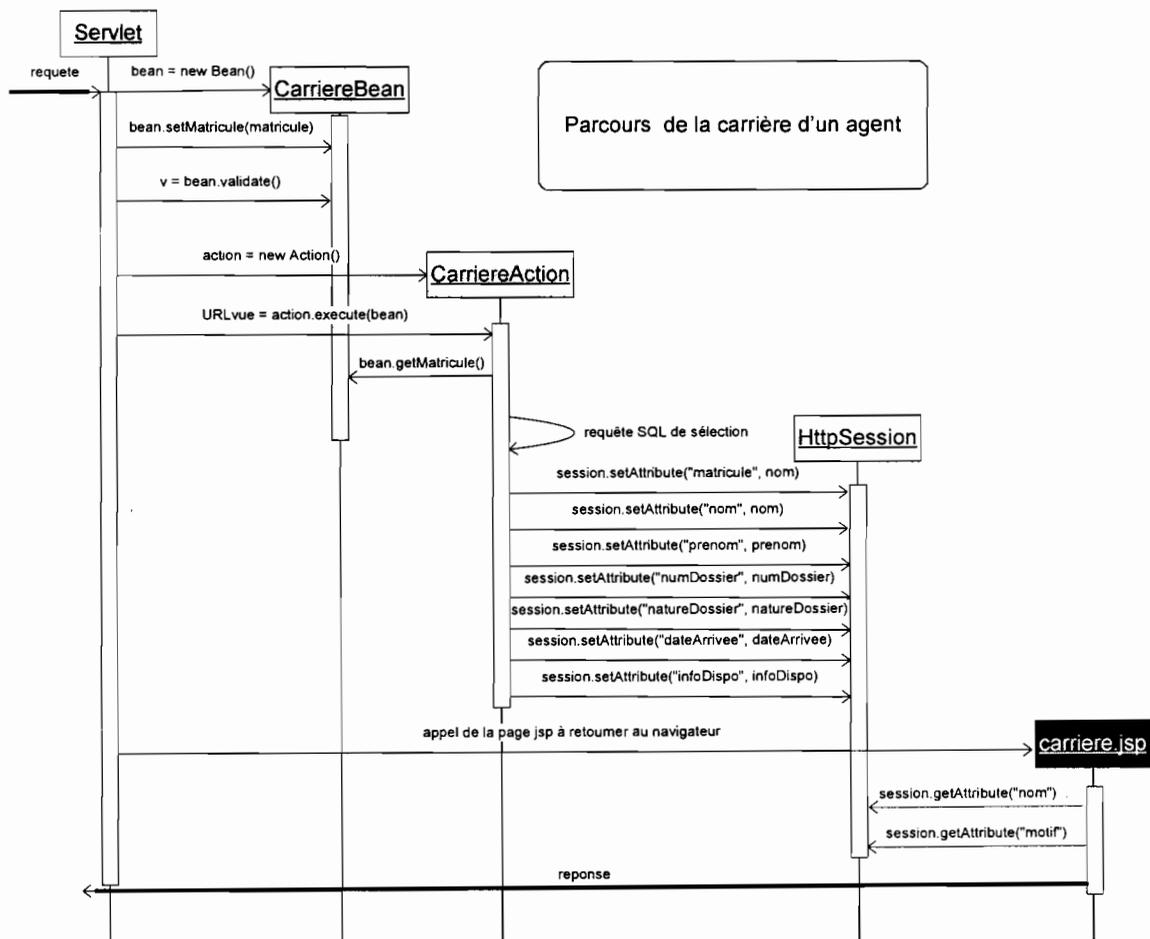
### a. Le diagramme de séquence générale

