

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU



FACULTE DU GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'INFORMATIQUE

Mémoire de Fin d'Etudes de MASTER ACADEMIQUE

Domaine : **Mathématiques et Informatique**

Filière : **Informatique**

Spécialité : **Systèmes Informatiques**

Présenté par
Saci Sarah
Nait larbi sylvia

Thème

Annotation des anciens manuscrits arabes numérisés et Extraction de métadonnées

Mémoire soutenu publiquement le 27/09/ 2016. devant le jury composé de :

Président : M Rahemani Rabah

Encadreur : M SOUALAH Rabah

Examineur : M Yacine younes

Examineur : M Radja Hakim

Résumé

Notre travail de fin d'étude s'inscrit dans le cadre d'extraction de métadonnées à partir des documents annotés ou transcrits.

Tout au long de notre projet nous avons essayé d'atteindre au mieux notre objectif, qui se résume en la mise à jour du catalogue des manuscrits arabes anciens, dans le but de les préserver tout en ayant accès à leur contenu.

Pour mener à bien notre travail, nous avons conçu notre système sur les pas de la méthode répétitive incrémentale UP, grâce à un ensemble de diagrammes UML très explicites. Nous avons représenté l'architecture et le fonctionnement de notre système en tenant compte des relations entre les concepts utilisés et l'implémentation qui en découle.

Après avoir passé par plusieurs jalons nous avons pu aboutir à la construction de notre application dont la fonctionnalité maitresse consiste à extraire des métadonnées à partir des fichiers XML générés au moment de l'annotation des images afin d'alimenter automatiquement les notices constituant un catalogue de manuscrits arabes en utilisant le concept de similarité structurelle.

Remerciements :

***Nous remercions en premier lieu ALLAH le tous puissant
De nous avoir illuminé et ouvert les portes du savoir, et de
nous avoir donné la volonté et le courage d'élaborer ce travail.***

Nous tenons à exprimer nos remerciements les plus vifs à

notre encadreur Monsieur SOUALAH,

Qui a su nous guider et nous aider dans ce travail

Avec beaucoup de gentillesse et

Qui nous a permis de découvrir un domaine très

Intéressant celui des Manuscrits arabes.

Qu'il trouve ici notre estime et notre profond respect.

Nous tenons également à remercier messieurs les membres du

jury,

Qui ont bien voulu accepter de porter leur jugement sur ce

modeste travail

Et Que nous souhaitons à la mesure de leur satisfaction

Nous remercions également toutes les personnes qui, de près

ou de loin, ont participé à l'élaboration de ce mémoire.

Dédicaces

Je dédie ce modeste travail à :

*L'homme de ma vie, mon exemple éternel, mon soutien moral
et source de joie et de bonheur, celui qui s'est toujours sacrifié
pour me voir réussir, que dieu te garde dans son vaste paradis,
à toi mon père*

*A la lumière de mes jours, la source de mes efforts, la flamme
de mon cœur, ma vie et mon bonheur ; maman que j'adore.
Aux personnes dont j'ai bien aimé la présence dans ce jour, à
tous mes frères et mes sœurs, mes nièces, et mes neveux, je
dédie ce travail dont le grand plaisir leurs revient en premier
lieu pour leurs conseils, aides, et encouragements.*

*A celui que j'aime beaucoup et qui m'a soutenue tout au long
de ce projet : mon fiancé Tahar*

Sans oublier mes beaux-parents que j'aime.

A mon binôme Sara et toute la famille Saci

A Toutes mes copines houda, souad et dyhia

*Et à tous ceux qui ont contribué de près ou de loin pour que ce
projet soit possible, je vous dis merci.*

Nait larbi SYLIA

Dédicace

Je dédie ce Mémoire à

Mes très chers parents : sans votre affection, vos conseils, vos sacrifices, vos encouragements, ce travail n'aurait jamais pu être réalisé. Je vous présente ma pleine gratitude et mon profond respect, je ne vous remercierai jamais assez que Dieu vous donne une longue vie accompagné d'une bonne santé, je vous aime énormément.

A mon frère Yacine et mes sœurs : Hayat, Karima. Merci pour tout, pour votre amour, la confiance et l'énergie que vous m'aviez donnée.

A toute ma famille.

A ma tante kheloudja, son mari djamel et ses enfants meziane, kenza, ilyes, younes, et mon adorable serine.

A ma très chère amie et mon binôme, Sylia et toute sa famille.

Aux personnes qui m'ont toujours aidé et encouragé, qui étaient toujours à mes côtés, au long de mon chemin d'études supérieures, mes inoubliables amis: Kamel, Souad, Houda, Dyhia, Nina.

Saci Sarah

Sommaire

Introduction générale.....	1
-----------------------------------	----------

Chapitre I :

Généralités sur les manuscrits arabes

I.1. Introduction.....	3
I.2. Généralités sur les manuscrits.....	3
I.2.1. Définition d'un manuscrit	3
I.2.2. Supports des manuscrits.....	4
a. Supports d'écritures.....	4
b. Les instruments.....	5
c. L'encre.....	5
d. Les méthodes.....	5
I.2.3. Présentation des manuscrits arabe anciens.....	5
a. Les bibliothèques les plus réputées en conservation de manuscrits arabes.....	6
b. Caractéristique d'un manuscrit arabe.....	6
I.3. La numérisations des manuscrits arabes anciens.....	8
I.3.1. Définition de numérisation.....	8
I.3.2. Outils de numérisation.....	8
I.3.3. Les modes de numérisation des manuscrits.....	9
a. Mode image.....	9
b. Mode texte.....	9
I.3.4. Objectifs de la numérisation des manuscrits arabes anciens.....	9
I.4. Image.....	10
I.4.1. Notion d'image	10
I.4.2. Définition d'une image numérique.....	11
I.4.3. Types d'images.....	11
I.4.4. Définition d'un pixel.....	12
I.4.5. Caractéristique d'une image numérique.....	12
a. Codage d'une image.....	12
b. Résolution d'image.....	13
c. Définition d'une image.....	14
I.5. Conclusion.....	15

Chapitre II :

Annotation des manuscrits

II.1. Introduction.....	16
II.2. Qu'est ce que l'annotation.....	16
II.3. Objectifs de l'annotation.....	16
II.4 Délimitation graphique d'une zone d'annotation sur l'image.....	17
II.5 .Le Processus d'annotation.....	17
II.6. Format des Annotations.....	17
II.7. Choix du format d'encodage.....	18
II.7.1. Définition de la TEI.....	18
II.7.2. Objectifs de la TEI.....	19
II.7.3. La TEI guidelines pour la description de manuscrits.....	19
II.7.4. Structure globale TEI.....	20
II.8. Conclusion.....	21

Chapitre III :

Le catalogue des manuscrits arabes

III.1. Introduction.....	23
III.2. Catalogage.....	23
III.2.1. Définition d'un catalogue.....	23
III.2.2. Utilité de catalogage des manuscrits.....	24
III.2.3. Les formats du catalogue bibliographique.....	24
III.2.3.1. Format Marc.....	24
III.2.3.2. Format ISBD.....	25
III.2.3.3. Format EAD.....	25
III.2.4. Type de notice d'un catalogue.....	26
III.2.5. Structure de la notice bibliographique.....	27
III.2.6. Informatisation du catalogue.....	27
a. Avantage de l'informatisation du catalogue.....	27
b. Quelle informatisation pour le catalogue des manuscrits ?	28

c. Pourquoi l'automatisation du catalogue bibliographique des manuscrits ?	29
III.3. Les métadonnées.....	29
III.3.1. Définition du concept de «Métadonnée »	30
III.3.2. Métadonnées pour les documents numériques.....	30
III.3.3. Intérêt des métadonnées.....	31
III.4. Extraction d'information : Généralité.....	32
III.4.1. Extraction de métadonnées.....	32
III.4.2 .Principe de mise à jour ou d'intégration d'une métadonnée dans le la catalogue.....	33
III.5. Conclusion.....	35

Chapitre IV :

Conception

IV.1. Introduction.....	36
IV.2. Le Processus Unifié(UP)	36
IV.2.1. Définition.....	36
IV.2.2. Les caractéristique du processus unifié.....	36
IV.2.3. Cycle de vie du processus unifié.....	38
IV.2.4. Modélisation du système.....	40
IV.2.4.1. Expression des besoins.....	40
IV.2.4.2. Analyse.....	42
IV.2.4.3. Conception.....	45
IV.2.4.4. Implémentation.....	47
IV.2.4.5. Test.....	47
IV.3. Structure des documents annotés.....	49
IV.3.1. DTD est un document de définition de données ?	49
IV.3.2. C'est quoi un document valide ?	49
IV.3.3. Avantages de la DTD.....	49
IV.3.4. Document DTD réalisé.....	50
a. La DTD pour le fichier XML cachet.....	50
b. La DTD pour le fichier XML correction.....	50
c. La DTD pour le fichier XML ecouteurs.....	50
d. La DTD pour le fichier XML editeur.....	50
e. La DTD pour le fichier XML head.....	51
f. DTD pour le fichier XML histoire.....	51
g. DTD pour le fichier XML incipit/explicit.....	51
h. DTD pour le fichier XML MsIdentifier.....	52

i. DTD pour le fichier XML Rubrique.....	52
j. DTD pour le fichier XML Sommaire.....	52
k. DTD pour le fichier XML Transcription.....	52
IV.4. Le principe d'extraction de métadonnées.....	53
IV.5. Le concept de similarité structurelle.....	56
IV.5.1. Optimisation du calcul de similarité.....	57
IV.6. Conclusion.....	58
 Chapitre V:	
Réalisation	
V.1 Introduction.....	59
V.2. Fonctionnement général de l'application.....	59
V.3. Dictionnaire de données.....	59
V.4. Connexion à la base de données.....	63
V.5. Présentation des interfaces graphiques et leur fonctionnement.....	64
a. Interface d'accueil.....	65
b. Interface d'authentification.....	65
c. espace utilisateur.....	67
d. Interface annotation.....	68
e. Interface de la métadonnée correction.....	69
f. Espace Mediateur.....	72
g. Interface mise à jour du catalogue.....	73
h. Interface espace visiteur.....	74
V.6. Présentation des autres tables de base de données.....	77
V.7. Modèle logique de données pour les tables de la base de données.....	82
V.8. Conclusion.....	83
Conclusion générale.....	84
Bibliographie.....	85
Annexe.....	90
A.1. Présentation d'UML.....	90
A.2. Présentation d'XML.....	93
A.3. Présentation du langage de programmation.....	97
A.4. Présentation de DOM.....	99

Liste des Figures

Figure I. Papyrus avec ornementation	4
Figure I.2. Résolution d'une image	14
Figure I.3. Définition d'une image.....	15
Figure III.1. Illustration de 1ere cas.....	34
Figure III.2. Illustration de 2ème cas.....	34
Figure III.3. Illustration de 3ème cas	35
Figure IV.1. Cycle de vie du processus UP	39
Figure IV.2. Diagramme de cas d'utilisation	41
Figure IV.3. Diagrammes de séquence annotation.....	43
Figure IV.4 : Diagramme de séquence de métadonnée éditeur.....	44
Figure IV.5 : Diagramme de séquence de mise à jour du catalogue.....	45
Figure IV.6. Diagramme de classe.....	46
Figure IV.7. Diagramme de déploiement.....	47
Figure V.1. Architecture générale du système	59
Figure V.2. Page d'accueil.....	65
Figure V.3. Interface d'authentification du Médiateur.....	66
Figure V.4. Interface d'authentification de l'utilisateur.....	66
Figure V. 5. Interface d'Espace Utilisateur.....	67
Figure V.6. Interface annotation.....	69
Figure V.7. Interface de la métadonnée correction.....	70
Figure V.8. Interface Espace Médiateur.....	73

Figure V.9. Interface mise à jour du catalogue.....	74
Figure V.10. Interface espace visiteur	75
Figure V.11. Interface de lecture	76

Introduction générale

Les manuscrits arabes constituent une source précieuse pour l'humanité et contient le savoir de nos ancêtres dans divers sujets, certains de ces documents remontent à une époque lointaine.

En effet, l'unicité et la rareté de ces documents, les réduisent en objets fragiles, cependant leur préservation est nécessaire dans le but d'améliorer leur accès et leur sauvegarde à cause de nombreuses manipulations qui peuvent les détériorer et les endommager.

Pour pallier ce danger, et faire connaître les manuscrits et leurs contenus à un public élargi, la numérisation se propose comme une solution irréfutable. Elle permet d'évaluer la nature du fond documentaire afin de rendre la consultation de ces documents plus aisée et plus conviviale, et elle leur sert de support de substitution de qualité, ainsi elle préserve les collections existantes. Cette technique a apporté dans ce domaine une amélioration indiscutable.

Le constat étant fait, nous nous retrouvons dans l'obligation de dématérialiser les anciens manuscrits, ce qui renforce leur maintien en diffusant largement leur contenu. Mais, à ce niveau le catalogue demeure établi sur un modèle classique qui subit le déficit informationnel dû à la précarité des manuscrits et à leur état dégradé.

Par conséquent, nous proposons d'associer les experts et les érudits pour améliorer le contenu du catalogue en mettant à leur disposition les images des manuscrits numérisés sous une plateforme d'annotation. Cette dernière pourra être exploitée automatiquement afin d'enrichir le contenu du catalogue.

L'objectif de notre travail est de détailler l'ensemble des aspects de cette plateforme et de mettre en œuvre une telle solution.

Pour mener à terme notre projet, nous l'avons réparti en 05 chapitres dont:

- Le premier chapitre : comprend des généralités sur les manuscrits arabes, et leur numérisation.
- Le deuxième chapitre : porte sur l'annotation des manuscrits arabes, ainsi que sur le format d'encodage choisi pour la structuration des documents annotés.

- Le troisième chapitre : est consacré pour le catalogage des manuscrits arabes, aussi pour le concept d'extraction de métadonnées mettant à jour le catalogue suivant le degré de similitude entre deux document XML.
- Le quatrième chapitre : qui est lui-même divisé en deux partie, la première étant la modélisation du système, tandis que la seconde partie est réservée pour la représentation de la DTD et au calcul de similarité structurelle.
- Quant au dernier chapitre : il traite l'implémentation et la concrétisation du système, en représentant certaines interfaces graphiques, la structure de la base de données, ainsi que quelques lignes de code.

.

I.1. Introduction :

Parmi les richesses antiques devant être préservées, les manuscrits arabes anciens, qui représentent une véritable source du savoir, vu la masse importante d'informations dont ils regorgent, ce sont les incontestables témoins de la présence d'un texte d'une époque donnée, ils véhiculent les connaissances de cette ère. Seulement leur archivage sans y avoir accès reste infécond.

De ce fait, la numérisation est la solution la plus idéale pour les consulter sans les fragiliser, c'est une étape importante pour amener une plus value aux manuscrits, en leur permettant de se démarquer en termes de diffusion et de valorisation de l'information. En effet, l'idée est de dématérialiser les manuscrits, de les rendre immédiatement consultable électroniquement et de les conserver de façon sécuritaire.

Dans ce chapitre nous décrirons les divers aspects de la description des manuscrits arabes anciens et leur numérisation sous forme d'images numériques.

I.2. Généralités sur les manuscrits :

I.2.1. Définition d'un manuscrit :

Le manuscrit est un objet complexe : à la fois textuel, graphique et spatial, désigne tout ouvrage écrit à la main, de manière calligraphique ou non jusqu'à l'avènement de l'imprimante. [\[1\]](#)

Un manuscrit est, littéralement, un texte « écrit à la main », sur un support souple, que ce soit par son auteur (« manuscrit autographe ») ou par un copiste, avant l'invention de l'imprimerie. Avant la mise au point et la diffusion de l'imprimerie, à partir du milieu du XV^e siècle, tous les livres étaient des manuscrits. Au-delà de cette période, le manuscrit peut être utilisé pour des textes de diffusion restreinte ou pour des documents préparatoires (prise de notes, brouillon...). [\[6\]](#)

Les manuscrits arabes anciens traitent en général, les domaines suivants :

- L'histoire
- La théologie musulmane.
- L'astrologie.
- La littérature arabe.
- La dissertation en droit (tahrir fi l-fiqh).
- La médecine.

- La pharmacopée.

Notons par ailleurs, qu'un manuscrit est tout d'abord une œuvre produite d'une manière artisanale, utilisant des matériaux rares et coûteux. Il est de ce fait un document archéologique.

I.2.2. Supports des manuscrits : [\[10\]](#)

Les anciens manuscrits existent depuis plusieurs siècles. Leur survie et leur endurance face aux dégradations sont en grande partie dues à :

a. Supports d'écritures :

Les matières utilisées pour la confection de manuscrits étaient de deux genres différents : d'un côté, les matières dures, pierres, métaux, os, bois, sur lesquels on gravait plutôt qu'on n'écrivait; de l'autre, les substances de nature végétale ou animale, d'abord feuilles et écorces d'arbres, ou peaux d'animaux, puis les produits manufacturés dérivés : tissus, papyrus, parchemin, papier, matières plus propres à recevoir l'écriture.



Figure I.1: Papyrus avec ornementation

b. Les instruments:

Les instruments dont on se servait pour écrire différaient selon la matière employée. Le style ou stylet, tige de métal ou d'os, pointue d'un côté, plate de l'autre pour effacer, servait pour les tablettes enduites de cire ou formées de lamelles de plomb. Le pinceau s'adaptait rationnellement aux tablettes de bois et à l'écriture

hiéroglyphique¹ des Egyptiens; et c'est encore l'instrument en usage chez les Chinois. On écrivait, à proprement parler, d'abord avec un roseau apprêté (*Calamus*) et taillé en guise de plume, puis avec des plumes d'oiseaux, surtout d'oie, et même avec des plumes métalliques qui ne constituent nullement une invention moderne.

c. L'encre:

Les encres étaient de plusieurs couleurs. La plus communément employée, la noire, était souvent rehaussée par une encre de couleur, notamment la rouge, réservée plus spécialement pour des lettres initiales, les titres d'ouvrages ou de chapitres pour désigner les lignes ainsi écrites. L'encre bleue (et aussi, mais plus rarement, la verte et la jaune) servait parfois à de mêmes usages, dans le but d'obtenir plus de variété et d'agrément pour l'œil. L'or et l'argent étaient usités même pour l'écriture dans des manuscrits luxueux, mais plus généralement pour la peinture des initiales et la décoration des volumes.

d. Les méthodes

A l'aide du compas et de la règle, on traçait des raies verticales pour établir des marges en limitant l'espace pour l'écriture, puis des raies horizontales pour la distance des lignes entre elles. Ces raies étaient, pendant longtemps, tracées à sec, au moyen d'un style; puis, vers le XI^e siècle, au crayon; enfin, on régla souvent l'écriture avec des lignes rouges, et cet usage passa ensuite, mais à titre de luxe, au livre imprimé et se maintint jusqu'au XVIII^e siècle. Le grattoir était un des accessoires obligés de tout scribe.

