

RAPPORT DE STAGE DE FIN D'ETUDES

Pour l'obtention de la

«Licence Appliquée en Sciences et Technologies de l'Information et de
Communication (LASTIC)»

Présenté par :

Mohamed Bouaziz

Titre

Mise en place d'une solution de Gestion Electronique de
Documents (GED) Open Source pour l'ISET de Zaghouan

Soutenu le : 30 juin 2018

Devant le jury :

Président : Mr. Ezzeddine Ben Braiek

Encadreur : Mme. Imen Ammari

Rapporteur : Mme. Lobna Kriaa

Année Universitaire : 2017 / 2018

Résumé

Le présent travail, effectué au sein de l'ISSET de Zaghouan, s'inscrit dans le cadre d'un projet de fin d'études pour l'obtention du diplôme de Licence Appliquée en Sciences et Technologies de l'Information et de la Communication de l'Université Virtuelle de Tunis.

Notre projet a pour objectif de mettre en place un système de Gestion Electronique de Documents (GED) et d'archivage sous la plateforme Alfresco. Ce travail consiste à mettre en place un système informatisé facilitant l'accès à l'information et assurant une conservation pérenne des documents conçue selon les nouvelles technologies d'Alfresco.

Mots clés : GED, Archivage, Alfresco.

Abstract

This work, performed within ISET Zaghouan, is a part of the project graduation in Applied degree in Sciences and Technologies of Information and Communication of the Virtual University of Tunis.

Our project has for objective to implement an Enterprise Content Management ECM system and Records Management in the Alfresco platform. This work consists into developing a computerized system that facilitates access to information and guarantees the long-term preservation of documents within the new technologies of Alfresco.

Keywords: ECM, Records Management, Alfresco.

Dédicace

Je dédie le présent travail à toute personne ayant contribué par une parole, un geste ou même une pensée, à cette finalité.

Ceci étant en général, vient maintenant une particulière distinction bien méritée pour mes parents sans qui, je n'y serai jamais arrivé.

Et pour ma chère Amel, source de ma motivation.

Remerciement

En cette fin d'année et à l'achèvement de ce travail, je tiens tout d'abord à remercier Dieu le tout puissant, qui m'a donné la force et la patience pour accomplir ce modeste travail.

Je tiens à remercier toute personne m'ayant accompagné dans cette journée

Je tiens à remercier mon encadrant industriel à l'ISET de Zaghouan M. Boubaker Ali et le directeur et tout le personnel de l'ISET pour leur coopération

Je suis reconnaissant également à Mme. Ammari Imen, mon encadrante pédagogique à l'UVT durant le déroulement du projet pour son suivi, son aide précieuse et ses conseils

MERCI

Table des matières

Introduction générale	1
Chapitre 1 : Cadre général du projet	2
Introduction	2
1. Présentation de l'organisme d'accueil	2
2. Problématique	3
3. Contexte du projet	3
Chapitre 2 : Etat de l'art	4
Introduction	4
1. Les systèmes GED	4
1.1. Définition	4
1.2. Avantages de la GED	4
1.3. Etapes de la GED	5
2. Présentation d'Alfresco	5
2.1. Définition	5
2.2. Historique	6
2.3. Les versions d'Alfresco	6
2.4. Les avantages d'Alfresco	6
3. Alfresco vis-à-vis de ses concurrents	6
3.1. Nuxeo	6
3.2. Knowledge Tree	7
3.3. Exo-Dms (Document Management System)	7
4. Choix d'Alfresco	8
5. Utilisation d'Alfresco	8
6. Personnalisation d'Alfresco	12
6.1. Méthodologie et technologies adoptées	12
6.2. Architecture Alfresco	12
7. Intégration du module « Activiti »	14
8. Solution adoptée	15
Conclusion	15
Chapitre 3 : Analyse et spécification des Besoins	16
Introduction	16
1. Objectifs	16
2. Spécification des besoins fonctionnels	16
2.1. Besoins des utilisateurs	16
2.2. Besoins des administrateurs	16

3. Spécification des besoins non fonctionnels	17
4. Modélisation des besoins	17
4.1. Diagramme de paquetages	17
4.2. Diagrammes des cas d'utilisation	18
4.3. Description des cas d'utilisation	20
Conclusion	24
Chapitre 4 : Conception	25
Introduction	25
1. Conception globale	25
1.1. Schéma d'architecture	25
1.2. Diagramme de déploiement	26
2. Conception détaillée	26
2.1. Diagramme de navigation	27
2.2. Modélisation du Contenu	28
2.3. Comportement du système	29
2.4. Conception des processus métiers	30
Conclusion	34
Chapitre 5 : Réalisation	35
Introduction	35
1. Environnement de travail	35
2. Description de l'application	35
2.1. Installation	36
2.2. Description des Interfaces utilisateurs	37
3. Chronogramme :	45
Conclusion générale et perspectives	46
Bibliographie	47
ANNEXES	48
Annexe I :	49
Annexes II :	51

Table des figures

Figure 1 : Organigramme de l'ISET	3
Figure 2 : Système GED	5
Figure 3 :Le logo de Nuxeo	7
Figure 4 : Le logo de Knowledge Tree	7
Figure 5 : Le logo d'Exo	8
Figure 6 :Le logo d'Alfresco	8
Figure 7 : Page d'authentification	9
Figure 8 : Menu principal d'alfresco	9
Figure 9 : Tableau de bord Alfresco share	10
Figure 10 : Espace de travail	10
Figure 11 : Tableau de liste des rôles	11
Figure 12 : Le Versionning de documents	11
Figure 13 : La console d'administration	12
Figure 14 : L'architecture d' Alfresco	13
Figure 15 : Alfresco Repository	13
Figure 16 : Menu principal d'Activiti	15
Figure 17 : Menu principal d'Activiti	15
Figure 18 : Diagramme de paquetages	17
Figure 19 : Diagramme de cas d'utilisation « Gestion des règles de classement »	18
Figure 20 : Diagramme de cas d'utilisation « Gestion des modèles types »	19
Figure 21 : Diagramme de cas d'utilisation « Gestion de compte»	19
Figure 22 : Diagramme de cas d'utilisation « Gestion des archives »	20
Figure 23 : Diagramme de séquence « Démarrer une règle de classement »	22
Figure 24 : Diagramme de séquence « Consulter son espace Alfresco »	23
Figure 25 : Diagramme d'activités « Stopper une règle de classement »	23
Figure 26 : Diagramme d'activités « archiver les documents »	24
Figure 27 : Architecture deux tiers	25
Figure 28 : Modèle de déploiement	26
Figure 29 : Diagramme de navigation de l'interface	27
Figure 30 : Tableau des moyens de classification des documents selon Alfresco	28
Figure 31 : Conception du modèle de données : Cas du service « Achat »	28
Figure 32 : Diagramme de séquence « démarrer règle de classement »	29
Figure 33 : Diagramme de séquence « ajouter document »	30
Figure 34 : Processus métier « service achat »	31
Figure 35 : Processus métier « service ressources humaines »	31
Figure 36 : Processus métier « service scolarité »	32
Figure 37 : Processus métier « service stage »	32
Figure 38 : Processus métier « service bibliothèque »	32
Figure 39 : Processus métier « Service Examen »	33
Figure 40 : Processus métier « Service maintenance »	33
Figure 41 : Processus métier « Service Finance »	34
Figure 42 : Choix du type d'installation	36
Figure 43 : Configuration du serveur Tomcat	37
Figure 44 : Interface d'accueil de login	38
Figure 45 : Tableau de bord de l'administrateur	38
Figure 46 : interface Gestionnaire des sites	39

Figure 47 : Interface de Gestionnaire de rôles	39
Figure 48 : Interface de création des règles de classement (administrateur)	40
Figure 49 : interface principale des règles de classement (utilisateur)	40
Figure 50 : Interface de création de modèles personnalisés	41
Figure 51 : Interface d'attribution de modèles/aspects	42
Figure 52 : Interface d'édition des propriétés	42
Figure 53 : Site RM de l'archivage	43
Figure 54 : Gestion des règles de classement des archives	44
Figure 55 : Moteur de recherche des documents du site RM	44
Figure 56 : Chronogramme du projet	45
Figure 57 : Code xml de création de modèle	49
Figure 58 : Code xml de déclaration de modèle créé	50
Figure 59 : configuration du serveur	51

Introduction générale

De nos jours, nous constatons une croissance exponentielle de la masse de documents et la quantité d'informations partagées, au sein des entreprises. La gestion de ces documents, leur classement fastidieux et pénible, la dépense d'énergie qui en découle, tout ceci demeure un problème persistant au quotidien. De même, la recherche d'information, devient une tâche indispensable à la performance et au développement des organisations quelles que soient leurs tailles, leurs missions, et leurs secteurs d'activité.

Dans ce contexte, l'archivage revêt désormais un caractère particulier et requiert une méthodologie rigoureuse pour faire face à ces problèmes, et organiser, d'une manière systématique et efficace, tous les documents et données dans une entreprise. Cette méthode d'archivage a pour but d'identifier les différents types de documents, les classer selon des critères bien définis, les conserver et en garder une trace tout le long de leur cycle de vie. Pour faire face aux besoins d'archivage, une nécessité exige le déploiement d'un système de Gestion Electronique des Documents (GED), introduisant le travail collaboratif et le partage de fichiers.

Dans ce cadre, s'inscrit notre projet de fin d'études qui consiste à déployer un système GED open-source pour l'Institut Supérieur des Etudes Technologiques (ISET) de Zaghouan. Il s'agit de mettre en place une solution de système de GED répondant à ses besoins en termes de gestion des documents et facilitant l'archivage et le classement de documents numériques. Nos Principaux défis à relever :

- La perte de productivité : plusieurs minutes par jour et par personne sont aujourd'hui perdues pour le classement et la recherche de documents papiers.
- L'explosion des coûts de stockage : coût de stockage en matière de papier dont le prix ne cesse d'augmenter et coût de stockage en m³ d'emplacement dédié [1].
- Cloisonnement des services dus à l'impossibilité de partager les informations stockées dans les classeurs papiers et sur les postes de travail en temps réel.

Le présent rapport décrit notre travail réalisé. Il s'articule autour d'une introduction générale suivie de cinq chapitres. L'introduction générale donne un aperçu sur l'organisme d'accueil et explique quelques notions liées au contexte général du projet. Le premier chapitre est consacré au cadre général du projet. Le deuxième chapitre est consacré à l'état de l'art, pour mieux définir et cerner les besoins. Le troisième chapitre intitulé ; Analyse et spécification des besoins ; introduit la problématique et donne un aspect aux besoins. Il définit également le processus de développement et la méthodologie à suivre. De même, il fait l'objet d'une description du projet à travers une capture des besoins fonctionnels et une analyse de ces derniers. Le quatrième chapitre spécifie les besoins techniques et l'architecture adoptée. Il concerne aussi la conception préliminaire et la conception détaillée des fonctionnalités attendues de notre projet. Le cinquième chapitre porte sur l'implémentation et l'intégration de la solution logicielle en présentant les différentes interfaces utilisateur. Il comporte aussi une description des outils de travail et les tests des cas d'utilisation. Enfin, nous concluons ce rapport en présentant des perspectives futures pour notre travail.

Chapitre 1 : Cadre général du projet

Introduction

Ce premier chapitre est consacré à la présentation de l'organisme d'accueil de notre projet de fin d'études l'ISET de Zaghouan tout en précisant ses activités et son organigramme ainsi que la problématique et le contexte du projet.

1. Présentation de l'organisme d'accueil

Crée par la loi n° 1535 du 25 juin 2002 ; l'Institut Supérieur des Etudes Technologiques de Zaghouan (ISET de Zaghouan) constitue un centre de compétence complet, ouvert à l'environnement extérieur et bien situé dans une ville au carrefour de grands axes de communication du nord au cap bon jusqu'au centre du pays.

MISSION

La mission de l'ISET de Zaghouan comme tout autre institut d'études technologiques se résume comme suit :

Assurer la formation de cadres moyens (licenciés) pour les besoins des entreprises aussi bien au niveau du secteur industriel (secondaire) que du secteur des services (tertiaire).

Promouvoir le recyclage et la formation continue au profit des cadres exerçant dans les entreprises (formation ponctuelle, formation continue etc...)

Mettre en place un partenariat avec les entreprises et les organisations professionnelles, participation des professionnels à la formation et à l'encadrement des étudiants, réalisation conjointe de programmes de recherches appliquées et de transfert de technologie.

L'implantation de l'ISET de Zaghouan constitue un about majeur pour le tissu économique et industriel de la région. En bref, l'ISET de Zaghouan va constituer un pôle de rayonnement technologique régional.

DEPARTEMENTS

L'ISET de Zaghouan, a pour objectif de former des licenciés capables de répondre aux besoins des entreprises régionales. L'établissement a pour vocation principale la formation dans les domaines secondaires et tertiaires suivants :

- 1) Génie des Procédés(GPr)
 - Procédés Chimiques (PC)
 - Procédés Agroalimentaires (PA)
- 2) Technologie de l'Informatique (TI)
 - Multimédia et Développement Web (MDW)
 - Réseaux et Services Informatique (RSI)
 - Développement des systèmes d'information (DSI)
- 3) Sciences Economiques et Gestion (AF)
 - Gestion Comptable et Financière(CF)
 - Gestion de la Production(Gp)



Institut Supérieur des Etudes Technologiques de Zaghouan Organigramme Général

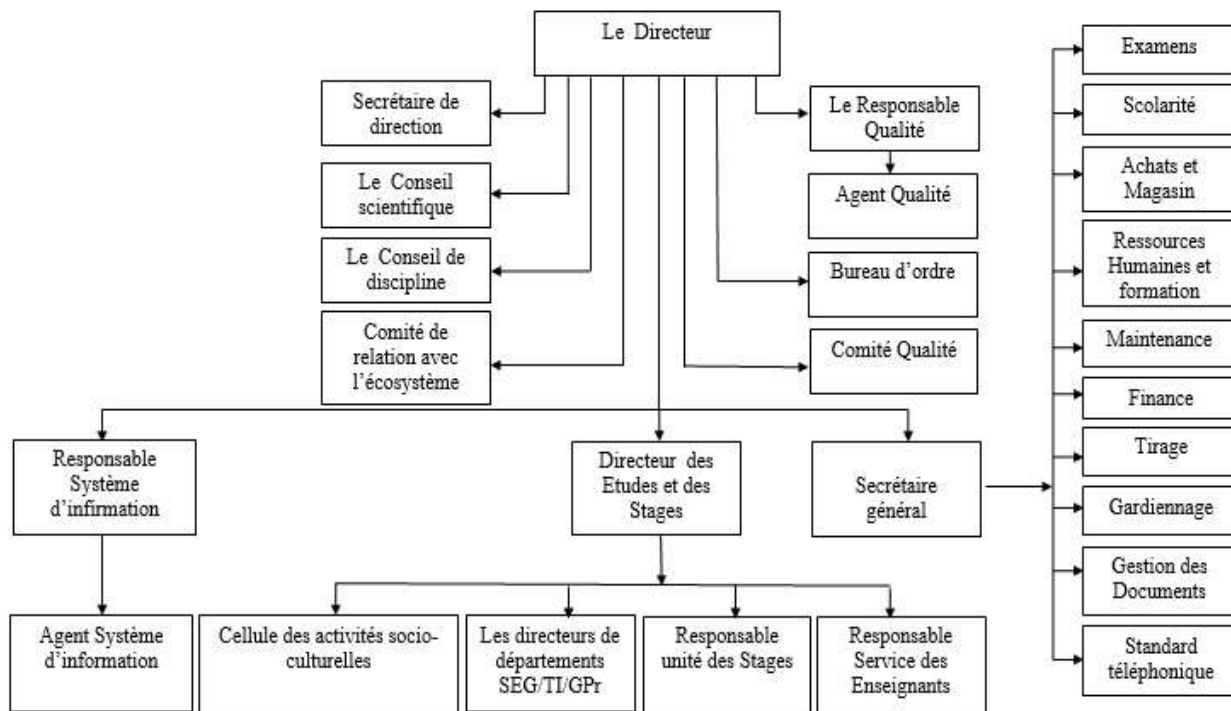


Figure 1 : Organigramme de l'ISET

2. Problématique

Les documents occupent une importance capitale dans les entreprises, d'où leur gestion doit être suivie minutieusement et prudemment. Ils contiennent des informations généralement de grande valeur. Donc une première étape nécessite de garder une version électronique de ces documents et puis ceux-ci doivent être intégrés dans un système de GED. Mais, avec toutes les informations et les fonctionnalités intégrées dans les GED actuels, il faudra, impérativement, trouver un moyen efficace de classer les documents numériques, suivant des méthodes bien définies et bien structurées. Dans cette visée, l'ISET de Zaghouan voudrait développer une solution GED qui s'occupe du classement des documents selon les deux modes Automatique et Semi-automatique. Le système de GED interviendra dans les étapes de classement et de stockage des documents afin d'automatiser ces tâches et d'obtenir un bon résultat.

3. Contexte du projet

Notre stage de fin d'études, effectué à l'ISET de Zaghouan, consiste à concevoir et réaliser une solution GED afin d'obtenir une méthode de classement et d'archivage qui mettra fin aux problèmes quotidiens de gestion des documents et faciliter la recherche et l'exploitation des documents.

Chapitre 2 : Etat de l'art

Introduction

Ce chapitre présente, en premier lieu le contexte du projet. Une deuxième partie sera consacrée pour l'étude de l'existant. Cette partie effectuera une étude sur les systèmes GED open-source actuels, pour aboutir à la problématique qui a suscité le besoin de notre projet et à la solution proposée.

Une autre partie sera consacrée à la présentation du système de GED Alfresco choisi et ses avantages par rapport à ses concurrents.

Une dernière partie de ce chapitre sera consacrée à la justification du choix de la solution adoptée pour mettre en place notre solution.

1. Les systèmes GED

1.1. Définition

1) La gestion électronique des documents (GED) :

La gestion électronique des documents désigne un procédé informatisé visant à organiser et à gérer des informations et des documents électroniques au sein d'une organisation. Le terme GED désigne également les logiciels qui permettent la gestion des contenus documentaires [2].

Il y a quatre étapes majeures dans une technique de GED : l'acquisition, le classement, le stockage et la diffusion.

2) Système de gestion de Contenu dans l'entreprise (ECM) :

La gestion de contenu d'entreprise (Entreprise Content Management) désigne également la structuration de l'ensemble des informations numériques dans l'entreprise [3]. Parmi les fonctionnalités d'un ECM nous citons, l'intégration d'une démarche de GED, le suivi des documents numérique et leur indexation, le workflow, le travail collaboratif (groupware),.

Il existe plusieurs ECM tels que SharePoint, Alfresco, Nuxeo.

1.2. Avantages de la GED

Le système de GED est une amélioration pour l'entreprise. Il permet l'archivage de la masse d'informations papiers et de la mettre en consultation avec, souvent, un outil de versionning.

En plus, il permet une diffusion beaucoup plus rapide et plus large, et donc l'entreprise va économiser en matière de coût d'espaces de stockages et d'impression et augmenter la productivité de ses employés [4].

1.3. Etapes de la GED

Les quatre étapes de la GED sont :

1) L'acquisition

C'est la première étape pour la mise en marche de GED. Elle permet la transformation des documents en papiers en une version numérique, en fonction du débit adéquat (carte d'acquisition scanner, etc.). A l'issue de cette phase de numérisation, l'entreprise définira ensuite une charte de nommage des documents pour que les avantages de la GED puissent en profiter largement.

2) Le classement

L'indexation constitue un élément clé à la description d'un document numérique. Ainsi, elle rend le document plus facile à exploiter. Il existe deux types d'indexations : l'indexation par type (formats du document), et l'indexation par mots clés (visant le contenu du document pour optimiser la recherche). L'entreprise doit définir une indexation relative à ses besoins, ceci étant très important pour l'organisation et le classement des documents dans le système de GED.

3) Le stockage

Il faut percevoir un support de stockage adapté au volume des documents. L'entreprise doit prendre des mesures de sécurité en cas de panne, d'incendie ou de perte. Il est impératif ainsi de planifier des sauvegardes automatiques et de différer les lieux des supports de stockages.