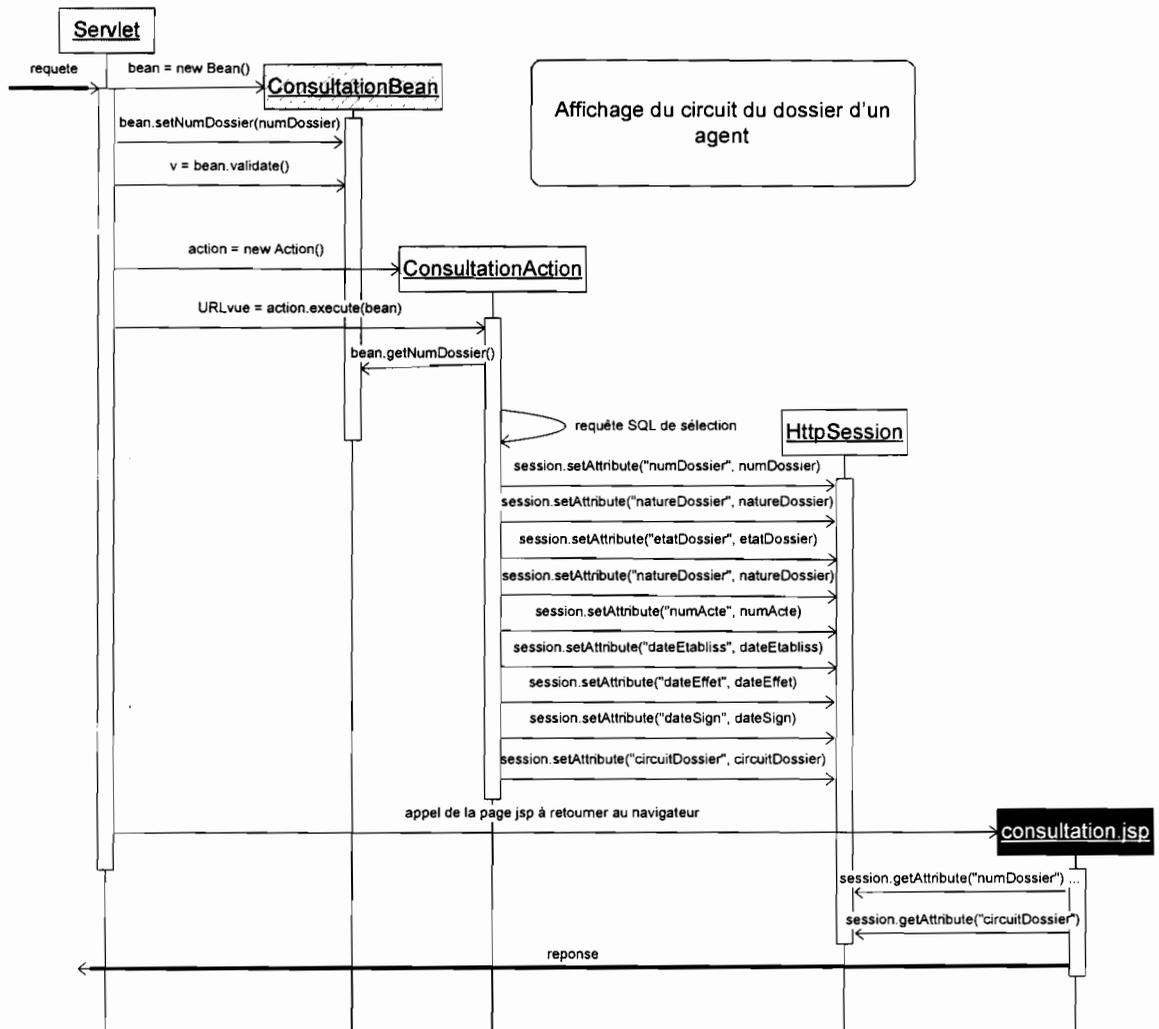


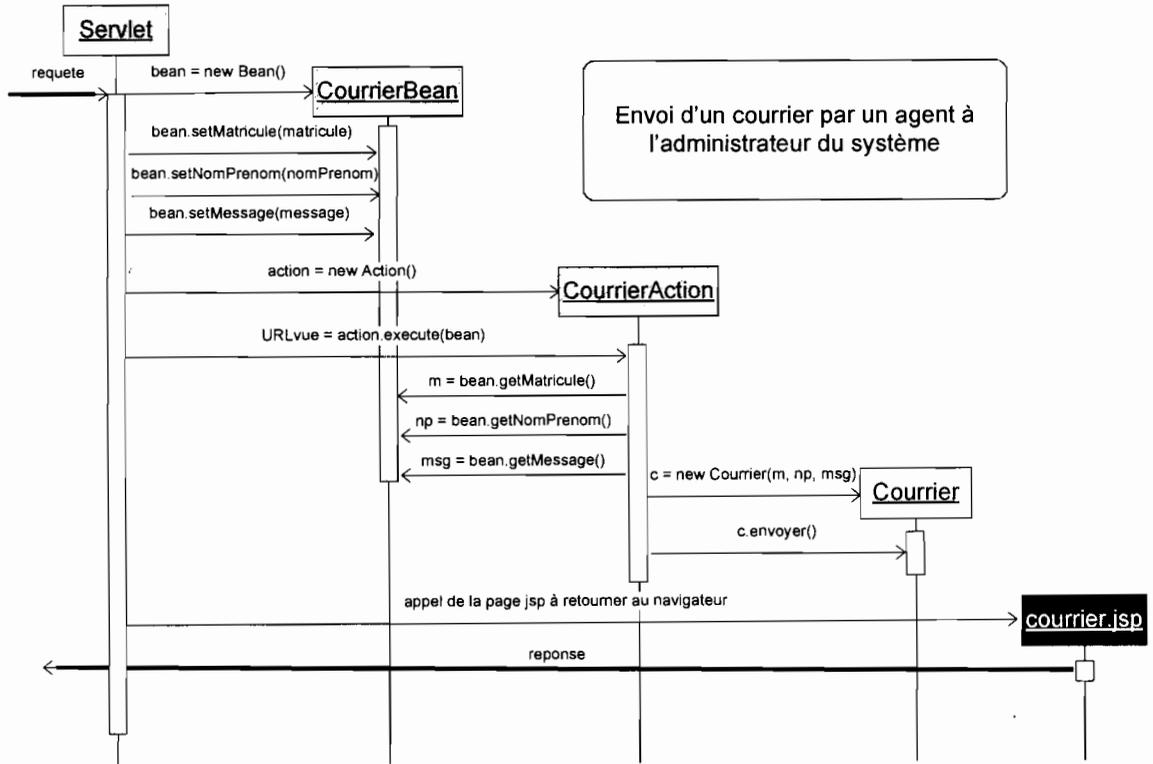
b. Quelques exemples

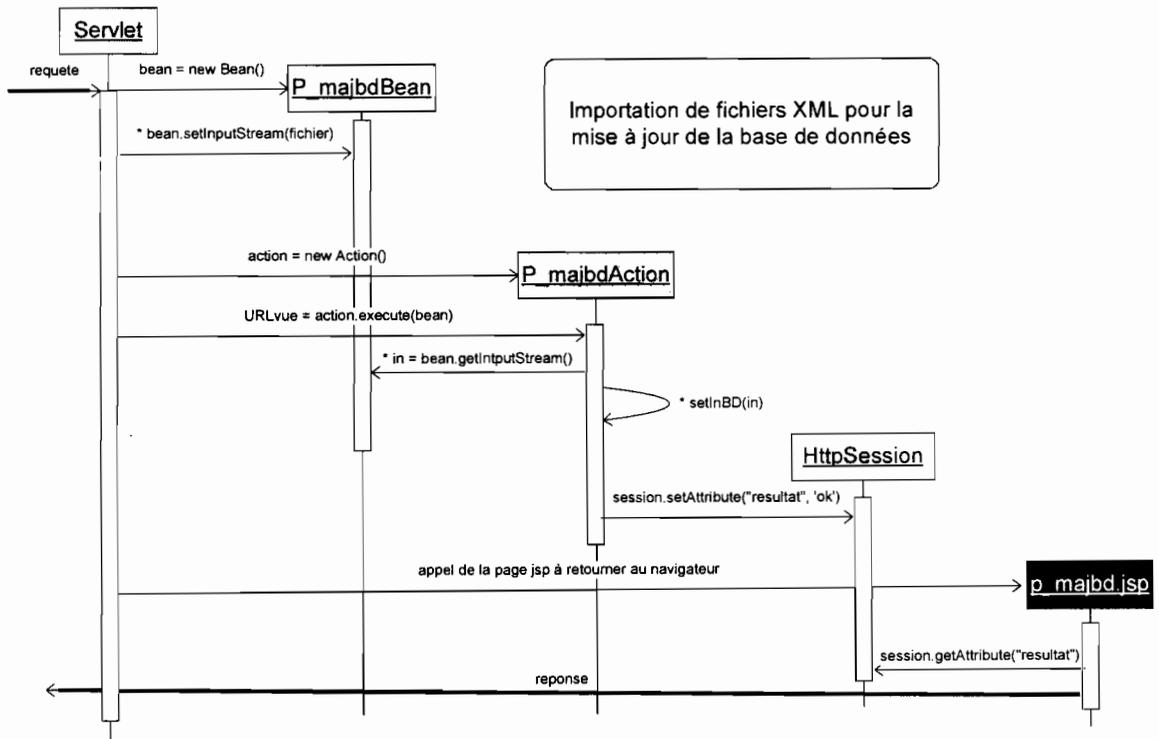


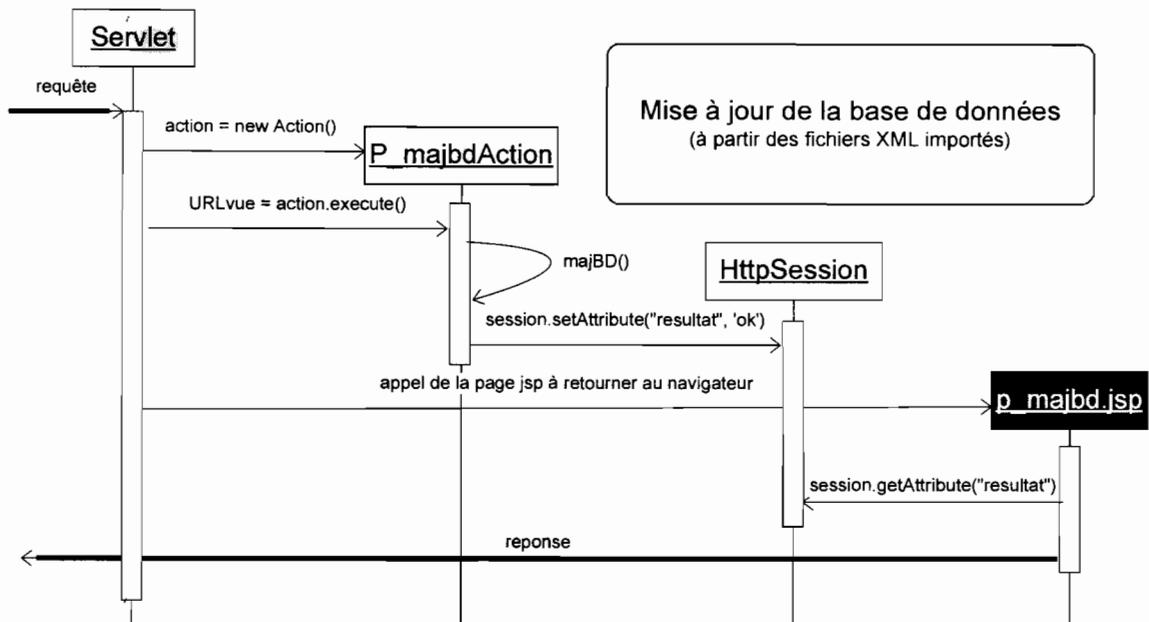




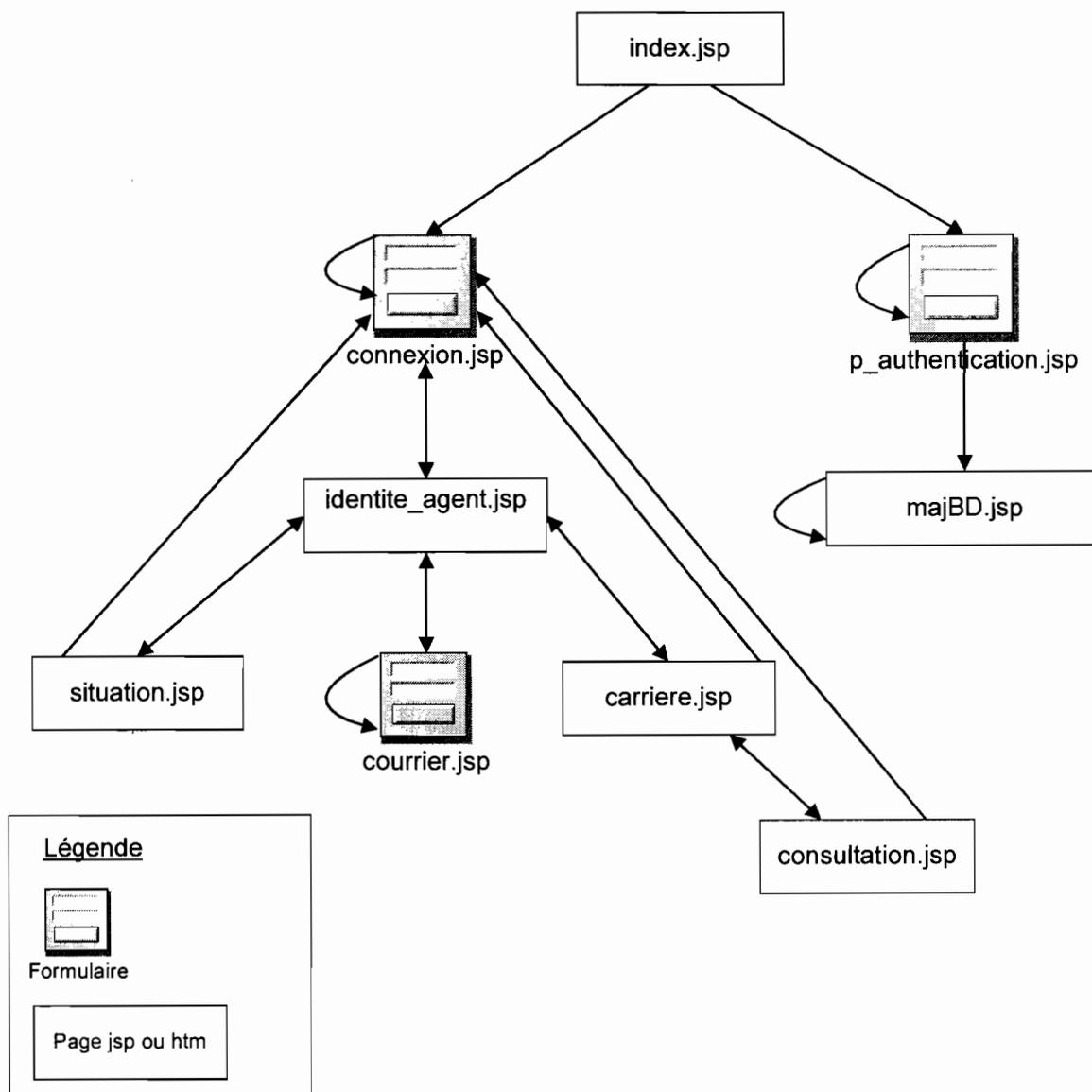




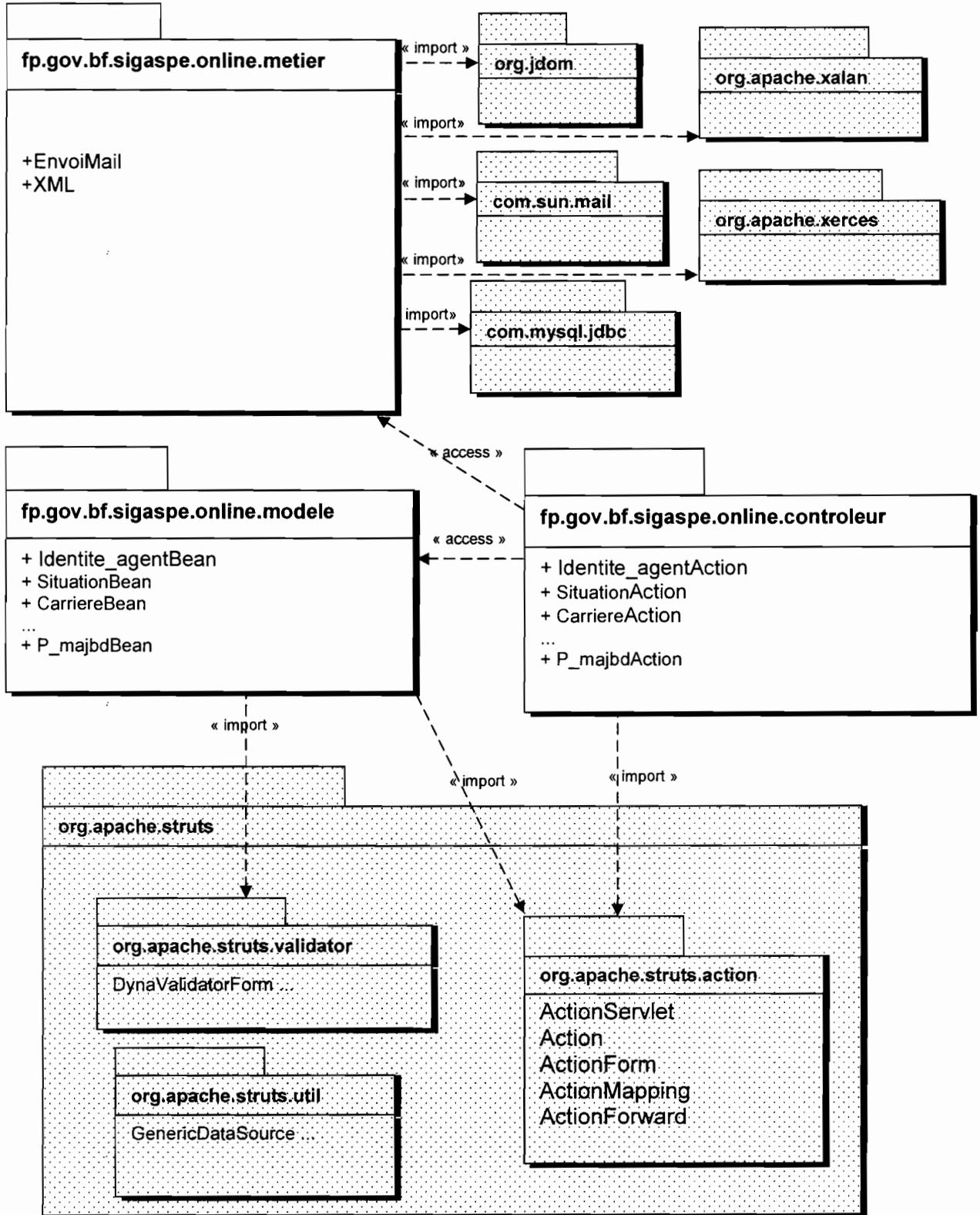




#### 4. Diagramme d'enchaînement des pages Web

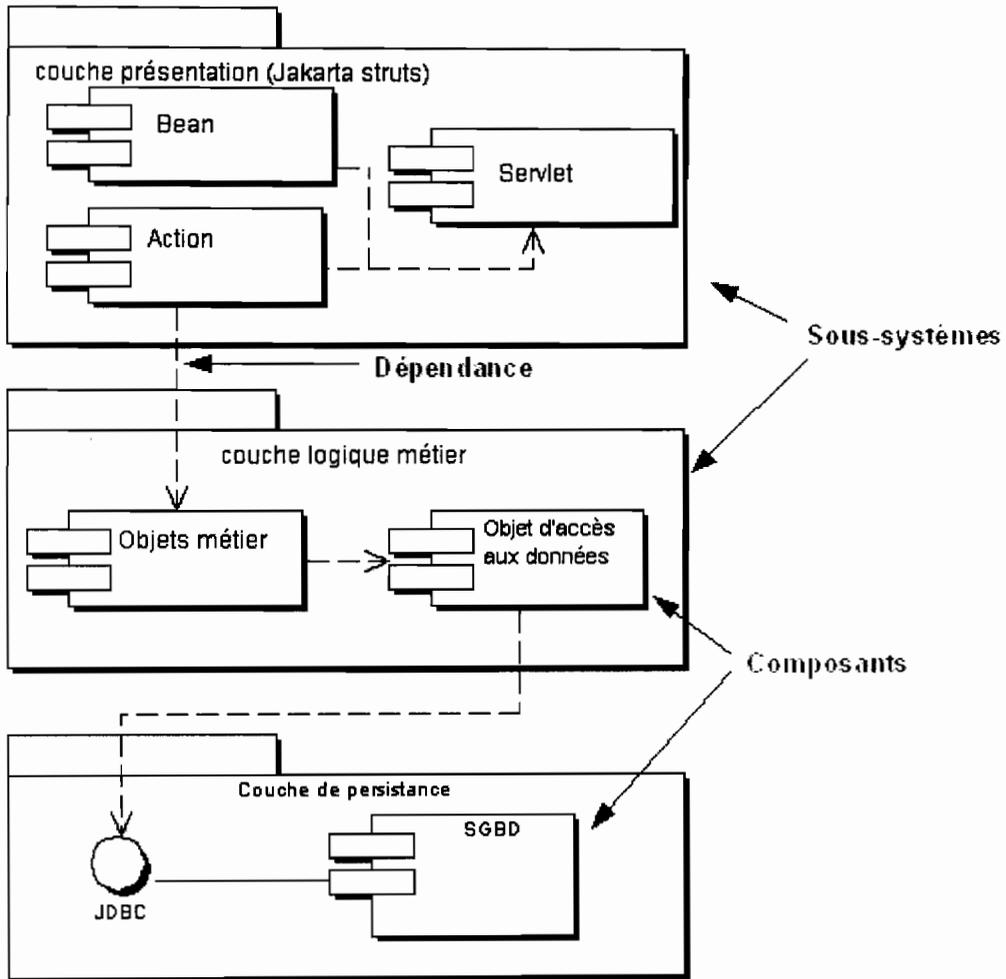


### 5. Diagramme de paquetage



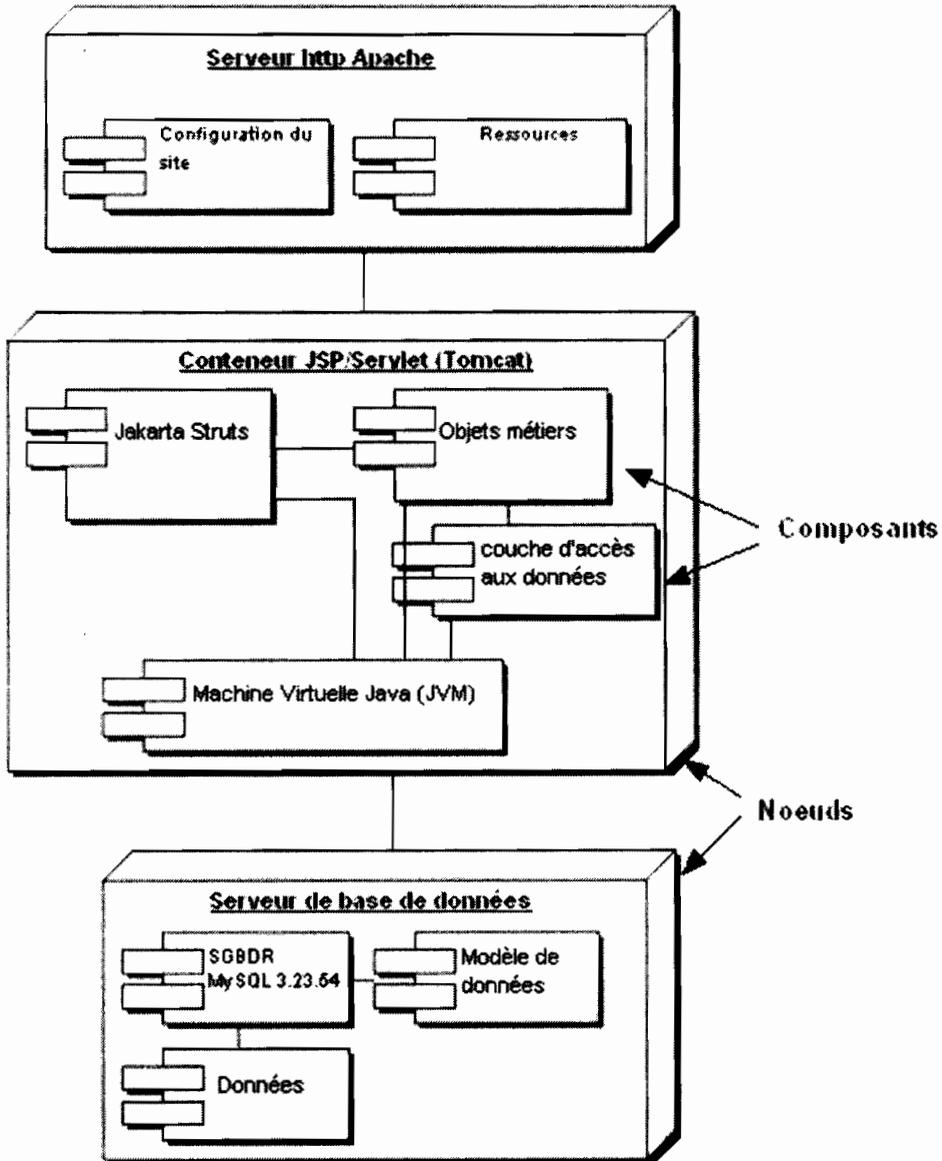
- Paquetage télécharger sur Internet, non disponible dans le J2sdk
- Paquetage développé, propre à l'application

## 6. Diagramme de composants



## IV. IMPLEMENTATION

### 1. Diagramme de déploiement



## 2. Procédure d'installation de l'application

L'application a été testée sous Windows XP service pack 2 et sous la redHat Entreprise WS avec les logiciels suivant :

- Apache Tomcat 5.0.27 : téléchargeable gratuitement à l'url <http://archive.apache.org/dist/tomcat/tomcat-5/archive/v5.0.27/>
- j2sdk 1.4.2 : téléchargeable gratuitement sur <http://java.sun.com/j2se/1.4.2/download.html>
- MySQL 3.23.54 et 4.1.9 téléchargeable gratuitement sur <http://dev.mysql.com/downloads/mysql/4.0.html>
- OpenCms 5.0.1 également téléchargeable gratuitement sur <http://www.opencms.org/opencms/en/download/archive.html>

Pour faciliter l'installation de l'application, nous avons regroupé les fichiers d'installation dans un répertoire que nous avons nommé *sigaspe-fonction-publique* dont le contenu est le suivant :

- *sigaspe.zip* : contient les pages dynamiques jsp, les images, les pages statiques html et les fichiers de configuration de l'application ;
- *bibliotheque* : c'est le répertoire qui contient toutes les bibliothèques (.jar) nécessaire au fonctionnement de l'application ;
- *script-sql.sql* : ce fichier texte contient les instructions SQL pour la création de la base de données.

Après avoir installé les logiciels ci-dessus (Apache Tomcat, j2sdk, MySQL, OpenCms) :

- lancer OpenCms ;