I.2.3. Présentation des manuscrits arabes anciens :

Le manuscrit est le véritable témoin de la présence d'un texte à une époque donnée. Il véhicule les connaissances de cette ère. Il serait donc, intéressant de pouvoir accéder au contenu de ces manuscrits. Pour ce faire, il faudrait mettre en place un système de description adéquat des manuscrits.

Un manuscrit est un document unique, qu'il faut examiner depuis sa fabrication, jusqu'à l'intervention des lecteurs et des possesseurs successifs. Par conséquent, les rédacteurs de notices descriptives de manuscrits doivent relever tous les aspects caractérisant les manuscrits, relatifs à sa composition et à son contenu. [\[1\]](#)

¹ les caractères qui la composent représentent en effet des objets divers, naturels ou produits par l'homme.

a. Les bibliothèques les plus réputées en conservation de manuscrits arabes : [3]

Au total, 30 814 unités de conservation sont actuellement accessibles au lecteur. Parmi les plus connues :

- Bibliothèque de l'université de BIRMINGHAM.
- Bibliothèque de France (exemple le manuscrit arabe 328).
- Bibliothèque de la mosquée TELXASHAYA (exemple « coran d'othman »).
- La Bibliothèque de Sanaa au Yémen.
- Bibliothèques de Tombouctou au Mali.
- Les différentes Zaouiyas en Algérie,
- La Bibliothèque nationale d'Algérie (plus de 5000 manuscrits, celle du centre d'Adrar)

b. Caractéristique d'un manuscrit arabe :

Les manuscrits arabes, comme les manuscrits de différentes langues, ont des caractéristiques communes mais chacun de ces manuscrits a sa propre identité. Parfois le même élément qui est en commun avec les autres diffère dans son contenu.

Les manuscrits arabes diffèrent selon leurs régions (manuscrits orientaux ou maghrébins), l'appartenance religieuse de la communauté qui l'en a produit (manuscrits arabo-islamiques et arabo-chrétiens) et la période de leur achèvement (création).

Les caractéristiques qu'ils peuvent avoir sont les suivantes : [4]

✚ La présentation : tous les manuscrits arabes commencent au verso du premier feuillet, tandis que les rectos sont normalement réservés au « frontispice² », à l'inscription du titre et du nom de l'auteur, et parfois au cachet, au médaillon ou au nom du commanditaire du livre, etc.

² Illustration placée au regard de la page de titre d'un livre.

- ✚ Le début du texte peut être accompagné d'un décor particulier à la première page ou aux deux premières pages.
- ✚ Le début du texte et aussi le début de chaque chapitre ou section du texte ainsi que le début de chaque sourate dans le Coran, est toujours précédé par le "Basmala" formule préliminaire conventionnelle (Au nom de Dieu, le Clément, le Miséricordieux).
- ✚ Le texte est normalement écrit avec de longues lignes, sauf dans le cas du texte poétique où on peut trouver deux colonnes dans la même page. On trouve certaines exceptions dans les manuscrits arabo-chrétiens, le texte étant parfois présenté dans deux ou trois colonnes et éventuellement en deux ou trois langues différentes.
- ✚ L'encre utilisée : plusieurs couleurs sont utilisées dans l'écriture des manuscrits. il existe des règles à suivre dans l'usage des encres de couleur ; la rouge par exemple est conseillé pour l'écriture des noms propres, des nombres, des citations, des termes techniques et pour le texte commenté dans les ouvrages d'exégèse. Mais on trouve aussi des copies du Coran écrites avec de l'encre argentée et où le titre est doré. Les lignes sont parfois tracées à l'aide d'un instrument en couleur marron différente de l'encre du texte.
- ✚ Le texte encadré : en ce qui concerne le manuscrit coranique, le texte encadré apparaît au 8ème siècle, tandis que pour les textes non coraniques, l'apparition du cadre ne se trouve que plus tard. Le texte se présente normalement entouré par des cadres de différents styles soit ornementé, soit normal.
- ✚ Dans les manuscrits islamiques, le texte peut, dans quelques cas, se présenter non seulement sur des lignes horizontales mais aussi dans les marges, sur des lignes verticales ou en obliques

I.3. La numérisation des manuscrits arabes anciens :

La numérisation des manuscrits est une pratique récente, rendue possible par les nouvelles technologies du traitement de l'information.

On sous entend par numérisation, la dématérialisation. C'est inévitablement l'un des passages obligés de tout projet visant à optimiser la gestion des manuscrits anciens.

I.3.1. Définition de numérisation :

Numériser est l'art délicat de transformer un document en données informatiques. Cette technique consiste à convertir une information analogique en information numérique. Elle permet de reproduire le plus fidèlement possible un document sur écran ou sur fac-similé (papier). [\[5\]](#)

I.3.2. Outils de numérisation : [\[6\]](#)

Pour numériser une image, on discrétise la hauteur et la largeur, et on convertit, pour chaque point, les niveaux de lumière, soit globalement, soit pour chaque couleur primaire. L'échantillonnage de l'espace s'effectue de trois manières différentes :

- Un appareil photographique numérique utilise un transducteur à transfert de charge en forme de matrice à deux dimensions, avec un capteur par pixel. Le système transfère successivement les charges de chaque ligne, créant un signal électrique corrélé aux impulsions de transfert, et l'on peut ainsi quantifier le signal pour chaque élément capteur.
- Un scanner utilise généralement un transducteur à transfert de charge linéaire, dont les capteurs sont espacés d'une distance correspondant à la résolution transversale maximale. Le système transfère les charges de la ligne comme dans le cas précédent, puis il actionne un moteur qui fait avancer la ligne de la distance correspondant à la résolution souhaitée.
- Un scanner rotatif utilise un seul transducteur, qui avance lentement au-dessus de l'image montée sur un cylindre tournant. Le capteur, parcourant ainsi toute l'image, produit un signal électrique qui peut être converti en données numériques à chaque impulsion d'un signal corrélé à la rotation du cylindre.

I.3.3. Les modes de numérisation des manuscrits :

a. Mode image

Le texte contenu dans une page d'un manuscrit est représenté sur un mode photographique.

Ce type de document est obtenu par la numérisation directe du document. On obtient ainsi une copie électronique du document appelée image. Cette méthode est simple à réaliser et d'un coût relativement faible, mais elle génère cependant des fichiers encombrants.

b. Mode texte :

La numérisation en mode texte consiste à coder le texte de l'image numérisée sous forme de caractères. Pour ce faire, des outils de reconnaissance de caractères sont mis en œuvre. Ces outils récupèrent le contenu du document numérisé d'une "image" et génèrent un format texte.

I.3.4. Objectifs de la numérisation des manuscrits arabes anciens : [\[6\]](#)

Les objectifs de numérisation et archivage sont nombreux, elle remplit plusieurs fonctions :

- **Favoriser l'accès aux manuscrits :**

La numérisation permet de faciliter l'accès à l'information en offrant de nouveaux modes de consultation pour le public, car la majorité des manuscrits arabes originaux se trouve en unique exemplaire et ils sont fragiles.

- **Diffuser les manuscrits :**

La numérisation permet alors, l'accès multiple et simultané aux documents numérisés

- **Facilité d'accès :**

La numérisation offre une grande souplesse de diffusion, les documents deviennent alors consultables à distance facilement.

- **Sauvegarde des manuscrits :**

Toutefois, la conservation des manuscrits sous forme numérique assure une certaine longévité.

Grace à la numérisation, on évite la manipulation physique des manuscrits, leur conservation est ainsi améliorée. De plus, il est possible de les reproduire et de les diffuser sans dégradation.

- **Participation à la recherche d'information :**

La numérisation permet l'échange de connaissances et de compétences professionnelles. Grâce à la liaison entre les fichiers images et les fichiers textes s'y rapportant, il est plus facile et plus rapide de choisir les fichiers nécessaires à une utilisation ultérieure. Il sera alors possible d'utiliser les images des manuscrits numérisés pour une multitude de projets, tels que: sites Internet, publications, création de base d'image ... Etc

I.4. Image :

Comme

c'est mentionné précédemment, la numérisation en mode image est la méthode la plus simple à réaliser. De ce fait nous avons opté pour la définition de quelques notions que nous avons jugées utiles.

I.4.1 Notion d'image :

Une image est une représentation visuelle, de quelque chose (objet, être vivant et/ou concept).

C'est un Ensemble de points ou d'éléments représentatifs de l'apparence d'un objet, formés à partir du rayonnement émis, réfléchi, diffusé ou transmis par l'objet, c'est une représentation d'un objet matériel donnée par un système optique. [\[11\]](#)

I.4.2. Définition d'une image numérique :

Le terme d'image numérique désigne, dans son sens le plus général, toute image qui a été acquise, traitée et sauvegardée sous une forme codée représentable par des nombres (valeurs numérique) [\[20\]](#)

Cette image numérique est constituée de pixels contenant chacun différentes informations (intensité lumineuse, couleur...). Ces informations seront codées dans une grille échelonnée, le niveau de gris, de 0 à 63 par exemple. [\[7\]](#)

I.4.3. Types d'images : [\[4\]](#)

Il existe 2 sortes d'images numériques, les images matricielles et les images vectorielles.

- **L'image matricielle (bitmap) :**

Elle est formée d'une grille de points ou pixels. Chacun pouvant avoir une couleur différente. Une image matricielle est caractérisée notamment par

- sa dimension en pixels
- sa résolution
- son mode colorimétrique.

Les images vues sur un écran de télévision ou une photographie sont des images matricielles.

Nous obtenons également des images matricielles à l'aide d'un appareil photo numérique, d'une caméra vidéo numérique ou d'un scanner.

- **L'image vectorielle :**

Elle n'est pas composée de pixels mais définie par des fonctions mathématiques qui décrivent des lignes, des courbes etc. Dans ce cas on manipule des objets et non des pixels. Par exemple, un cercle est décrit par une fonction de type (cercle, position du centre, rayon). Ces images sont essentiellement utilisées pour réaliser des schémas ou des plans.

Ces images présentent deux avantages :

- Elles occupent peu de place en mémoire.
- Elles peuvent être redimensionnées sans perte d'information.

I.4.4. Définition d'un pixel : [8]

Une image numérique est constituée d'un ensemble de points appelés pixels (abréviation de *PICTure Element*) pour former une image. Le pixel représente ainsi le

plus petit élément constitutif d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image.

Remarque :

Une image bitmap contient un nombre fixe de pixels en hauteur et en largeur. Sa dimension en pixels correspond au nombre total de pixels qui la constituent.

I.4.5. Caractéristique d'une image numérique

a. Codage d'une image : [9]

Une image est donc représentée par un tableau à deux dimensions dont chaque case est un pixel. Pour représenter informatiquement une image, il suffit donc de créer un tableau de pixels dont chaque case contient une valeur. La valeur stockée dans une case est codée sur un certain nombre de bits déterminant la couleur ou l'intensité du pixel, on l'appelle profondeur de codage (parfois *profondeur de couleur*).

Les images informatisées se présentent en plusieurs niveaux de couleurs

Niveau	Couleurs
1 bit	2 couleurs (noir et blanc)
4 bits	16 couleurs (16 dégradés de gris allant du noir au blanc ou bien 16 couleurs différentes)
8 bits	256 couleurs ou nuances de gris (256 dégradés de gris allant du noir au blanc ou bien 256 couleurs différentes)
16 bits	65 536 couleurs (16 dégradés de gris allant du noir au blanc ou bien 16 couleurs différentes)

24 bits ou vraies couleurs	16 777 216 couleurs (cette représentation permet de représenter une image en définissant chacune des composantes (RGB, pour rouge, vert et bleu). Chaque pixel est représenté par un entier comportant les trois composantes, chacune codée sur un octet, c'est-à-dire au total 24 bits)
30 bits	1 073 741 824 couleurs

b. Résolution d'image : [\[8\]](#)

La résolution d'une image est définie par le nombre de pixels par unité de longueur. Elle s'exprime en dpi (dots per inches) ou ppp (points par pouce). Un pouce = 2,54 centimètres. La résolution d'une image numérique définit le degré de détail qui va être représenté sur cette image. Une image de résolution élevée compte un plus grand nombre de pixels (elle contient plus d'informations), elle est donc plus volumineuse qu'une image basse résolution de mêmes dimensions. La question de la résolution se pose:

- Soit au moment de numériser une image (à quelle résolution scanner)
- Soit au moment de créer un nouveau fichier image avec l'application adéquate (Photoshop par exemple)

La résolution permet ainsi d'établir le rapport entre la définition en pixels d'une image et la dimension réelle de sa représentation sur un support physique (écran, papier...).

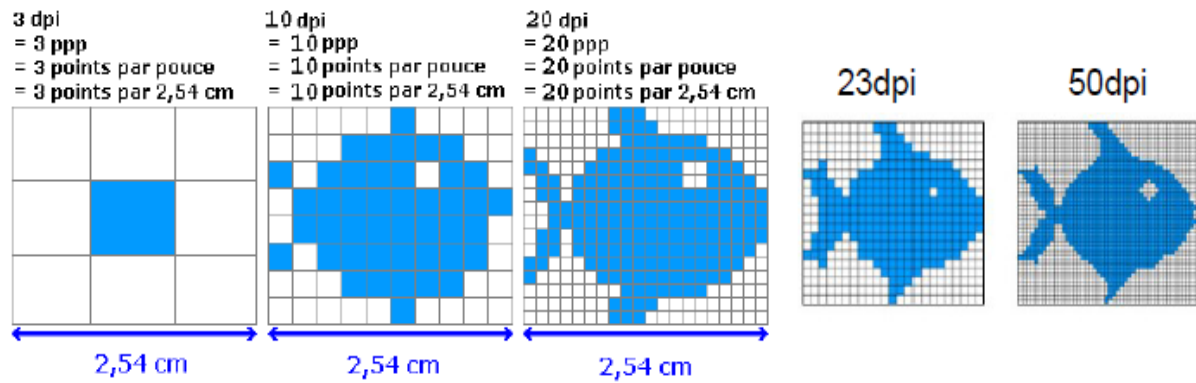


Figure I.2: Résolution d'une image [8]

c. Définition d'une image :

On appelle définition le nombre de points (pixels) constituant une image c'est-à-dire sa «dimension informatique» : c'est le nombre de colonnes de l'image que multiplie son nombre de lignes. Une image possédant 10 colonnes et 11 lignes aura une définition de 10 x11. [8]

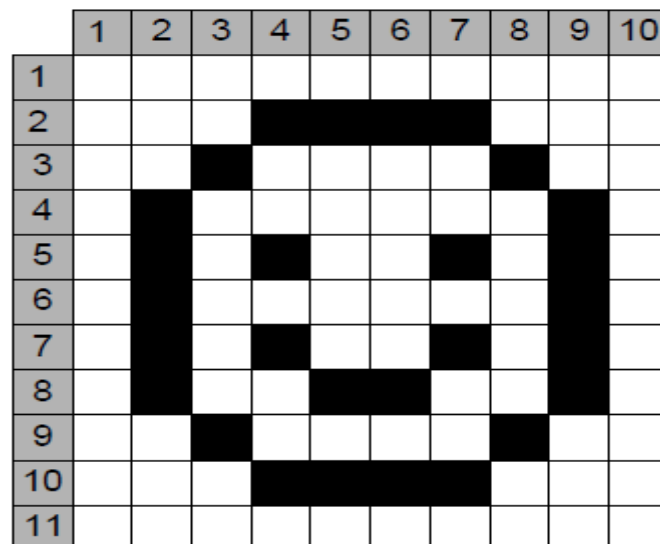


Figure I.3: Définition d'une image [8]

I.5.Conclusion :

Les manuscrits arabes représentent un véritable trésor englobant des secrets d'histoire d'une époque lointaine, leur valeur est inestimable, et leur préservation reste

incontournable, c'est pourquoi leur numérisation est indispensable, c'est une véritable alternative à l'édition traditionnelle.

La dématérialisation entraîne un meilleur suivi et une meilleure traçabilité des manuscrits, elle permet aussi de limiter au maximum le temps de recherche, et elle assure un accès multiple, quoiqu'elle conduise à la perte de l'originalité de ces œuvres.

Cependant, nous sommes amenés à effectuer la transcription des manuscrits arabes anciens, ce qui sera l'essentiel de notre prochain chapitre.

II.1. Introduction :

La numérisation des manuscrits est une technique de dématérialisation d'objets, elle permet non seulement la diffusion du contenu des manuscrits et leur mise à disposition au grand public, mais aussi leur préservation de tout acte de détérioration ou d'endommagement.

Les manuscrits arabes une fois numérisés, il ne reste qu'à les interpréter et les rendre beaucoup plus compréhensibles, en associant à certaines de leurs parties des notes ou plus généralement, une représentation textuelle de leur contenu ce que nous appelons annotation.

Dans le présent chapitre nous traiterons le concept d'annotation, l'encodage des ressources numérisées ainsi que le concept de format d'encodage de documents.

II.2. Qu'est ce que l'annotation :

Une annotation est un commentaire, une note, une explication ou tout autre remarque externe qui peut être attachée à un texte ou à une partie de celui-ci, elle peut être vue comme une métadonnée puisqu'elle fournit une information supplémentaire sur une donnée existante. [\[21\]](#)

Annoter une image est la pratique consistant à ajouter un texte lié à une zone délimitée graphiquement sur une image. Elle est définie par :

- Une donnée, c'est-à-dire l'annotation en tant que telle (un texte, un lien...).
- Des métadonnées (date de création, auteur de l'annotation...).
- Un lien à une zone, c'est-à-dire une portion/sélection du manuscrit. [\[14\]](#)

II.3. Objectifs de l'annotation :

- Faciliter la compréhension.
- Mettre à disposition du plus grand nombre de chercheurs ou curieux des sources primaires.
- La relecture d'un document.
- L'annotation est utilisée à des fins d'extraction de métadonnées, dans le but d'alimenter le catalogue des manuscrits arabes.