4) La diffusion des documents

La diffusion des documents se fait par plusieurs moyens : internet ou intranet, dossier partagé protocole FTP, CMIS, etc.

Certains éditeurs d'outils GED proposent une consultation client-léger, de l'entrepôt de données, via un navigateur web [5].

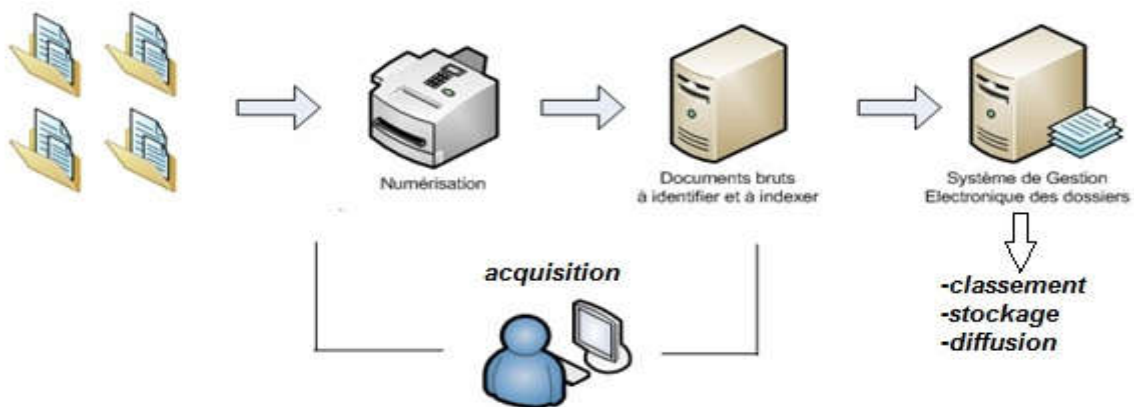


Figure 2 : Système GED

2. Présentation d'Alfresco

2.1. Définition

Alfresco est un système de gestion de contenu (ECM), open source en langage JAVA, créé en 2005 par John Newton et John Powell, les co-fondateurs d'Alfresco software, situé en Grande-Bretagne.

2.2. Historique

Alfresco est le produit d'une équipe composée du co-fondateur de Documentum et d'anciens d'Intervowen, deux entreprises leaders dans le domaine de la gestion des documents et qui ont comme but de trouver des solutions pour les documents qui sont non structurés, depuis 1990. En effet, Alfresco n'est pas né pour répliquer une solution open source vis-à-vis des solutions propriétaires comme Microsoft SharePoint, FileNet d'IBM ou Documentum, mais même de les surpasser [6].

2.3. Les versions d'Alfresco

Alfresco se présente sous forme de trois versions :

- Alfresco Community:

Une version téléchargeable, modifiable et utilisable, distribuée en licence libre (LGPL)

- Alfresco Enterprise:

Une version commerciale, démarque essentiellement de la version Community, et elle y rajoute le support technique et d'autres fonctionnalités complétant une solution GED. Ainsi nous pouvons tester cette version, Alfresco offre 30 jours gratuits puis il oblige l'achat [7].

- Alfresco Version Cloud

Cette version permet de partager le contenu tout en centralisant les données sur les serveurs Cloud (Amazon Web Services aux états unis). Elle offre une gestion complète et un niveau de sécurité supérieur.

2.4. Les avantages d'Alfresco

Alfresco représente la GED par excellence puisqu'il offre une grande gamme de produits, (une version gratuite, version payante, une application iPad, iPhone et Android gratuite pour pouvoir même gérer à distance). En outre, « Alfresco Share » facilite la création des sites.

3. Alfresco vis-à-vis de ses concurrents

3.1. Nuxeo

Est une plateforme open source de gestion de contenu qui permet aux architectes et aux développeurs de faciliter, concevoir, déployer et exploiter les applications métiers. Sur site ou sur le Cloud, la technologie d'Enterprise Content Management (ECM) de Nuxeo offre des solutions intégrées de gestion de documents.

➤ Points forts

Basée sur Java et doté d'une architecture logicielle moderne, Nuxeo est conçu pour être modulaire et extensible, se démarquant ainsi des solutions d'ECM traditionnelles.

➤ Points Faibles

L'utilisation semble hautement compliquée pour l'utilisateur simple. La configuration est compliquée vu que nous devons modifier des fichiers XML[8].



Figure 3 :Le logo de Nuxeo

3.2. Knowledge Tree

Knowledge tree est une solution de GED open-source, disponible également en version gratuite et payante. Elle utilise les classiques PHP MySQL. Ainsi, elle propose les fonctionnalités qu'on doit attendre de ce type de logiciel.

➤ Points forts :

Knowledge tree est facile à utiliser doté d'une interface agréable offrant des fonctionnalités suffisantes.

➤ Points faibles :

Knowledge tree n'intègre ni Microsoft Office ni la signature électronique, dans sa version gratuite[9].



Knowledge Tree

Figure 4 : Le logo de Knowledge Tree

3.3. Exo-Dms (Document Management System)

Exo est un éditeur français fondé en 2003 par le français Benjamin Mestralet, notamment via un financement du DoD (ministère de la défense américain). La société est de nos jours présente en France, en Ukraine, au Vietnam, et en Tunisie.

Exo-Platform fournit une suite de logiciels dont nous citons : eXo Portal, la gestion documentaire (eXo DMS), la gestion de contenu web (eXo WCM).

Au sein de cette suite, eXo DMS, qui est écrit en Java, occupe une place centrale, étant donné qu'il s'agit de la brique destinée à stocker l'ensemble des fichiers.

➤ Points forts :

eXo DMS offre une interface standard qui intègre nativement les facilités de l'explorateur Windows : cliquer- glisser, raccourci clavier, affichage selon plusieurs modes et autres[10].

Il permet l'intégration bureautique bien poussée, avec un plug-in spécifique pour MS Office et une intégration Open Office.

Il offre la possibilité d'adapter les interfaces utilisateurs en fonction des profils.

➤ Points faibles :

Exo-Platform permet d'attacher directement à exo-Platform qui est un portail d'entreprise présentant beaucoup de fonctionnalités, et il ne peut pas être pris en compte hors de cette suite.



Figure 5 : Le logo d'Exo

4. Choix d'Alfresco

Après cette étude des principaux systèmes GED open-source, nous avons choisi Alfresco pour sa mise en valeur sur le travail collaboratif, et ses fonctionnalités complètes de la GED. Il supporte différentes plateformes : Windows, Mac et Linux.

Parmi les nouveautés de la version 5 d'Alfresco que nous allons utiliser nous citons :

- L'interactions sociales via des chaînes de publications sur les réseaux sociaux (Facebook, Tweeter).
- De nouvelles options de sociabilité présentes.
- L'accès via mobile ou tablette et aussi la modification et la, consultation de contenu.
- La collaboration avec des utilisateurs via Alfresco Cloud.



Figure 6 :Le logo d'Alfresco

5. Utilisation d'Alfresco

Le téléchargement de la version gratuite d'Alfresco Community Edition se fait sur le lien : <https://www.alfresco.com/fr/products/community/download>

Après son installation (voir Chapitre réalisation) Alfresco peut être accédé par deux moyens différents :

- L'interface web : via un navigateur web (Internet Explorer, Chrome, Opera Firefox, ...)
- L'interface WebDAV : par un logiciel spécifique WebDAV (Web-based Distributed Authoring and Versioning).

WebDAV : C'est une extension du protocole http, son rôle est de rendre possible l'écriture à travers le web et pas seulement la lecture (ressemble au protocole FTP).

L'interface web offre le plus de possibilités : la possibilité d'importer des fichiers, de créer des espaces, de partager des documents, d'interagir avec d'autres utilisateurs, etc.

L'interface WebDAV est plus limitée vu qu'elle se présente sous la forme d'un explorateur de fichier.

1) Authentification :

En saisissant l'url suivant : <http://127.0.0.1:8080/alfresco/> dans le navigateur, une fenêtre d'authentification apparaisse. Par défaut, nous disposons d'un seul utilisateur, l'administrateur.



Figure 7 : Page d'authentification

2) Menu principal :

Le menu principal d'Alfresco contient les éléments suivants :

Online documentation, Alfresco share, Alfresco WebDav, Alfresco WebScripts (administrateur seulement), Alfresco Administration Console (administrateur seulement), Alfresco Forums, Alfresco Jira, CMIS 1.0 Atompub Service Document, CMIS 1.0 Web Services WSDL Document, CMIS 1.1 Atompub Service Document, CMIS 1.1 Browser Biding URL.

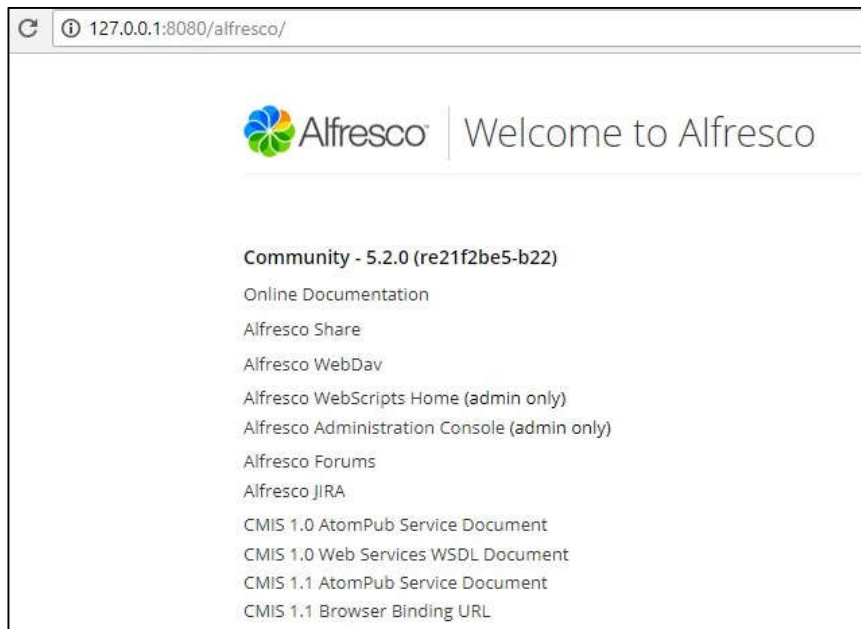


Figure 8 : Menu principal d'alfresco

3) Tableau de bord :

Le tableau de bord d'Alfresco share présente beaucoup de fonctionnalités que nous allons aborder. Ainsi il peut être configuré et l'utilisateur peut choisir les éléments à afficher sur son tableau tels que : les tâches à réaliser, guide de démarrage, liste des espaces, et autres.

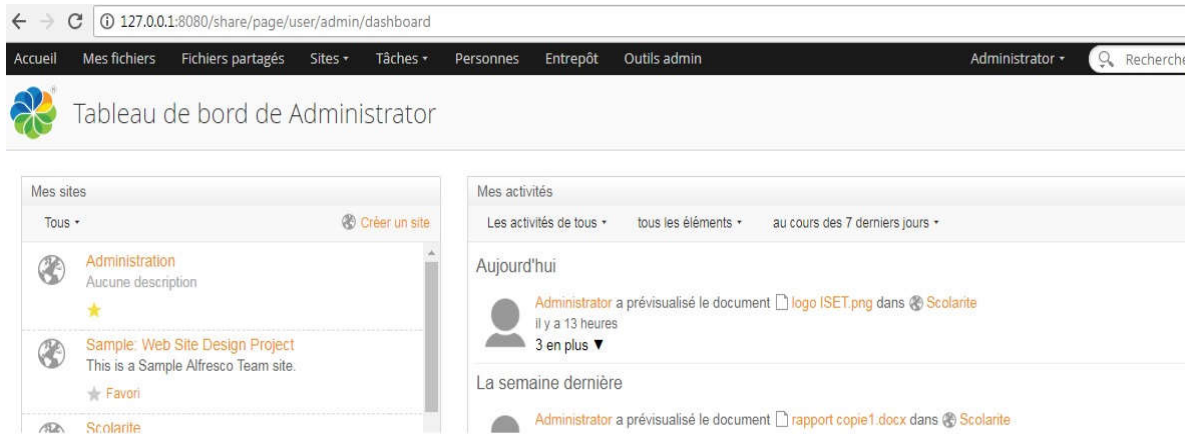


Figure 9 : Tableau de bord Alfresco share

4) Espaces de travail :

Un espace de travail (ou espace documentaire) est constitué d'un ensemble de sous-espaces (dossiers) ou de documents. Un simple clic sur le titre d'un contenu ou d'un espace permet d'y accéder directement. Dans le cas d'un contenu, cela donne la main pour télécharger le document.

Nous pouvons gérer la vue, créer un espace, ajouter un contenu, avec le menu au-dessus de l'espace principal. Les actions disponibles sur les contenus et les sous-espaces varient selon le rôle qui nous a été attribué (voir Figure 11).



Figure 10 : Espace de travail

5) Les rôles :

Les rôles procurent des droits particuliers à l'utilisateur pour un espace ou un contenu donné. Dans Alfresco, il existe 4 rôles dont les droits sont résumés dans la figure suivante :

	Lecteur	Contributeur	Collaborateur	Gestionnaire
Voir tous les espaces et continue	X	X	X	X
Mettre à jour/Editer			X	X
Ajouter des contenus		X	X	X
Gérer les droits			X	X
S'approprier un contenu				X

Figure 11 : Tableau de liste des rôles

Le rôle du Gestionnaire qui a, tous les droits, est généralement donné à l'administrateur des espaces. Le gestionnaire affecte ensuite les rôles aux autres utilisateurs du système selon les droits qu'il souhaite leur attribuer[11].

6) La gestion des versions :

Le Versionning n'est pas activé par défaut. Il doit être 'activer en cliquant sur « Activer la gestion des versions» sur le vue « Détails » du contenu.

Lors de l'activation du Versionning, la version 0.1 s'affecte au document. Ainsi, chaque modification ou édition du contenu par les utilisateurs, une fenêtre s'affiche pour affecter la nouvelle version.



Figure 12 : Le Versionning de documents

7) Les actions :

Les actions dans Alfresco sont des actions basiques réalisables par l'utilisateur, telle que copier, coller un document, gérer les droits sur un espace, ajouter un aspect à un document, envoyer un courriel, etc. En outre, nous trouvons un assistant pour création des actions avancées en appuyant sur « Lancer une action » depuis la vue des actions.

Parmi les actions avancées les plus notables:

- Déplacer les éléments vers une destination, copier les éléments vers une destination, envoyer un courriel à des utilisateurs, etc[12].

8) Administration :

Dans Alfresco, l'administration peut se faire sur plusieurs niveaux, nous allons aborder une administration basique.

Via la barre de menu, nous accédons à la console d'administration par un clic sur Outils admin.

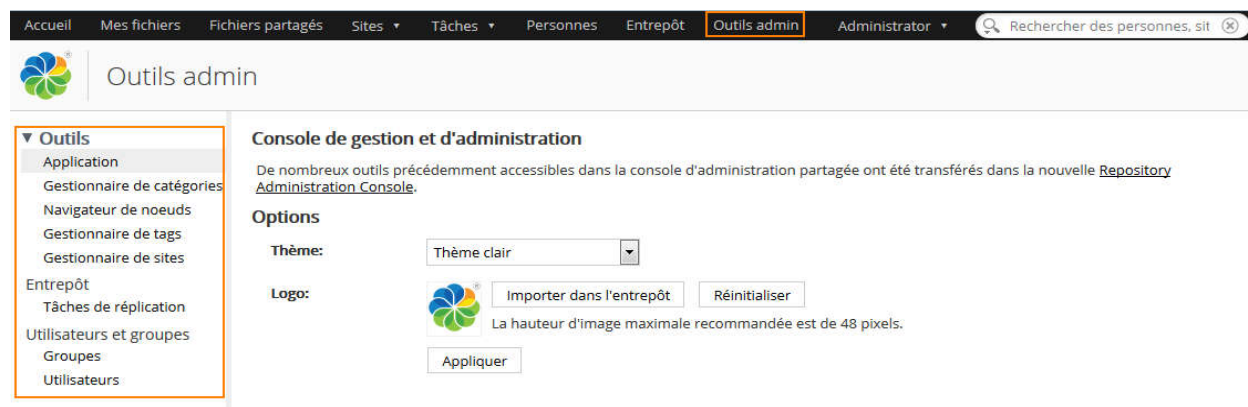


Figure 13 : La console d'administration

6. Personnalisation d'Alfresco

6.1. Méthodologie et technologies adoptées

Alfresco est un projet open-source, qui utilise des standards ouverts, basé sur la technologie J2EE.

Alfresco a développé une architecture moderne utilisant les derniers outils open-source pour l'optimisation des performances, et de la programmation orientée aspects (AOP), notamment dans le Framework Spring, facilitant ainsi la modularité et l'adaptabilité de l'application.

Spring est un conteneur dit « soft », c'est-à-dire une infrastructure semblable à un serveur d'application J2EE. Il prend donc en charge la création d'objets et leur mise en relation par l'intermédiaire d'un fichier de configuration qui décrit les objets à construire et les relations de dépendance entre eux.

Le grand avantage par rapport aux serveurs d'application est qu'avec Spring, les classes n'ont pas besoin d'implémenter une quelconque interface pour être prises en charge par le Framework (au contraire d'autres serveurs d'applications J2EE et des EJBs). C'est en ce sens que Spring est qualifié de conteneur soft[13].

6.2. Architecture Alfresco

L'architecture d'Alfresco, se repose sur quatre couches, nous les détaillons comme suit :

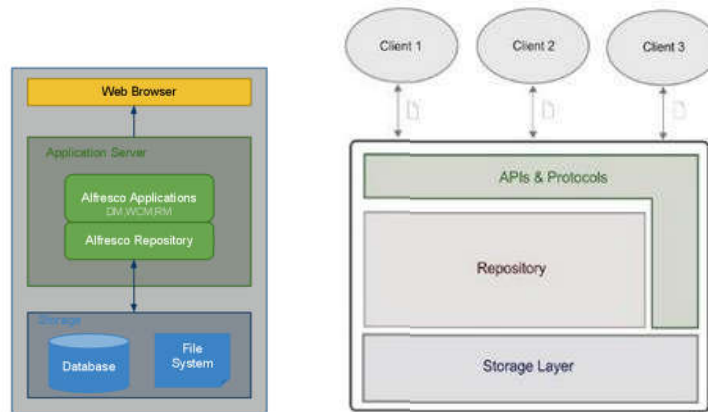


Figure 14 : L'architecture d'Alfresco

1) Couche de stockage (Storage Layer) :

Composée d'un SGBD relationnel (par défaut PostgreSQL) et un système de fichiers.

Un contenu dans Alfresco, est composé d'un document plus des métadonnées. Il utilise un système de fichier binaire (.bin) pour être indexé sur les moteurs de recherche open-sources Solr ou Lucene (selon de la version installée)[14].

- Modèle de données :

Les modèles de données dans Alfresco sont souples, personnalisables et extensibles.

Les composants d'un modèle :

- Le type du contenu : spécifie le contenu (un contenu ne peut avoir plus qu'un seul type).
Exemple : Devis, facture, rapport
- L'aspect : Qualifie un contenu (un contenu peut avoir un ou plusieurs aspects), donc la définition est réutilisable dans d'autres types. Exemple : Etudiant, Clients.
- Les propriétés et les associations : peuvent être affectées pour définir un type ou un aspect.

2) Couche d'entrepôt de données (Alfresco Repository) :

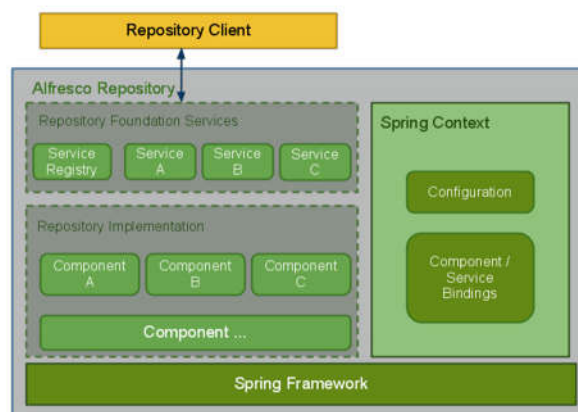


Figure 15 : Alfresco Repository

Cette couche se compose d'un moteur de gestion de contenu et des services associés. Les services reposent sur le Framework Spring. Ils sont exposés par des interfaces publics (Repository Foundation Services) et des composants qui les implémentent (Repository Implementation). Parmi les services qui permettant le pilotage du contenu stockés dans l'entrepôt Alfresco, nous citons principalement :

- Les actions et les règles :

Les règles de contenu s'appliquent aux espaces. Elles leurs ajoutent de l'intelligence. Nous pouvons les classer suivant trois aspects :

- Un évènement déclencheur
- Un ensemble de conditions
- Une action à appliquer

- Audit :

Permet la traçabilité des contenus et le suivi.