- importer le fichier compressé *sigaspe.zip* en demandant sa décompression (l'application est ainsi installée, il ne reste plus qu'à le configurer) ;
- après la décompression, OpenCms crée un répertoire nommé *sigaspe* contenant les fichiers décompressés. Ouvrez le fichier *sigaspe/config.xml* pour modifier le chemin d'accès à la base de données, définir le nom et le mot de passe de l'utilisateur MySQL qui a droit à la base de données de l'application (par défaut l'utilisateur est *root* sans mot de passe, *jdbc:mysql://localhost/bd\_concours* est le

chemin d'accès à la base de donnée et *com.mysql.jdbc.Driver* le pilote JDBC pour MySQL);

- copier les fichiers du répertoire *bibliotheque* dans le répertoire *lib* d'OpenCms ;
- créer la base de données en exécute les instructions SQL contenues dans *script-sql.sql*.

Redémarrer votre serveur Apache Tomcat et vérifier que l'application fonctionne en invoquant sur votre serveur l'url <http://127.0.0.1:8080/sigaspe/index.jsp>.

## CONCLUSION

L'inscription en ligne aux concours de la Fonction Publique, la consultation des situations administratives et carrière des agents de la Fonction Publique font partir des nombreux services en ligne que l'administration publique burkinabé compte proposer aux citoyens. J'ai été très honoré de participer à leur réalisation.

Ce stage d'ingénieur a été très enrichissant car il m'a permis d'approfondir mes connaissances de l'« l'approche objet », du « design pattern MVC », de la « framework Struts », et surtout sur le développement web coté serveur et client.

A l'issue de ce stage, nous laissons à la Délégation Générale à l'Informatique une application Web d'« inscription en ligne aux concours de la fonction publique » réalisée à 90% et une autre pour la « consultation des situations, positions et carrière des agents de la Fonction Publique » entièrement développée. Nous espérons que ces deux logiciels pourront d'être utilisés pour le bien des citoyens (fonctionnaires et futurs candidats aux concours de la Fonction Publique).