II.4. Délimitation graphique d'une zone d'annotation sur l'image :

La délimitation graphique d'une zone d'annotation sur l'image, est représentée par toutes formes géométriques en une ou deux dimensions qui pourraient servir de délimiteur. Vraisemblablement, les rectangles sont les types de formes les mieux adaptées à l'annotation de manuscrits.

II.5. Le processus d'annotation :

Nous pouvons résumer le processus d'annotation en quatre étapes principale :

1. Sélectionner la zone à annoter.
2. Déterminer le type de métadonnée.
3. Ecrire le texte correspondant.
4. Création du document annoté propre à la métadonnée présélectionnée.

II.6. Format des annotations :

Pour représenter les données (annotations), il faut choisir un modèle de structuration de l'information d'une façon claire et répondant aux besoins des utilisateurs, notamment en matière de recherche d'information. De ce fait, XML est le langage le plus approprié.

XML présente de nombreux intérêts pour le codage, que nous décrivons dans ce qui suit :

- XML peut être utilisé pour la description de tous types de document.
- XML rend possible des échanges de données entre des systèmes d'informations hétérogènes.
- XML n'est lié ni à un système d'exploitation ni à une famille de logiciel. Par défaut, les documents XML sont des documents texte dont le jeu de caractère est l'Unicode, c'est-à-dire un jeu de caractères très complet prenant en charge de nombreux systèmes d'écriture dont l'arabe.
- Il permet la consultation des données sous forme statique, après transformation (exemple : HTML).

II.7. Choix du format d'encodage :

Afin d'obtenir une bonne structuration des documents XML annotés servant à la mise à jour de la notice bibliographique, nous avons opté pour XML/TEI qui fournit des outils adaptés à la description des manuscrits et répond, de ce fait entièrement à nos besoins. Il se présente sous une forme souple et adaptable.

En effet, la structure modulaire de la TEI permet à l'utilisateur de choisir les outils qui lui conviennent.

II.7.1. Définition de la TEI:

"Text Encoding Initiative" est un projet international qui vise à la mise au point d'un ensemble de normes pour la préparation et l'échange de textes électroniques. La TEI est née lors d'une réunion organisée en 1987 au Vassar College (Poughkeepsie, New York) à laquelle ont participé diverses personnalités travaillant dans le domaine de l'archivage, de la structuration ou de l'analyse des textes électroniques.[\[6\]](#)

L'objectif était de mettre en place un nouveau format :

- Complet, qui permette un traitement efficace.
- Simple, clair et concret.
- Facile à utiliser sans logiciel particulier.
- Rigoureusement défini.
- Ouvert à des extensions définies par les utilisateurs.
- Compatible avec les standards existants ou en développement.[\[28\]](#)

La TEI a été créée officiellement en 1988 sous l'égide de l'Association for Computers and the Humanities³, de l'Association for Computational Linguistics⁴ et de l'Association for Literary and Linguistic Computing⁵.[\[27\]](#)

Text Encoding Initiative est un consortium qui propose des recommandations afin de normaliser l'encodage de toutes sortes de documents sous le format numérique. [\[13\]](#)

II.7.2. Objectifs de la TEI :

- Le principal objectif de la TEI est de faire les recommandations, qui devaient :

³ Société professionnelle pour les sciences humaines numériques

⁴ Société pour les personnes travaillant sur les problèmes impliquant le langage naturel et de calcul

⁵ Est une organisation des humanités numériques fondée à Londres en 1973

- être suffisamment précises pour représenter les caractéristiques textuelles d'un texte.
 - être simples, claires et concrètes.
 - être utilisables facilement par les chercheurs et ne pas nécessiter l'utilisation de logiciels spécifiques.
 - permettre une définition rigoureuse et des traitements efficaces des textes.
 - prévoir des extensions définies par l'utilisateur.
 - respecter les standards existants ou émergents.
- Faciliter la création, l'échange et l'intégration de données textuelles sous un format informatisé (TEI a été développée alors que le www n'existait pas).
 - Préférer les solutions générales à celles spécifiques à une discipline en même temps permettre la spécialisation et l'extension.
 - Faciliter la structuration des documents annotés.

II.7.3. La TEI guidelines pour la description de manuscrits : [\[13\]](#)

La TEI est un ensemble de guidelines (lignes directrices) : peu prescriptives représentant un consensus au sujet des distinctions significatives dans un vaste ensemble de matériaux textuels qui s'expriment en deux gros volumes de prose et un ensemble de définitions formelles.

Les *recommandations* de la TEI - *Text Encoding Initiative (TEI) guidelines* - s'adressent à tous ceux qui souhaitent échanger des informations stockées sous forme électronique. Elles mettent l'accent sur l'échange des données textuelles mais d'autres types de données comme les images et les sons sont également pris en compte.

Les *recommandations* peuvent être appliquées aussi bien pour créer de nouvelles informations que pour échanger des informations existantes.

Les *lignes directrices* fournissent le moyen de rendre explicites certaines caractéristiques d'un texte, de façon à faciliter le traitement de ce texte par des programmes informatiques pouvant s'exécuter sur des plates-formes différentes. Cette tâche d'explicitation est appelée *balisage* ou *codage*. La représentation d'un texte sur un ordinateur met toujours en œuvre une forme de balisage ou une autre.

La TEI tire son origine d'une part de l'anarchie qui règne dans la communauté scientifique en matière de format, et d'autre part du nombre croissant de traitements que les chercheurs opèrent sur les textes sous forme électronique.

II.7.4. Structure globale TEI :

La TEI (*Text Encoding Initiative*), utilise un ensemble très complexe de balises permettant de traiter de façon informatique des textes de tous genres : prose, poésie, théâtre, annotation...

La structure de base d'un texte TEI repose sur un document original. Un document est en général un livre imprimé, mais peut être également un manuscrit. La structure d'un texte TEI décrit le contenu d'un document manuscrit ou imprimé qui comprend une partie textuelle prépondérante.

Nous avons choisi d'encoder nos annotations des manuscrits en XML, en suivant les recommandations de la TEI P5 Guidelines.

Tout document conforme à la TEI P5 Guidelines comporte :

1. Un en-tête TEI (balisé comme un élément `<teiHeader>` assez comparable à une fiche de catalogue.
2. La transcription du texte lui-même (balisé comme un élément `<text>`).

L'en-tête TEI contient des informations analogues à celles que l'on trouve sur la page de titre d'un texte imprimé. Il contient jusqu'à quatre parties :

1. Une description bibliographique du texte électronique.
2. Une description de la manière dont il a été codé.
3. Une description non-bibliographique du texte (le « profil » du texte).
4. Un historique de révision.

Un texte TEI peut être unitaire (une œuvre isolée) ou composite (un recueil d'œuvres, comme une anthologie). Dans un cas comme dans l'autre, le texte peut éventuellement comporter des pièces liminaires ou des annexes. Entre les deux se trouve

le corps du texte qui, dans le cas d'un texte composite, peut comporter des groupes, chacun contenant encore des groupes ou des textes.

Exemple:

```
<TEI>
  <teiHeader> [informations contenues dans l'en-tête TEI ]
</teiHeader>,
  <text>
    <front> [textes préliminaires...] </front>,
    <body> [corps du texte...] </body>
    <back> [annexes... ] </back>
  </text>
</TEI>
```

II.8. Conclusion :

L'annotation permet de représenter l'information de façon claire en ajoutant des corrections, ainsi que d'autres types d'information aux textes manuscrits.

Dans le souci d'une bonne représentation des données et une meilleure structuration de l'information, nous avons eu recours à la TEI.

Ainsi nos documents annotés auront tous la même structure que celle du catalogue, afin de faciliter son alimentation.

Le prochain chapitre portera sur l'extraction de métadonnées utile pour la mise à jour du catalogue.

III.1. Introduction :

Un manuscrit est une œuvre unique, cela sous entend la nécessité de mise en place de méthodes et d'outils de conservation adéquats afin d'assurer une longévité de consultation et d'accès à cette ressource. Pour cela, la mise en place des compagnes de numérisation est indispensable, ces dernières sont initiées par les instituts chargés de la conservation de ces biens qui expriment deux besoins principaux:

- Le besoin d'une conservation à l'abri.
- La possibilité de créer des bibliothèques numériques accessibles aux experts ou au grand public mettant en œuvre des fonctionnalités adaptées et proposant des outils d'exploration.

Dans le secteur des bibliothèques, toutes les activités professionnelles reposent sur la création et la gestion des catalogues. Le passage des catalogues papiers aux catalogues numériques est un projet ambitieux qui permettra d'accéder aux manuscrits numérisés.

Dans la partie suivante nous montrons un aperçu sur les catalogues des manuscrits arabes.

III.2. catalogage :

Le Catalogue des manuscrits s'inscrit dans un contexte entièrement nouveau. Il a en effet connu d'importantes transformations au cours de ces dix dernières années, soit dans le contenu des notices, ou bien dans leur forme.

III.2.1. Définition d'un catalogue :

_____ Un catalogue peut être défini comme un ensemble de notices de documents rédigées et présentées selon des normes sous forme papier ou sous un format électronique. Il permet de rechercher un document et de le localiser.

Tout document doit être catalogué, - Cataloguer, c'est décrire chaque document dans le catalogue en lui associant une notice bibliographique de telle sorte qu'il puisse être retrouvé (à partir de divers critères de recherche), et identifié (sans le confondre avec un autre document). [\[22\]](#)

Le catalogue est un outil de description, d'identification et de localisation des collections d'un établissement.

III.2.2. Utilité de catalogage des manuscrits :

Un catalogue est une liste d'éléments composant une collection d'objets. Cette liste, établie suivant un ordre déterminé, est destinée à faciliter la recherche, l'identification et la localisation de ces objets.

Le catalogue a pour but :

- de décrire un document et de permettre au lecteur l'accès à ce document.
- d'indiquer la localisation des documents à la différence de la bibliographie qui les recense sans les localiser. [\[6\]](#)

III.2.3. les Formats du catalogue bibliographique : [\[18\]](#)

Un format est une représentation formelle des données, il existe plusieurs formats normalisés de structuration et de représentation de notices bibliographique.

III.2.3.1. Format Marc :

Un format Marc est la définition des champs et sous-champs utilisés et de leurs propriétés.

Des mêmes données peuvent être structurées de manière différente selon l'utilisation.

Dans un format Marc, les données sont structurées sous forme de champs identifiés par trois chiffres.

Chaque champ peut-être divisé en sous-champs identifiés par le caractère "\$" suivi d'un caractère alphabétique ou numérique.

Les sous-champs sont précédés de plusieurs caractères (généralement deux) précisant le contenu du champ appelés indicateurs.

A chaque zone Marc (champ ou sous-champ) sont associés des propriétés :

- **Obligatoire :** Si un champ est obligatoire, il doit être présent au moins une fois dans la notice. Si un sous-champ est obligatoire, il ne peut y avoir de champ sans au moins une occurrence de ce champ.

- Répétable : Un champ non répétable ne peut apparaître qu'une fois dans une notice, et un sous-champ une fois par champ.

III.2.3.2. Format ISBD :

Le concept de Description bibliographique internationale normalisée ou ISBD :

- précise les éléments requis pour une description bibliographique établie par une agence bibliographique nationale.
- prescrit leur ordre de présentation, en les répartissant entre huit zones cohérentes.
- définit une ponctuation pour les délimiter, cette ponctuation étant utilisée comme un codage.
- donne des règles pour leur transcription à partir de sources d'informations clairement identifiées.

Elaboré et maintenu par l'IFLA, l'ISBD garantit une description bibliographique rigoureuse et fiable, facilement intégrable dans des catalogues multimédia et échangeable entre établissements au niveau international.

III.2.3.3. Format EAD:

C'est un format basé sur le langage XML qui permet de structurer des descriptions de manuscrits ou de documents d'archives.

Ce format a été initialement conçu pour permettre le traitement rétrospectif des instruments de recherche existants, documents imprimés ou produits avec des outils de traitement de texte dans la perspective d'une édition papier. Ceci explique que :

- La structure générale du modèle reste proche de celle des documents imprimés.

- Les règles d'utilisation sont peu contraignantes, afin de pouvoir être adaptées à des contextes divers.
- Le modèle combine des éléments de description structurée et des éléments d'encodage de texte.

Il existe d'autres formats et normes pour la représentation des notices tel que :

- Format UNIMARC, un tel format est très utile pour des catalogueurs professionnels.
- REMIS (PREservation Metadata Implementation Strategies), Il sert à exprimer des informations sur les documents numériques en vue de leur pérennisation.

III.2.4. Type de notice d'un catalogue: [\[16\]](#)

Un catalogue se présente comme un ensemble de fiches appelées notices bibliographiques.

➤ **Notices bibliographiques**

La notice bibliographique est la fiche descriptive du document. Elle contient l'ensemble des éléments de description d'un document.

Elle est composée de plusieurs champs, ces champs sont des zones qui permettent l'interrogation et le repérage du document. Et ils peuvent varier d'un catalogue à l'autre : auteur, titre et sujet. [\[17\]](#)

➤ **Données locales**

Ce sont les données qui sont propres à une bibliothèque et correspondent aux usages et modes de travail de chaque établissement. Elles peuvent être saisies:

- Soit dans les champs de données locales .
- Soit dans les données d'exemplaire (n° d'inventaire, cote, etc.).

Les données locales doivent pouvoir être ajoutées dès qu'elles sont disponibles : cotes validées, cotes magasin, secteur documentaire, rayon.

Les deux parties de la notice (données bibliographiques et données locales) doivent être indépendantes l'une de l'autre : elles doivent pouvoir être mises à jour séparément.

➤ **Notices de gestion**

Elle contient les informations minimales pour permettre l'identification d'un titre.

➤ **Notices autorités**

Une notice d'autorité permet le contrôle des accès à une notice bibliographique. Elle contient des informations sur le point d'accès (dates biographiques, notes, sources de la saisie).

III.2.5. Structure de la notice bibliographique : [\[15\]](#)

Les données bibliographiques d'une notice se répartissent en zones

- Zones descriptives permettant d'accéder à la notice par le titre ;
- Zones d'accès normalisé permettant d'accéder à la notice le plus souvent par l'intermédiaire de la liste de vedettes ou des autorités ;
- Zones de liens permettant de lier une notice bibliographique à une autre notice bibliographique.

II.2.6. Informatisation du catalogue:

La numérisation des catalogues change l'art du possible dans le domaine des bibliothèques. En effet, elle permet la création de nouveaux services aussi bien pour les lecteurs que pour les professionnels.

Le passage des catalogues papiers aux catalogues informatisés pose le problème de la rétro-conversion⁶ des données. Mais cela n'empêche de dire que l'informatisation est la technologie qui a amélioré l'utilisation et la mise à jour du catalogue traditionnel.

a. Avantage de l'informatisation de catalogue :

L'informatisation des catalogues de manuscrits présente de nombreux avantages :

⁶ la transformation des catalogues papiers en dossiers numériques et leur intégration dans un seul dossier.

- Les notices pouvant être mises à jour et complétées .
- L'insertion de liens entre les notices et d'autres documents, comme des reproductions numériques ou des notices d'autorité.
- Les possibilités d'interrogation inégalées par le croisement des critères de recherche.
- La consultation à distance.

b. Quelle informatisation pour le catalogue des manuscrits ? [\[1\]](#)

Notre étude porte sur les manuscrits arabes, nous allons donc nous limiter à ce type de manuscrits. Les projets d'informatisation du livre ancien proposent en général trois types de solutions opérationnelles différentes, qui se résument comme suit :

1) La première solution s'agit d'un catalogue en format texte que le chercheur, expert dans le domaine et supposé ne rencontrer aucun problème de concepts ou de recherche d'information, utilisera en faisant appel à un moteur de recherche un style de traitement de texte. Elle répond à l'objectif de base et aux contraintes de temps de mise en ligne du catalogue imposé aux divers projets.

2) La deuxième solution utilise une structuration forte des données. Il s'agit de faire appel à un système de balisage défini après analyse de la structure de la notice. Cette solution est coûteuse car nécessite un personnel qualifié et l'opération est relativement lente, mais elle fournit aisément une notice et une mise à jour facile de la base.

3) La troisième solution est plutôt un mixte des deux solutions: cette solution a pour objectif de faire intervenir le moins possible l'homme dans le processus de mise en ligne du catalogue. En effet, le catalogue en format texte est utilisé en entrée et un système de reconnaissance automatique des caractères est généralement, mis en place. Il a pour rôle la transformation des images en texte, qui est ensuite indexé de manière automatique. Le programme procède à l'analyse typographique des notices.

Il crée de ce fait, des index de noms, de langue, de date...etc. Un catalogue à structuration forte utilisant un système de balisage est alors obtenu.

c. Pourquoi l'automatisation du catalogue bibliographique des manuscrits?

Un ouvrage mal catalogué peut être considéré comme un ouvrage perdu.[\[1\]](#)

Le catalogue automatisé est destiné pour:

- L'amélioration de l'utilisation des anciens catalogues manuels : ce qui permet d'obtenir des accès rapides aux fonds documentaires des institutions de conservation de documents.
- Faciliter la recherche, en effet il permet de repérer la disponibilité d'un document dans un premier temps, puis l'identification et la localisation de manuscrit.

Grace aux nouvelles technologies, l'accès à distance aux manuscrits exige tout une chaîne de processus, dont la création de métadonnées est indispensable.

Les métadonnées sont un instrument qui transforme les données brutes en connaissances. Elles représentent une valeur ajoutée à l'information en permettant leur compilation et leur repérage. Malgré la différence de structure, tous les types de métadonnées poursuivent un objectif commun : offrir des éléments de description pour faciliter l'accès à des ressources données en fournissant toutes les informations les concernant.

Dans ce qui suit nous allons voir les métadonnées de près.

III.3. Les métadonnées

L'objectif de notre travail est de permettre aux utilisateurs d'accéder aux manuscrits arabes numérisés.

L'accès à une ressource numérisée se fait selon un critère d'accès spécifique. En effet le lecteur commence sa recherche à l'aide d'une demande particulière. Il peut être le titre manuscrit, son auteur ou toute autre caractéristique manuscrite décrite dans les métadonnées. Mais, qu'est ce qu'alors une métadonnée ?