- Workflow :

Alfresco intègre un moteur de Workflow BPM (Business process management) il comprend une étape (acceptation ou refus) et deux utilisateurs (celui qui dépose le document et celui qui refuse ou valide).

- Sécurité :

Les permissions définissent les droits d'accès et les droits d'opérations sur les contenus. Ils se basent sur le système ACEGI de Spring et sont conçus pour définir des rôles (Gestionnaire, Collaborateur, Contributeur, et Lecteur).

Parmi les apports en sécurité, le verrouillage lors l'édition d'un document pour limiter les risques de conflit[15].

3) Couche d'APIs et protocoles :

Les applications clientes communiquent avec Alfresco via les APIs et les protocoles. L'exposition de ces protocoles et APIs assurent l'interopérabilité d'Alfresco vis-à-vis des différentes applications du système d'information.

L'entrepôt d'Alfresco est accessible par de différents protocoles, sans installation sur les postes des clients.

Les protocoles : FTP, webDAV, LDAP, NFS, CIFS, CMIS.

Les APIs : SOAP, web scripts, java API[16].

4) Couche des applications Clients :

Deux applications web sont déployées dans le serveur d'applications :

- alfresco.war : l'explorateur d'Alfresco
- share.war : l'espace collaboratif d'Alfresco

7. Intégration du module « Activiti »

La qualité des processus d'une entreprise constitue un avantage concurrentiel majeur. Activiti offre la possibilité de modéliser, créer et déployer des processus métiers, et définir des formulaires associés aux

tâches métiers. Activiti se base sur le moteur de Workflow BPMN (Business Process Management Notation), permettant de déployer des circuits de validation avancés.

Nous avons choisi d'installer cet outil pour modéliser les processus métier des différents services de l'établissement et ainsi dégager les principaux documents et Workflow de notre système GED[17].

C'est un module supplémentaire auxiliaire à Alfresco, après son installation sur le serveur Tomcat, il est accessible séparément sur l'adresse : <http://127.0.0.1:8080/activiti-app/>

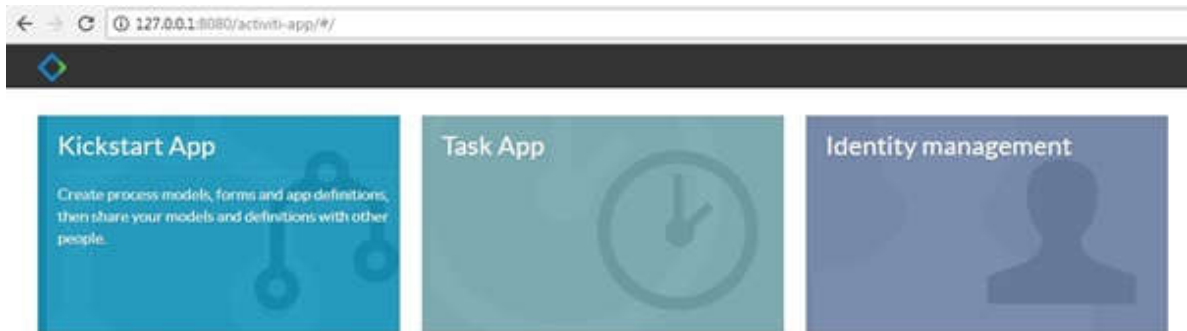


Figure 16 : Menu principal d'Activiti

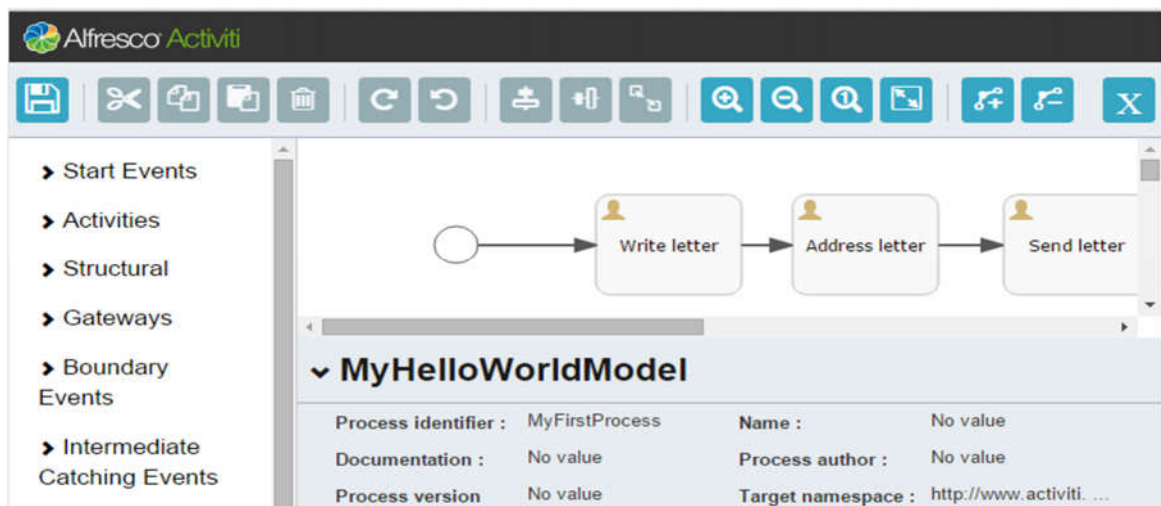


Figure 17 : Menu principal d'Activiti

8. Solution adoptée

D'après cette étude théorique et pratique sur Alfresco, il était impératif de choisir la solution la plus convenable à l'implémentation de notre solution de GED. Pour y parvenir nous devons faire des personnalisations dans le code source du système et y rajouter de nouveaux modèles de données qui nous aideront à mieux établir la solution et atteindre nos objectifs fixés.

Conclusion

Dans ce chapitre nous avons décrit le fonctionnement du système de gestion électronique de documents et son apport à l'établissement. Nous avons également présenté Alfresco comme solution à adopter, et ainsi de découvrir des fonctionnalités utiles. Le chapitre 2 présentera l'aspect fonctionnel, le contexte du projet et ses objectifs .

Chapitre 3 : Analyse et spécification des Besoins

Introduction

Cette étape va nous permettre d'élaborer une typologie de documents produits dans l'institut et viser les différentes contraintes techniques et fonctionnelles. L'objectif de notre travail est de concevoir une application serveur de GED Alfresco facilitant la gestion de documents (automatisation des processus acquisition, extraction des métadonnées, gérer règles de classement).

1. Objectifs

Nous devons préciser les attentes particulières des utilisateurs, concernant le système à mettre en place. Pour y parvenir, les objectifs ci-dessous ont été fixés :

- Etudier l'existant en matière d'organisation des documents, analyser les problèmes vécus, et attribuer des modèles de données propres à l'établissement.
- Analyse et automatisation des processus métiers.
- Proposer une solution optimale qui facilite les différentes tâches de gestion des documents en outre la tâche de classification et d'archivage.
- Automatiser la gestion des documents et faire naître un aspect d'intelligence à l'application, par l'affectation des modèles personnalisés (des types des aspects, des métadonnées).

2. Spécification des besoins fonctionnels

Le projet consiste à mettre en place une solution GED Alfresco. En premier lieu, elle doit permettre de modéliser les processus métiers des utilisateurs du système. En deuxième lieu, elle doit permettre aux utilisateurs de classer les documents suivant des règles de classement qui s'exécutent en deux modes (Automatique, Semi-Automatique).

2.1. Besoins des utilisateurs

- S'authentifier avec un compte au serveur Alfresco. Si l'utilisateur ne possède pas déjà un compte Alfresco, il doit contacter l'administrateur pour lui créer un compte.
- Lister les règles de classement et leurs propriétés.
- Démarrer ou Stopper une ou plusieurs règles de classements.
- Manipuler les documents (ajout, modification des modèles types et des métadonnées).
- Consulter l'espace personnel via le navigateur web.

2.2. Besoins des administrateurs

- Gérer les règles de classement (Création, Modification, Suppression).
- Gérer les processus métiers et tâches (workflow).

- Gérer les modèles de données (Types, aspects, des métadonnées).
- Administrer modifier les paramètres de la connexion (Serveur Alfresco).
- Affecter des types de comptes aux utilisateurs

3. Spécification des besoins non fonctionnels

En plus des besoins fonctionnels, plusieurs considérations et contraintes additionnelles doivent être prises en compte au cours de la réalisation de notre projet. Ces besoins se résument au niveau de :

- L'interface utilisateur simple et facile à utiliser qui montre proprement les fonctionnalités du système.
- La charte graphique qui inclut l'insertion du logo de l'ISET de Zaghouan.
- La montée en charge en particulier dans le nombre d'utilisateurs (Scalabilité du système).
- La fourniture d'un manuel d'utilisation décrivant les fonctionnalités de l'application.
- La portabilité de la solution avec les différentes plateformes (Linux, Windows, Mac).

4. Modélisation des besoins

4.1. Diagramme de paquetages

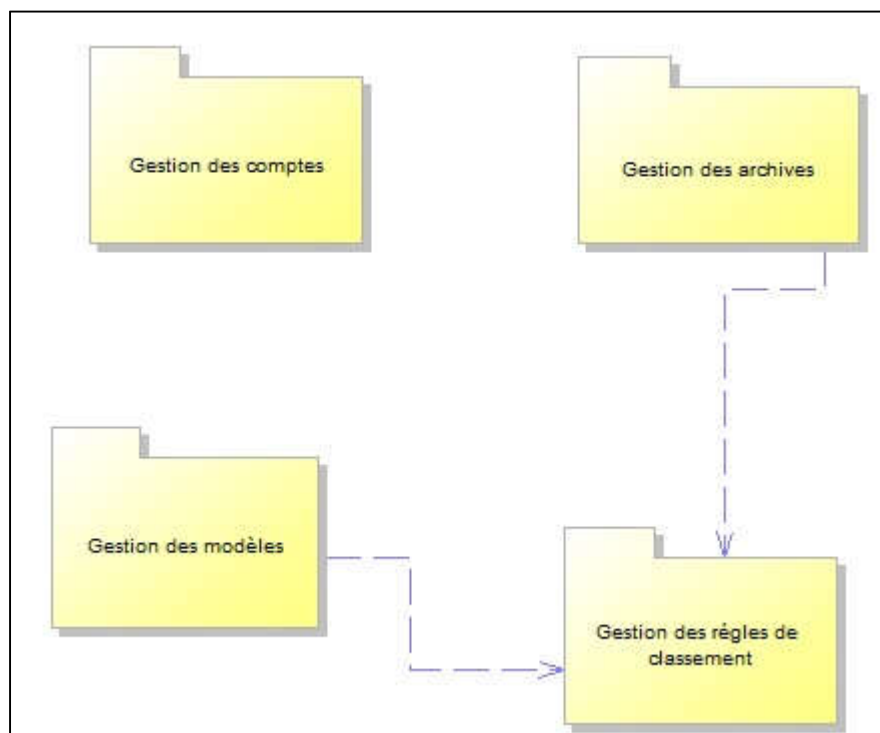


Figure 18 : Diagramme de paquetages

Ce diagramme décrit exhaustivement les exigences fonctionnelles du système. Il se compose de quatre paquetages et englobe les fonctionnalités essentielles du système. Nous avons identifié les quatre paquetages suivants

- Paquetage 1 : Gestion des règles de classement
- Paquetage 2 : Gestion des modèles
- Paquetage 3 : Gestion des comptes
- Paquetage 4 : Gestion des archives

4.2. Diagrammes des cas d'utilisation

4.2.1. Paquetage « Gestion des règles de Classement »

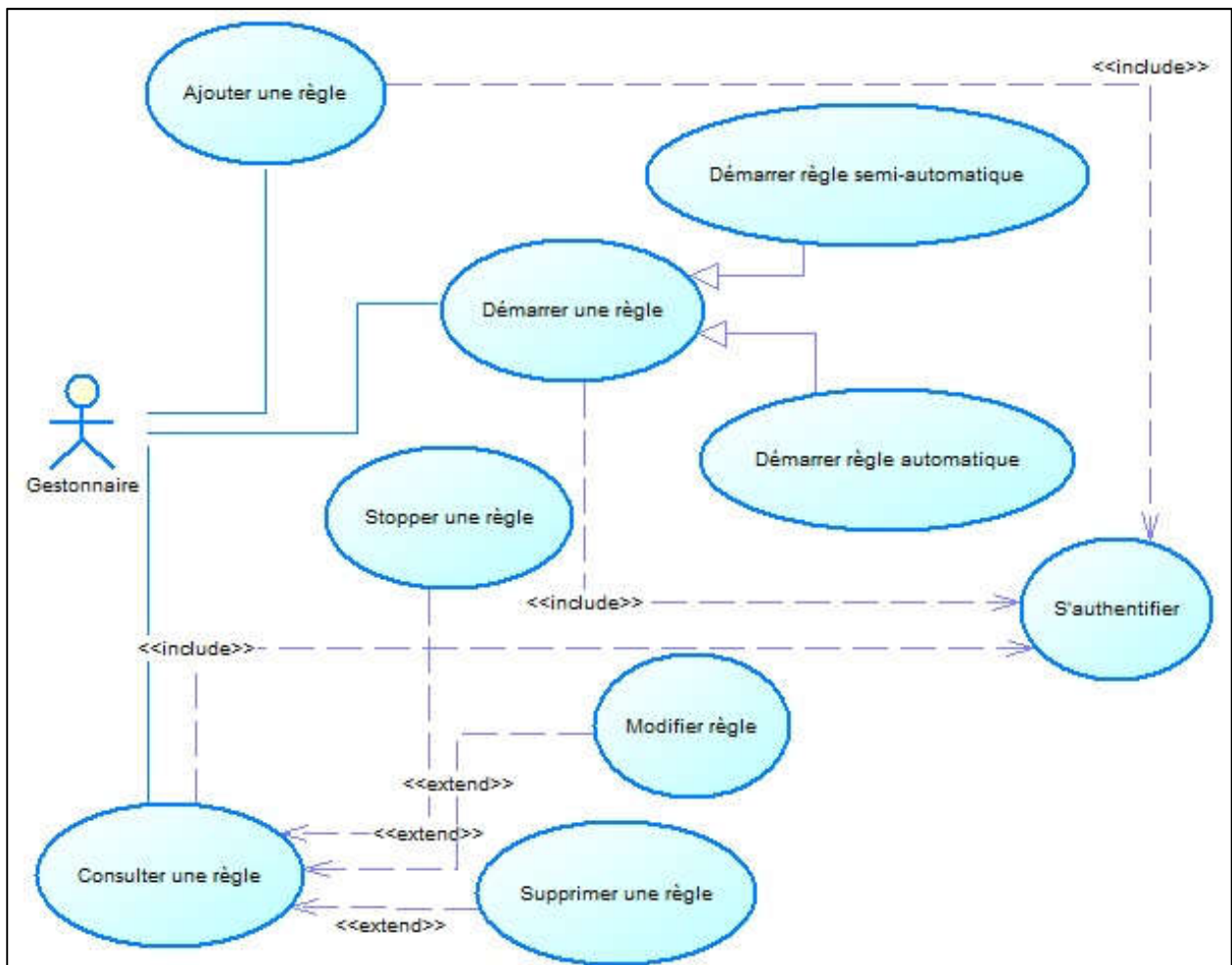


Figure 19 : Diagramme de cas d'utilisation « Gestion des règles de classement »

Use Cases (Gestionnaire) :

- S'authentifier
- Démarrer une règle
- Stopper une règle
- Consulter une règle
- Ajouter une règle
- Supprimer une règle
- Modifier une règle

4.2.2 Paquetage Gestion des modèles

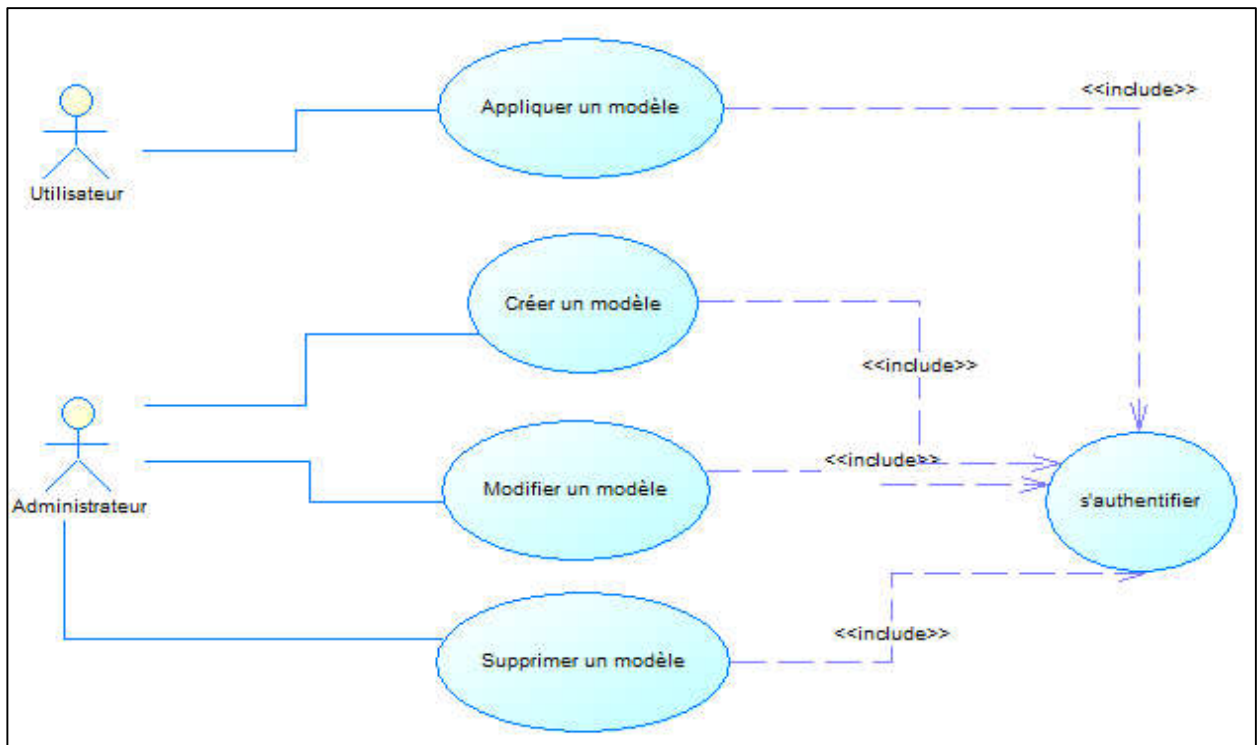


Figure 20 : Diagramme de cas d'utilisation « Gestion des modèles types »

Use Cases (Utilisateur) :

- S'authentifier
- Appliquer modèles.