## ANNEXES

## ANNEXE-1 : QUELQUES ECRANS DES INSCRIPTIONS EN LIGNE

## – Page d'accueil

**Liste des concours ouverts**

**Comment s'inscrire au concours**  
 Cette page donne un aperçu des concours auxquels vous pouvez vous inscrire « en ligne ». Avant de procéder à votre inscription électronique il est important que vous lisiez attentivement l'avis de concours. Vous le trouverez en cliquant sur l'intitulé du concours. [\[suite\]](#)

**Personnes déjà inscrites**  
 Les personnes déjà inscrites ont la possibilité de suivre leur dossier afin de savoir si leur inscription a été acceptée, vous avez également la possibilité de réimprimer un récépissé d'inscription. [Cliquez ici pour consulter votre dossier \(réservé aux personnes déjà inscrites\)](#)

**Concours direct**

Intitulé du Concours	Session	Date de début des inscriptions	Date de fin des inscriptions	Inscription
<a href="#">Agents Techniques Agriculture Spécialisés</a>	2005	01-10-2005	01-10-2005	<input type="button" value="S'inscrire"/>
<a href="#">Techniciens Supérieurs en Pédagogie</a>	2005	01-10-2005	01-10-2005	<input type="button" value="S'inscrire"/>
<a href="#">Agents Techniques Elevage</a>	2005	01-09-2005	01-09-2005	<input type="button" value="S'inscrire"/>

**Concours professionnel**

**Ministère de ESSRS**

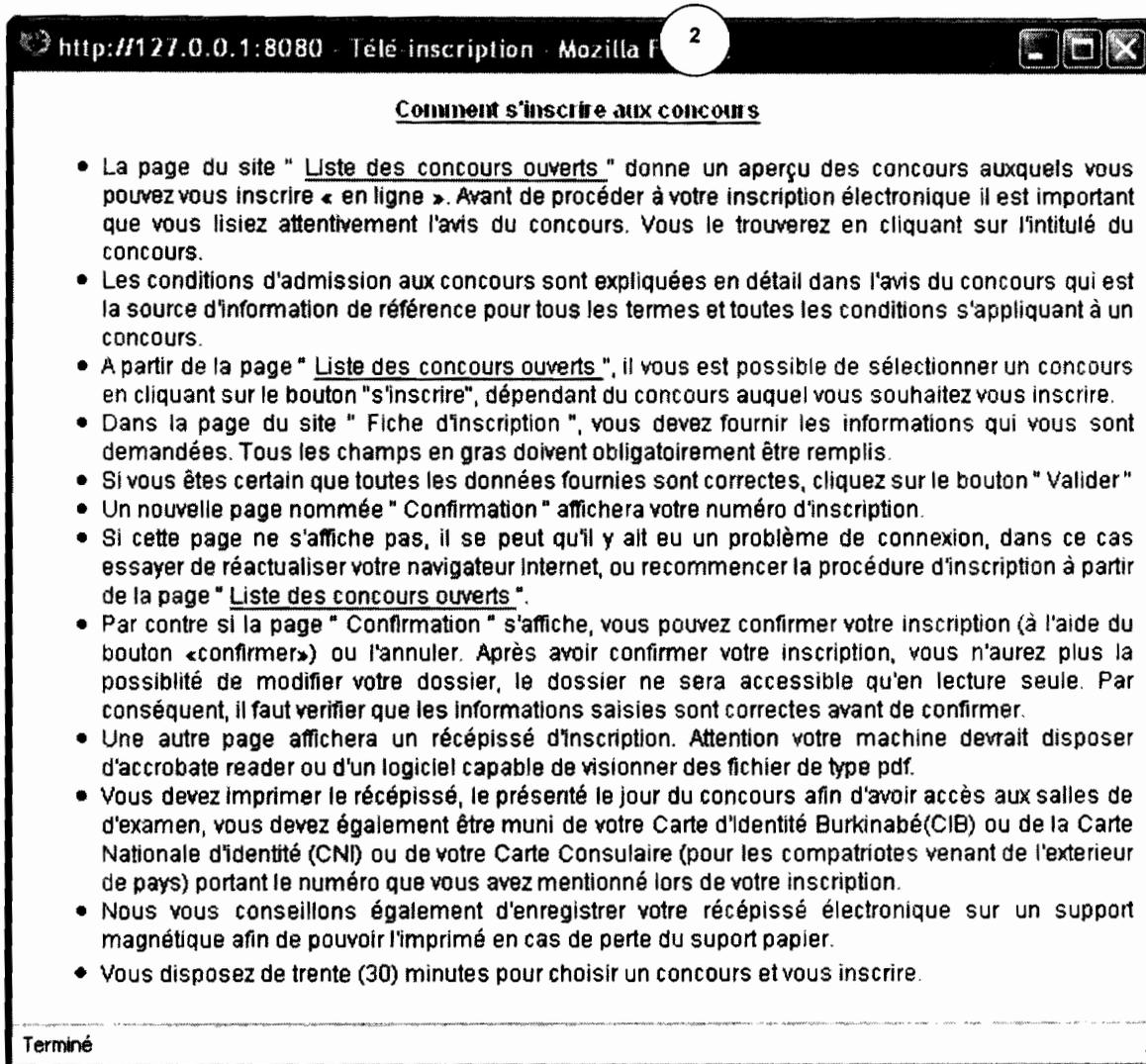
Intitulé du Concours	Session	Date de début des inscriptions	Date de fin des inscriptions	Inscription
<a href="#">Assistants FJA</a>	2005	01-07-2005	01-07-2005	<input type="button" value="S'inscrire"/>

**Ministère de la santé**

Intitulé du Concours	Session	Date de début des inscriptions	Date de fin des inscriptions	Inscription
<a href="#">Agents GSP</a>	2005	01-09-2005	01-09-2005	<input type="button" value="S'inscrire"/>

Terminé

– Description de la procédure d'inscription

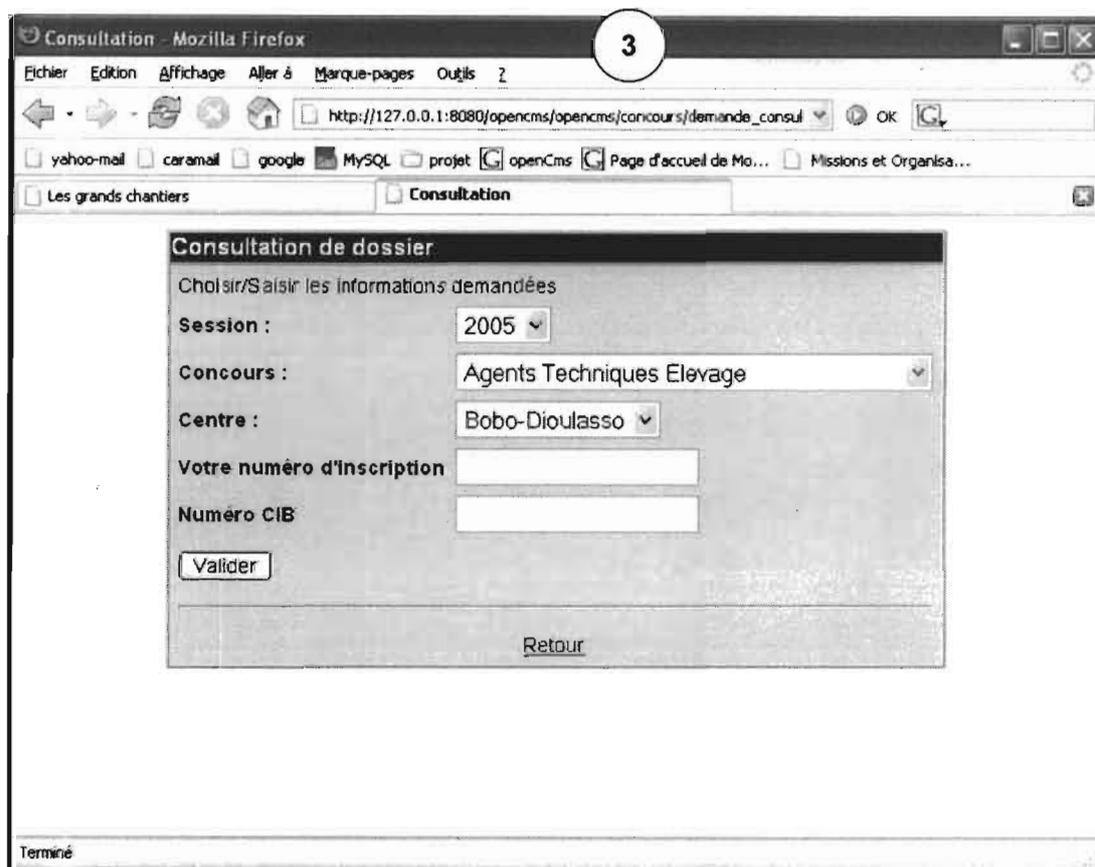


The screenshot shows a web browser window with the address bar containing 'http://127.0.0.1:8080 - Télé inscription - Mozilla F'. The page title is 'Comment s'inscrire aux concours'. The content consists of a list of instructions for registration, followed by the word 'Terminé' at the bottom left of the page content area.

**Comment s'inscrire aux concours**

- La page du site "Liste des concours ouverts" donne un aperçu des concours auxquels vous pouvez vous inscrire « en ligne ». Avant de procéder à votre inscription électronique il est important que vous lisiez attentivement l'avis du concours. Vous le trouverez en cliquant sur l'intitulé du concours.
- Les conditions d'admission aux concours sont expliquées en détail dans l'avis du concours qui est la source d'information de référence pour tous les termes et toutes les conditions s'appliquant à un concours.
- A partir de la page "Liste des concours ouverts", il vous est possible de sélectionner un concours en cliquant sur le bouton "s'inscrire", dépendant du concours auquel vous souhaitez vous inscrire.
- Dans la page du site " Fiche d'inscription ", vous devez fournir les informations qui vous sont demandées. Tous les champs en gras doivent obligatoirement être remplis.
- Si vous êtes certain que toutes les données fournies sont correctes, cliquez sur le bouton " Valider "
- Une nouvelle page nommée " Confirmation " affichera votre numéro d'inscription.
- Si cette page ne s'affiche pas, il se peut qu'il y ait eu un problème de connexion, dans ce cas essayer de réactualiser votre navigateur Internet, ou recommencer la procédure d'inscription à partir de la page "Liste des concours ouverts".
- Par contre si la page " Confirmation " s'affiche, vous pouvez confirmer votre inscription (à l'aide du bouton «confirmer») ou l'annuler. Après avoir confirmé votre inscription, vous n'aurez plus la possibilité de modifier votre dossier, le dossier ne sera accessible qu'en lecture seule. Par conséquent, il faut vérifier que les informations saisies sont correctes avant de confirmer.
- Une autre page affichera un récépissé d'inscription. Attention votre machine devrait disposer d'acrobate reader ou d'un logiciel capable de visionner des fichiers de type pdf.
- Vous devez imprimer le récépissé, le présenter le jour du concours afin d'avoir accès aux salles de d'examen, vous devez également être muni de votre Carte d'identité Burkinabé(CIB) ou de la Carte Nationale d'identité (CNI) ou de votre Carte Consulaire (pour les compatriotes venant de l'extérieur de pays) portant le numéro que vous avez mentionné lors de votre inscription.
- Nous vous conseillons également d'enregistrer votre récépissé électronique sur un support magnétique afin de pouvoir l'imprimer en cas de perte du support papier.
- Vous disposez de trente (30) minutes pour choisir un concours et vous inscrire.

Terminé

– Demande de consultation d'un dossier

3

Consultation - Mozilla Firefox

Fichier Edition Affichage Aller à Marque-pages Outils ?

http://127.0.0.1:8080/opencms/opencms/concours/demande\_consul OK

yahoo-mail caramail google MySQL projet openCms Page d'accueil de Mo... Missions et Organisa...