III.3.1. Définition du concept de « Métadonnée »

L'étude et l'analyse humaines ou automatiques d'un document peuvent fournir des « indications » quant à l'usage des informations qu'il représente. Pour permettre l'exploration du document et définir les dites indications, il peut être approprié d'utiliser des informations définissant le plus précisément possible l'indication en question ou son contexte. Ces indications sont des informations appelées « métadonnées ». Une métadonnée est littéralement une donnée sur une donnée ou un document. C'est un ensemble structuré d'informations décrivant une ressource quelconque. Une métadonnée peut être utilisée dans la gestion, la description, la préservation de collections de ressources de natures différentes. [23]

La définition des métadonnées la plus simple et consensuelle est la suivante : les métadonnées sont de l'information sur l'information électronique.

Une métadonnée peut être utilisée à des fins diverses:

- la description et la recherche de ressources
- la gestion de collections de ressources
- la préservation des ressources

III.3.2. Métadonnées pour les documents numériques :

Un document numérique est une suite de fichiers : il est décrit par **un identifiant unique** et **un ensemble de métadonnées**.

Les outils de recherche d'informations opérationnels actuellement, attribuent généralement à chaque document une simple liste de mots clés pour permettre leur recherche. Les travaux étudiant la création, la structuration et la modélisation des descripteurs soutiennent par contre que c'est en organisant les descripteurs dans une structure, que des requêtes complexes pourront être posées. Ainsi la recherche d'informations portant sur ces documents numériques devient plus riche. Les documents numériques contiennent de nombreuses métadonnées implicites et/ou explicites, classées selon trois principaux types : [24]

➤ Des métadonnées descriptives pour :

- Donner une description bibliographique approfondie et détaillée dans un format normalisé permettant l'échange de données.

- Rattacher le document à l'original ou à différentes versions d'un document.
- Donner accès à la copie numérique.
- **Des métadonnées de structure** pour :
 - Rattacher les fichiers d'un même document entre eux.
 - Reconstituer la structure du document : connaître tous les fichiers qui composent un document (fichiers textes, images...).
 - Connaître la relation physique entre ces fichiers (ordre d'affichage, fichier cible donnant accès à l'ensemble).
- **Des métadonnées administratives** pour :
 - Gérer les droits : d'accès (droits d'auteur, confidentialité) et d'usage (droits d'impression, de reproduction, de modification...).
 - Préserver les informations techniques nécessaires à la lecture des fichiers.
 - Garantir l'intégrité des fichiers et le suivi de leurs éventuelles modifications.

Dans notre cas nous avons utilisé les métadonnées de structure, qui servent à connaître l'organisation de l'information. Il y a deux niveaux de structure : un niveau logique et un niveau physique.

- Le niveau logique définit les liens entre les éléments qui ont un sens pour l'utilisateur : numéro de page, titres, et chapitre, etc.
- Le niveau physique qui définit comment sont enregistrés les objets numériques.

III.3.3. Intérêt des métadonnées : [\[23\]](#)

Devant la multiplication de la publication des documents numériques, l'apparition des métadonnées présente de nombreux intérêts, parmi lesquels:

- faciliter la gestion et l'archivage de l'information (informations sur le cycle de vie des documents, gestion des collections de ressources, gestion des archives électroniques).
- Gestion et protection des droits (les droits de propriété intellectuelle, les droits d'accès à des pages web).

- Authentification d'un texte (encoder une signature électronique pour valider un texte sur Internet).
- Faciliter l'interopérabilité (partage et échange des informations).
- Faciliter la recherche d'informations .

Nous avons parlé précédemment de catalogage et de métadonnées, pour enfin arriver à notre objectif, qui est l'extraction de métadonnées. Ces dernières vont participer à la mise à jour du catalogue. De ce fait la partie suivante portera sur l'extraction de métadonnées.

III.4. Extraction d'informations : Généralités

L'extraction d'information (EI) consiste à remplir automatiquement des formulaires ou une banque de données à partir de textes écrits en langue naturelle. Elle s'oppose classiquement à la recherche documentaire (ou recherche d'information RI) qui vise à retrouver dans une base de documents un ensemble de documents pertinents au regard d'une question. L'extraction met en œuvre une analyse du texte pour interpréter et construire une représentation formelle qui permettra d'apporter automatiquement des réponses précises à l'utilisateur. Il ne s'agit donc pas simplement de sélectionner un fragment brut du texte, mais de mettre des éléments en relation pour restituer une information complète et structurée. [\[25\]](#)

III.4.1. Extraction de métadonnées:

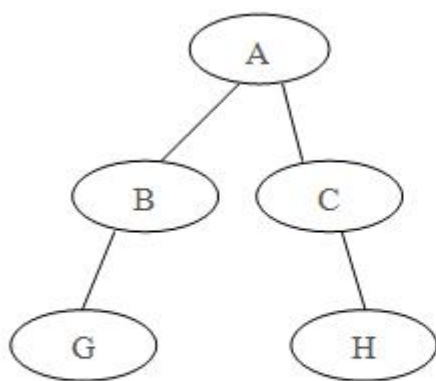
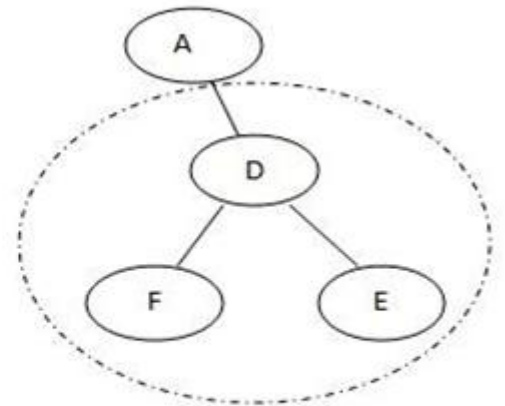
L'extraction de métadonnées peut être considéré comme un cas particulier de l'extraction d'informations, elle est basée sur les informations que l'on peut extraire des documents, ces informations peuvent être différentes mais elles portent toutes sur l'apparence visuelle du document, soit dans sa totalité, soit dans certaines régions le constituant. L'extraction se fait à partir des documents XML précédemment générés par les annotations et la transcription des images de manuscrits arabes. [\[26\]](#)

Les métadonnées ainsi extraites seront comparées au contenu de la notice bibliographique. Cette comparaison est rendue possible par le concept de similarité structurelle qui calcule le degré de similitude entre deux arbres XML pour ensuite compléter et mettre à jour le catalogue.

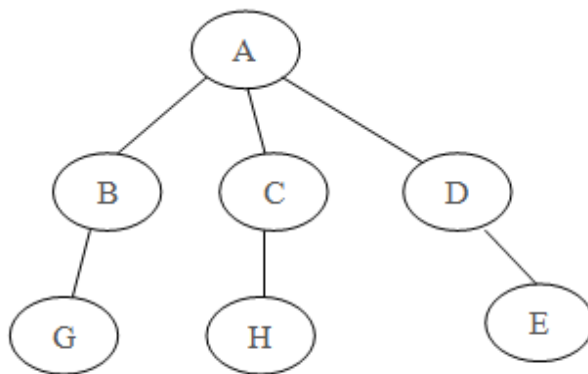
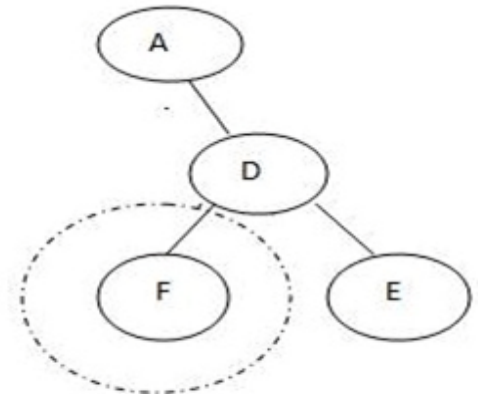
III.4.2. Principe de mise à jour ou d'intégration d'une métadonnée dans le catalogue [\[1\]](#)

En général, à chaque document annotation, ou transcription correspond une arborescence dans la notice bibliographique. Mais, il peut avoir des cas de dissemblance, à ce moment là, le problème d'hétérogénéité des deux documents se pose, ce dernier doit être réglé avec le calcul de degré de similitude entre les arborescences des deux fichiers. La similarité structurelle, permet d'évaluer la nécessité d'intégrer la métadonnée dans le catalogue. Cependant, nous nous retrouvons face à trois cas possible:

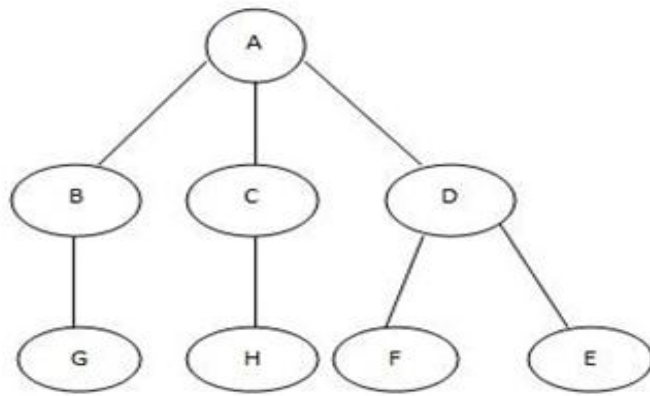
Cas 1 : L'élément annoté ou transcrit est inexistant dans la notice bibliographique, ce qui nécessite l'intégration totale du document-annotation ou du document transcription dans la notice.

**Notice bibliographique****Document annoté ou transcrit****Figure -III.1-: Illustration du 1er cas**

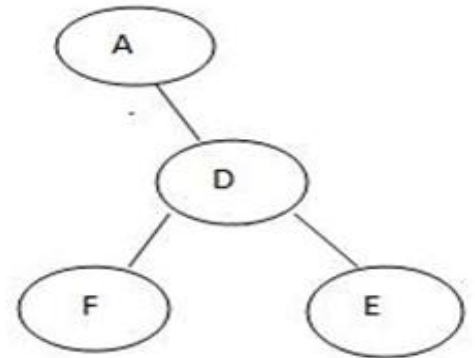
Cas 2 : Une partie du document annoté ou transcrit est inexistante dans la notice bibliographique, ce qui induit à compléter cette dernière.

**Notice bibliographique****Document annoté ou transcrit****Figure -III.2- : Illustration du 2ème cas**

Cas 3: L'élément du document annotation ou du document transcription figure dans la notice bibliographique mais leur contenu est non identique, il s'agit de la modification du contenu de la notice bibliographique par celui du document annoté ou du document transcrit.



Notice bibliographique



Document annoté ou transcrit

Figure -III.3-: Illustration du 3ème cas

III.5. Conclusion :

L'objectif de ce chapitre est de présenter le catalogue des manuscrits arabes informatisé accessibles hors les murs des lieux de conservation, d'introduire le concept de métadonnées constituant une forme de données structurées, servant à rendre l'information plus claire. Et enfin de montrer la facilité qu'apporte l'extraction automatique des métadonnées pour la mise à jour de notre catalogue, permettant une fiabilisation, actualisation et pertinence des informations diffusées.

N'empêche que l'inexistence d'une norme standard de catalogage pour les manuscrits arabes reste une anomalie vu que de nombreuses méthodes de catalogage ont été mises au point.

Dans le prochain chapitre nous allons présenter la démarche de modélisation suivie pour concevoir de notre système.

IV.1. Introduction

La modélisation est le pilier de toute activité qui conduit au déploiement de logiciel de qualité. Les modèles sont construits pour spécifier la structure et le comportement attendu d'un système, pour visualiser et contrôler son architecture et pour mieux comprendre son organisation.

Dans ce chapitre, nous entamons la modélisation de notre système. Pour cela, nous avons choisi d'utiliser la démarche UP, en s'appuyant sur le langage de modélisation orienté objet UML (Unified Modeling Language), qui permet de bien représenter les aspects statiques et dynamiques de notre projet, par la série des diagrammes qu'il offre, par la suite nous allons développer le concept de similarité structurelle et son intervention dans notre système.

IV.2. Le Processus Unifié (UP) :

IV.2.1. Définition :[\[19\]](#)

Pour définir le processus unifié, nous allons simplement définir les deux termes qui le composent :

- **Processus** : Suite continue d'opérations constituant la manière de fabriquer. En d'autres termes, c'est une succession de tâches dans le but d'accomplir un travail, un projet.
- **Unifié** : Etre amené à l'unité, se fondre en un tout. En fait, les méthodes d'analyse et de conception orientées objet, étaient variées jusqu'à leur unification.

Le processus unifié est un processus de développement logiciel itératif, centré sur l'architecture, Piloté par des cas d'utilisation et orienté vers la diminution des risques. 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'entreprises, à différents niveaux de compétences et à différentes tailles de l'entreprise.

IV.2.2. Les caractéristiques du processus unifié [\[19\]](#)

D'après les auteurs d'UML, un processus de développement qui possède ces qualités devrait favoriser la réussite d'un projet. Cependant, dans le cadre de la modélisation d'une application informatique, les auteurs d'UML préconisent d'utiliser une démarche Itérative et incrémentale, guidée par les besoins des utilisateurs du système, Centrée sur l'architecture logicielle, et pilotée par les risques.

❖ UP est itératif et incrémental

Le projet est découpé en itérations ou étapes de courte durée qui permettent de mieux suivre l'avancement globale. A la fin de chaque itération une partie exécutable du système est produite, de façon incrémentale (par ajout).

❖ UP est guidé par les cas d'utilisation d'UML

Le but principal d'un système informatique est de satisfaire les besoins du client. Les cas d'utilisation permettent d'illustrer ces besoins qui servent de fil rouge, tout au long du cycle de développement (itératif et incrémental):

- A chaque itération de la phase d'analyse, il faut clarifier, affiner et valider les besoins des utilisateurs.
- A chaque itération de la phase de conception et de réalisation, il est très important de veiller à la prise en compte des besoins des utilisateurs.
- A chaque itération de la phase de test, vérifier que les besoins des utilisateurs sont satisfaits.

❖ UP est centré sur l'architecture

Une architecture adaptée est la clé de voûte du succès d'un développement. Elle décrit des choix stratégiques qui déterminent en grande partie les qualités du logiciel. Tout système complexe doit être décomposé en partie modulaire afin d'en faciliter la maintenance et l'évolution.

❖ UP est piloté par les risques

Les risques majeurs du projet doivent être identifiés au plus tôt mais surtout levés le plus rapidement.

IV.2.3. Cycle de vie du processus unifié : [\[19\]](#)

L'objectif d'un processus unifié est de maîtriser la complexité des projets informatiques en diminuant les risques. UP est un ensemble de principes génériques adapté en fonctions des spécificités des projets.

- **L'architecture bidirectionnelle** : UP gère le processus de développement par deux axes. (figure IV.1).
- **L'axe vertical** : représente les principaux enchaînements d'activités, qui regroupent les activités selon leur nature. Cette dimension rend compte l'aspect

statique du processus qui s'exprime en termes de composants, de processus, d'activités, et d'enchaînements.

- **L'axe horizontal** : représente le temps et montre le déroulement du cycle de vie du processus, cette dimension rend compte de l'aspect dynamique du processus qui s'exprime en terme de cycles, de phases, d'itérations et de jalons.

Le processus unifié se déroule en quatre phases, incubation, élaboration, construction et transition. Chaque phase répète un nombre de fois une série d'itérations. Et chaque itération est composée de cinq activités : capture des besoins, analyse, conception, implémentation et test.

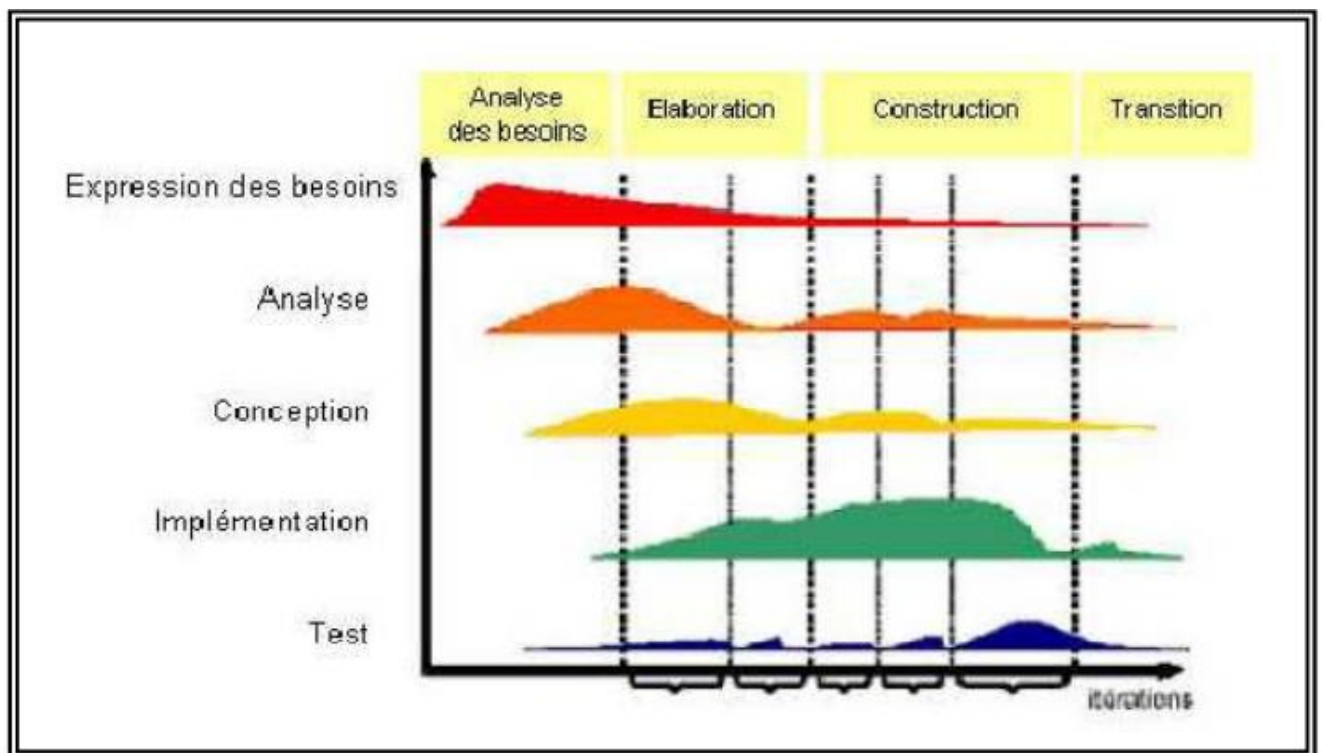


Figure IV.1: Cycle de vie du processus UP [19]

➤ **Incubation:**

C'est la première phase du processus unifié. Il s'agit de délimiter la portée du système, c'est-à-dire tracer ce qui doit figurer à l'intérieur du système et ce qui doit rester à l'extérieur, identifier les acteurs, lever les ambiguïtés sur les besoins et les exigences nécessaires dans cette phase. Il s'agit aussi d'établir une architecture candidate, c'est à dire que pour une première phase, on doit essayer de construire une architecture capable de

fonctionner. Dans cette phase, il faut identifier les risques critiques susceptibles de faire obstacles au bon déroulement du projet.