Use Cases (Administrateur) :

- S'authentifier
- Créer un modèle
- Supprimer modèle type
- Modifier modèle type

4.2.3 Paquetage Gestion de compte

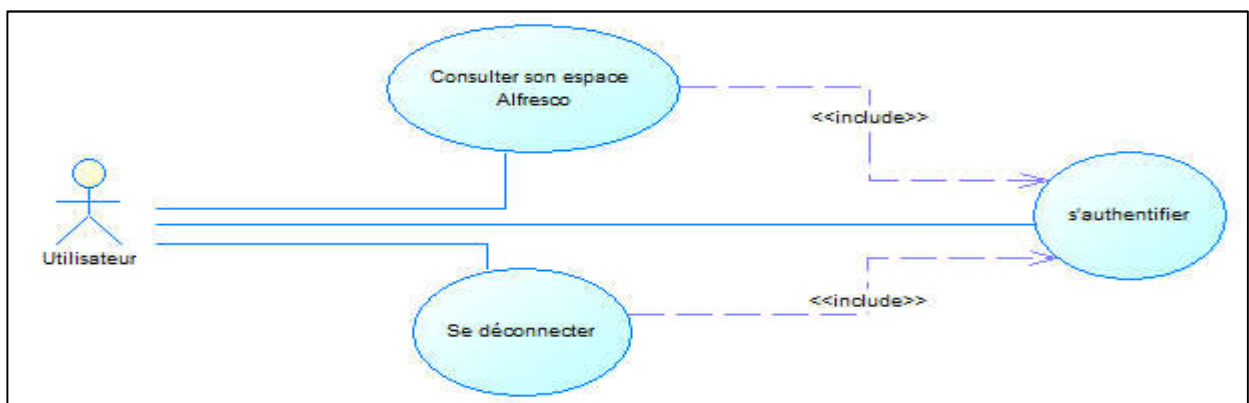


Figure 21 : Diagramme de cas d'utilisation « Gestion de compte »

Use Cases (Utilisateur) :

- S'authentifier
- Consulter l'espace Alfresco
- Se déconnecter

4.2.4. Paquetage Gestion des archives

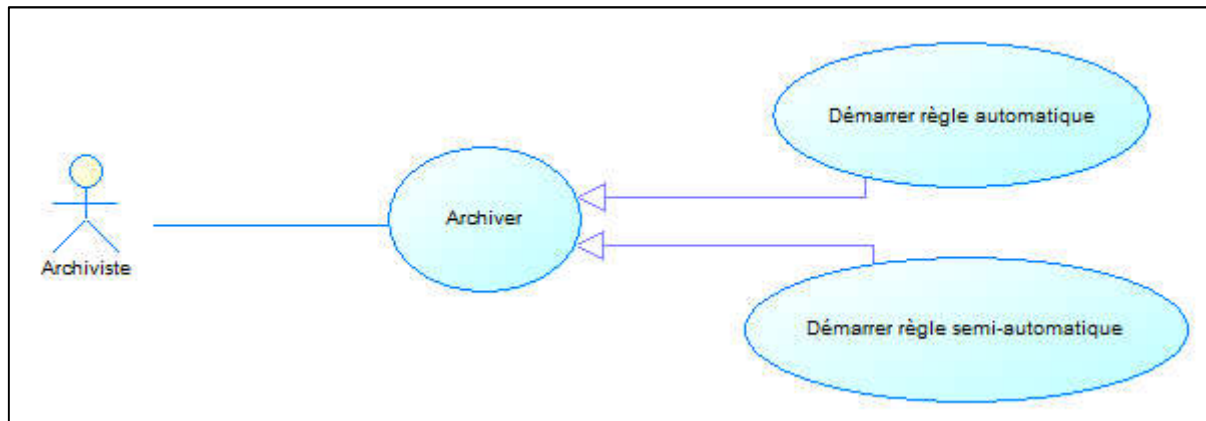


Figure 22 : Diagramme de cas d'utilisation « Gestion des archives »

Use Cases (Archiviste) :

- Archiver

4.3. Description des cas d'utilisation

Dans cette section nous allons présenter les différents scénarios d'utilisation et les interactions système-utilisateur, avec des diagrammes de séquences pour décrire les différents cas d'utilisations déjà cités.

4.3.1. Descriptions textuelles

- Cas d'utilisation « S'authentifier »

Nom : s'authentifier

Objectif : Permet d'établir une connexion au serveur Alfresco et de s'identifier (Utilisateur normal / Administrateur) pour accéder à l'espace personnel ou à l'historique

Acteurs : utilisateur, administrateur

Précondition : L'utilisateur doit avoir un compte Alfresco.

Scénario nominal :

- 1) L'utilisateur remplit le formulaire d'identification avec son login et son mot de passe.
- 2) L'utilisateur clique sur « Connexion ».
- 3) Le système vérifie les informations entrées par l'utilisateur
- 4) La page d'accueil s'affiche

Scénario d'exception :

E1. Si les informations de connexion sont invalides

L'enchaînement E1 démarre au point 3. Le système affiche un message d'erreur et le scénario nominal reprend à l'action 1.

Post-Condition : Une session est ouverte, l'utilisateur peut ainsi choisir parmi les fonctionnalités visibles pour lui sur l'interface.

- Cas d'utilisation « Ajouter une règle »

Nom : Ajouter une règle de classement

Objectif : Permet d'ajouter une nouvelle règle de classement, de définir ses propriétés et son mode de classement (automatique ou semi-automatique).

Acteur : Gestionnaire

Précondition : L'utilisateur se connecte en tant que Gestionnaire.

Scénario nominal :

- 1) Le Gestionnaire se connecte avec un compte administrateur.
- 2) Le Gestionnaire accède à l'interface de gestion des règles de classement.
- 3) Le Gestionnaire clique sur le bouton « Ajouter une règle »,
- 4) Le système affiche un formulaire
- 5) Le Gestionnaire remplit les propriétés d'une règle (Nom de la règle, Mode de classement, Répertoire de numérisation, Format des documents numérisés, Format de documents classés, types et aspects associés au fichier).
- 6) Le Gestionnaire enregistre la règle.
- 7) Le système affiche l'ensemble des règles disponibles.

Scénario d'exception :

- 1) Un message d'erreur est affiché si on clique sur « enregistrer règle » sans remplir correctement les propriétés nécessaires de la règle.
- 2) Les paramètres de la base de données doivent être corrects et la connexion établie pour ajouter une règle.

Post-Condition : Une nouvelle règle de classement est créée.

- Cas d'utilisation « Ajouter un modèle »

Nom : Ajouter un modèle.

Objectif : Permet d'ajouter un nouveau modèle qui s'introduit dans les propriétés des fichiers. Il peut comprendre des types et des aspects qui affectent les règles de classement semi-automatique et l'indexation des archives.

Acteur : administrateur

Précondition : Administrateur identifié.

Scénario :

- 1) L'utilisateur se connecte avec un compte administrateur.
- 2) L'administrateur accède à l'interface de gestion des modèles.
- 3) L'administrateur clique sur le bouton « Ajouter un modèle ».
- 4) Le système affiche un formulaire
- 5) L'administrateur saisit les propriétés d'un modèle (Nom de modèle, création des types et d'aspects).
- 6) L'administrateur valide les données saisies,
- 7) Le système affiche l'ensemble des modèles disponibles.

Exception :

- 1) Un message d'erreur est affiché si on clique sur « enregistrer modèle » sans remplir correctement les propriétés nécessaires.
- 2) Les paramètres de la base de données doivent être corrects et la connexion établie afin d'ajouter un nouveau modèle.

Post-Condition : Un nouveau modèle est créé.

4.3.2. Les diagrammes de séquences

- Cas d'utilisation « Démarrer une règle de classement »

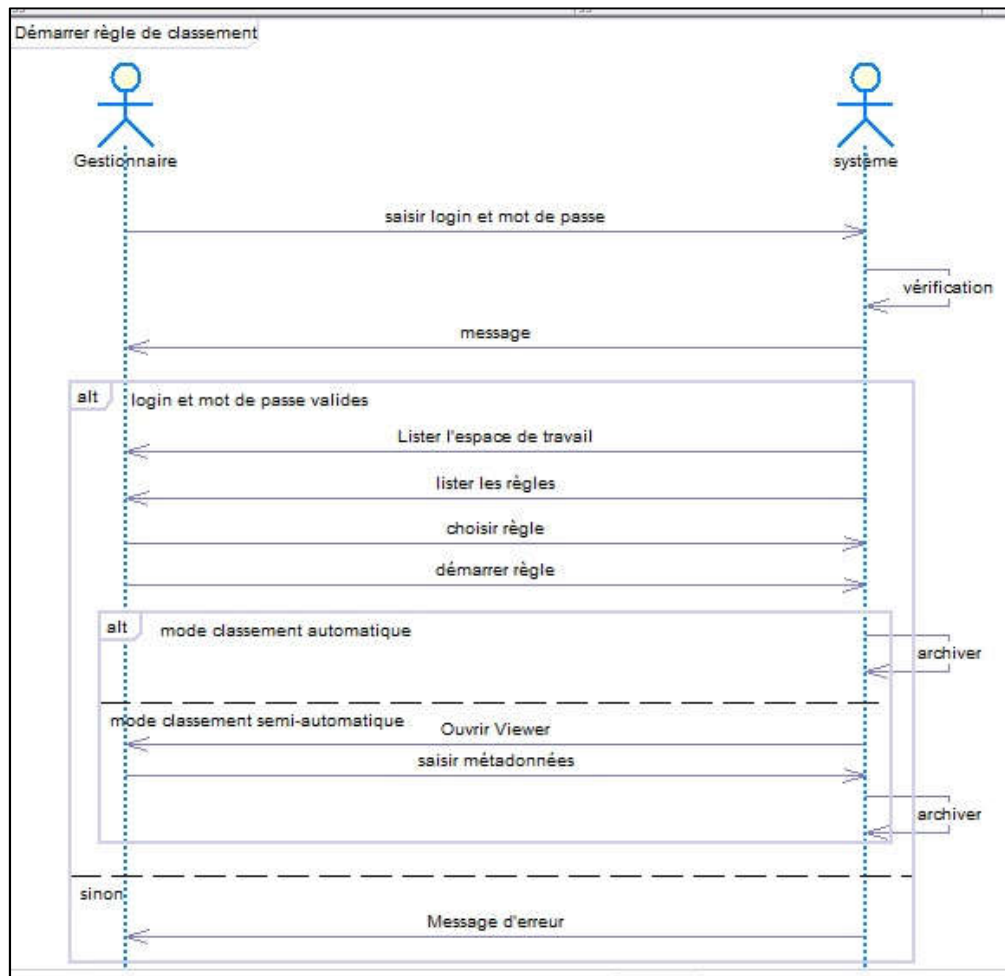


Figure 23 : Diagramme de séquence « Démarrer une règle de classement »

- Cas d'utilisation « Consulter son espace Alfresco »

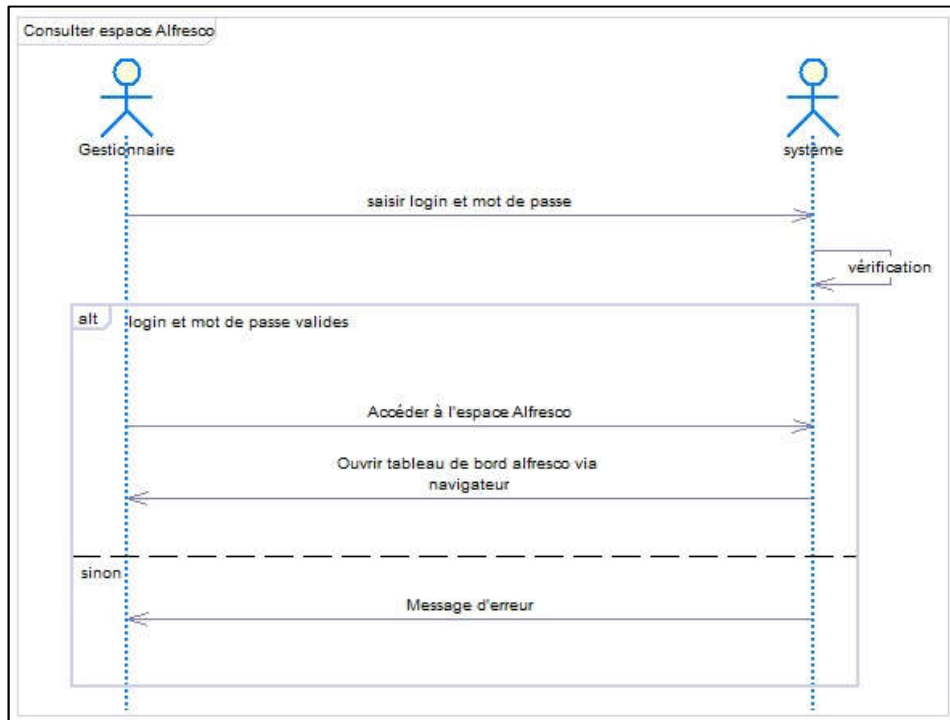


Figure 24 : Diagramme de séquence « Consulter son espace Alfresco »

4.3.3. Les diagrammes d'activités

- Cas d'utilisation « Stopper une règle de classement »

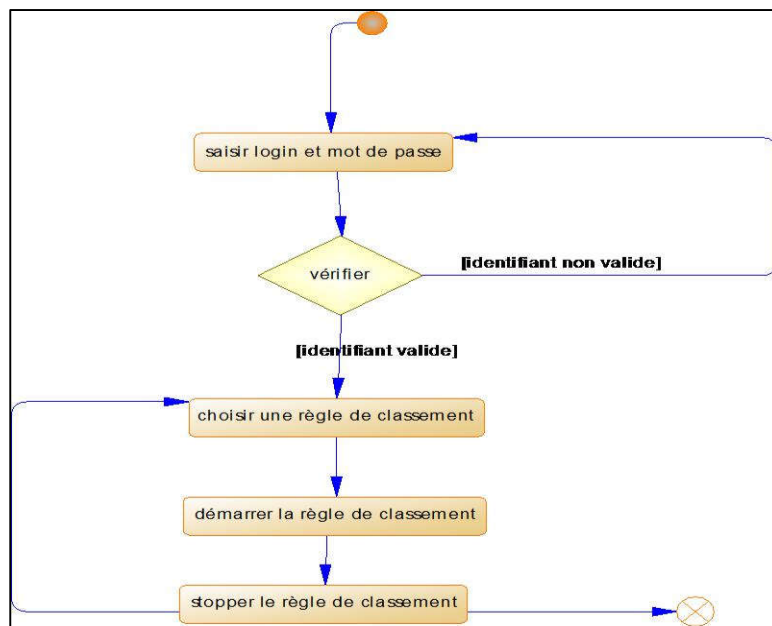


Figure 25 : Diagramme d'activités « Stopper une règle de classement »

- Cas d'utilisation « archiver les documents »

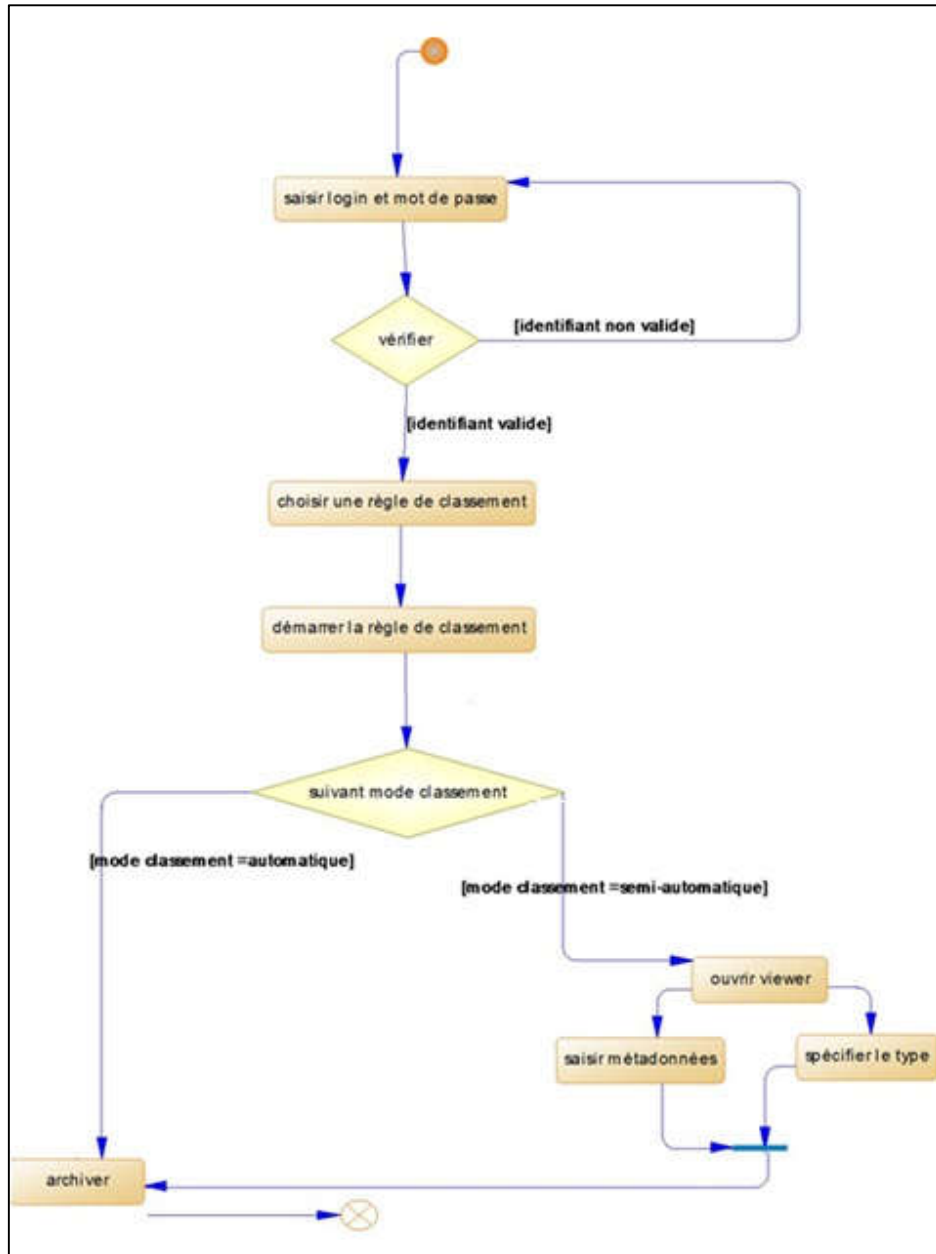


Figure 26 : Diagramme d'activités « archiver les documents »

Conclusion

Dans ce chapitre, nous avons présenté les besoins auxquels le système doit répondre. Nous avons aussi décrit les différents cas d'utilisation du système et nous les avons formellement illustrés par des diagrammes de cas d'utilisation UML.

Nous avons aussi achevé la partie spécification et analyse de ce projet. Dans le chapitre suivant nous allons passer à la partie de conception. Le chapitre suivant nous mettra dans le cadre de la conception détaillée du projet.

Chapitre 4 : Conception

Introduction

Afin d'atteindre les résultats envisagés, nous avons organisé nos besoins, exprimés dans le chapitre spécification, selon une méthodologie de conception qui va faciliter la phase de réalisation. Après la mention de l'architecture adoptée pour concevoir notre solution, nous présenterons les diagrammes relatifs.

1. Conception globale

A cette étape nous traitons la conception préliminaire qui consiste à l'intégration du modèle d'analyse fonctionnelle dans l'architecture technique. Ensuite, nous allons passer à la conception détaillée qui décrit la façon de réaliser chaque objet du système.

1.1. Schéma d'architecture

Nous allons utiliser une architecture en deux tiers cette architecture est parfaitement bien adaptée aux besoins des systèmes départementaux, dans la mesure où les concepts et les processus manipulés n'existent qu'une seule fois au sein d'un département de l'établissement.

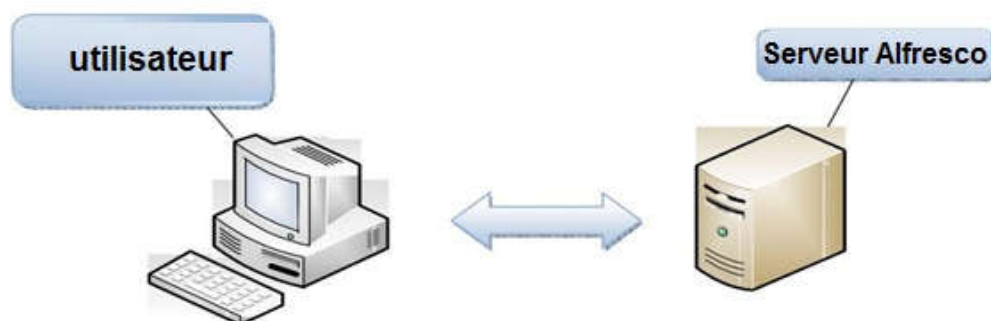


Figure 27 : Architecture deux tiers

Cette architecture s'articule autour d'une architecture client-serveur dans sa configuration la plus basique, c'est une architecture traditionnelle dans laquelle le client demande une ressource et le serveur la lui fournit. Cela signifie que le serveur ne fait pas appel à d'autres applications afin de fournir le service.