Les grands chantiers Consultation

**Consultation de dossier**

Choisir/Saisir les informations demandées

Session : 2005

Concours : Agents Techniques Elevage

Centre : Bobo-Dioulasso

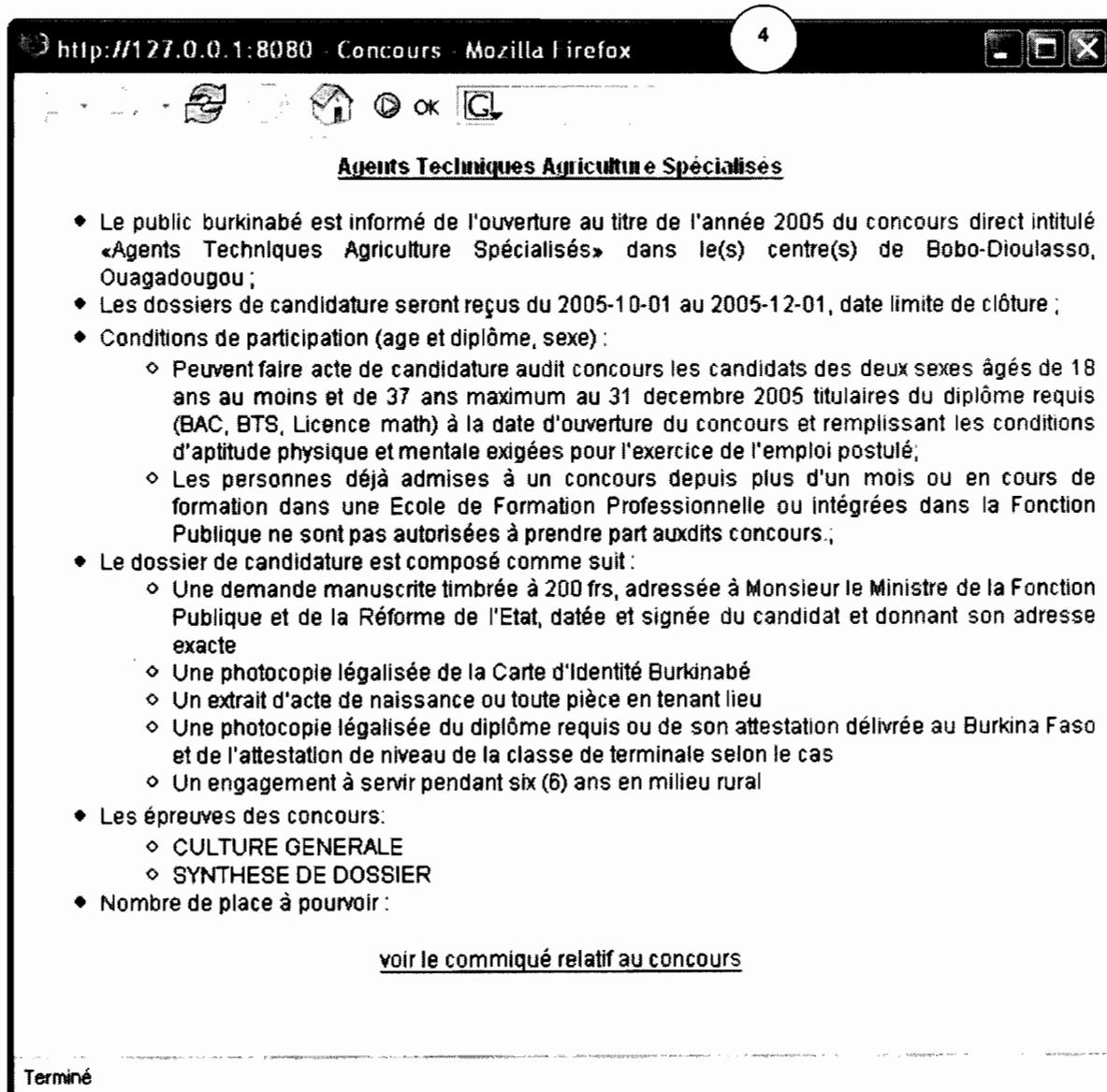
Votre numéro d'inscription

Numéro CIB

Valider

[Retour](#)

Terminé

– Affichage du communiqué relatif à un concours

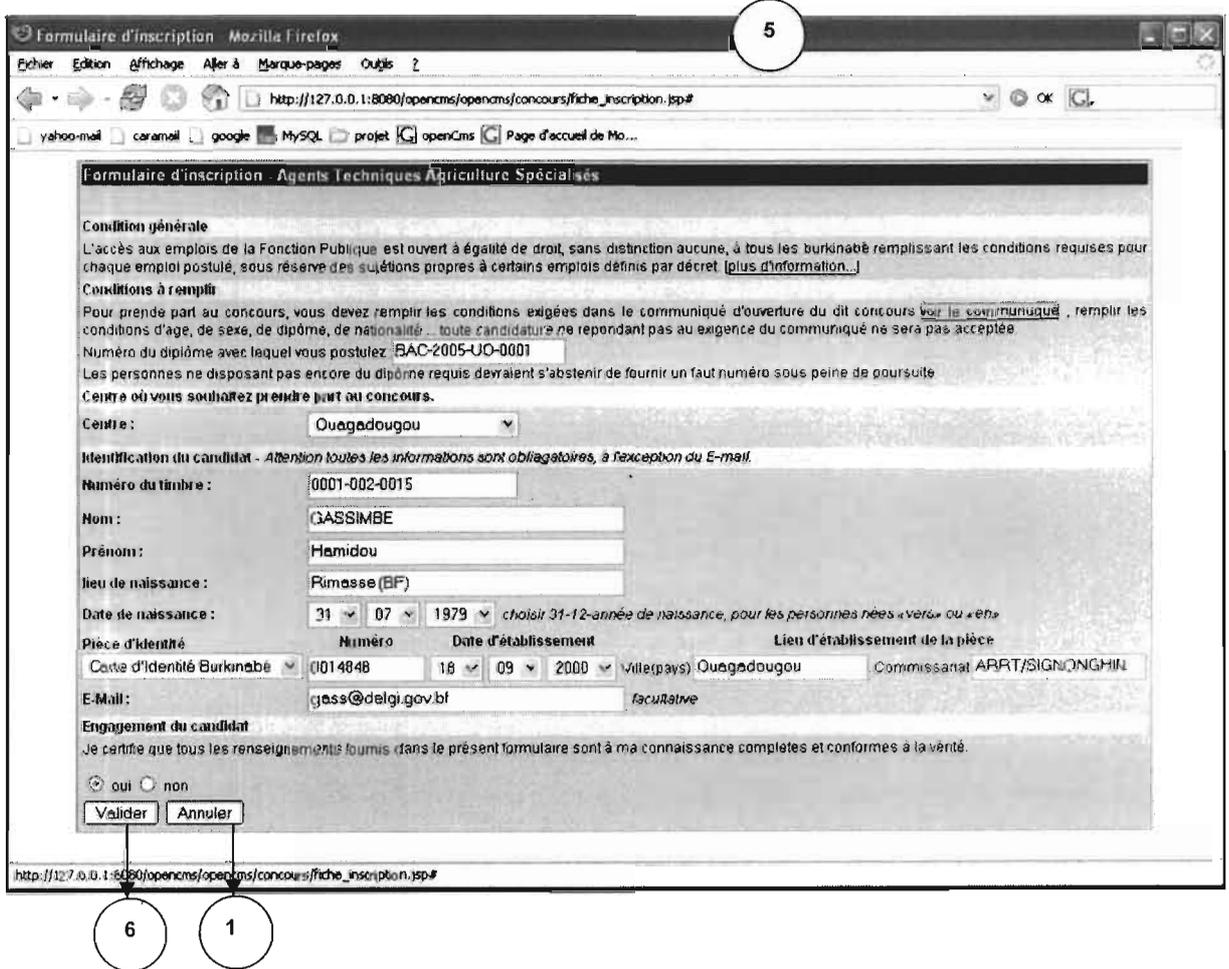
The screenshot shows a Mozilla Firefox browser window with the address bar containing 'http://127.0.0.1:8080 - Concours - Mozilla Firefox'. The page title is 'Agents Techniques Agriculture Spécialisés'. The content is a recruitment notice with the following details:

- ◆ Le public burkinabé est informé de l'ouverture au titre de l'année 2005 du concours direct intitulé «Agents Techniques Agriculture Spécialisés» dans le(s) centre(s) de Bobo-Dioulasso, Ouagadougou ;
- ◆ Les dossiers de candidature seront reçus du 2005-10-01 au 2005-12-01, date limite de clôture ;
- ◆ Conditions de participation (age et diplôme, sexe) :
  - ◇ Peuvent faire acte de candidature audit concours les candidats des deux sexes âgés de 18 ans au moins et de 37 ans maximum au 31 decembre 2005 titulaires du diplôme requis (BAC, BTS, Licence math) à la date d'ouverture du concours et remplissant les conditions d'aptitude physique et mentale exigées pour l'exercice de l'emploi postulé;
  - ◇ Les personnes déjà admises à un concours depuis plus d'un mois ou en cours de formation dans une Ecole de Formation Professionnelle ou intégrées dans la Fonction Publique ne sont pas autorisées à prendre part auxdits concours ;
- ◆ Le dossier de candidature est composé comme suit :
  - ◇ Une demande manuscrite timbrée à 200 frs, adressée à Monsieur le Ministre de la Fonction Publique et de la Réforme de l'Etat, datée et signée du candidat et donnant son adresse exacte
  - ◇ Une photocopie légalisée de la Carte d'Identité Burkinabé
  - ◇ Un extrait d'acte de naissance ou toute pièce en tenant lieu
  - ◇ Une photocopie légalisée du diplôme requis ou de son attestation délivrée au Burkina Faso et de l'attestation de niveau de la classe de terminale selon le cas
  - ◇ Un engagement à servir pendant six (6) ans en milieu rural
- ◆ Les épreuves des concours :
  - ◇ CULTURE GENERALE
  - ◇ SYNTHESE DE DOSSIER
- ◆ Nombre de place à pourvoir :