➤ **Elaboration :**

C'est la deuxième phase du processus. Après avoir compris le système, dégagé les fonctionnalités initiales, précisé les risques critiques, le travail à accomplir maintenant consiste à stabiliser l'architecture du système. Il s'agit alors de raffiner le modèle initial de cas d'utilisation, voire capturer de nouveaux besoins, analyser et concevoir la majorité des cas d'utilisation formulés, et si possible implémenter et tester les cas d'utilisation initiaux.

➤ **Construction :**

Dans cette phase, il faut essayer de capturer tous les besoins restants car il n'est pratiquement plus possible de le faire dans la prochaine phase. Ensuite, continuer l'analyse, la conception et surtout l'implémentation de tous les cas d'utilisation. A la fin de cette phase, les développeurs doivent fournir une version exécutable du système.

➤ **Transition :**

C'est la phase qui finalise le produit. Il s'agit au cours de cette phase de vérifier si le système offre véritablement les services exigés par les utilisateurs, détecter les défaillances, combler les manques dans la documentation du logiciel et adapter le produit à l'environnement (mise en place et installation).

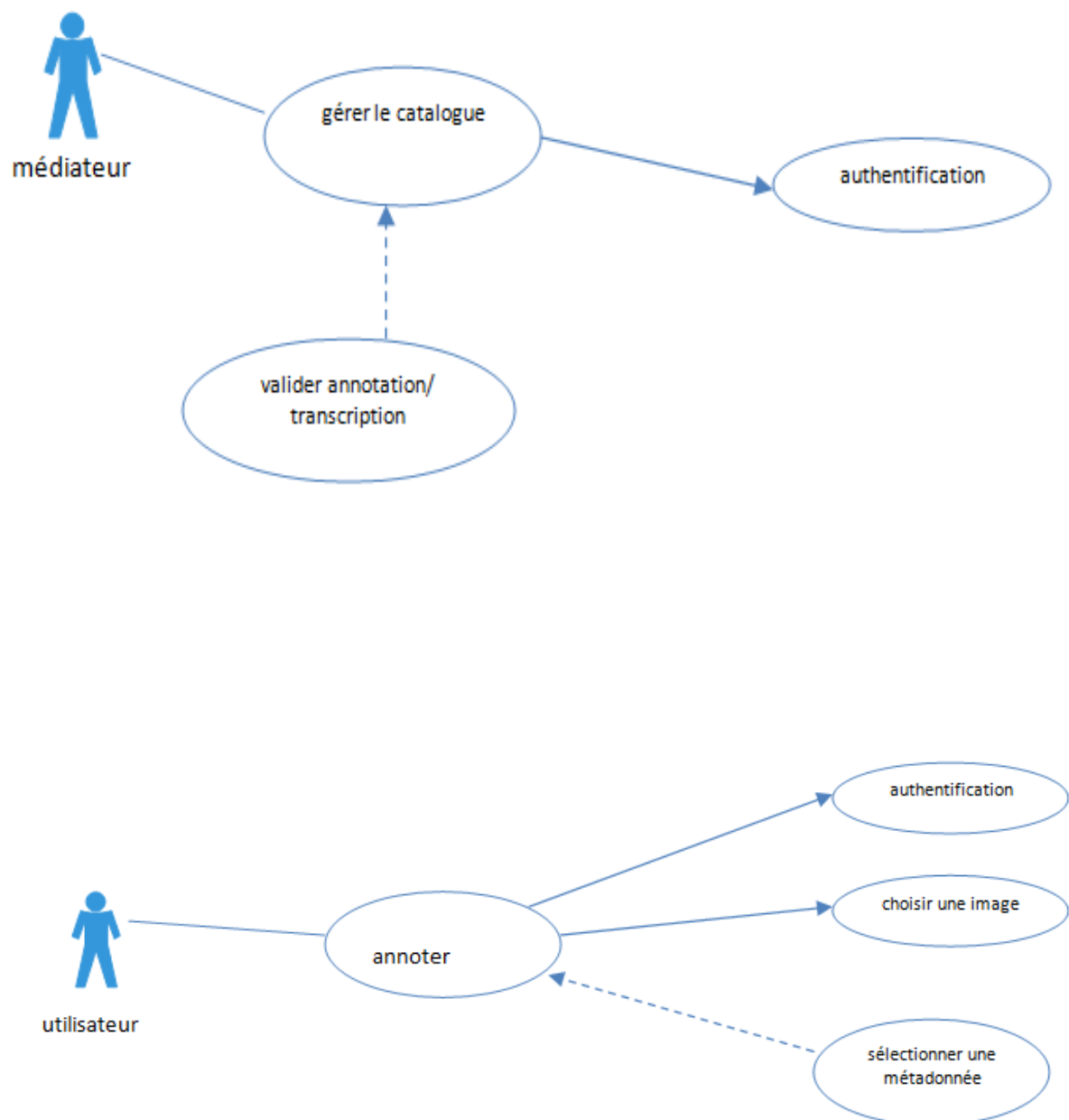
IV.2.4. Modélisation du Système :

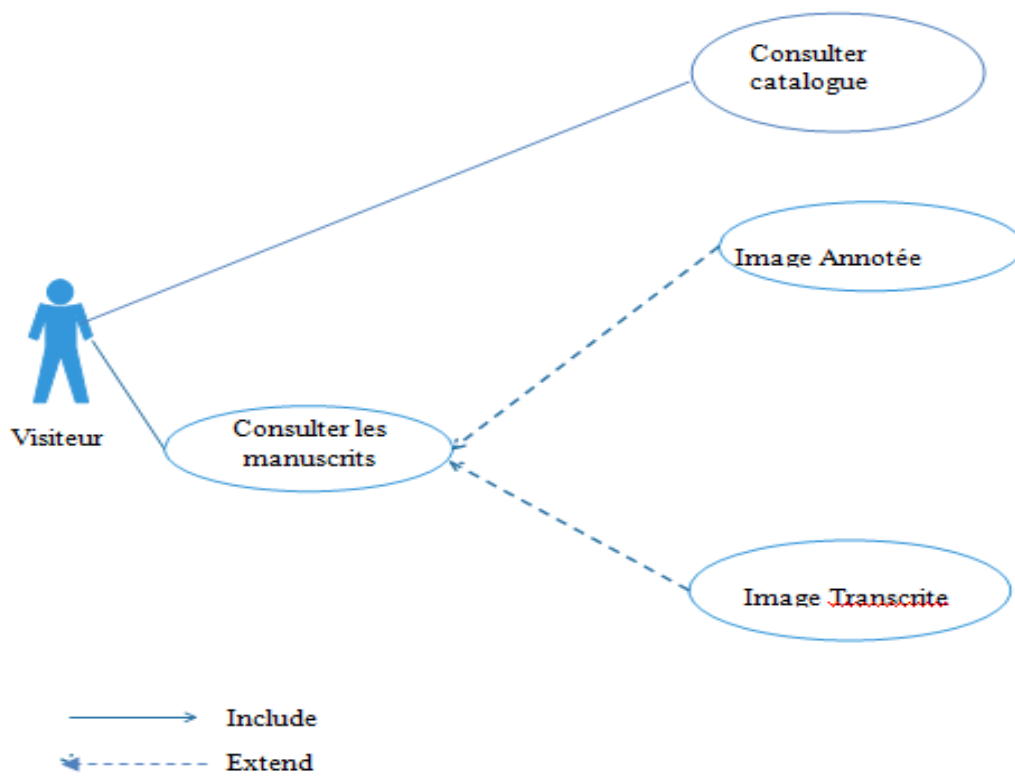
Pour mener efficacement notre modélisation, nous avons besoins de toutes les représentations du produit logiciel.

IV.2.4.1. Expression des besoins :

L'expression des besoins permet de définir les besoins principaux et leurs fonctions, évaluer les besoins fonctionnels (les cas d'utilisation), et saisir les besoins non fonctionnels.

Le modèle de cas d'utilisation présente le système du point de vue de l'utilisateur et représente les besoins du client sous forme de cas d'utilisation et d'acteur.





IV.2.4.2. Analyse

Le but principal de l'analyse est d'examiner les besoins et les exigences du client pour élaborer un schéma de conception de la solution. L'objectif de cette exploration étant de décrire comment se déroulent les actions entre les acteurs ou objets.

Nous allons montrer les interactions d'objets par le biais des diagrammes de séquences.

Le premier diagramme à présenter est celui permettant de décrire le processus d'annotation :

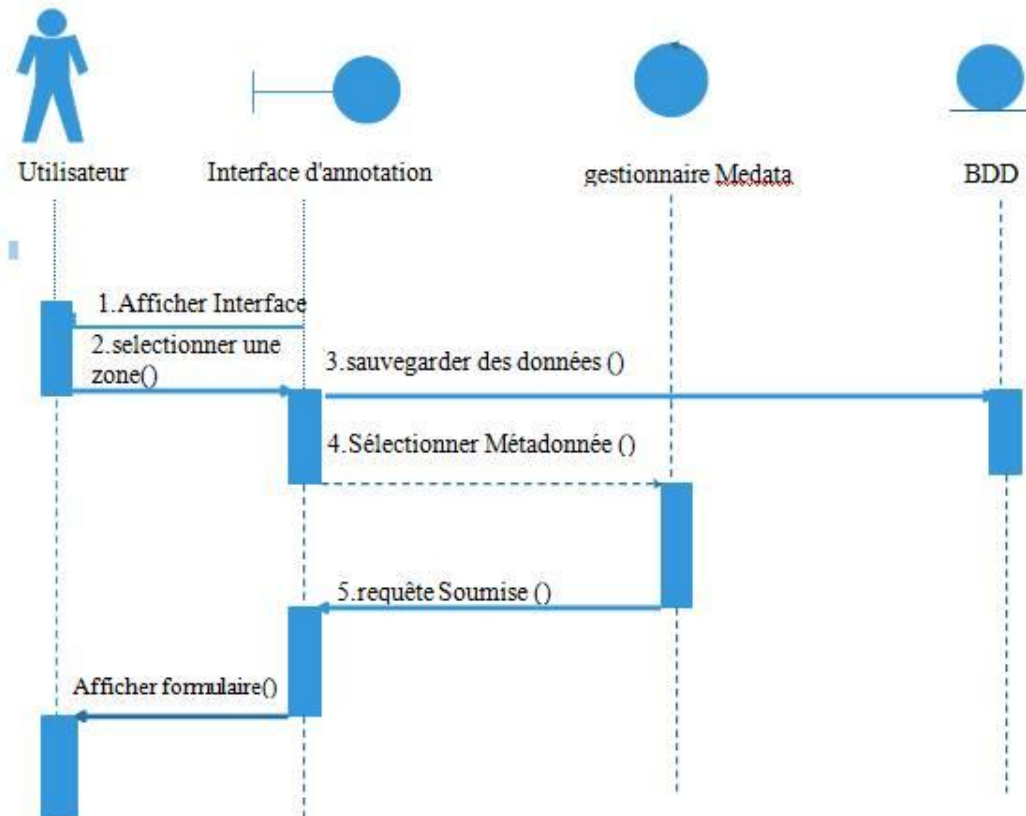


Figure IV.3: Diagramme de séquence annotation

Le second diagramme étant celui illustrant, l'acte de remplir le formulaire de la métadonnée éditeur et créer le fichier XML propre à cette métadonnée.

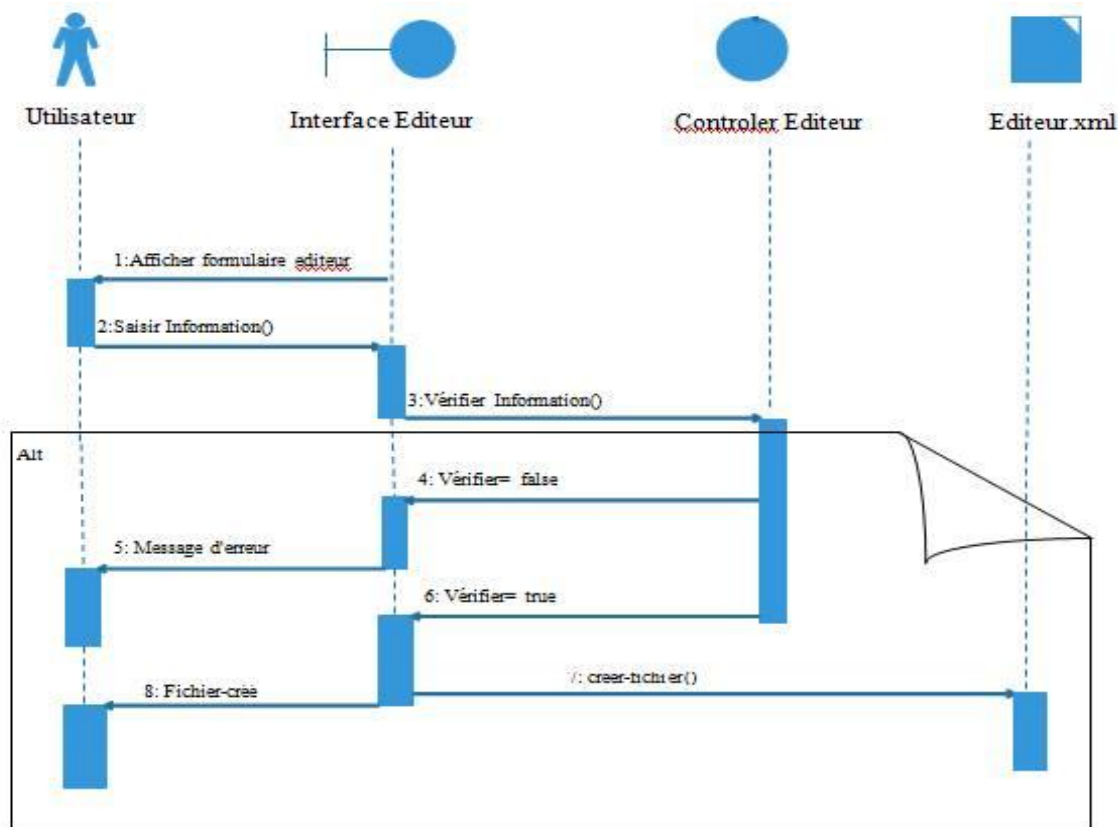


Figure IV.4: Diagramme de séquence de métadonnée éditeur

Le troisième et dernier diagramme à représenter, consiste en le diagramme de mise à jour du catalogue propre aux manuscrits arabes numérisés :

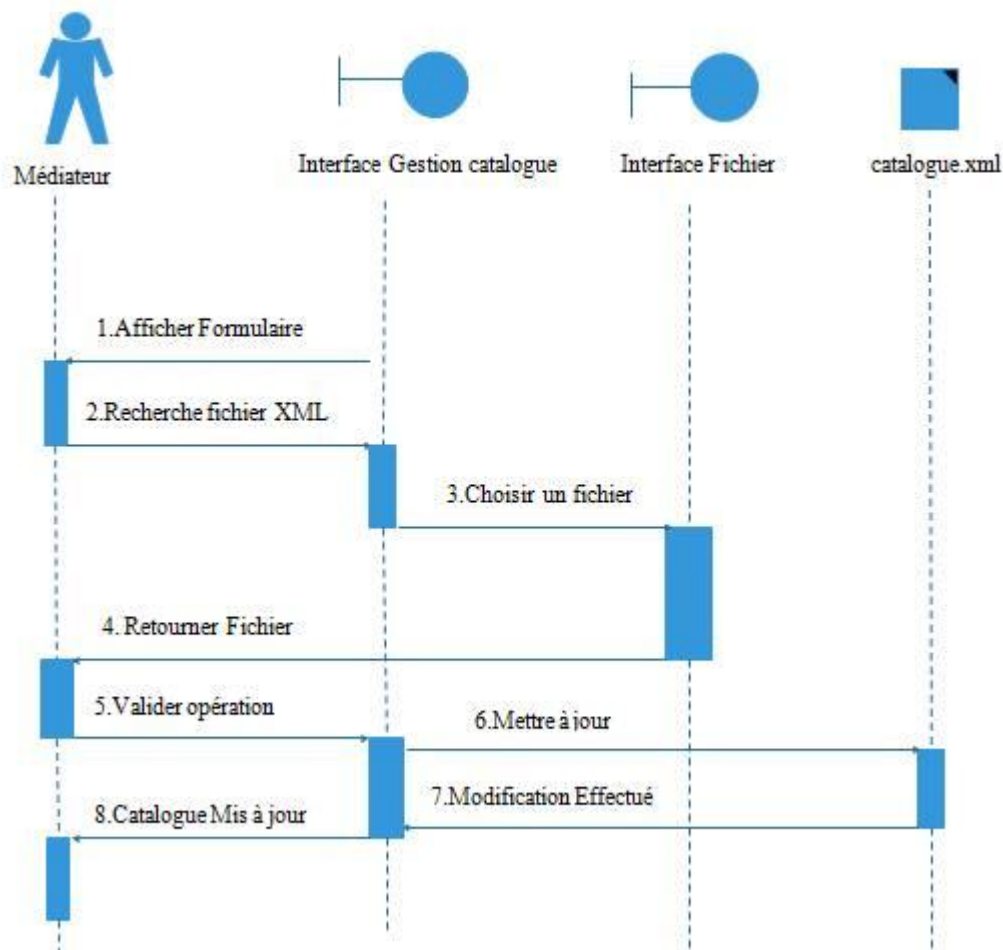


Figure IV.5:Diagramme de séquence mise à jour du catalogue

IV.2.4.3. Conception

La conception permet de déduire les contraintes liées au langage de programmation, à l'utilisation des composants et au système d'exploitation. Elle détermine les principales interfaces.

Elle constitue un point de départ à l'implémentation en lui créant une abstraction transparente après l'avoir décomposé en sous-systèmes. Nous pouvons illustrer ceci avec le diagramme de classe.