Dans ce cas la base de données est installée une seule fois dans un serveur et l'application et elle ne doit pas être installée dans tous les postes des clients. Les clients accèdent au serveur via le réseau. La figure ci-dessus décrit l'aperçu de cette architecture.

Avantages de la solution :

- Pas de duplication de données sur les postes.
- Gain de temps au niveau des échanges de données grâce à l'inexistence d'un serveur de traitement.
- Les mises à jour se font en temps réel.
- Disponibilité des informations fiables en temps souhaité pour tous les acteurs dans le système.

- Répartition des tâches entre les différents acteurs du système.
- Diminution des retards qui apparaissent dans l'élaboration des documents.
- Centralisation des données.

Inconvénients :

- Une grande charge sur le serveur due aux traitements applicatifs.
- Contrôle permanent est nécessaire pour éviter les pannes et pour la sécurité données.

1.2. Diagramme de déploiement

Le modèle de déploiement est le premier niveau de conception vu que c'est lui qui permet d'organiser les environnements de travail. Evidemment, le diagramme de déploiement décrit la disposition physique des ressources matérielles qui constituent le système et décrit la répartition des différents composants sur ces matériels. Ainsi nous allons modéliser notre architecture par un diagramme de déploiement :

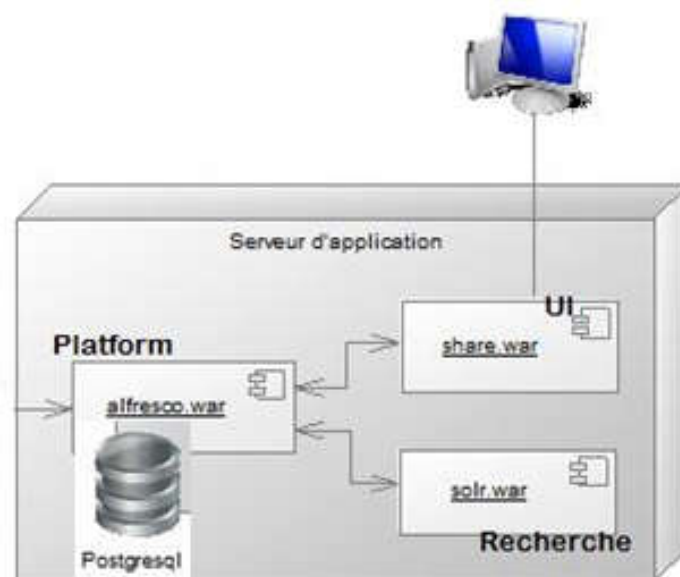


Figure 28 : Modèle de déploiement

2. Conception détaillée

Dans cette section nous allons décrire les couches de l'application. En premier lieu, nous allons décrire la structure de l'interface GUI, puis nous allons présenter les diagrammes de séquences relatifs aux métiers et persistance de notre application.

2.1. Diagramme de navigation

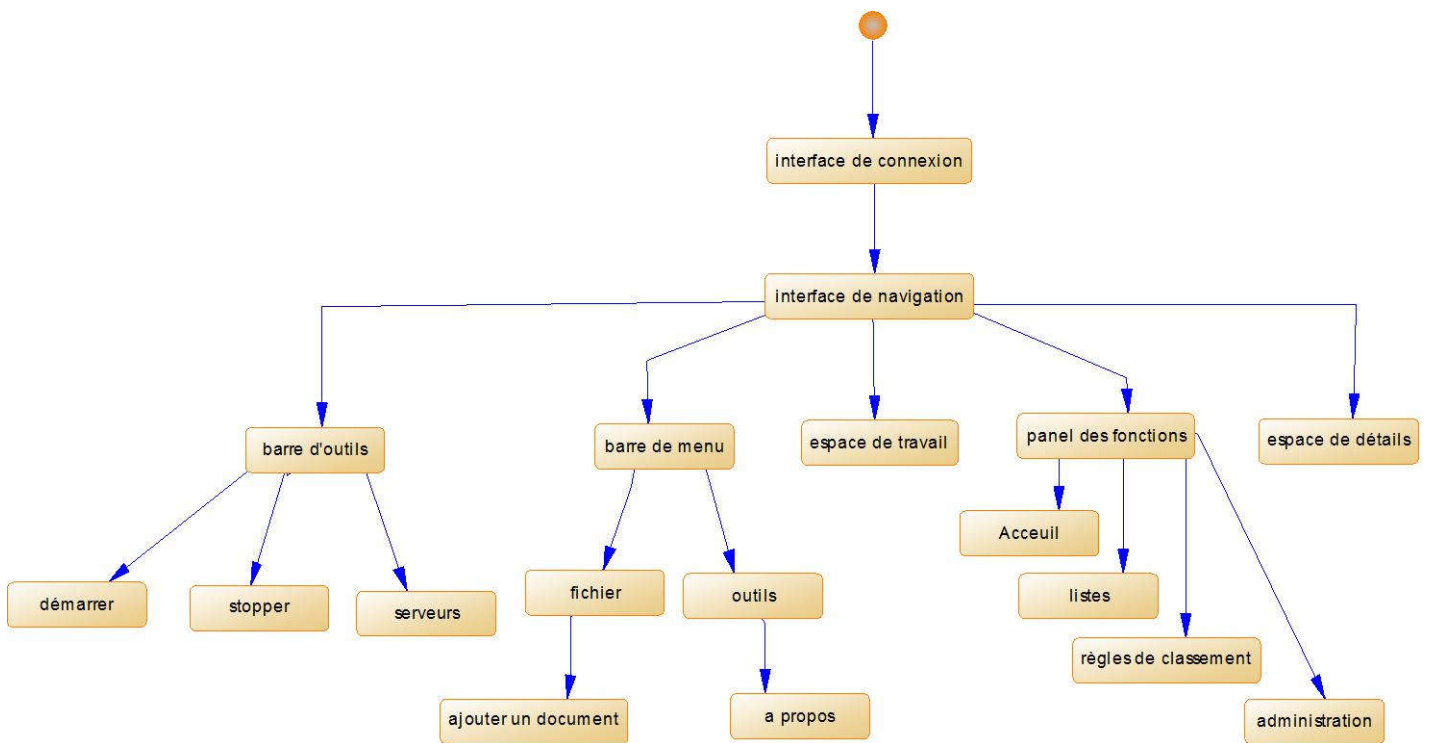


Figure 29 : Diagramme de navigation de l'interface

Description de l'interface :

- **Barre de menu :**
 - Fichier : Ajout d'un document dans l'espace personnel d'Alfresco.
 - Outils :
 - A propos : détail sur l'application (résumé, version, auteur).
- **Barre d'outils :**
 - Démarrer : pour le choix d'une règle et la démarrer.
 - Stopper : pour stopper une règle démarrée.
 - Serveurs : configuration des serveurs et administration.
- **Boutons de fonctionnalités :**
 - Accueil : pour retourner sur l'interface d'accueil.
 - Listes : gérer les listes qui vont être comme des modèles(types/aspects) lors du saisi des métadonnées.
 - Règles de classement : démarrer, stopper et gérer les règles de classement.
 - Mes documents : accéder à l'espace personnel d'Alfresco via le navigateur.
 - Administration : Configuration des serveurs Alfresco.

Les cas d'utilisation décrits dans le chapitre précédent décrivent les interactions entre le système et les acteurs (utilisateur et administrateur). Lors de ces interactions, les acteurs engendrent des services en envoyant des requêtes au système, qui lui répond à son tour par l'envoi de réponses. Ainsi, le système est vu comme étant une boîte noire. A présent, nous allons découper le système et découvrir son fonctionnement interne à l'aide des diagrammes de comportement.

2.2. Modélisation du Contenu

2.2.1. Moyens de classification

Alfresco permet de gérer de multiples moyens de classification de documents tout en permettant une organisation adaptée à la nature des documents archivés (factures, devis, journaux, contrats, rapports, etc.)

Nous pouvons classer les documents soit par une simple arborescence des espaces, Tags, ou par des catégories[18]. Ci- dessous une figure montrant le tableau des moyens de classification dans Alfresco :

	Arborescence	Tags	Catégories
Contrôle d'accès	x	-	-
Console d'administration	-	x	x
Hiérarchisable	x	-	x
Multicritères	-	x	x
Accès rapide	-	x	-

Figure 30 : Tableau des moyens de classification des documents selon Alfresco

2.2.2. Exemple de modèle de données personnalisé

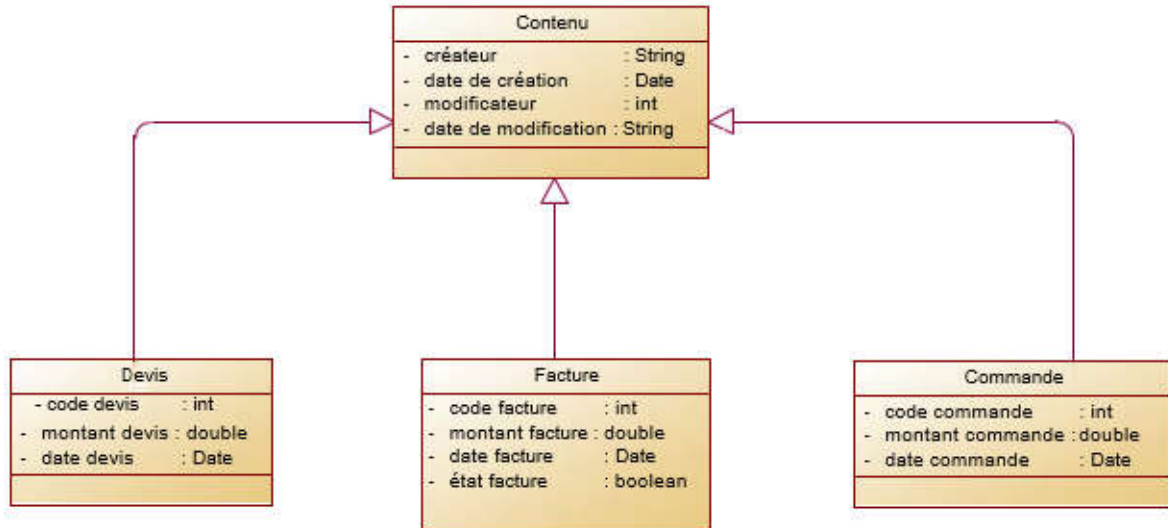


Figure 31 : Conception du modèle de données : Cas du service « Achat »

2.3. Comportement du système

1) Diagramme de séquence « Démarrer une règle de classement »

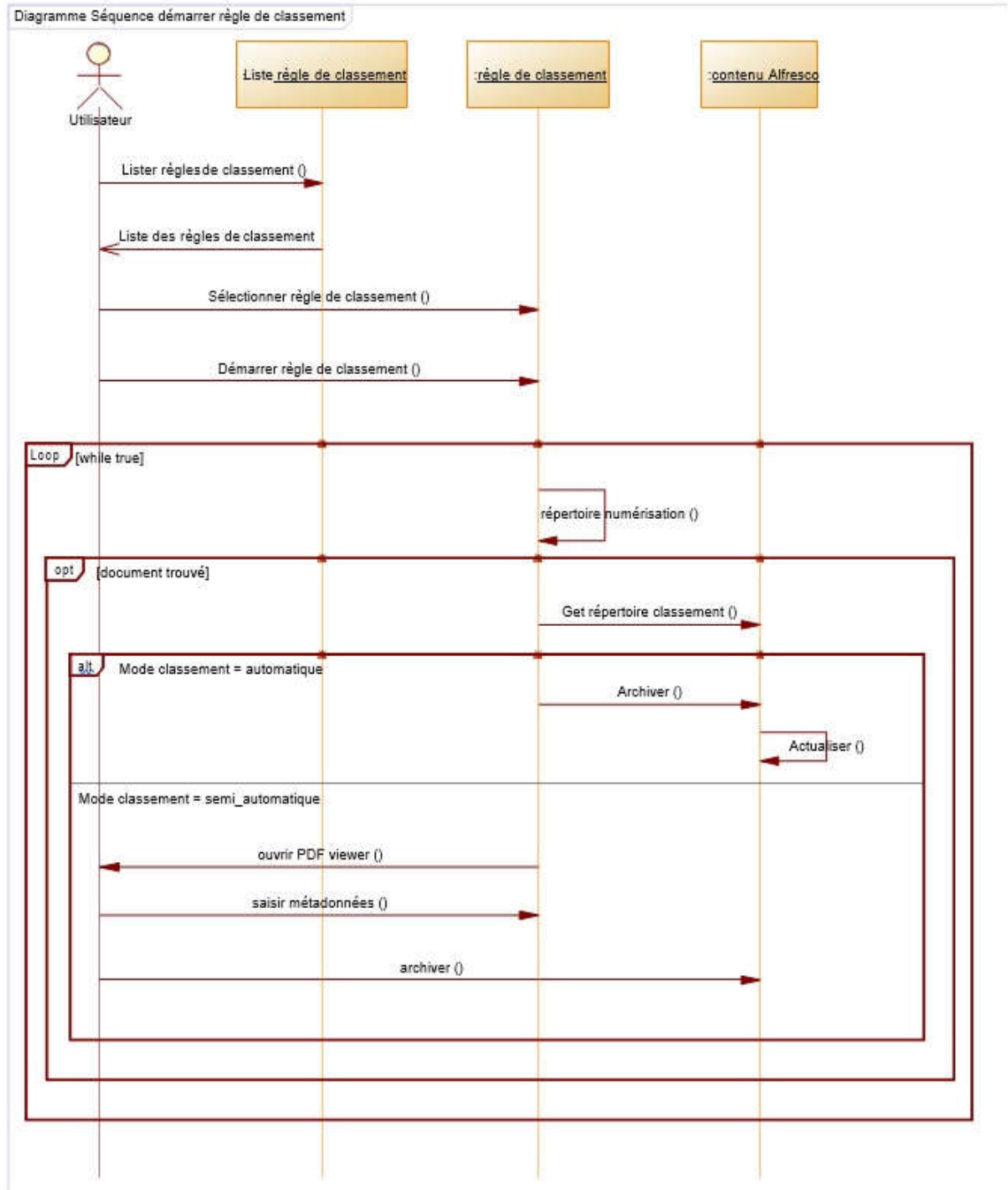


Figure 32 : Diagramme de séquence « démarrer règle de classement »

2) Diagramme de séquence « Ajouter un document à son espace personnel Alfresco »

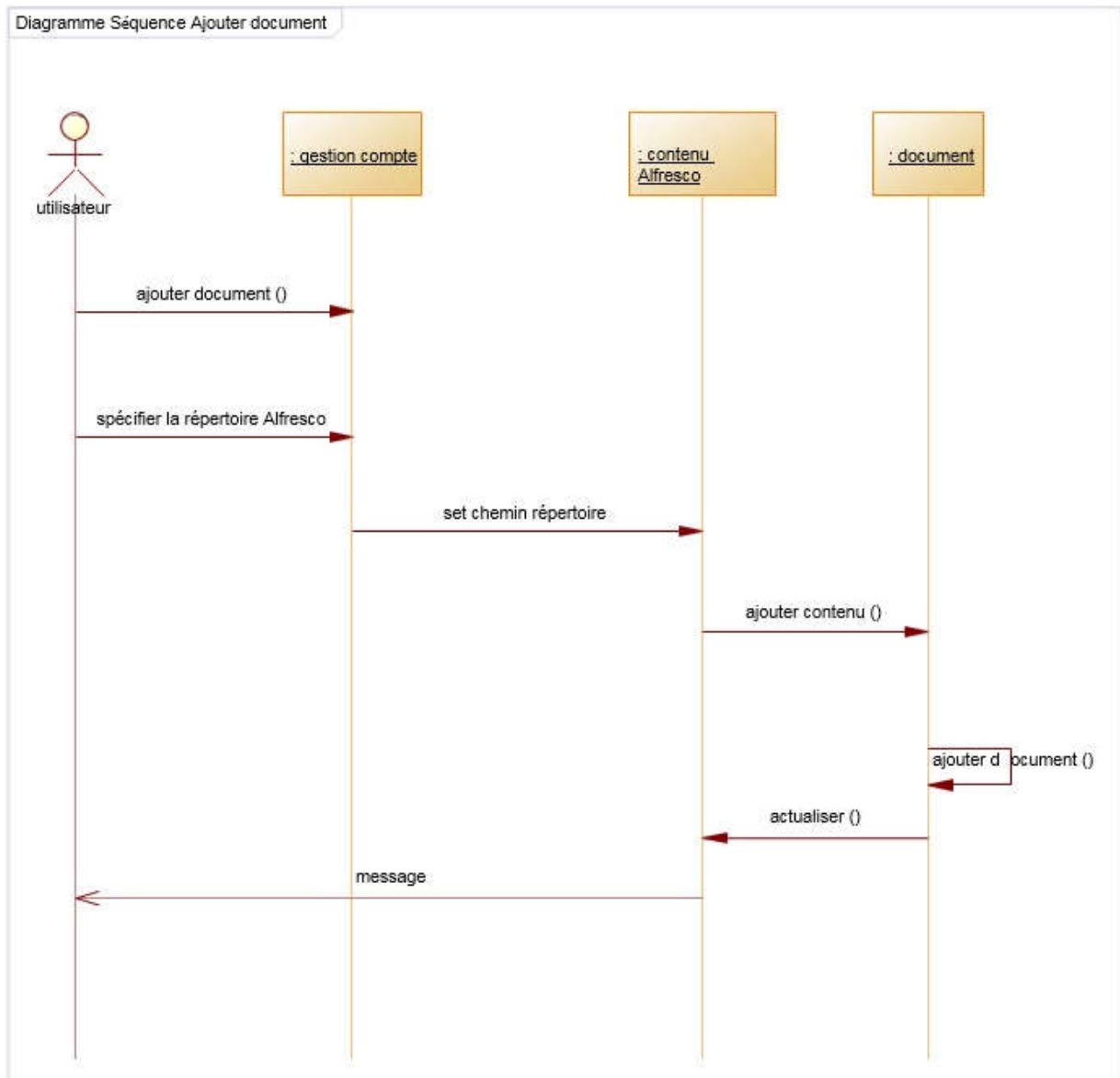


Figure 33 : Diagramme de séquence « ajouter document »

2.4. Conception des processus métiers

Il y a plusieurs langages différents de modélisation de processus. Aujourd’hui, il y a un grand consensus sur le standard BPMN (Business Process Management Notation). Utilisant une notation standardisée, le modèle BPMN est compréhensible par tous les participants de l’initiative, depuis l’analyste métier qui définit le modèle, jusqu’au développeur responsable de la technologie d’automatisation, et finalement, jusqu’aux utilisateurs de l’entreprise qui vont gérer et monitorer le processus. De plus, une représentation BPMN d’un processus métier peut être traduite vers les langages d’exécution comme BPEL (Business Process Execution Language). Nous allons concevoir ceci via le module « Activiti » d’Alfresco[19].