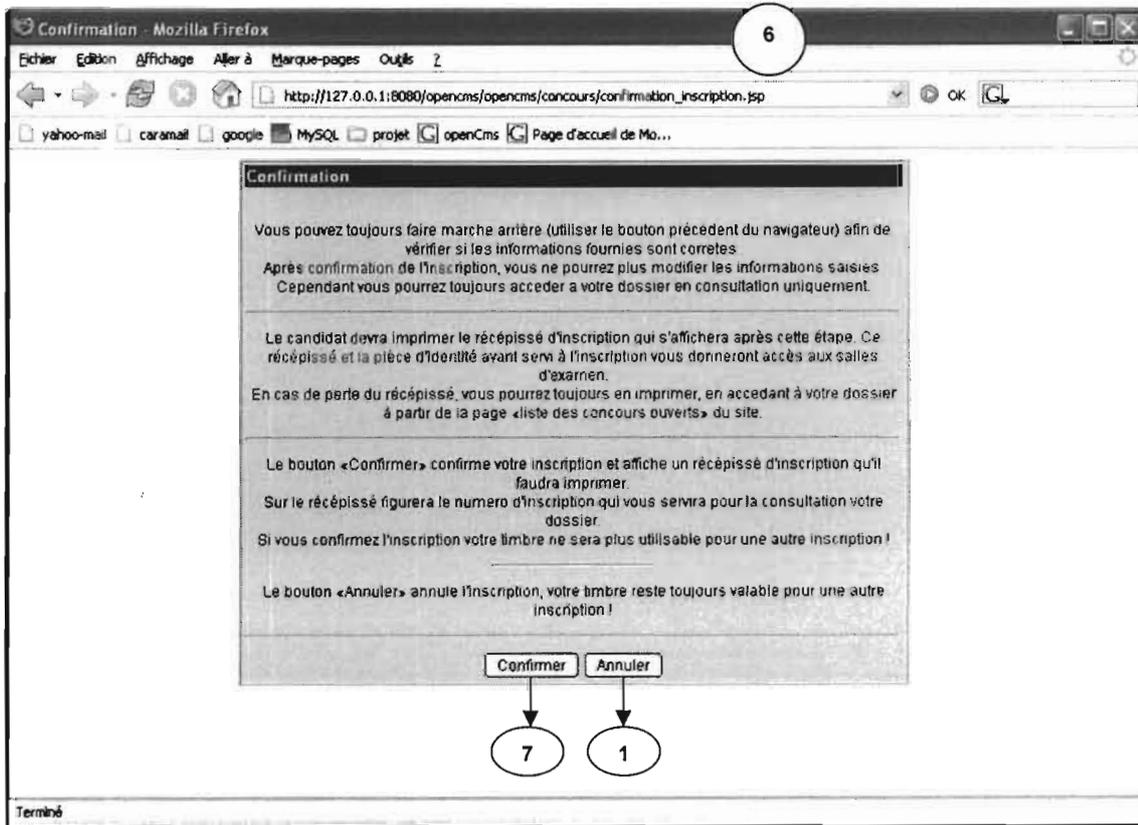
voir le communiqué relatif au concours

Terminé

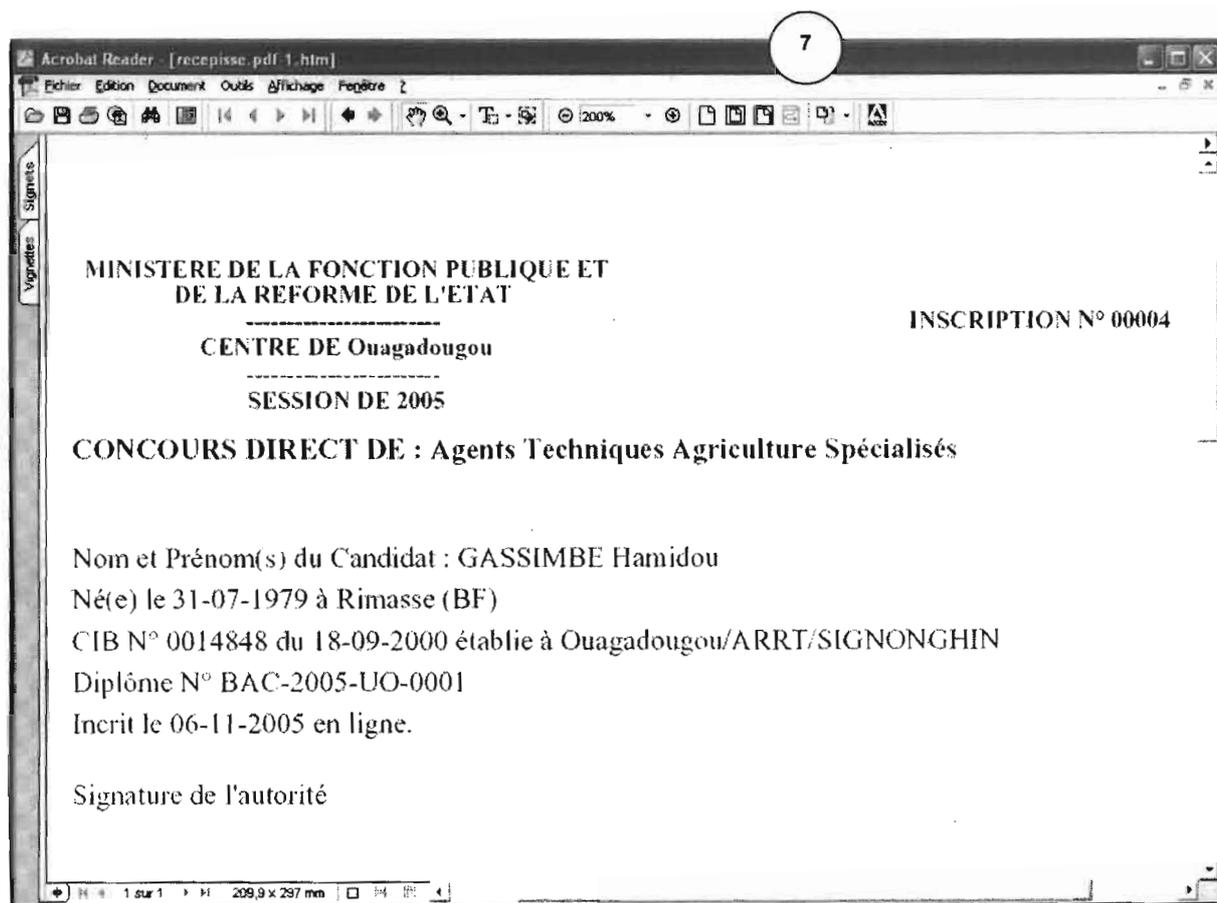
– Formulaire d'inscription



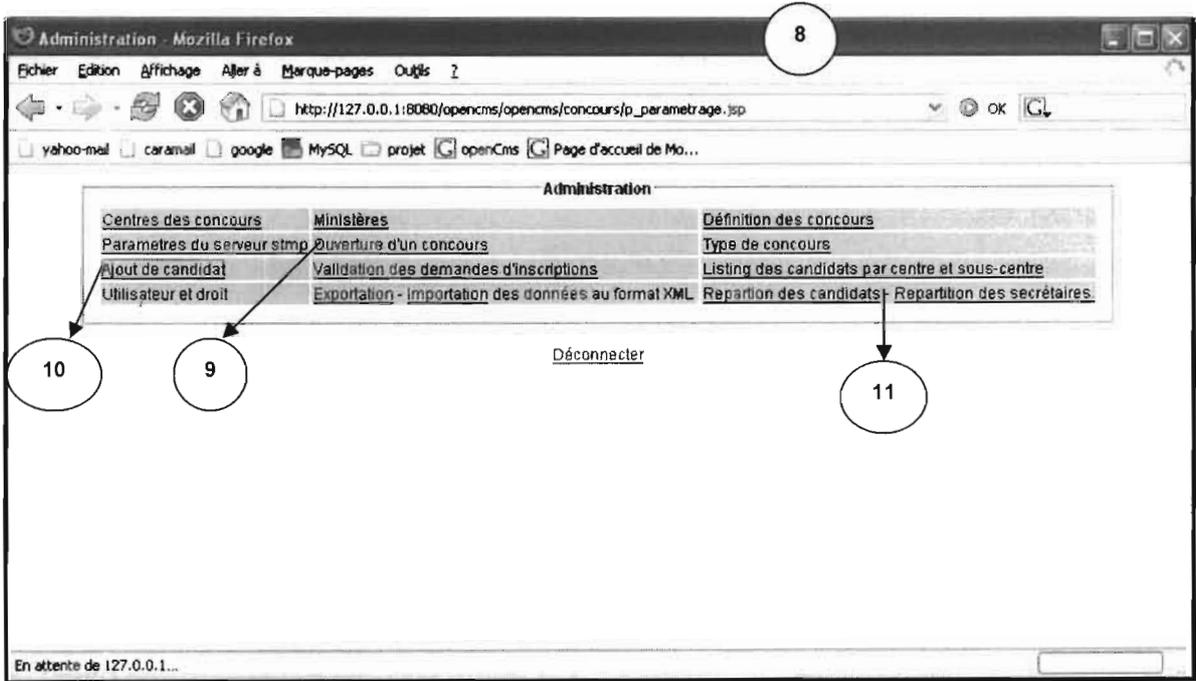
– Formulaire de confirmation de sa demande d'inscription



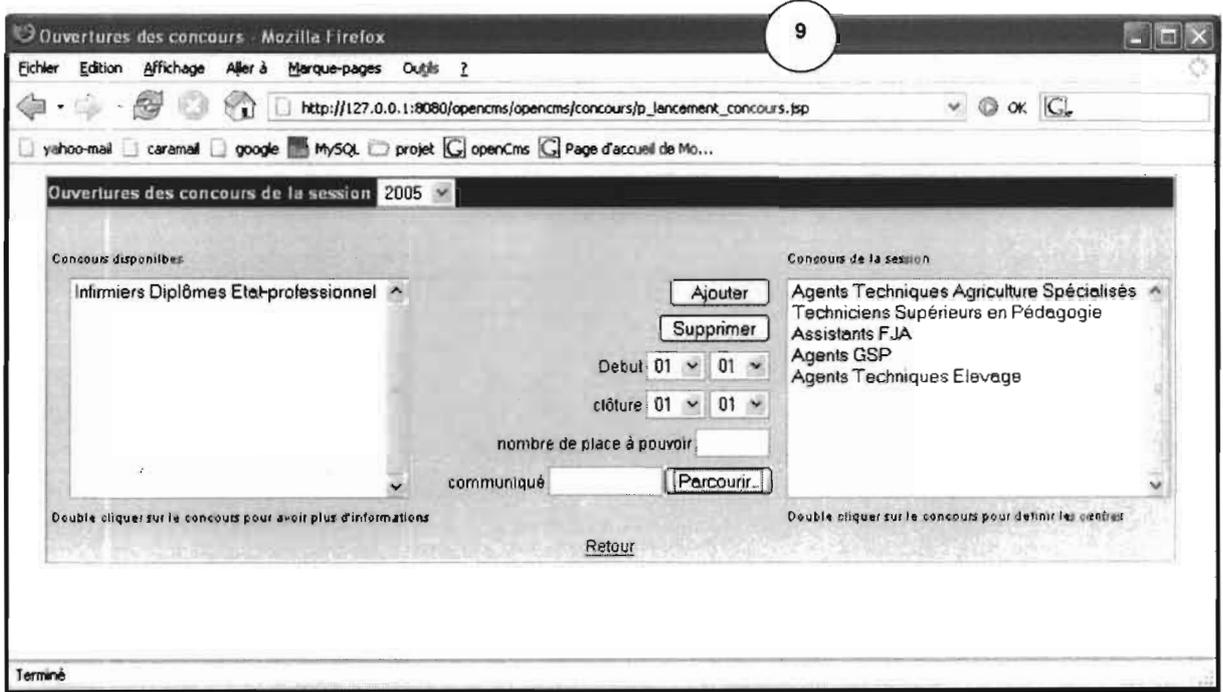
– Edition d'un récépissé d'inscription



– Page d'administration du système



– Interface d'ouverture d'un concours



- Formulaire d'ajout des candidats inscrits *offline*

Inscription des candidats - Mozilla Firefox

10

http://127.0.0.1:8080/opencvms/opencvms/concours/p\_ajout\_candidat.jsp

Identification du candidat - Attention tous les champs sont obligatoires

session : 2005

concours : Agents Techniques Agriculture Spécialisés

centre : Bobo-Dioulasso

N° inscription : 20000

Nom : SANA

Prénom : Alioune

lieu de naissance : Ouagadougou

Date de naissance : 10 12 2005

CIB, CNI, Consulaire ...	Numéro	Date d'établissement	Lieu d'établissement
01258	10 05 2000		Bobo-Dioulassa/Coleama