IV.2.4.4. Implémentation

L'implémentation est le résultat de la conception, il s'agit d'écrire des lignes de programme pour mettre en œuvre les composants au sein du système. Les objectifs principaux de l'implémentation sont de produire les classes et les sous-systèmes sous forme de codes sources, de définir les relations entre les composants logiciels et matériels du système, d'une part, et la distribution physique du traitement, d'autre part.

Nous avons modélisé ceci à l'aide du diagramme de déploiement qui présente la disposition physique des nœuds dans un système réparti

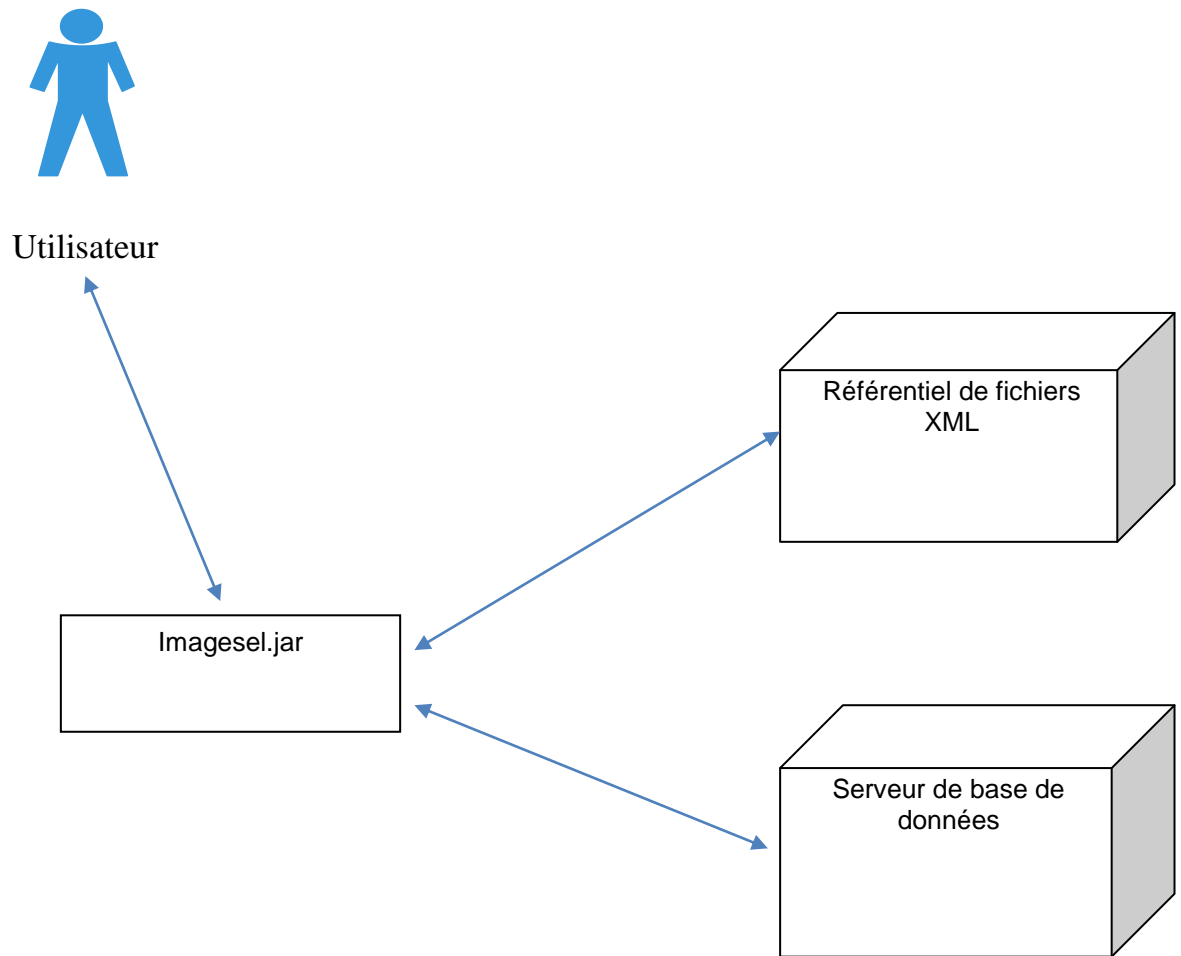


Figure IV.7 : Diagramme de déploiement

IV.2.4.5. Test

Les tests permettent de vérifier des résultats de l'implémentation en testant la construction. Pour mener à bien ces tests, il faut les planifier pour chaque itération, les implémenter en créant des cas de tests, effectuer ces tests et prendre en compte le résultat de chacun.

IV.3. Structure des documents annotés:

Comme nous l'avons vu précédemment, notre système permet la génération de fichiers XML.

XML est un **langage de description** qui permet de décrire et structurer un ensemble de données selon un jeu de règles et des contraintes définies.

Les fichiers XML créés à partir de l'annotation des images doivent bien respecter une structure de données. Pour cela nous avons utilisé le système de définition de type de document DTD qui vérifie la validité de nos documents, et qui répond amplement à nos besoins.

Avant de foncer tête baissée dans cette partie, il est indispensable de revenir sur quelques termes qui seront importants pour la suite de ce chapitre.

IV.3. 1. DTD est un document de définition de données ? [\[29\]](#)

Une définition est un ensemble de règles que l'on impose au document. Ces règles permettent de décrire la façon dont le document XML doit être construit. Elles peuvent être de natures différentes. Par exemple, ces règles peuvent imposer la présence d'un attribut ou d'une balise, imposer l'ordre d'apparition des balises dans le document ou encore, imposer le type d'une donnée (nombre entier, chaîne de caractères, etc.).

IV.3.2. C'est quoi un document valide ? [\[29\]](#)

Un document valide est un document bien formé conforme à une définition. Cela signifie que le document XML respecte toutes les règles qui lui sont imposées.

IV.3.3. Avantages de la DTD :

- Les DTD sont très utilisées pour leur simplicité et efficacité dans la description des langages de description documentaire.
- Elles viennent compléter les données contenues dans un document XML pour qu'elles soient organisées de manière normalisée et puissent être partagées suivant un modèle commun.
- Elles donnent une organisation logique au document XML.

IV.3.4. Documents DTD réalisés :

Nous allons présenter maintenant la DTD que nous avons réalisé pour nos fichiers XML.

a. La DTD pour le fichier XML cachet :

```
<!DOCTYPE TEI [  
<!ELEMENT TEI (seal)>  
<!ATTLIST TEI xmlns CDATA #REQUIRED>  
<!ELEMENT seal (P)>  
<!ELEMENT P (name)>  
<!ELEMENT P (#PCDATA)>  
<!ATTLIST P id (red|black|brown|green|gold|silver) #REQUIRED>  
<!ELEMENT name (#PCDATA)>  

```

b. La DTD pour le fichier XML correction :

```
<!DOCTYPE TEI [  
<!ELEMENT TEI (choice)>  
<!ATTLIST TEI xmlns CDATA #REQUIRED>  
<!ELEMENT choice (sic, corr)>  
<!ELEMENT sic (#PCDATA)>  
<!ELEMENT corr (#PCDATA)>  

```

c. La DTD pour le fichier XML ecouteurs :

```
<!DOCTYPE TEI [  
<!ELEMENT TEI (personGrp)>  
<!ATTLIST TEI xmlns CDATA #REQUIRED>  
<!ELEMENT personGrp (persname)>  
<!ELEMENT persname (#PCDATA)>  

```

d. La DTD pour le fichier XML editeur :

```
<!DOCTYPE TEI [  
<!ELEMENT TEI (editionStmt)>  
<!ATTLIST TEI xmlns CDATA #REQUIRED>  
<!ELEMENT editionStmt (editor, edition)>  
<!ELEMENT editor (#PCDATA)>  
<!ELEMENT edition (placename, date)>  
<!ELEMENT placename (#PCDATA)>  
<!ELEMENT date (#PCDATA)>  

```

e. La DTD pour le fichier XML Head :

```
<!DOCTYPE TEI [  
<!ELEMENT TEI (head)>  
<!ATTLIST TEI xmlns CDATA #REQUIRED>
```

```
<!ELEMENT head (#PCDATA)>
]
>
```

f. La DTD pour le fichier XML histoire :

```
<! DOCTYPE TEI [
<!ELEMENT TEI (history)>
<!ATTLIST TEI xmlns CDATA #REQUIRED>
<!ELEMENT history (origin, provenance, aquisition)>
<!ELEMENT origin (date, origPlace)>
<!ELEMENT date (#PCDATA)>
<!ATTLIST date calendar (hegire|gregorien)>
<!ELEMENT origPlace (#PCDATA)>
<!ELEMENT provenance (sp)>
<!ELEMENT provenance (#PCDATA)>
<!ELEMENT sp (speaker)>
<!ELEMENT speaker (#PCDATA)>
<!ELEMENT aquisition (#PCDATA)>
]>
```

g. La DTD pour le fichier XML incipit/explicit :

```
<! DOCTYPE TEI [
<!ELEMENT TEI (incipit, explicit)>
<!ATTLIST TEI xmlns CDATA #REQUIRED>
<!ELEMENT incipit (#PCDATA)>
<!ELEMENT explicit (#PCDATA)>
]>
```

h. La DTD pour le fichier XML Ms Identifier :

```
<! DOCTYPE TEI [
<!ELEMENT TEI (msIdentifier)>
<!ATTLIST TEI xmlns CDATA #REQUIRED>
<!ELEMENT msIdentifier (country, sttlement, institution, repository, collection,
idno)>
<!ELEMENT country (#PCDATA)>
<!ELEMENT settlement (#PCDATA)>
<!ELEMENT institution (#PCDATA)>
<!ELEMENT repository (#PCDATA)>
<!ELEMENT collection (#PCDATA)>
<!ELEMENT idno (#PCDATA)>
]>
```

i. La DTD pour le fichier XML Rubrique :

```
<!DOCTYPE TEI [  
  <!ELEMENT TEI (rubric)>  
  <!ATTLIST TEI xmlns CDATA #REQUIRED>  
  <!ELEMENT rubric (#PCDATA)>  

```

j. La DTD pour le fichier XML sommaire :

```
<!DOCTYPE TEI [  
  <!ELEMENT TEI (summury)>  
  <!ATTLIST TEI xmlns CDATA #REQUIRED>  
  <!ELEMENT summury (#PCDATA)>  

```

k. La DTD pour le fichier XML transcription :

```
<!DOCTYPE TEI [  
  <!ELEMENT TEI (teiHeader, text)>  
  <!ATTLIST TEI xmlns CDATA #REQUIRED>  
  <!ELEMENT teiHeader (EMPTY)>  
  <!ELEMENT text (body)>  
  <!ELEMENT body (line)>  
  <!ELEMENT line (author*, country*, sttlement*, institution*, repository*,  
  colletion*, idno*, origPlace*, provenance*, speaker*, aquition*, publisher*,  
  pubPlace*, title*, head*, editor*, corr*, sic*, perspnGrp*, rubric*, summury*,  
  fw*, incipit*, explicit*, bibl*, note* >)>  
  <!ELEMENT author (persName)>  
  <!ELEMENT persName (#PCDATA)>  
  <!ELEMENT country (#PCDATA)>  
  <!ELEMENT sttlement (#PCDATA)>  
  <!ELEMENT institution (#PCDATA)>  
  <!ELEMENT repository (#PCDATA)>  
  <!ELEMENT collection (#PCDATA)>  
  <!ELEMENT idno (#PCDATA)>  
  <!ELEMENT origPlace (#PCDATA)>  
  <!ELEMENT provenance (#PCDATA)>  
  <!ELEMENT speaker (#PCDATA)>  
  <!ELEMENT aquition (#PCDATA)>  
  <!ELEMENT publisher (#PCDATA)>  
  <!ELEMENT pubPlace (#PCDATA)>  
  <!ELEMENT title (#PCDATA)>
```

```
<!ELEMENT head (#PCDATA)>
<!ELEMENT editor (#PCDATA)>
<!ELEMENT corr (#PCDATA)>
<!ELEMENT sic (#PCDATA)>
<!ELEMENT personGrp (personName)>
<!ELEMENT persName (#PCDATA)>
<!ELEMENT rubric (#PCDATA)>
<!ELEMENT summury (#PCDATA)>
<!ELEMENT fw (#PCDATA)>
<!ATTLEST fw type (pageNum|header|footer)>
<!ELEMENT incipit (#PCDATA)>
<!ELEMENT explicit (#PCDATA)>
<!ELEMENT bibl (#PCDATA)>
<!ELEMENT note (#PCDATA)>
]>
```

La DTD nous a servit de support pour s'assurer de la validité des documents annotés. Ce qui nous facilite l'extraction automatique de métadonnées déjà traitée dans le chapitre précédant.

IV.4. Le principe d'extraction de métadonnées :

L'objectif final de notre travail est de compléter la notice bibliographique, qui demeure le plus souvent dans un état fragmentaire au vue de l'indisponibilité de l'information. Cependant il est indispensable d'évaluer la nécessité d'intégrer le contenu du document annotation ou celui du document transcription dans la notice, en faisant recours au concept de similarité structurelle.

Une fois l'évaluation faite, nous procédons à l'extraction de métadonnées à partir des documents annotés ou ceux transcrits afin de mettre à jour le catalogue.

L'extraction diffère suivant ces trois cas de figure:

Cas 1 : Lorsque l'élément annoté ou transcrit n'existe pas dans la notice bibliographique

Exemple :

Document annoté :

```
<author>
  <persName xml:lang="ara">
    <foreName></foreName>
    <surName></surName>
    <addName type="kunya"></addName>
    <NickName type="Ism chohra"> </NickName>
    <addName type="nisbah"></addName>
  </persName>
  <birth>
    <date calendar="Hegire"></date>
    <placeName> </placeName>
  </birth>
  <death>
    <date calendar="Hegire"> </date>
    <placeName></placeName>
  </death>
</author>
```

Notice bibliographique :

```
Elément 1 # de l'élément auteur
Elément 2 # de l'élément auteur
Elément 3 # de l'élément auteur
...
Elément n # de l'élément auteur
```

- Dans ce cas l'élément auteur doit être copié en entier dans la notice.

Cas 2 : Une partie de l'élément annoté ou transcrit n'existe pas dans la notice bibliographique

Exemple :Élément annoté :

```

<author>
  <persName xml:lang="ara">
    <foreName></foreName>
    <surName></surName>
    <addName type="kunya"></addName>
    <NickName type="Ism chohra"> </NickName>
    <addName type="nisbah"></addName>
  </persName>
  <birth>
    <date calendar="Hegire"></date>
    <placeName> </placeName>
  </birth>
  <death>
    <date calendar="Hegire"> </date>
    <placeName></placeName>
  </death>
</author>

```

- Dans ce cas, la partie manquante de l'élément auteur est "persName" pour cela, seulement cette partie sera copiée dans la notice bibliographique.

Cas 3 : Le contenu de l'élément annoté ou transcrit est différent de celui de la notice bibliographique.

Exemple:Document annoté :

```

<author>
  <persName xml:lang="ara">
    <foreName>صادق</foreName>
    <surName>محمد</surName>
    <addName type="kunya">گتکونی چشتی</addName>
    <NickName type="Ism chohra"> </NickName>
    <addName type="nisbah">بن فتح الله</addName>
  </persName>
  <birth>

```

Notice bibliographique :

```

Elément 1 # de l'élément auteur
Elément 2 # de l'élément auteur
Elément 3 # de l'élément auteur
...
<author>
  <birth>
    <date calendar="Hegire"></date>
    <placeName> </placeName>
  </birth>
  <death>
    <date calendar="Hegire"> </date>
    <placeName> </placeName>
  </death>
</author>
Elément n # de l'élément auteur

```

Notice bibliographique :

```

Elément 1 # de l'élément auteur
Elément 2 # de l'élément auteur
Elément 3 # de l'élément auteur
...
<author>
  <persName xml:lang="ara">
    <foreName></foreName>
  <surName></surName>
  <addName type="kunya"/>

```

```

<date calendar="Hegire"></date>
<placeName> </placeName>
</birth>
<death>
  <date calendar="Hegire"> </date>
  <placeName></placeName>
</death>
</author>

```

```

<NickName type="Ism chohra"/>
<addName type="nisbah"/>
</persName>
<birth>
  <date calendar="Hegire"></date>
  <placeName> </placeName>
</birth>
<death>
  <date calendar="Hegire"> </date>
  <placeName></placeName>
</death>
</author>

```

Élément n # de l'élément auteur

Dans ce dernier cas, nous remarquons que le contenu de l'élément auteur est différent dans les deux documents, pour cela, nous procédons au remplacement du contenu de l'élément auteur de la notice par celui du document annoté ou transcrit.

IV.5. Le concept de similarité structurelle :

Généralement pour comparer deux termes, on utilise un dictionnaire. Mais dans un contexte hiérarchique, il faut trouver une mesure de ressemblance qui considère à la fois, l'information structurelle et l'information sémantique. Ainsi, la similarité structurelle de deux nœuds va dépendre non seulement de leur similarité ontologique (sémantique) mais aussi de leur contexte hiérarchique.

Sur la base de la similitude entre chaque paire d'éléments, nous avons utilisé l'équation générale de calcul de la similarité entre deux arbres XML.

La Similarité entre deux arbre T_1 et T_2 est calculée par la formule suivante:

$$\text{Similarité}(T_1, T_2) = \frac{\sum_{i=1}^n \sum_{j=1}^m \text{sim}(e_{1i}, e_{2j})}{\text{Max}(|T_1|, |T_2|)}$$

$\text{sim}(e_{1i}, e_{2j})$ représente la similarité des nœuds e_{1i} et e_{2j} . Les nœuds e_{1i} et e_{2j} appartiennent respectivement aux arbres T_1 et T_2 . $|T_1|$ et $|T_2|$ sont les tailles (nombre de nœuds) respectives n et m des arbres T_1 et T_2 . La division par $\text{Max}(|T_1|, |T_2|)$ permet de normaliser le résultat de la sommation.

- la similarité entre deux nœuds ne peut prendre que deux valeurs :
 - Zéro (0), si le nœud n'existe pas dans la notice bibliographique (absence de la métadonnée).
 - Un (1), si le nœud existe (cas d'une mise à jour de la métadonnée).