a) Service achat

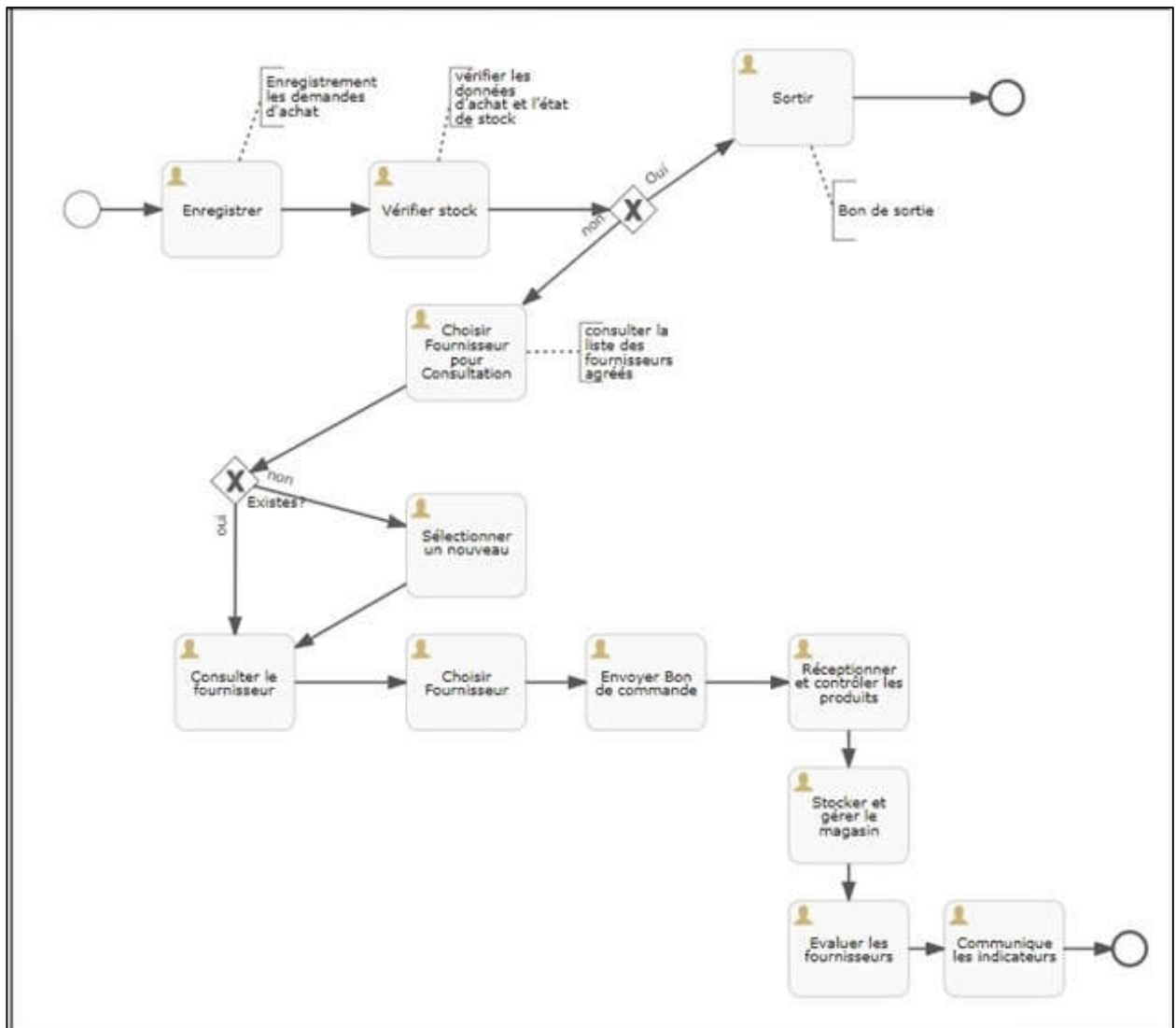


Figure 34 : Processus métier « service achat »

b) Service ressources humaines :

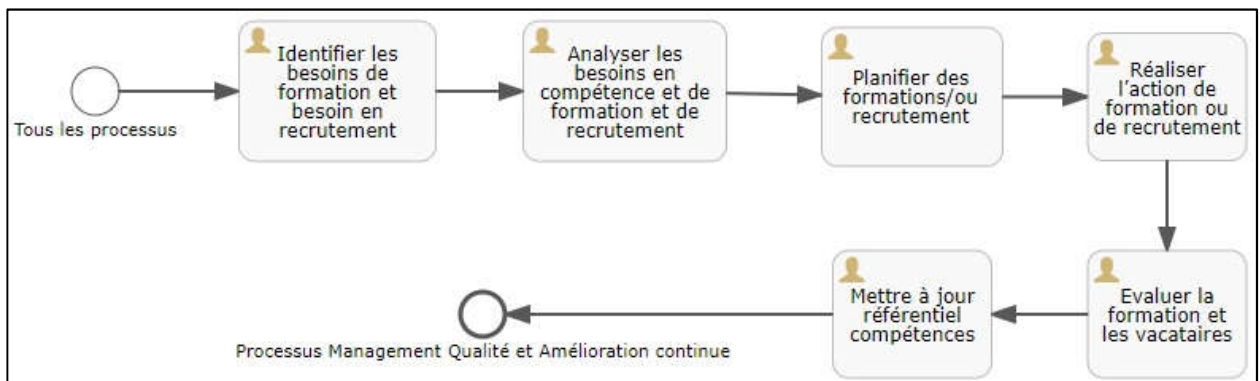


Figure 35 : Processus métier « service ressources humaines »

c) Service scolarité :

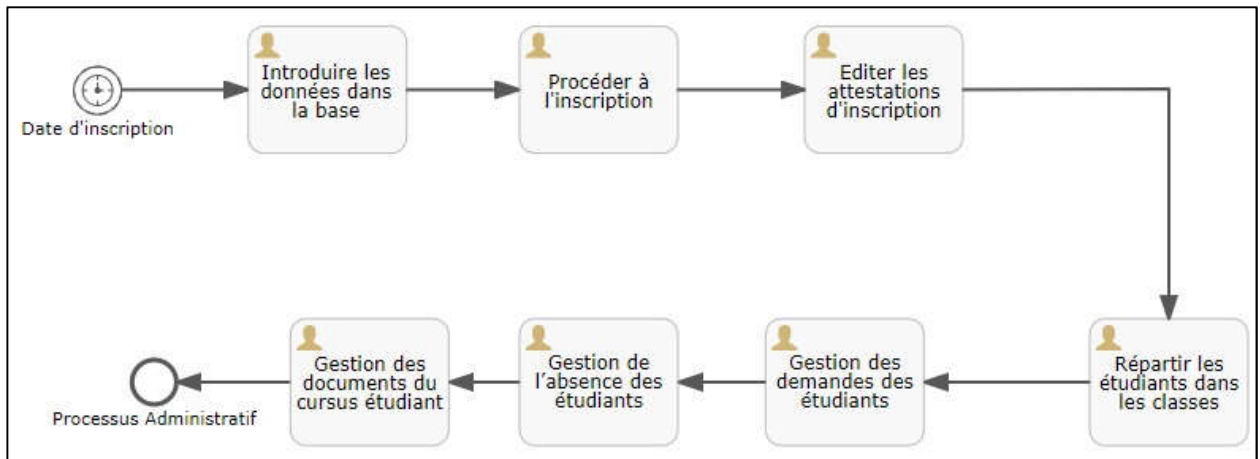


Figure 36 : Processus métier « service scolarité »

d) Service Stage :

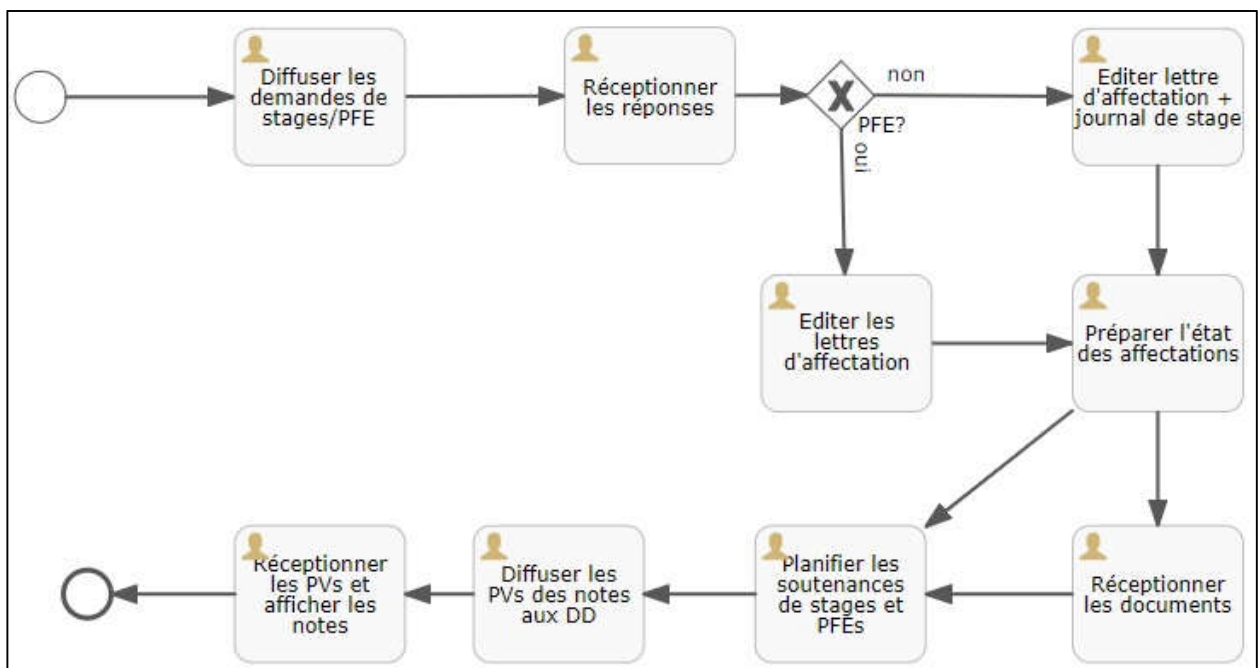


Figure 37 : Processus métier « service stage »

e) Service ressources Bibliothèque :

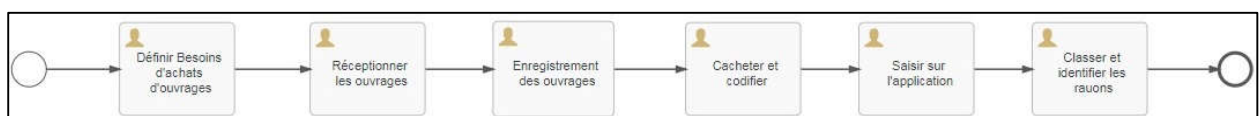


Figure 38 : Processus métier « service bibliothèque »

f) Service Examen :

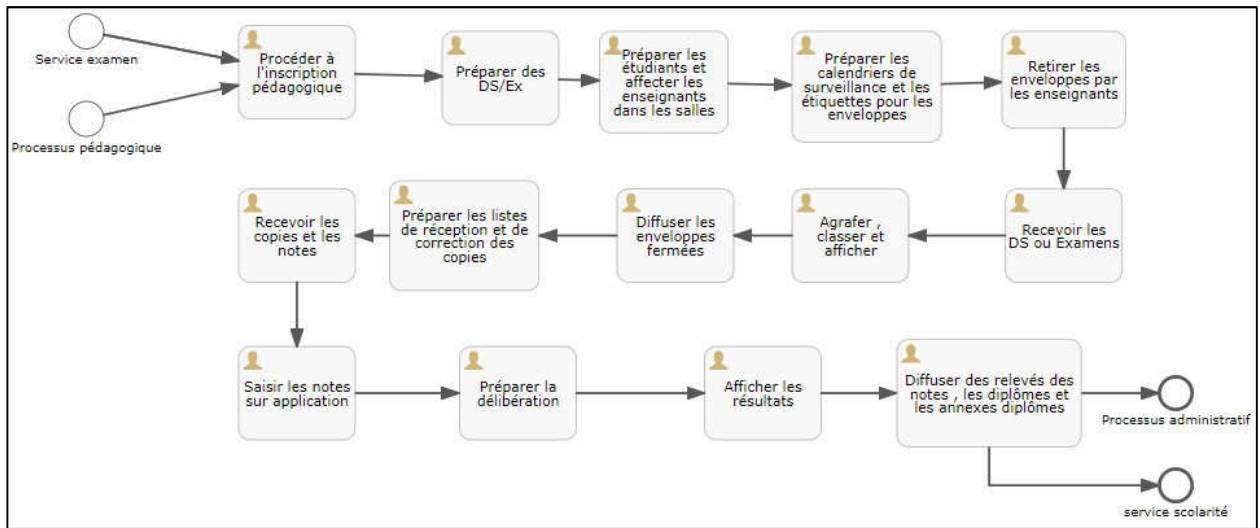


Figure 39 : Processus métier « Service Examen »

g) Service maintenance :

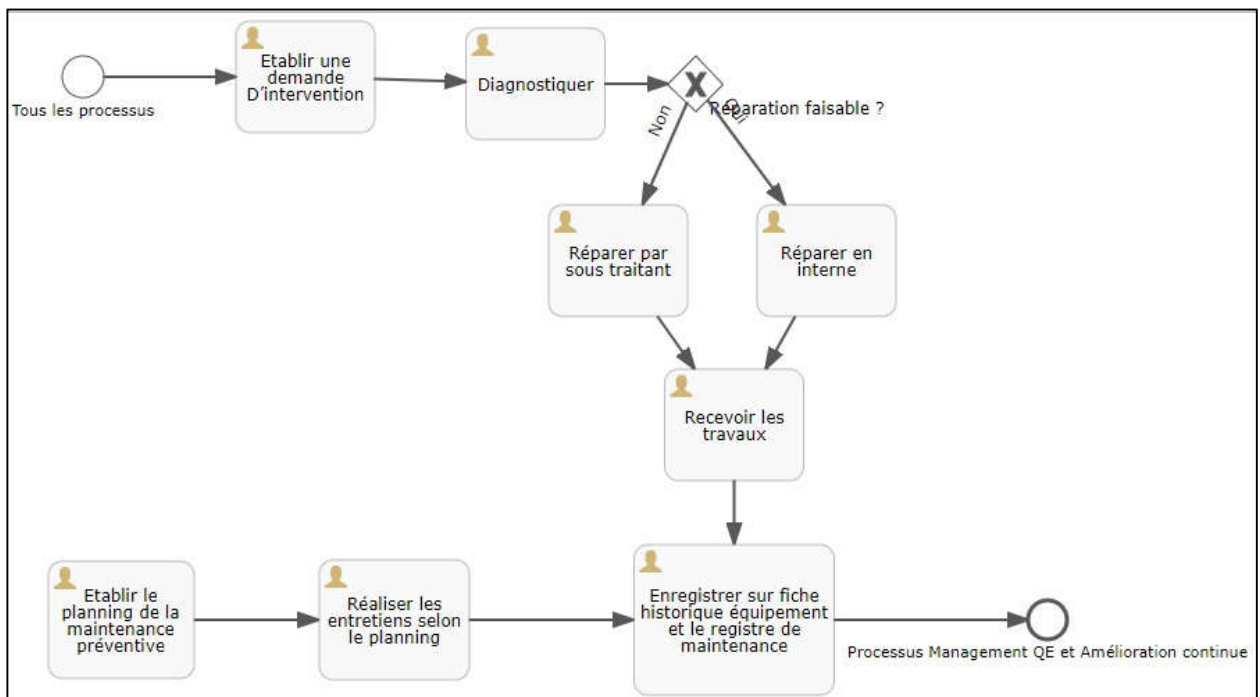


Figure 40 : Processus métier « Service maintenance »

h) Service Finance :

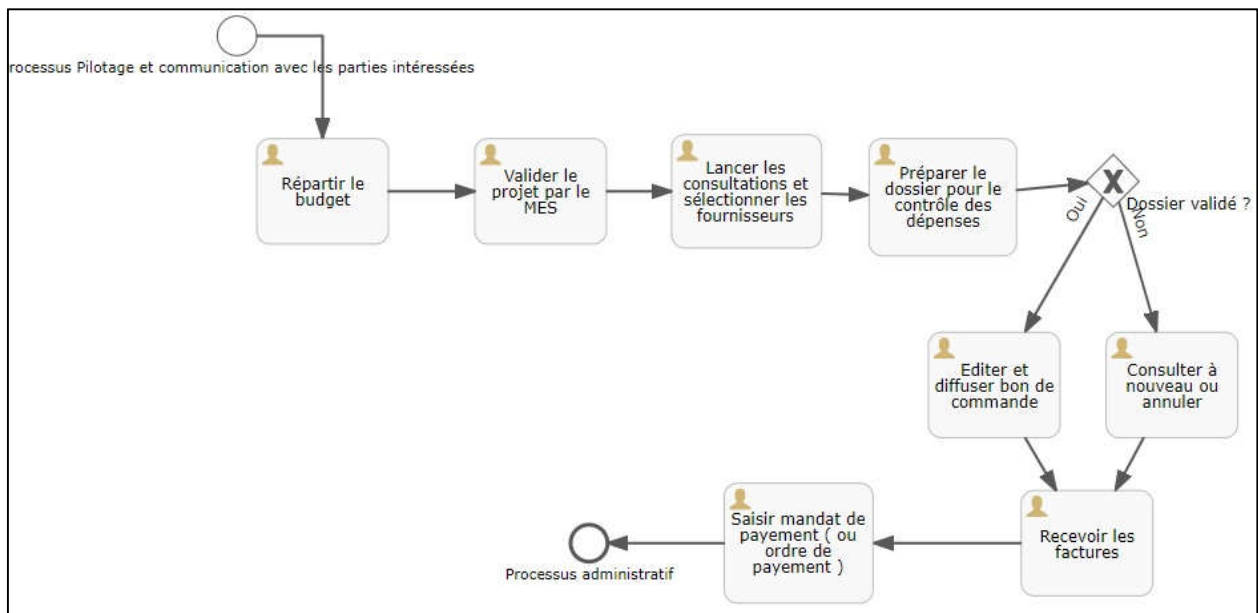


Figure 41 : Processus métier « Service Finance »

Les processus métiers vont nous aider par la suite à dégager les principaux types de documents et fichiers ainsi que les principaux Workflows à implémenter par la suite dans l’application.

Conclusion

Tout le long de ce chapitre, nous avons décrit les différentes parties de l’architecture utilisée par des diagrammes et nous avons décrits les principaux processus métiers.

Dans le chapitre suivant nous abordons de façon détaillée la phase de réalisation de notre solution GED et nous exposons le travail réalisé.

Chapitre 5 : Réalisation

Introduction

Ce chapitre traite la réalisation de notre projet. Il sera composé de trois parties : la première partie s'intéresse à l'environnement matériel et logiciel ainsi que les choix technique de développement de notre application, la deuxième partie est une description des différentes interfaces de l'application et la dernière partie sera le chronogramme du travail réalisé.

1. Environnement de travail

Dans cette partie, nous allons justifier notre choix pour les différents outils de travail afin de réaliser les différentes parties de notre application selon l'architecture adoptée client-serveur.

- Environnement matériel :

Comme station de développement, nous avons utilisé un ordinateur de bureau Fujitsu. Il est équipé d'un processeur Intel Core i5 et de 4 Go de mémoire vive et d'une version 64 bits de Windows 7.:

- Environnement logiciel :

- **Tomcat 7** : est un conteneur libre de servlets et JSP Java EE, il est issu du projet Jakarta qui est un projet principal de l'Apache Software Foundation. Tomcat est à la fois serveur Web et conteneur d'applications Java. Il prend en charge des pages HTML statiques les pages JSP et fonctionne sur presque la moitié des sites Web existants. Il est à noter qu'Apache est un logiciel Open Source qui est disponible gratuitement sur Internet.
- **Activiti 6.0** : Module intégré sur le serveur Tomcat est un outil d'édition/conception de processus métiers automatisés se basant sur le langage BPMN 2.0 (Business Process Management Notation)
- **Postgresql** : est un système de gestion de base de données relationnel (SGBD). Il permet de stocker des données en masse et pouvoir les utiliser à volonté en les triant comme nous le souhaitons.
- **PowerAMC** : Pour la phase de conception de notre projet, nous avons recourt à ce logiciel pour mettre en place les différents diagrammes UML utiles pour concevoir les différentes parties de conception de l'application.
- **Brackets** : est un éditeur de texte conçu spécialement pour le développement orienté web, il est très adéquat pour le traitement des fichiers xml dans Alfresco [15] .

2. Description de l'application

Dans cette partie nous allons expliquer le fonctionnement de notre application et nous allons présenter quelques interfaces de l'application avec des écrans témoins.

2.1. Installation

Lors de l'installation à partir du fichier exécutable, nous nous trouvons avec deux options :

- Installation facile :

Permet une configuration par défauts des serveurs, nous avons juste à introduire le mot de passe du compte administrateur.

- Installation avancées (que nous allons utiliser) :

Permet de configurer manuellement les serveurs (port et les services associés). Aussi, nous avons la possibilité d'ajouter des composants supplémentaires au cours de l'installation.