Valider Effacer

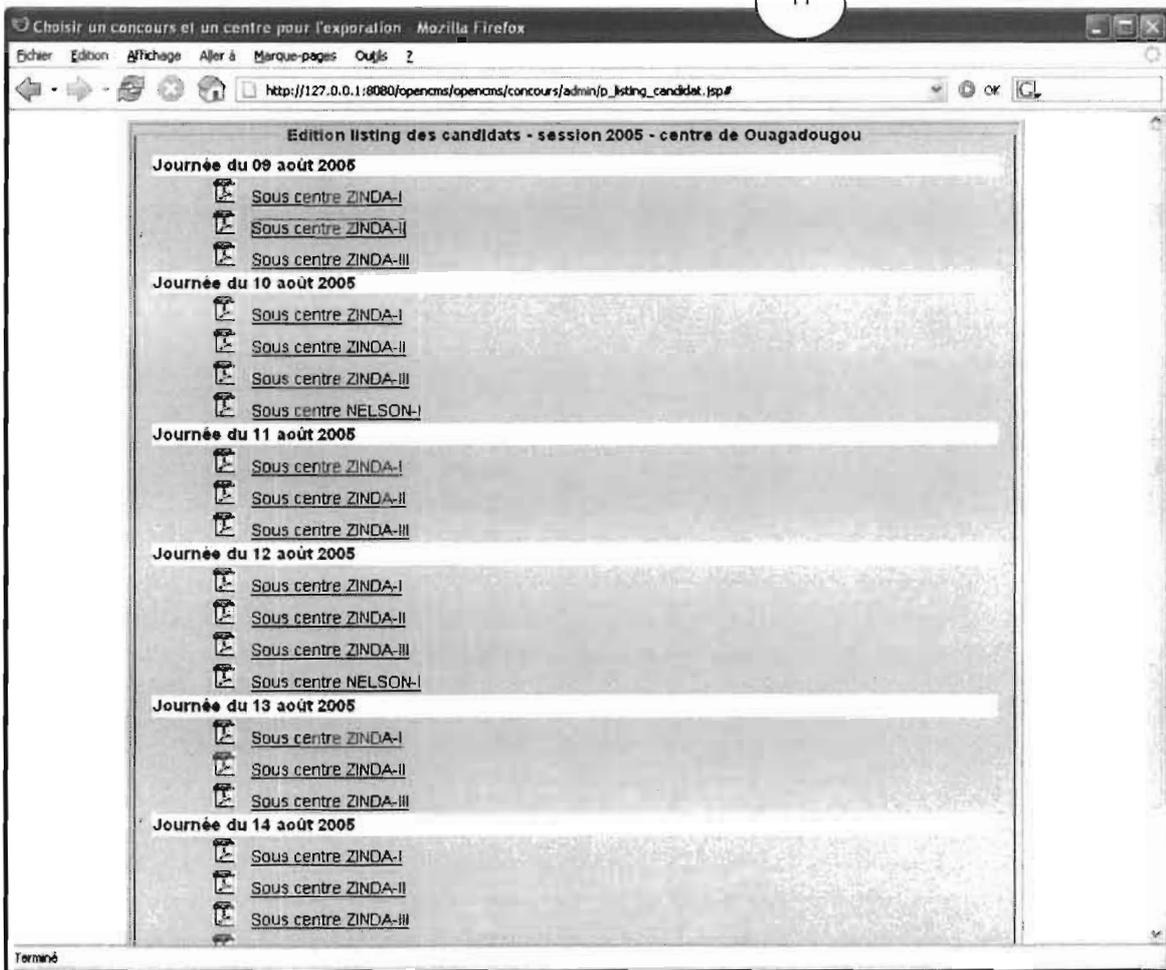
[Retour](#)

[Administration](#)

Terminé

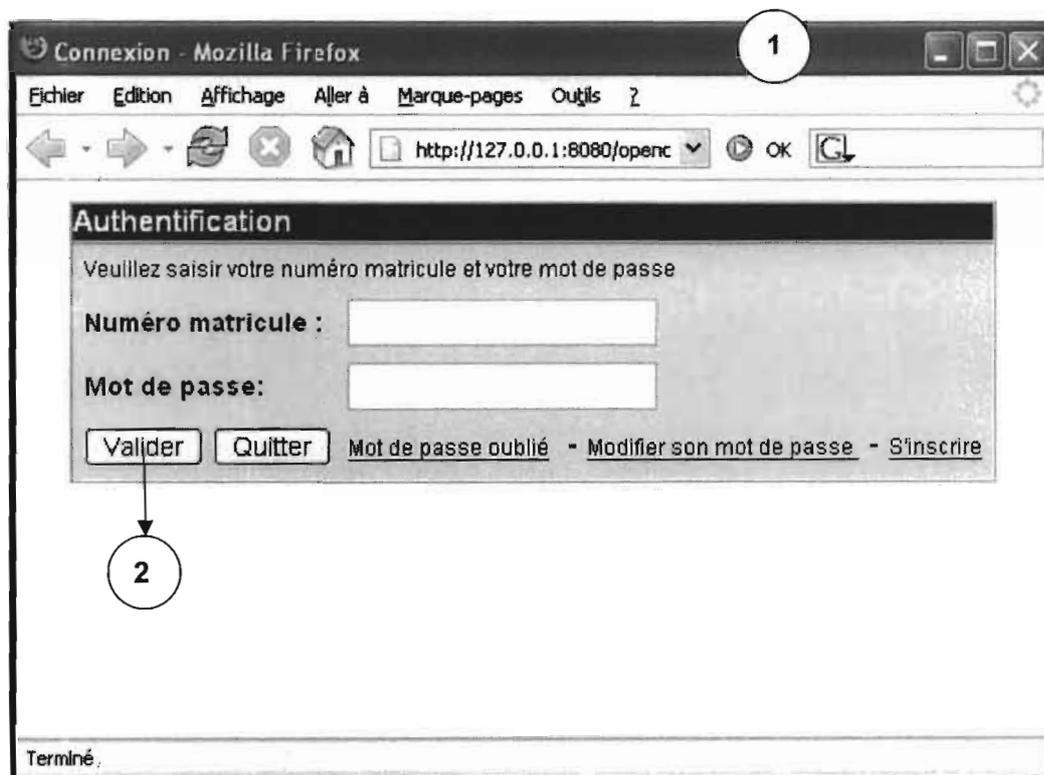
- Interface d'édition des listings de réparation des candidats dans les sous-centres

11



## ANNEXE-2 : QUELQUES ECRANS DE CONSULTATION DES SITUATIONS ADMINISTRATIVES DES AGENTS DE LA FONCTION PUBLIQUE

- Interface d'accès à l'application



- Identité de l'agent

The screenshot shows a web browser window with the address bar displaying "http://177.0.0.1:8080 Identité agent Mozilla Firefox". A circled number "2" is in the top right corner of the browser window. The main content is a form titled "Identité de l'agent" with the following fields:

Matricule			
Nom	COMPORE		
Prénom(s)	L. J. L.	Nom de Jeune Fille	
Date de naissance	03-05-19...	Lieu de naissance	...
Nationalité	Burkinabè	Sexe	M
Situation SND	Dispensé(e)	début SND	fin SND
Diplôme le plus élevé			

Below the form are four buttons: "Situation et position administrative", "Carrière", "Nous écrire", and "Fermer le dossier". Arrows point from these buttons to circled numbers: 3, 4, 5, and 1 respectively. The word "Terminé" is visible in the bottom left corner of the browser window.

- Affichage de la situation et position administratives de l'agent

http://127.0.0.1:8080 - Situation et position administratives - Mozilla Firefox

3

Situation et position administratives

Matricule: 1005 Nom et prénom: COMPAORE, J...

Situation administrative

Mode Rec.		date	26-05-1975
Qualité	Fonctionnaire titulaire	date	01-10-1975
Corps	[Fonctionnaire titulaire]		
Cadre	[Fonctionnaire titulaire]		
Cat/ech.	r1	Classe	[Fonctionnaire titulaire]
Echelon	01	date	11-12-2003

Position administrative

Position	Activité	depuis la
Se position	Exerçant	depuis la
Motif	[Fonctionnaire titulaire]	

Page précédente → 2

Prenez soin de fermer votre dossier pour éviter que d'autres personnes n'y accèdent. Fermer le dossier → 13

Terminé

- Affichage des différents dossiers d'un agent

http://127.0.0.1:8080 Consultation des dossiers - Mozilla Firefox

4

Consultation des dossiers

Agent

Matricule : ...

Nom et Prénom(s) : COMPAORE I...

Dossiers traités ou en cours de traitement

Code de l'acte	Nature	Date d'arrivée	
...	Avancement d'échelon		information non disponible
...	Avancement d'échelon		information non disponible
...	Reversement	10-03-2000	information non disponible
...	Avancement d'échelon	13-08-2001	information non disponible
...	Avancement d'échelon	13-08-2001	information non disponible
...	Avancement d'échelon	13-08-2001	information non disponible
...	Avancement d'échelon	13-08-2001	information non disponible
...	Avancement d'échelon	13-08-2001	information non disponible
...	Avancement d'échelon	13-08-2001	information non disponible
...	Décoration	21-05-2004	<a href="#">cliquer ici pour voir le dossier</a>

10 dossier(s) trouvé(s)

Page précédente

2

12

Fermer le dossier

Prenez soin de fermer votre dossier pour éviter que d'autres personnes n'y accèdent.

Terminé

- Consultation de l'historique du circuit d'un dossier

http://127.0.0.1:8080 Consultation d'un dossier - Mozilla Firefox

6

Consultation du dossier Numéro 2004.07752

Nature du dossier : Décoration  
Position actuelle du dossier : SOLDE  
Etat du dossier : Validé

Références de l'acte relatif au dossier :

N° de l'acte	Date d'établissement	Date d'effet	Date de signature
2004-07752	07-06-2004	11-12-2003	16-06-2004

Circuit du dossier

Ce circuit retrace l'historique de votre dossier : date, service traiteur du dossier

Date d'envoi	Position du dossier
25-05-2004	Bureau d'ordre
28-05-2004	Chef de service
08-06-2004	Rédacteur
09-06-2004	Contrôle Financier
09-06-2004	Circuit
11-06-2004	OPE
14-06-2004	DGFP
15-06-2004	Secrétaire Général
26-06-2004	Circuit