Cette formule est sans doute une solution efficace pour la comparaison de deux arbres XML, pour faire une mise à jour complète du catalogue. Reste que nous connaissons très bien la structure de notre catalogue, pour cela nous avons pensé à optimiser cette solution.

IV.5.1. Optimisation du calcul de similarité :

Pour diminuer le nombre de contrôles inutiles effectués lors de la comparaison avec la formule de calcul de similarité précédemment indiquée, nous avons pensé à rendre cette dernière plus optimale. Sachant qu'un arbre est structuré en niveaux hiérarchiques,

le principe est de comparer chaque nœud d'un niveau donné aux nœuds du même niveau, ainsi la formule précédente devient :

$$sim(T_1, T_2) = \frac{\sum_{i,j=1}^n sim(e_{1i}, e_{2j})}{Max(|T_1|, |T_2|)}$$

Les trois cas cités précédemment sont repérés à la suite de l'application de l'algorithme ci-après

Algorithme :

Entrée :

- Document-Annotation/transcription (T1)
- Notice bibliographique (T2)

Sortie :

- Notice bibliographique mise à jour.

Calcul similarité :

{

 Calculer la similarité Sim (T₁, T₂);

 Si Sim (T₁, T₂) = 0

 Alors

 Intégrer le contenu du document-annotation/transcription (T₁) dans la notice (T₂)

 Sinon

 Remplacer le contenu de la métadonnée de la notice (T₂) par le contenu du document-annotation/transcription.

 Finsi

}

IV.6. Conclusion

Le langage UML nous apporte une aide à toutes les étapes d'un projet. Il nous offre ainsi de nombreux avantages pour l'analyse et la conception d'un système.

Le couple UML et le processus unifié propose une approche pour conduire la réalisation de systèmes orientés objet.

Dans la seconde partie de ce chapitre, nous avons abordé l'une des technologies permettant de définir une structure stricte aux documents XML, qui est le Document Type Definition (DTD), comme nous avons évoqué une nouvelle notion assurant un calcul précis de degré de similitude entre deux arbres XML.

Le prochain chapitre sera consacré à la réalisation de notre système, en présentant notre base de données les différentes fonctionnalités de notre application à travers ses diverses interfaces.

V.1.Introduction

L'implémentation est la phase la plus importante après la conception, elle consiste à transformer le modèle conceptuel établi précédemment en des composants logiciels formant notre système.

Dans cette partie, nous allons présenter les différentes phases de la réalisation de notre application puis expliquer son fonctionnement en présentant quelques interfaces illustratives, les structures de données utilisées, ainsi que quelques programmes réalisés.

V.2.Fonctionnement général de l'application :

Le but principal de notre système, est de mettre à jour le catalogue des manuscrits arabes numérisés, à partir de l'extraction de métadonnées des documents annotés ou transcrits. Sur ce, le schéma ci-après donne une idée globale sur son fonctionnement:

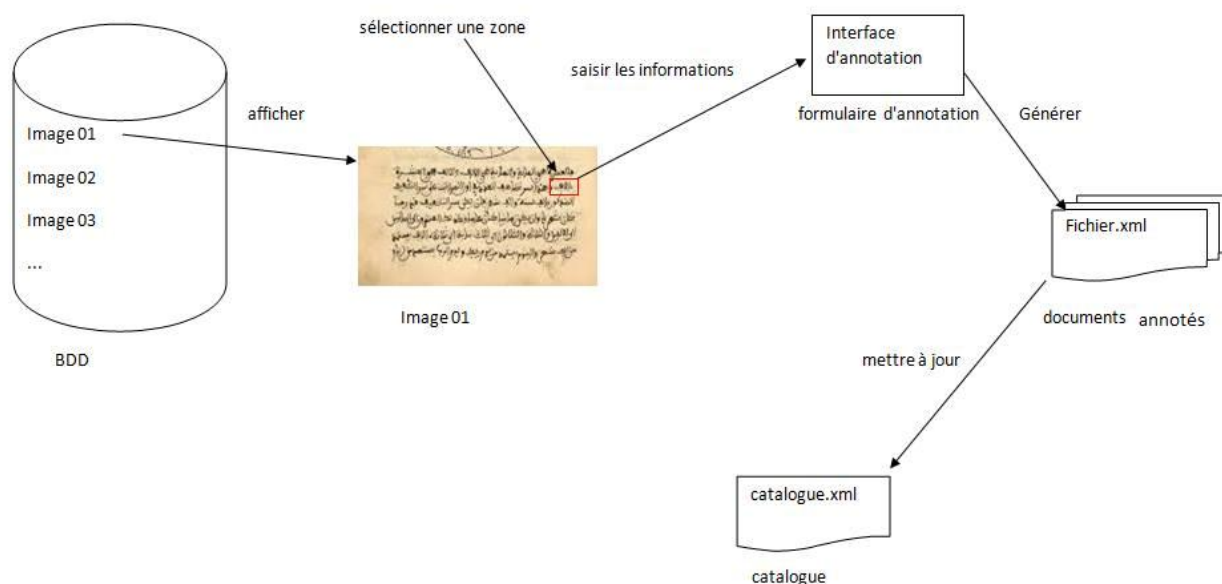


Figure V.1 : Architecture générale du système

V.3. Dictionnaire de données

Codification	Désignation	Type	Taille	Observation
Id	Identificateur	INTEGER	11	
id_image	Identificateur de l'image	INTEGER	11	

Nom	Nom de l'auteur	VARCHAR	500	
Prenom	Prénom de l'auteur	VARCHAR	500	
Kunya	Kunya de l'auteur	VARCHAR	500	
date_n	Date de naissance de l'auteur	DATE	8	jj/mm/aaaa
lieu_n	Lieu de naissance de l'auteur	VARCHAR	500	
date_d	Date de décès de l'auteur	DATE	8	jj/mm/aaaa
lieu_d	Lieu de décès de l'auteur	VARCHAR	500	
nom_cel	Nom de célébrité	VARCHAR	500	
Nisbah	Le rapport de l'auteur	VARCHAR	500	
Couleur	Couleur du cachet	VARCHAR	255	
Forme	Forme du cachet	VARCHAR	255	
proprietaire	Propriétaire du	VARCHAR	500	

	cachet			
Titre	Titre du chapitre	VARCHAR	500	
startX	Abscisse de départ de sélection	INTEGER	11	
startY	Ordonnée de départ de sélection	INTEGER	11	
endX	Abscisse d'arrivée de sélection	INTEGER	11	
endY	Ordonnée d'arrivée de sélection	INTEGER	11	
type_categorie	Type de catégorie	VARCHAR	60	
Id_categorie	Identificateur de catégorie	INTEGER	11	
type_ano_tran	Annotation ou Transcription	INTEGER	11	
texte_org	Texte originale	VARCHAR	500	
Corr	Correction	VARCHAR	500	
Debut	Début du manuscrit	VARCHAR	500	

Fin	Fin du manuscrit	VARCHAR	500	
Ecout	Ecouteur	VARCHAR	500	
nom_e	Nom de l'éditeur	VARCHAR	500	
lieu_e	Lieu de naissance de l'éditeur	VARCHAR	500	
date_e	Date de naissance de l'éditeur	DATE	8	jj/mm/aaaa
lieu_h	Lieu du manuscrit	VARCHAR	500	
date_h	Date du manuscrit			
aquisition	Acquisition du manuscrit	VARCHAR	500	
provenance	Provenance du manuscrit	VARCHAR	500	
Lissane	Porte parole	VARCHAR	500	
nom_image	Nom de l'image	VARCHAR	250	
Image	image	LONGBLOB		

Type	Type	VARCHAR	250	
Texte	Contenu de l'information	VARCHAR	500	
Place	Emplacement de l'information	VARCHAR	250	
Nom_u	Nom de l'utilisateur	VARCHAR	250	
Prenom_u	Prénom de l'utilisateur	VARCHAR	250	
Login	Nom d'accès au compte utilisateur	VARCHAR	250	
m_passe	Mot de passe de l'utilisateur	VARCHAR	20	
principale	Langue principale du manuscrit	VARCHAR	500	
secondaire	Langue secondaire du manuscrit	VARCHAR	500	
utilisateur	Nom d'accès au compte médiateur	VARCHAR	250	

password	Mot de passe de médiateur	VARCHAR	20	
Pays	Pays de détention	VARCHAR	500	
Ville	Ville de détention	VARCHAR	500	
institution	Institution de conservation	VARCHAR	500	
Entrepot	Entrepôt de conservation	VARCHAR	500	
collection	Collection du manuscrit	VARCHAR	500	
Ident	Identificateur du manuscrit	VARCHAR	50	
publicateur	Nom de publicateur	VARCHAR	500	
lieu_p	Lieu de publication	VARCHAR	500	
date_p	Date de publication	DATE	8	jj/mm/aaaa
Ref	Référence du manuscrit	VARCHAR	500	

Rubric	Rubrique	VARCHAR	500	
Sommaire	Sommaire du manuscrit	VARCHAR	500	
Text	Contenu textuel du manuscrit	VARCHAR	500	
Tilte_t	Titre du manuscrit	VARCHAR	500	
Type_t	Type du titre	VARCHAR	50	
Place_t	Emplacement du titre	VARCHAR	50	
id_auteur	Identificateur de l'auteur	INTEGER	11	
id_cachet	Identificateur du cachet	INTEGER	11	
id_chapitre	Identificateur du chapitre	INTEGER	11	
id_correction	Identificateur d'une correction	INTEGER	11	
id_debutfin	Identificateur de début ou de fn du manuscrit	INTEGER	11	
id_ecouteur	Identificateur d'écouteur	INTEGER	11	
id_editeur	Identificateur	INTEGER	11	

	d'éditeur			
id_information	Identificateur d'information	INTEGER	11	
id_langue	Identificateur de langue	INTEGER	11	
id_origine	Identificateur d'origine du manuscrit	INTEGER	11	
id_publication	Identificateur de publication	INTEGER	11	
id_reference	Identificateur de référence	INTEGER	11	
id_rubrique	Identificateur de rubrique	INTEGER	11	
id_sommaire	Identificateur de sommaire	INTEGER	11	
id_texte	Identificateur de texte	INTEGER	11	
id_titre	Identificateur de titre	INTEGER	11	
id_histoire	Identificateur d'histoire	INTEGER	11	

V.4.Connexion a la base de données :

Nous avons utilisé la base de donnée MySql, pour cela nous avons eu recours à la librairie "com.mysql.jdbc_5.1.5.jar". Nous avons mis le code de connexion dans une classe "connection.Java" permettant d'interagir avec la base de données.

➤ Classe Connection.Java

```
public class connection {
public Connection getConnection(){
Connection c=null;
try{
```

```
Class.forName("com.mysql.jdbc.Driver");
String
url="jdbc:mysql://localhost:3306/database?useUnicode=yes&characterEncoding=UTF-
8";

    c=DriverManager.getConnection(url,"root","");

}
catch (Exception ex){
JOptionPane.showMessageDialog(null,"erreur de connection:"+ex.toString());

}
return c;

}
}
```

Les données qui doivent être insérées à la base de données sont multilingues, ce qui nécessite un réglage au niveau des paramètres de la base de données. Ce réglage consiste en la modification de l'ancienne URL :

`jdbc:mysql://localhost:3306/database?zeroDateTimeBehavior=convertToNull`

Par la ligne suivante :

`jdbc:mysql://localhost:3306/database?useUnicode=yes&characterEncoding=UTF-8\`

Cela permet de modifier le type d'encodage pour insérer les données en langue arabes.

V.5. Présentation des interfaces graphique et leurs fonctionnements :

La conception des interfaces de l'application est une étape très importante puisque toutes les interactions avec le cœur de l'application passent à travers ces interfaces, nous devons alors guider l'utilisateur avec les messages d'erreurs et de notifications si besoin, ainsi présenter un système complet.

Dans la conception des interfaces de notre application nous avons respecté un ensemble des choix ergonomiques comme la lisibilité, la compréhensibilité, etc. Dans ce qui suit, une présentation des captures écrans des plus importantes interfaces de l'application.

a. Interface d'accueil

Cette première capture présente l'interface d'accueil. Elle contient les différentes fonctions du système, on y trouve trois espaces : espace visiteur, espace utilisateur, espace médiateur, ainsi qu'un hyperlien permettant d'ajouter un nouveau utilisateur.

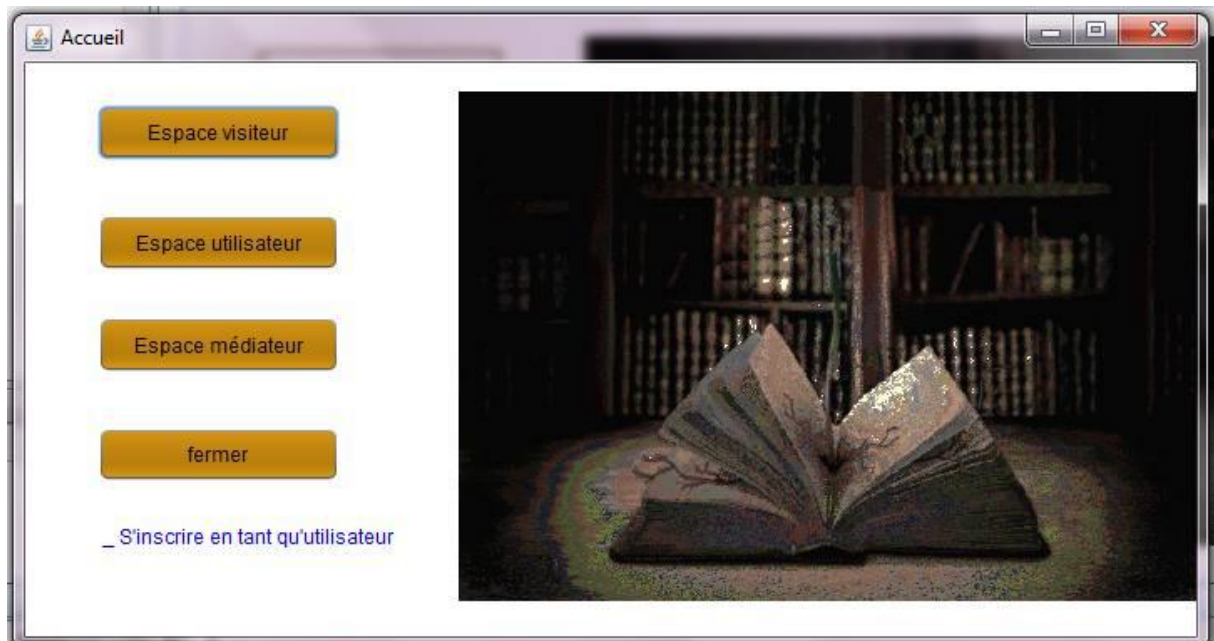


Figure V.2. Page d'accueil

b. Interface d'authentification

Elle est présentée par deux interfaces, l'une d'elles est destinée à l'utilisateur et l'autre au médiateur. A travers ces deux fenêtres l'utilisateur et le médiateur s'authentifient pour utiliser l'application. Cette étape met en valeur l'aspect sécurité.



Figure V.3. Interface d'authentification médiateur**Figure V.4. Interface d'authentification utilisateur**

L'authentification repose sur la vérification des informations saisies à celles dans les tables de la base de données. Les tables contenant les informations relatives à l'authentification sont la table médiateur pour l'authentification du médiateur et la table inscription en ce qui concerne l'utilisateur.

Table mediateur :

Nom de champ	Type	Null	Index
utilisateur	VARCHAR(250)	NON	Clé primaire
Password	VARCHAR(250)	NON	

Table inscription :

Nom de champ	Type	Null	Index
nom_u	VARCHAR(250)	NON	
prenom_u	VARCHAR(250)	NON	

Login	VARCHAR(250)	NON	Clé primaire
m_passe	VARCHAR(250)	NON	

c. Espace utilisateur

Elle permet à l'utilisateur d'effectuer l'annotation ou la transcription des images des manuscrits arabes numérisés, ces images sont récupérées depuis la table images de notre base de données.



Figure V.5. Interface d'espace utilisateur

Table image

Nom de champ	Type	Null	Index

id_image	INTEGER(11)	NON	Clé primaire
nom_image	VARCHAR (250)	NON	
Image	LONGBLOB	NON	

d. Interface annotation

Elle permet à l'annotateur d'effectuer son annotation, en sélectionnant une zone sur l'image affichée, ensuite choisir une métadonnée cela permettra d'afficher l'interface liée à la métadonnée sélectionnée, de remplir les champs de l'interface et d'enregistrer les coordonnées de la zone sélectionnée et les champs remplis respectivement dans les table coordonnées et l'une des tables métadonnée(auteur, titre, écouleur, éditeur, ...etc.) dans la base de donnée suite à une validation, cette validation permettra non seulement la sauvegarde dans la base de donnée mais aussi la création du fichier XML propre à la métadonnée précédemment sélectionnée.