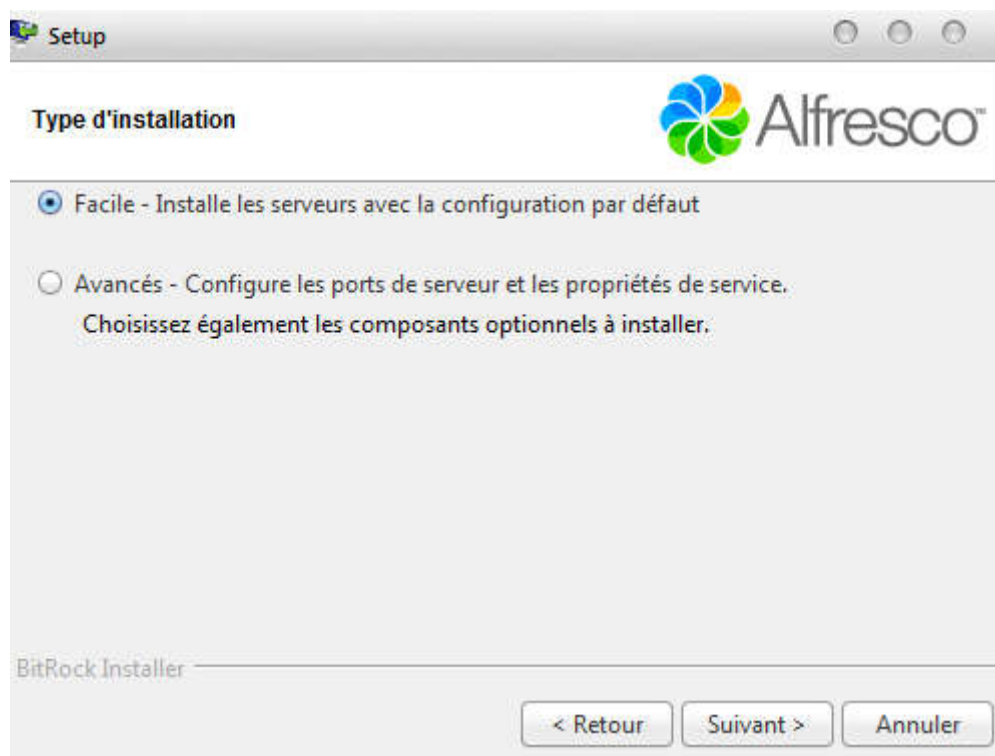


Figure 42 : Choix du type d'installation

Alfresco permet de choisir les composants axillaires pour un fonctionnement souhaité :

- Java
- PostgreSQL
- SharePoint
- Web Quick Start
- Open Office

Configuration des serveurs :

A cette étape, l'assistant d'installation nous demande d'entrer le port du serveur postgresSQL (par défaut c'est 5432) et les paramètres du serveur web TomCat comme indiqué sur la figure ci-dessous.

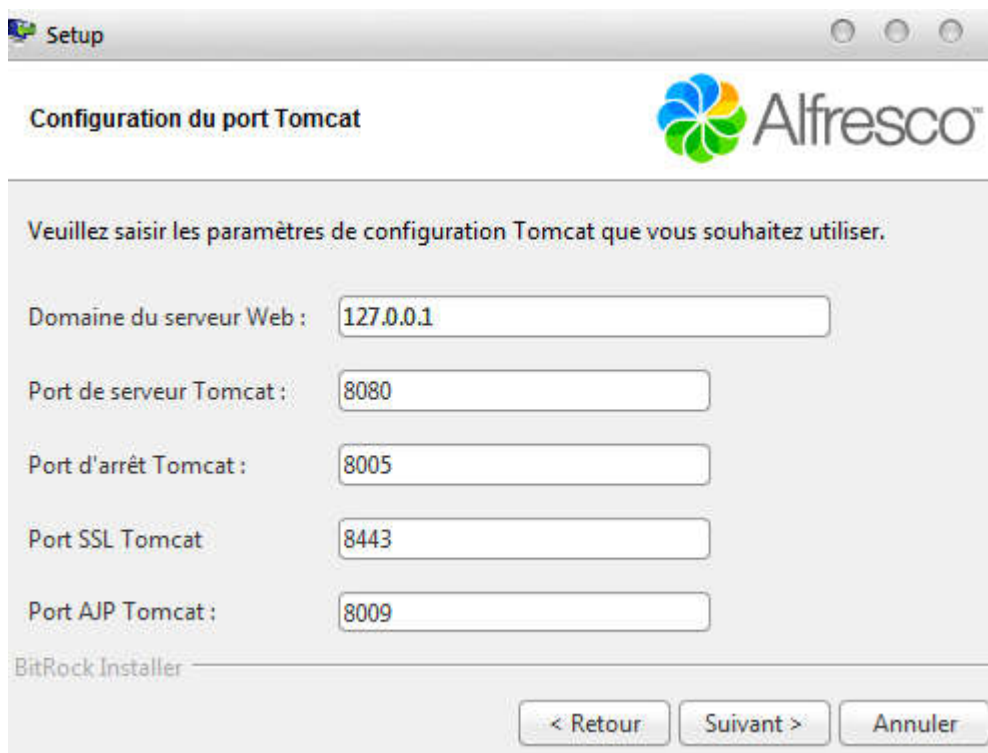


Figure 43 : Configuration du serveur Tomcat

- Autres paramètres à mentionner :

Après la configuration de Tomcat et postgresSQL, nous allons configurer d'autres services relatifs au fonctionnement d'Alfresco :

- Port du service RMI (par défaut 50500).
- Port du serveur FTP (par défaut 21).
- Mot de passe administrateur.
- Port du protocole SharePoint et OpenOffice si nous avons choisi d'installer ces modules supplémentaires. (port SharePoint par défaut 7070, OpenOffice 8100).

Note : toutes configurations écrites précédemment peuvent être modifiées ultérieurement dans le fichier `alfresco-global.properties` dont le chemin est emplacement est :

(Le dossier d'installation d'Alfresco)\Alfresco\tomcat\shared\classes

2.2. Description des Interfaces utilisateurs

1) Interface utilisateur Accueil

La figure ci-après illustre l'interface d'accueil d'authentification après les modification du code sources d'Alfresco pour adapter une charte graphique personnalisée propre à nous.

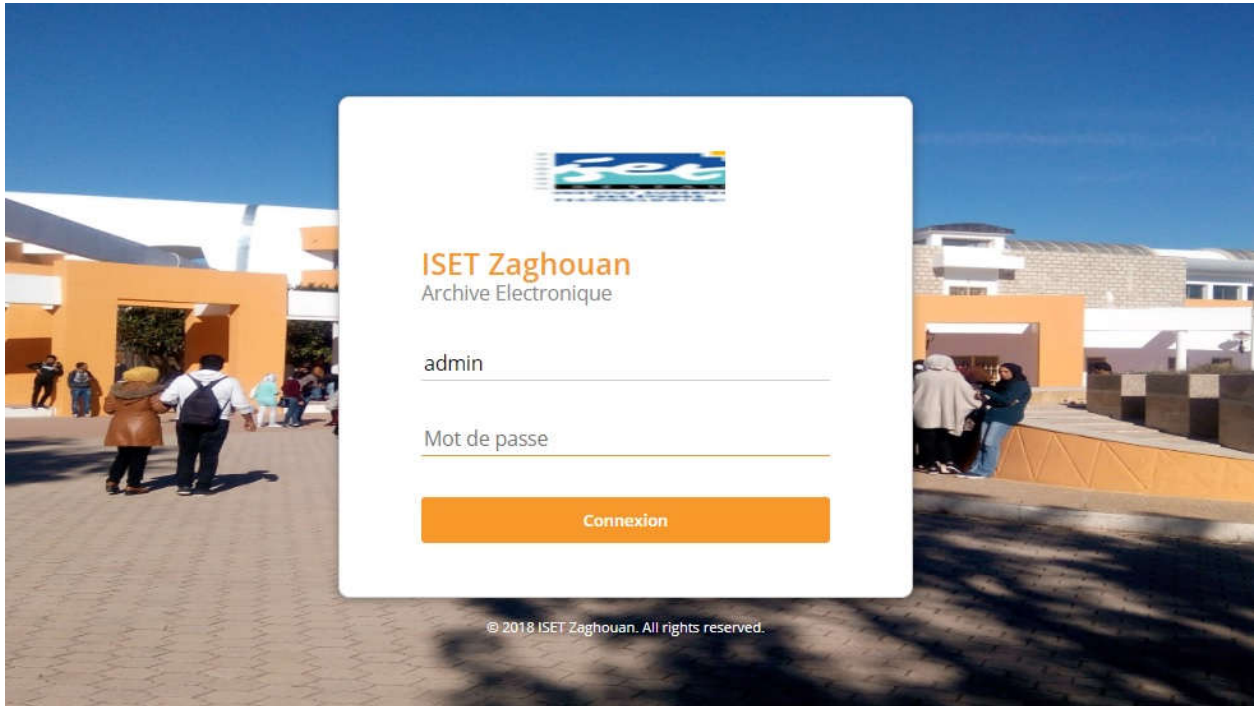


Figure 44 : Interface d'accueil de login

2) Interface Tableau de bord

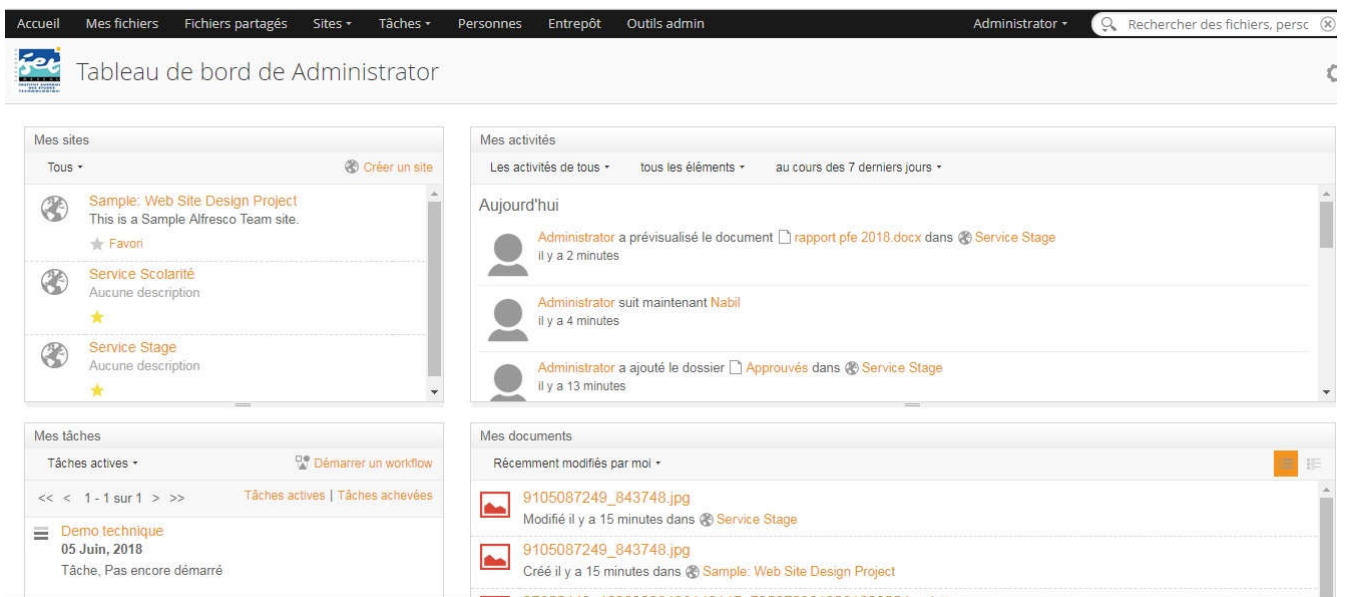


Figure 45 : Tableau de bord de l'administrateur

Nous avons personnalisé le thème (insérer le logo de l'ISET) et le menu du tableau de bord de l'espace administrateur global.

3) Interface Gestionnaire de site

Nom du site	Description du site	Visibilité	Je suis un gestionnaire du site	Actions
Service Achat		Modéré	Oui	Actions
Archive		Modéré	Oui	Actions
Le Directeur		Modéré	Oui	Actions
Service Finance		Modéré	Oui	Actions
Service Scolarité		Modéré	Oui	Actions
Sample: Web Site Design Project	This is a Sample Alfresco Team site.	Public	Oui	Actions
Service Stage		Modéré	Oui	Actions

Figure 46 : interface Gestionnaire des sites

Nous avons créé les sites des services et les avons configuré de façon à ce que chaque utilisateur n'appartenant pas à un site, doit impérativement avoir une autorisation d'accès de la part d'un administrateur.

4) Interface de Gestionnaire de rôles

Recherchez et sélectionnez des utilisateurs, attribuez-leur des rôles et ajoutez-les à ce site.

1. Recherche de 'res' : 1 résultat(s).
 Nabil (Responsable Stage) [Sélectionner]

2. Définir le rôle de l'utilisateur : Vous n'avez pas encore sélectionné d'utilisateur.

3. Ajouter des utilisateurs au site : Nombre total d'utilisateurs ajoutés 1.
 Nabil Gestionnaire [✓]

Figure 47 : Interface de Gestionnaire de rôles

Nous avons choisi de donner le rôle de « Gestionnaire » à chaque utilisateur dans son propre site, et le rôle de « Contributeur » pour le reste des sites.

5) Les interfaces des règles de classement et des modèles de types/aspects

a) Interface de création des règles de classement relative à un administrateur :

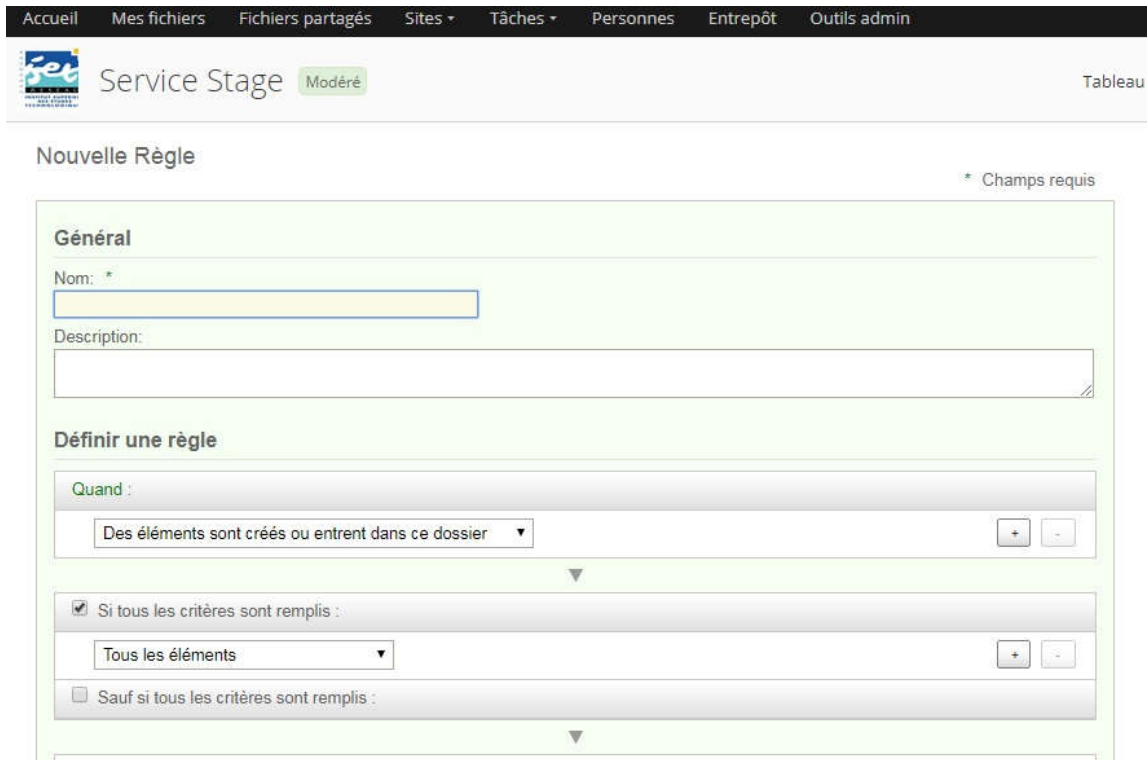


Figure 48 : Interface de création des règles de classement (administrateur)

b) Interface principale des règles de classement relative à un utilisateur :

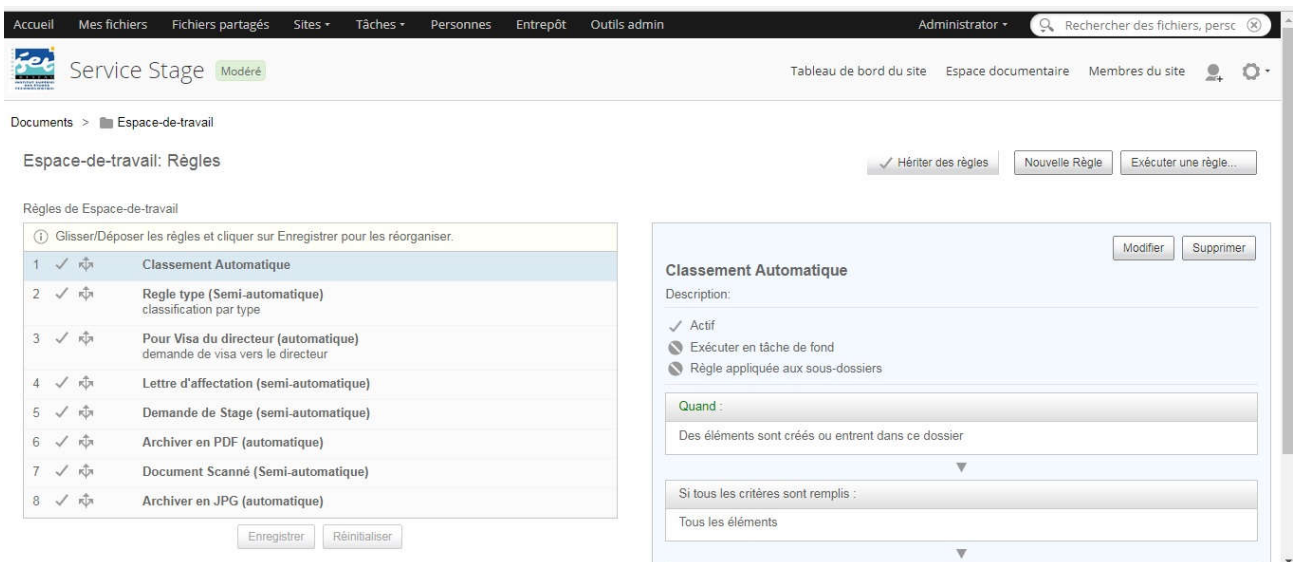


Figure 49 : interface principale des règles de classement (utilisateur)

L'utilisateur peut à partir de cette interface activer ou désactiver et ajouter/supprimer les règles de classement automatiques et semi-automatiques (conditionner par des aspects) des documents selon ses besoins.

c) Interface de création de modèles

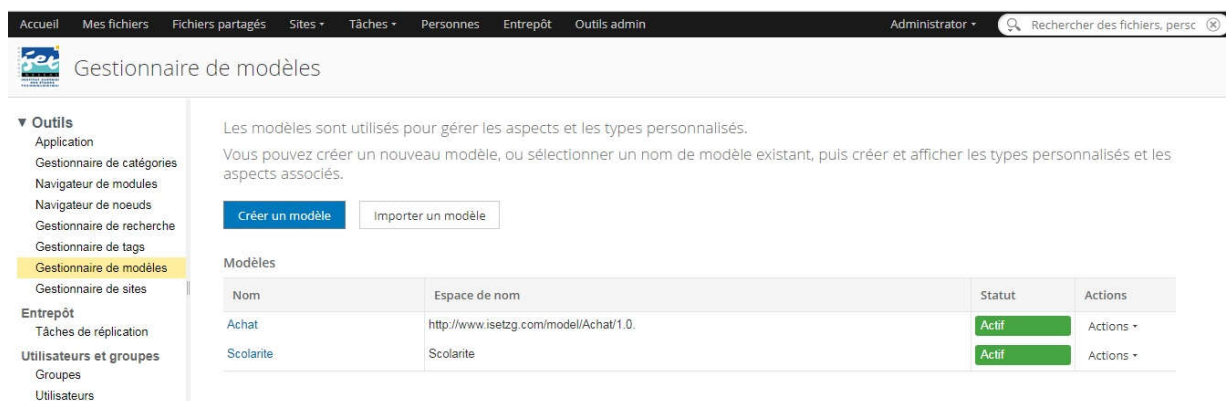


Figure 50 : Interface de création de modèles personnalisés

- Dictionnaire de données :

Il faut décrire les entités écrites précédemment (voir Figure 31 chapitre Conception), dans un dictionnaire de données pour qu'elles soient gérées par Alfresco.