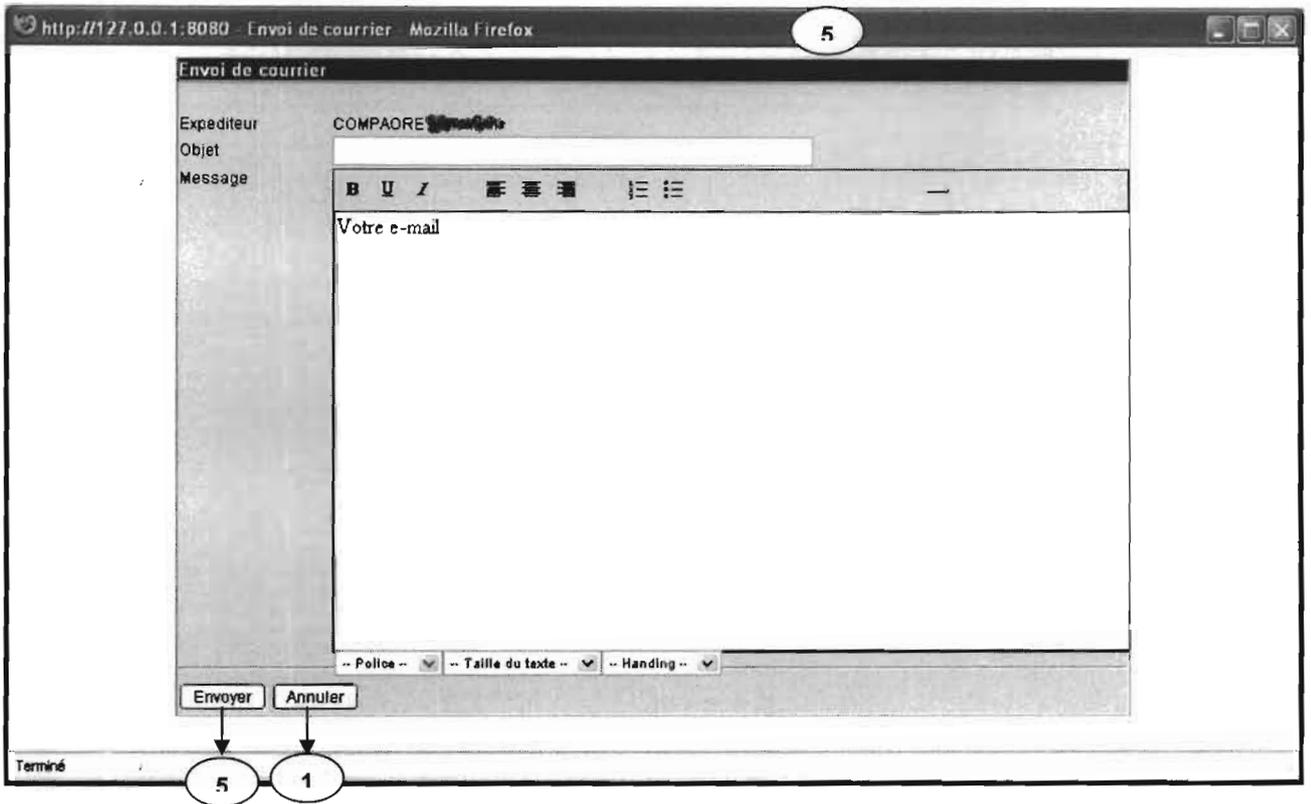
Page précédente → 2

Fermer le dossier → 1

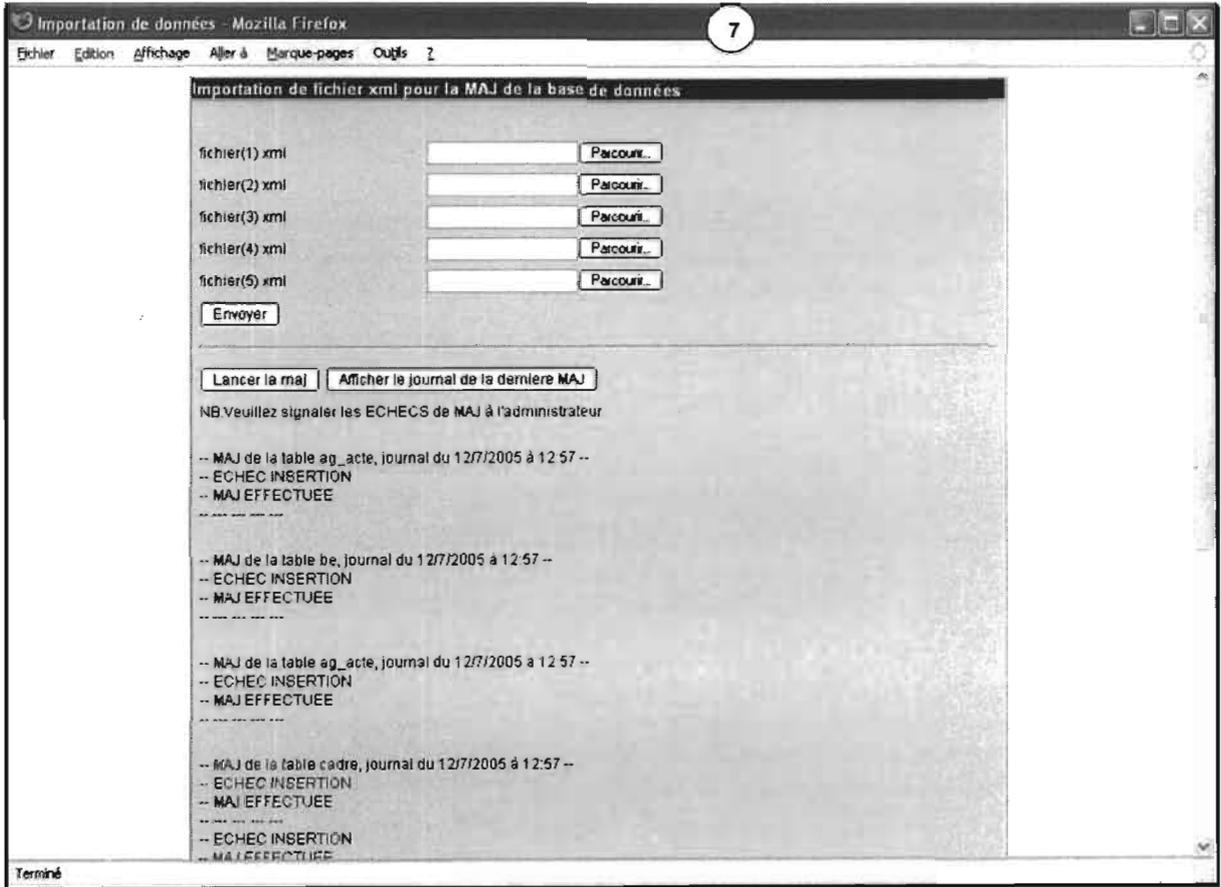
Prenez soin de fermer votre dossier pour éviter que d'autres personnes s'y accordent

Terminé

- Interface d'envoi de courrier électronique



- Interface de mise à jour de la base de données



## GLOSSAIRE

- Application Web

Une application Web est une application délivrée par un serveur Web sur un réseau.

Elle repose sur une architecture client/serveur à plusieurs niveaux (Tiers) :

- niveau interface utilisateur ;
- niveau contrôle ou logique ;
- niveau des données ou objets ;
- niveau « *inter-niveaux* » ou middleware.

Pour mettre en œuvre ces niveaux, un ensemble de technologies et de modèles sous-jacents sont disponibles. Au niveau interface utilisateur, les technologies à utiliser sont celles des pages Web : HTML, Javascript, styles CSS.

Le niveau contrôle est régi par les technologies de la génération dynamique de page Web souvent associées à un langage de programmation : PHP, JSP (Java)...

Le niveau des données peut être objet et/ou relationnel. Lorsque les données sont persistantes, une base de données est alors utilisée.

Enfin, pour faire communiquer ces technologies, des bibliothèques de composants (API) existent dans le langage de développement choisi. Par exemple, en java l'API JDBC (Java Data Base Connectivity) est utilisée pour se connecter à une base de données.

- Serveur Web : processus logiciel chargé d'interpréter des requêtes de pages et de renvoyer les pages demandées.
- Servlet : c'est une classe Java compilée (.class) qui s'exécute dans un conteneur JSP/Servlet hébergé dans un serveur Web, la Servlet est un composant qui étend les fonctionnalités d'un serveur Web de manière portable et efficace.
- JSP (Java Server Page) : une JSP est une page HTML contenant des morceaux de code Java et/ou des balises JSP spécifiques, permettant d'effectuer des traitements et d'intégrer des données, afin de générer dynamiquement des pages HTML.

- Conteneur Web JSP/Servlet : moteur d'exécution des Servlet/JSP coté serveur (exemple Apache Tomcat).
- Client léger/lourd : Le client léger désigne un poste utilisateur dont la fonction se limite à interpréter l'affichage de pages Web. Ce client léger, né avec les technologies du Web, s'oppose au client lourd des architectures client/serveur. Dans ce dernier cas, le poste client assume une part du traitement applicatif. Résultat, le client léger demande beaucoup moins de maintenance que le client lourd puisqu'il n'a pas à être mis à jour à chaque nouvelle version du logiciel. Le seul composant requis est un navigateur Web.
- Le développement coté serveur en java : c'est écrire des composants Web (Servlet, JSP) et des composants métiers en respectant une certaine architecture (MVC par exemple) afin d'optimiser la création et la maintenance des applications.
- EJB (Entreprise Java Bean) : c'est une API qui fournit un modèle de conception de composants métiers utilisés dans les applications basées sur une architecture distribuée n/tiers.
- JavaBean : un JavaBean est un Objet compilé qui offre des services à travers ses méthodes publiques. On peut par exemple les utiliser dans les JSP pour séparer le code source java du code HTML.
- Frameworks (exemple Struts): les frameworks (squelettes d'application) sont des structures logicielles qui définissent des cadres dans lesquels viennent s'insérer les objets et concepts spécifiques à une application. En pratique, un framework est un ensemble de classes et de mécanismes associés à une architecture logicielle qui fournit un ou plusieurs services techniques ou métiers aux applications qui s'appuient dessus.

## BIBLIOGRAPHIE

### ● Références bibliographiques

- Benjamin AUMAILLE. *J2EE Développement d'applications Web*. Nantes : Editions ENI, 2002, 361 p. ISBN 2-7460-1656-7
- Geoges Gardarin. *XML des bases de données aux services Web*. Paris : Dunod, 2002, 521 p. ISBN 2 10 006933 0
- George Reese. *Database Programming with JDBC and JAVA*. Sebastopol : O'Reilly, 2000, 368 p.
- Ben Laurie, Peter Laurie. *Apache La référence*. 3<sup>e</sup> édition. O'Reilly, Paris, 2003, 585 p. ISBN 2-84177-255-X
- SGG-CM. *Réforme Globale de l'Administration Publique*. Ouagadougou : Journal Officiel 2000, 302 p. Dépôt légal N° 3446 ;
- Ministère de la Fonction Publique et de Réforme de l'Etat. *Système Intégré de la Gestion Administrative et Salariale du Personnel de l'Etat (SIGASPE)*. Ouagadougou : MFPRE, 2002, 61p.

### ● Adresses Internet

- <http://java.sun.com/j2se/1.4.2/docs/api/index.html>
- <http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html>
- <http://www.labo-sun.com>
- <http://www.lab-oracle.com>
- <http://dev.mysql.com/doc/>
- <http://xmlfr.org/>
- <http://www.developpez.com>
- <http://www.opencms.com>
- <http://uml.free.fr/>
- <http://www.dotnetguru.org/articles/Persistence/livreblanc/ormapping.htm>
- <http://struts.apache.org/>
- <http://tomcat.apache.org/>
- <http://www.jdom.org/docs/apidocs/index.html>
- [http://www.oracle.com/technology/tech/pl\\_sql/index.html](http://www.oracle.com/technology/tech/pl_sql/index.html)
- <http://www.javascript.com/>
- <http://java.sun.com/products/jsp/>
- <http://java.sun.com/products/servlet/docs.html>
- <http://www.fonction-publique.gov.bf>
- <http://www.delqi.gov.bf>