Le dictionnaire de données Alfresco est un ensemble de modèles de base simple (Float, Double, Integer) et complexe (Folder, Content, Person) qui permettent de décrire la structure des contenus à traiter.

- Module Complémentaire « Editeur de modèles » :

Nous allons définir les nouvelles entités à partir des modèles de base dans un fichier CustomModel.xml.

A l'aide d'alfresco-mmt.jar, nous pouvons déployer et installer de nouveaux modèles AMP (Alfresco Module Package).

Cette extension permet de produire des fichiers XML graphiquement, sans avoir besoin de coder. Il génère automatiquement deux fichiers XML qu'on les trouve par défaut, sous le répertoire dictionnaire de données

- Fichier de Configuration : extForm.xml
- Fichier de Modèle : extModel.xml

Cette interface permet d'ajouter « importer » des modèles personnalisés que nous avons créé via le langage XML.

Il faut importer un fichier zip contenant deux fichiers XML, le fichier contenant le modèle et ses propriétés et attributs (types et aspects). (Voir Annexe I).

Ces modèles personnalisés sont clés pour les règles de classement semi-automatiques (par type de document) et l'attribution d'index pour les recherches avancées sur les archives.

d) Interface d'attribution de modèles/aspects :

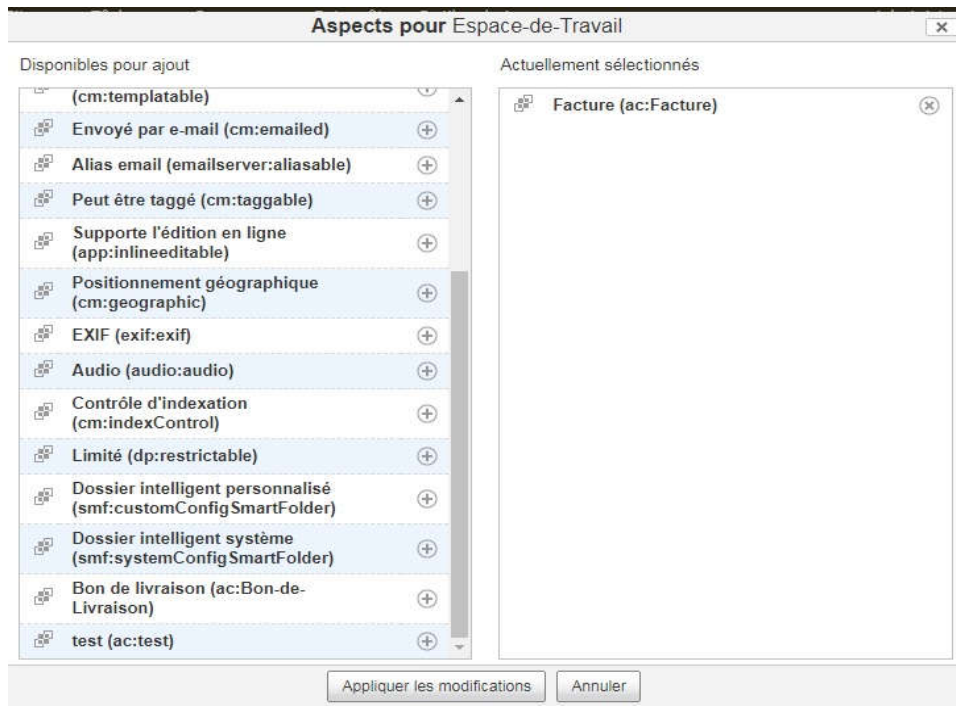


Figure 51 : Interface d'attribution de modèles/aspects

En attribuant le modèle « Achat » que nous avons créé- à notre espace de travail, nous constatant que ses aspects sont maintenant associables à nos documents (exemple aspect facture avec préfixe d'indexation « ac :Facture »).

e) Interface d'édition des propriétés des documents :

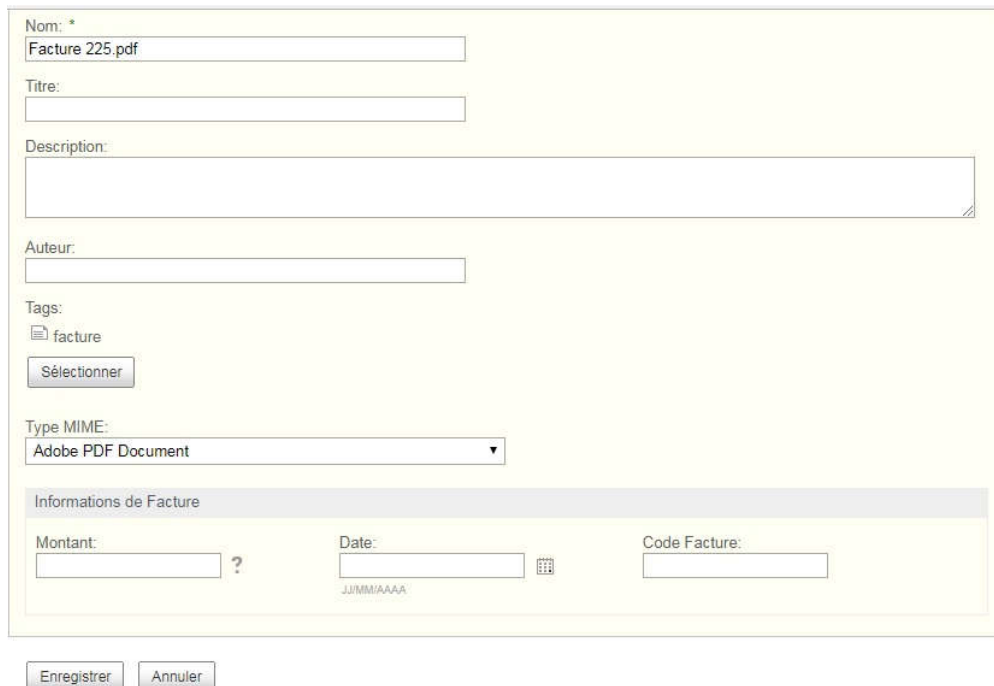


Figure 52 : Interface d'édition des propriétés

Comme nous pouvons le constater maintenant l'utilisateur en éditant les propriétés de ses documents peut voir apparaître les champs des propriétés de l'aspect de modèle attribuer « ac :Facture » : (Code Facture, date et montant).

f) Interface de création des étapes de conservation finale (Gestion des archives) :

Nous avons intégré le module RM 3.0 pour Alfresco (Record Manager) qui nous donne la main pour créer un site spécial pour l'archivage (site d'archive au lieu de site de collaboration pour les autres sites par défaut), ce site comporte des règles spéciaux plus poussées pour la gestion du cycle de vie des documents jusqu'à leur destruction finale et intègre un moteur de recherche dans les archives spécial rapides et proposant plusieurs critères et indexes.

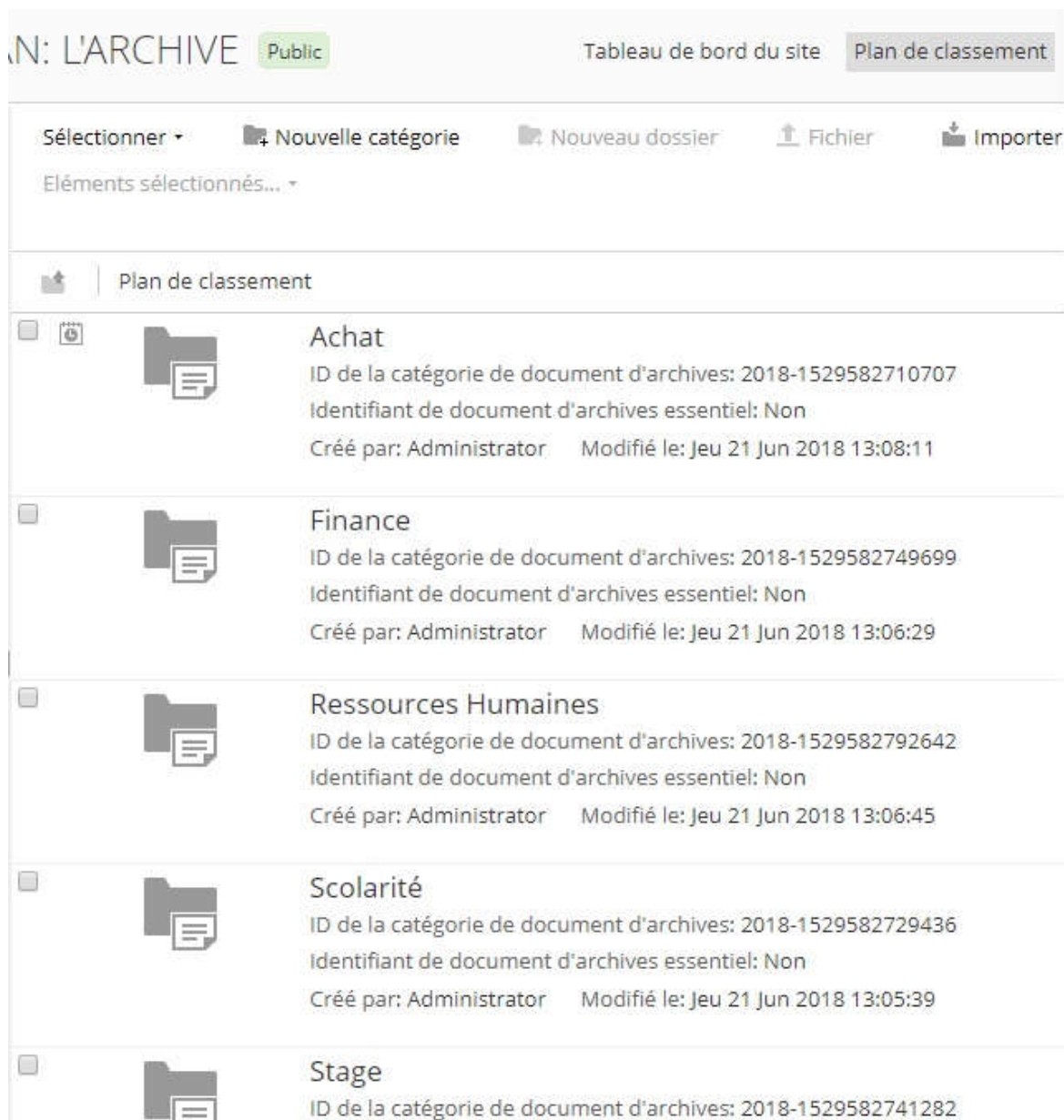


Figure 53 : Site RM de l'archivage

ISSET ZAGHOUAN: L'ARCHIVE Public Tableau de bord

Modifier la règle de conservation * Champs requis

Etapes

1 Début de conservation après 1 année(s)

Après une période de Année à partir de ou

Lorsque l'événement se produit :

Description de l'étape: *

Début de la conservation dans l'archive

2 Détruire après 5 année(s)

Après une période de Année à partir de ou

Lorsque l'événement se produit :

Conserver les métadonnées du document d'archives après sa destruction

Description de l'étape: *

Destruction de l'archive Après 5 ans de conservation.

Figure 54 : Gestion des règles de classement des archives

ISSET ZAGHOUAN: L'ARCHIVE Public Tableau de bord du site

Rechercher

Critères **Résultats**

Texte à rechercher

Rechercher par: Date de recherche:

- Contenu
- Document d'archives ▶
- Règle de conservation ▶

► Options de résultats

- Mots clés (texte et nom)
- Nom
- Titre
- Description
- Créateur
- Créé
- Modificateur
- Modifié
- Auteur

Figure 55 : Moteur de recherche des documents du site RM

3. Chronogramme :

La réalisation de l'application a pris du temps surtout pour la partie de documentation, il fallait mettre une démarche de travail et la suivre en se basant sur la partie relative à la conception pour arriver à implémenter un système sûr et fiable.

Nous avons développé les modules des règles de classement et les modèles pour l'indexation et la recherche intelligente ainsi que la coordination entre l'utilisation de l'application comme étant un utilisateur et administrateur. Nous pensions introduire un module de gestion de l'historique mais par contrainte de temps nous l'avons reporté.

Le chronogramme de la figure ci-dessous décrit la partition temporelle pour les différentes étapes du projet au cours de la période de réalisation.

	Février	Mars	Avril	Mai
Etude de documentation	[Barre rouge couvrant tout le mois]			
Analyse et conception		[Barre rouge couvrant tout le mois]		
Réalisation			[Barre rouge couvrant tout le mois]	
Rédaction du rapport		[Barre rouge couvrant tout le mois]		

Figure 56 : Chronogramme du projet

Conclusion

Dans ce chapitre, après une étude sur le choix technique et le choix de l'environnement de travail, nous avons présenté les aspects de réalisation de notre projet à travers les captures d'écrans montrant les différentes faces de l'application.

Conclusion générale et perspectives

Aujourd'hui, les systèmes d'ECM doivent satisfaire des besoins en constante évolution. Dans ce cadre, pour améliorer son organisation et sa flexibilité, l'ISET de Zaghouan a lancé un projet de gestion électronique de documents.

Le présent rapport détaille toutes les étapes par lesquelles nous sommes passées pour arriver au résultat attendu. Nous avons commencé par comprendre le contexte général du projet et identifier les différentes exigences du futur système. Nous avons préparé, par la suite, un planning de travail en respectant les priorités des besoins.

Malgré les contraintes du temps et les difficultés techniques qui se résument principalement dans la maîtrise des différents aspects techniques et architecturaux sur lesquels s'articule le sujet, nous avons réussi à réaliser une fiable application.

Ce travail nous a été très instructif, puisqu'il nous a permis de découvrir un nouveau domaine de travail qui est la gestion électronique des documents. Nous avons eu l'occasion de mettre en œuvre nos connaissances dans les bonnes pratiques de la gestion de projet vu que nous avons eu l'opportunité d'organiser son déroulement dès le début.

Ce travail, pourrait être amélioré par d'autres fonctionnalités, en effet, l'outil que nous avons construit est toujours ouvert à des améliorations telles que :

- L'ajout d'une nouvelle fonctionnalité qui permet d'exploiter les versement/destruction d'archives

- L'implémentation d'un workflow de cycle de vie afin d'assurer la transition de la ressource documentaire, de son versement à son sort final.

Aussi nous prévoyons un système de serveurs en mode cluster pour assurer la redondance et la disponibilité de l'application.

Bibliographie

- [1] <http://www.arcalys.com/archivage/stockage-darchives-physiques-vous-cherchez-un-local-pour-stocker-vos-archives/>
- [2] <http://www.amoks.com/rep-lexique/ido-325/ged.html>
- [3] <https://fr.slideshare.net/JM.Pascal/ecmdefinition-presentation>
- [4] <http://www.qualishare.fr/qualishare-logiciel-gse-metier-collaboratif-sharepoint/gestion-electronique-des-documents/>
- [5] https://fr.wikipedia.org/wiki/Gestion_électronique_des_documents
- [6] <https://fr.wikipedia.org/wiki/Alfresco>
- [7] <https://community.alfresco.com/docs/DOC-6139-enterprise-edition>
- [8] <https://www.nuxeo.com/fr/media-center/open-source-content-management-vendor-nuxeo-continues-strong-growth-path/>
- [9] <https://www.pulsar-informatique.com/services/ged-open-source/technologies-et-solutions-de-ged-open-source/knowledge-tree>
- [10] <https://www.exoplatform.com/digital-workplace/document-management-software>
- [11] <https://community.alfresco.com/thread/195645-role-utilisateur-dans-alfresco>
- [12] <https://docs.alfresco.com/5.2/references/dev-extension-points-actions.html>
- [13] <https://vdocuments.site/rapport-pfe-developpement-dun-connecteur-alfresco.html>
- [14] <https://koossery-tech.developpez.com/tutoriels/java/ecm/alfresco-jbpm/bible-developpeur-alfresco-jbpm-guide-complet-developpeur-alfresco-jbpm/?page=alfresco-developper-guide-alfresco-repository-foundation-service>
- [15] <https://docs.alfresco.com/5.0/concepts/dev-repository-intro.html>
- [16] <https://docs.alfresco.com/5.2/concepts/dev-api-intro.html>
- [17] <https://community.alfresco.com/docs/DOC-6043-workflow-with-activiti>
- [18] https://memic.ccsd.cnrs.fr/mem_00629437/document
- [19] <https://www.lucidchart.com/pages/fr/quest-ce-que-la-norme-de-mod%C3%A9lisation-des-processus-m%C3%A9tier>

ANNEXES

Annexe I :

Les Fichiers xml de création et de déclaration de modèles personnalisés (exemple modèle scolarité)

```

C:/Users/Nabil/Desktop/scolModel.xml (Premiers pas) - Brackets
Page Naviguer Débuguer Aide

1  <?xml version="1.0" encoding="UTF-8"?>
2  <!-- Definition of new Model -->
3  <model name="sco:scoomodel" xmlns="http://www.alfresco.org/model/dictionary/1.0">
4      <!-- Optional meta-data about the model -->
5      <description>ArchiveYourDocs</description>
6      <author>Administrator</author>
7      <version>1.0</version>
8      <!-- Imports are required to allow references to definitions in other models -->
9      <imports>
10         <!-- Import Alfresco Dictionary Definitions -->
11         <import uri="http://www.alfresco.org/model/dictionary/1.0" prefix="d" />
12         <!-- Import Alfresco Content Domain Model Definitions -->
13         <import uri="http://www.alfresco.org/model/content/1.0" prefix="cm" />
14     </imports>
15     <!-- Introduction of new namespaces defined by this model -->
16     <namespaces>
17         <namespace uri="scol.model" prefix="sco" />
18     </namespaces>
19
20     <types>
21         <!-- Enterprise-wide generic document type -->
22
23         <type name="sco:inscription">
24             <title>scol inscription</title>
25             <parent>cm:content</parent>
26             <properties>
27
28                 <property name="sco:inscrpdate">
29                     <title>Date d'inscription</title>
30                     <type>d:date</type>
31                     <mandatory>>false</mandatory>
32                 </property>
33
34                 <property name="sco:inscrpnom">
35                     <title>Nom etudiant</title>
36                     <type>d:text</type>
37                     <mandatory>>false</mandatory>
38                 </property>
39
40                 <property name="sco:inscrpniveau">
41                     <title>Niveau</title>
42                     <type>d:text</type>
43                     <mandatory>>false</mandatory>
44                 </property>
45
46             </properties>
47         </type>
48
49         <type name="sco:présence">
50             <title>scol ininscription</title>
51             <parent>cm:content</parent>
52             <properties>
53                 <property name="sco:presnom">
54                     <title>Nom etudiant</title>
55                     <type>d:text</type>
56                     <mandatory>>false</mandatory>
57                 </property>
58
59                 <property name="sco:presniveau">
60                     <title>Niveau</title>
61                     <type>d:text</type>
62                     <mandatory>>false</mandatory>
63                 </property>
64             </properties>
65         </type>
66     </types>
67 </model>

```

Figure 57 : Code xml de création de modèle

```
C:/Users/Nabil/Desktop/scol-model-context.xml (Premiers pas) - Brackets
page Naviguer Débuguer Aide

1 <?xml version='1.0' encoding='UTF-8'?>
2 <!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN" "http://www.springframework.org/dtd/spring-beans.dtd">
3 <beans>
4   <!-- Registration of new models -->
5
6   <bean id="example.dictionaryBootstrap" parent="dictionaryModelBootstrap" depends-on="dictionaryBootstrap">
7
8     <property name="models">
9       <list>
10        <value>alfresco/extension/scol/scolModel.xml.sample</value>
11      </list>
12    </property>
13  </bean>
14 </beans>
```

Figure 58 : Code xml de déclaration de modèle créé

Annexes II :

Configuration des fichiers .ini pour l'implémentation de l'application sur un serveur web sur le réseau de l'ISET.

Figure 59 : configuration du serveur