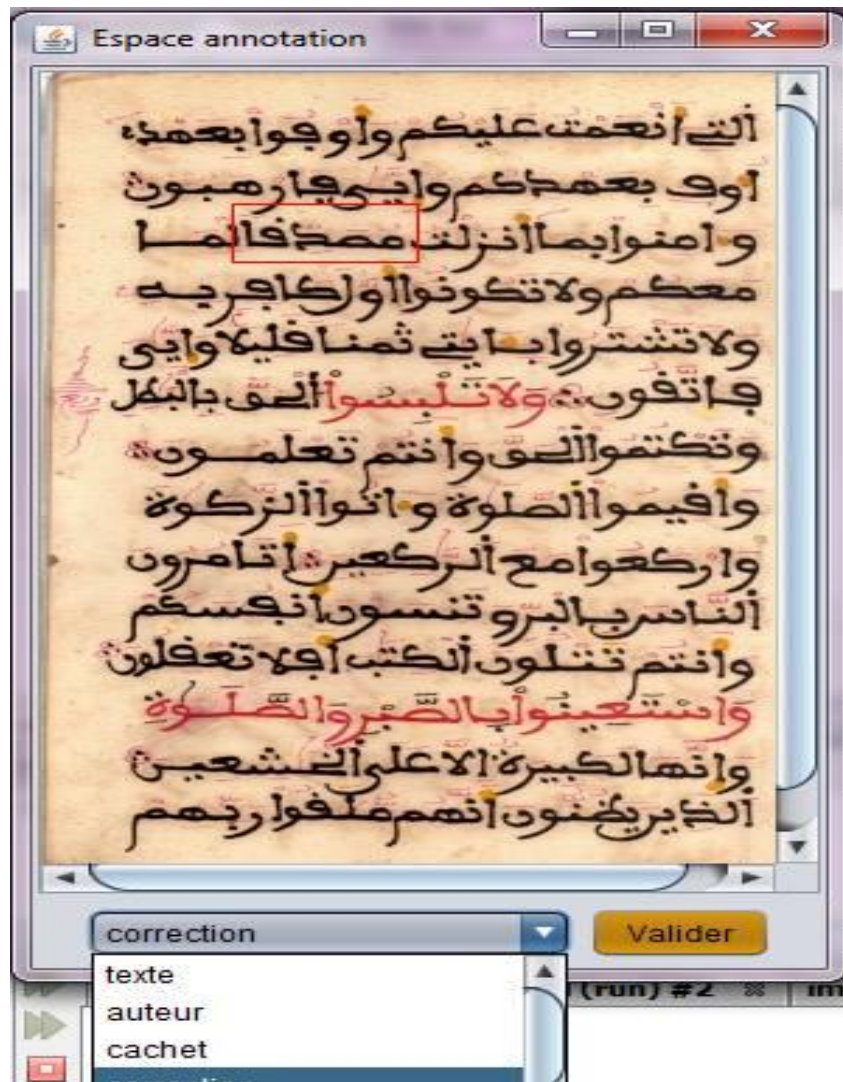


Figure V.6. Interface annotation

e. Interface de la métadonnée correction

Elle permet à l'annotateur de saisir les informations relatives à la métadonnée correction, de générer le fichier « correction.xml », et de sauvegarder les coordonnées de la zone annotée ainsi que les informations saisies dans la base de données suite à une validation.

Figure V.7. Interface de la métadonnée correction

➤ Exemple de fichier xml générer « correction.xml »

```
<?xml version="1.0" encoding="UTF-8"?>
- <tei xmlns="http://www.tei-c.org/ns/1.0">
  - <choice>
    <sic> مصدقاً </sic>
    <corr> مصدقاً </corr>
  </choice>
</tei>
```

Pour la création d'un document XML, nous avons utilisé l'API DOM, c'est un parseur XML, qui permet de lire un document XML et d'en extraire différentes informations.

Le code réalisé pour la création du fichier « correction.xml » et présenter ci-après :

```
//Étape 1 : récupération d'une instance de la classe "DocumentBuilderFactory"
final DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
try {
    // Etape 2 : création d'un parseur
    final DocumentBuilder builder = factory.newDocumentBuilder();
    // Etape 3 : création d'un Document
    final Document document= builder.newDocument();
```

// Etape 4 : création de l'Element racine

```
final Element racine = document.createElement("tei");
racine.setAttribute("xmlns", "http://www.tei-c.org/ns/1.0");
document.appendChild(racine);
```

// Etape 5 : création des elements fils

```
final Element choice = document.createElement("choice");
racine.appendChild(choice);
final Element sic = document.createElement("sic");
sic.appendChild(document.createTextNode(textarea.getText()));
choice.appendChild(sic);
final Element corr = document.createElement("corr");
corr.appendChild(document.createTextNode(correctionarea.getText()));
choice.appendChild(corr);
```

//Etape 6 affichage du resultat

```
final TransformerFactory transformerFactory = TransformerFactory.newInstance();
final Transformer transformer = transformerFactory.newTransformer();
final DOMSource source = new DOMSource(document);
```

//L'affichage dans un fichier

```
final StreamResult sortie = new StreamResult(new File(nom_fichier.xml));
```

//prologue

```
transformer.setOutputProperty(OutputKeys.VERSION, "1.0");
transformer.setOutputProperty(OutputKeys.ENCODING, "UTF-8");
transformer.setOutputProperty(OutputKeys.STANDALONE, "yes");
```

//formatage

```
transformer.setOutputProperty(OutputKeys.INDENT, "yes");
transformer.setOutputProperty("{http://xml.apache.org/xslt}indent-amount", "2");
```

//Affichage du fichier

```
transformer.transform(source, sortie);
```

➤ Table correction

Nom de champ	Type	Null	Index
--------------	------	------	-------

id_correction	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Texte_org	VARCHAR(500)	NON	
Corr	VARCHAR(500)	NON	

➤ **Table coordonnées**

Nom de champ	Type	Null	index
id	INTEGER(11)	NON	Clé primaire
startX	INTEGER(11)	NON	
startY	INTEGER(11)	NON	
endX	INTEGER(11)	NON	
endY	INTEGER(11)	NON	

f. Espace médiateur

Une fois le médiateur s'est authentifié, il accède directement à son espace ou il peut gérer le catalogue ou Gérer les images.



Figure V.8. Interface espace médiateur

La gestion des images se fait soit en rajoutant une image à la base de données ou bien en la supprimant de celle-ci. Tandis que la gestion du catalogue se fait en calculant la similarité structurelle entre deux arbres XML, l'un d'eux se présente dans le catalogue et l'autre sur les documents XML générés à partir de l'annotation et de la transcription des manuscrits arabes. Une fois ce calcul de similitude est effectué, le catalogue sera mis à jour.

g. Interface mise à jour du catalogue :

Elle comporte trois champs, le premier concerne la notice bibliographique à modifier, le second champ, est relié au bouton parcourir, ainsi en cliquant dessus nous choisissons le document annoté ou celui transcrit. Pendant que le troisième champs est réservé pour le catalogue à mettre à jour. Le calcul de similarité et l'alimentation du catalogue se fait suite à une validation.



Figure V.9. Interface mise à jour du catalogue

h. Interface espace visiteur :

Elle permet aux visiteurs de consulter le catalogue des manuscrits arabes numérisés, et d'accéder aux images numérisées, que se soit celles annotées ou bien transcrites.

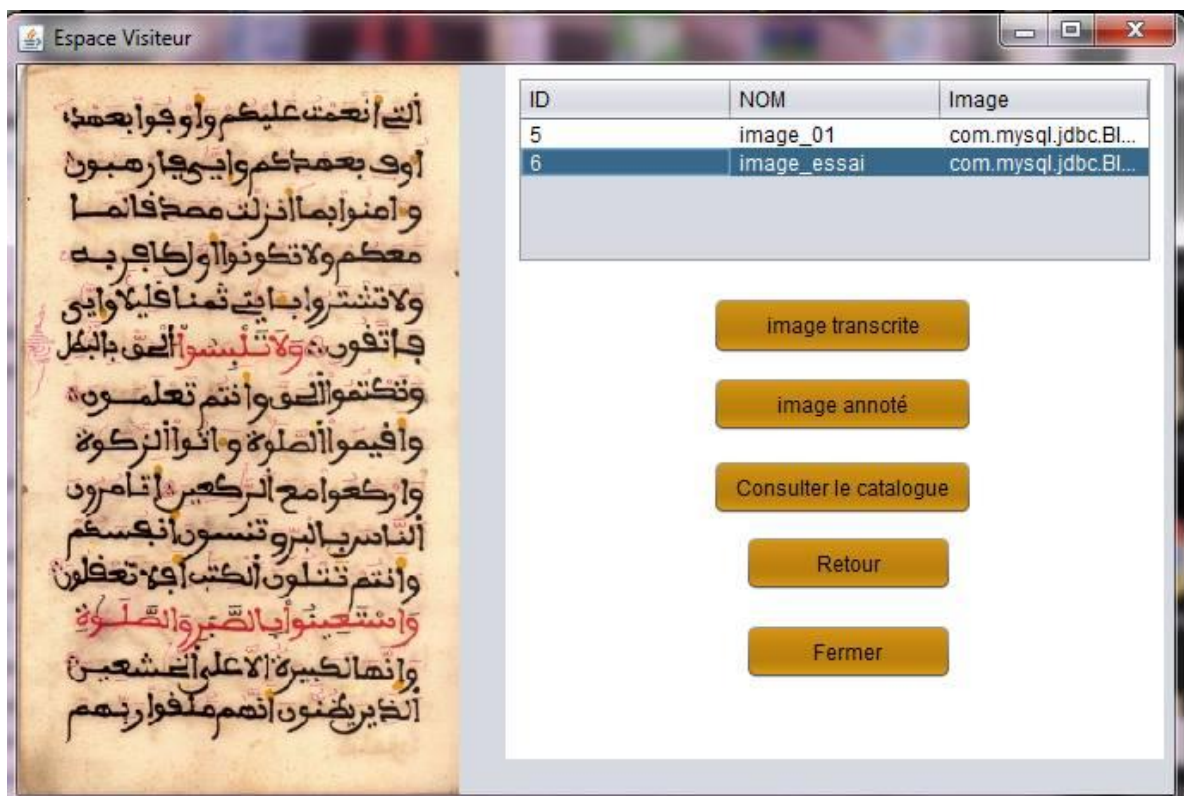


Figure V.10. Interface espace visiteur

Pour afficher l'ensemble des images annotées dans l'interface "Espace visiteur" nous utilisons la requête SQL suivante :

"SELECT DISTINCT id_image,nom_image,image FROM images JOIN coordonees ON coordonees.id_image = images.id_image and type_ano_tran= "+1

La lecture se fait sur une autre interface, l'interface de lecture en utilisant des infobulles :

➤ L'interface de lecture

La lecture se fait sur une autre interface, L'interface de lecture :



Figure V.11. Interface de lecture

Pour afficher le texte dans les info bulles nous devons d'abord récupérer la position du curseur, suite à un évènement "e" de la souris, une fois la position récupérée, nous passons les deux point e.getX et e.getY en paramètre pour la fonction ci dessous:

```
public boolean isIncluded(int x, int y){
    if( ((startX<=x) && (endX>=x)) && ((startY<=y) && (endY>=y)) )
        return true;
    return false;
}
```

Les quatre variables startX, endX, startY, et endY sont utilisées pour le stockage des valeurs concernant la zone sélectionnée récupérée depuis la table coordonnées.

Si la valeur de la fonction `isIncludes` est égale à vrai alors nous récupérons les données annotées depuis les tables métadonnées dans une variable "message" et nous l'affichons dans une info bulle avec la fonction:

```
setToolTipText(message);
```

V.6. Présentation des autres tables de base de données :

Nous avons vu précédemment un exemple complet d'annotation d'une métadonnée, mais notre système permet l'annotation de plusieurs métadonnées, cependant nous tenons à présenter ci-après les autres tables de la base de données.

Table cachet

Nom de champ	Type	Null	Index
id_chapitre	INTEGER (11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Couleur	VARCHAR(255)	NON	
Forme	VARCHAR(255)	NON	
Proprietaire	VARCHAR(500)	NON	

Table chapitre

Nom de champ	Type	Null	Index
id_chapitre	INTEGER(11)	NON	Clé primaire

id_image	INTEGER(11)	NON	Clé étrangère
Titre	VARCHAR(500)	NON	

Table debutfin

Nom de champ	Type	Null	Index
id_debutfin	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Debut	VARCHAR(500)	NON	
fin	VARCHAR(500)	NON	

Table Ecouleur

Nom de champ	Type	Null	Index
id_ecouteur	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Ecout	VARCHAR(500)	NON	

Table Editeur

Nom de champ	Type	Null	Index
---------------------	-------------	-------------	--------------

id_editeur	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
nom_e	VARCHAR(500)	NON	
lieu_e	VARCHAR(500)	NON	
date_e	DATE(8)	NON	

Table Histoire

Nom de champ	Type	Null	Index
id_histoire	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
lieu_h	VARCHAR(500)	NON	
date_h	DATE(8)	NON	
Aquisition	VARCHAR(500)	NON	

provenance	VARCHAR(500)	NON	
Lissane	VARCHAR(500)	NON	

Table Information

Nom de champ	Type	Null	Index
id_information	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Type	VARCHAR(250)	NON	
Texte	VARCHAR(250)	NON	
Place	VARCHAR(250)	NON	

Table langue

Nom de champ	Type	Null	Index
id_langue	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère

Principale	VARCHAR(500)	NON	
secondaire	VARCHAR(500)	NON	

Table Origine

Nom de champ	Type	Null	Index
id_origine	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Pays	VARCHAR(500)	NON	
ville	VARCHAR(500)	NON	
Institution	VARCHAR(500)	NON	
Entrepot	VARCHAR(500)	NON	
Collection	VARCHAR(500)	NON	

Ident	VARCHAR(50)	NON	
--------------	--------------------	------------	--

Table publication

Nom de champ	Type	Null	Index
id_publication	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
publicateur	VARCHAR(500)	NON	
lieu_p	VARCHAR(500)	NON	
date_p	DATE(8)	NON	

Table Reference

Nom de champ	Type	Null	Index
id_reference	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Ref	VARCHAR(500)	NON	

Table Rubrique

Nom de champ	Type	Null	Index
id_rubrique	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Rubric	VARCHAR(500)	NON	

Table Sommaire

Nom de champ	Type	Null	Index
id_sommaire	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Somaire	VARCHAR(500)	NON	

Texte

Nom de champ	Type	Null	Index
id_texte	INTEGER(11)	NON	Clé primaire

id_image	INTEGER(11)	NON	Clé étrangère
Text	VARCHAR(500)	NON	

Titre

Nom de champ	Type	Null	Index
id_titre	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère
Title_h	VARCHAR(500)	NON	
type_h	VARCHAR(50)	NON	
place_t	VARCHAR(50)	NON	

Auteur

Nom de champ	Type	Null	Index
id_auteur	INTEGER(11)	NON	Clé primaire
id_image	INTEGER(11)	NON	Clé étrangère

Nom	VARCHAR(250)	NON	
Prenom	VARCHAR(250)	NON	
Kunya	VARCHAR(250)	NON	
date_n	DATE(8)	NON	
lieu_n	VARCHAR(500)	NON	
date_d	DATE(500)	NON	
lieu_d	VARCHAR(500)	NON	
nom_cel	VARCHAR(500)	NON	
Nisbah	VARCHAR(500)	NON	

V.7. Modèle logique de données des tables de la base de données :

Auteur (id_auteur , # id_image , nom , prenom , kunya , date_m , lieu_m , date_d , lieu_d , nom_cel , nisbah);

Cachet (id_cachet , # id_image , couleur , forme , propriétaire);

Chapitre (id_chapitre , # id_image , titre);

Coordonnées (id , startX , startY , endX , endY , # id_image , type_categorie , id_categorie , type_ano_tran);

Correction (id_correction , #id_image , texte_org , corr);

Debutfin (id_debutfin , # id_image , debut , fin);

Ecouteur (id_ecouteur , # id_image , ecout);

Editeur (id_editeur , # id_image , nom_e , lieu_e , date_e);

Histoire (id_histoire , #id_image , lieu_h , date_h , aquisition , provenance , lissane);

Images (id_image , nom_image , image);

Information (id_information , # id_image , type , texte , place);

Inscription (login , nom_u , prenom_u , m_passe);

Langue (id_langue , # id_image , principale , secondaire);

Mediateur (utilisateur , password);

Origine (id_origine , # id_image , pays , ville , institution , entrepot , collection , ident);

Publication (id_publication , # id_image , publicateur , lieu_p , date_p);

Reference (id_reference , # id_image , ref);

Rubrique (id_rubrique , # id_image , rubric);

Sommaire (id_sommaire , # id_image , somaire);

Texte (id_texte , # id_image , text);

Titre (id_titre , # id_image , title_t , place_t , type_t);

V.8. Conclusion

Ce dernier chapitre conclut la dernière étape de notre projet, l'étape implémentation, dans lequel nous avons présenté la modélisation de notre base de données, le fonctionnement général de notre application par le moyen de quelques interfaces graphiques, et quelques programmes implémentés, cela nous a permis de donner une idée globale sur notre logiciel.

Nous estimons que ce dernier reflète ce qui a été conçu.

Conclusion générale

Au terme de ce travail qui a fait l'objet d'une longue expérience, nous avons pu améliorer nos connaissances et nos compétences tantôt dans le domaine applicatif et tantôt dans celui de la recherche.

Durant le présent projet, Il nous a été confié la mission, de mettre en place un système qui sert à annoter les images des anciens manuscrits arabes, de sauvegarder ces annotations sous forme de documents XML et par la suite d'en extraire les métadonnées suivant un concept bien défini, en l'occurrence celui de similarité structurelle reposant sur l'évaluation du degré de similitude entre deux arbres XML, le premier étant celui du document annoté, pendant que le second est celui de la notice bibliographique, tout en proposant une solution d'optimisation de ce calcul.

La finalité de ce processus étant d'alimenter le catalogue de manuscrits arabes numérisés.

Grâce à ce travail réalisé, nous avons pu mettre en place un moyen d'enrichir le catalogue des manuscrits arabes automatiquement.

Références Bibliographique

- [1] : SOUALAH M. Numérisation des manuscrits arabes : Catalogage et accès Multilingue, Thèse pour l'obtention de Magister l'Institut National de Formation I.N.I. Alger.2008
- [2] : Site officiel d'Asie centrale; Patrimoine Manuscrit et vie intellectuelle de l'Asie central islamique , 1999.
<http://asiecentrale.revues.org/563>
- [3] : Site officiel de dialogue Islam et science.
<http://islam-et-science.forumactif.org>.
- [4] : Kaileh2004, Kaileh H. L'accès à distance aux manuscrits arabes numérisés en mode image, Thèse de Doctorat de l'université Lumière LyonII.2004.
Disponible sur <http://theses.univ-lyon2.fr>
- [5] : Site officiel de Bibliothèque de Lecture Publique et Discothèque.
<http://bibliotheques-discotheque-verdun.fr>
- [6] : L'encyclopédie libre wikipedia.
<https://fr.wikipedia.org/wiki/Numérisation>
- [7] : Génie des Procédés", centre SPIN, Ecole des Mines de Saint-Etienne
- [8] : Raphaël Isdant - 2009.Traitement numérique de l'image.
Disponible sur :
file:///C:/Users/ordina/Downloads/2-traitement_numerique_de_limage.pdf
- [9] : Site officiel de l'infographie, Aout 2016
<http://www.commentcamarche.net/contents/1191-infographie>
- [10] : <http://www.cosmovisions.com/textManuscrit.html>
- [11] : Dictionnaire de français Larousse
<http://www.larousse.fr/dictionnaires/francais/image/41604>
- [12] : Thèse Pétra Bilane Contributions à l'indexation et à la reconnaissance des manuscrits syriaques ,Inter- face homme-machine [cs.HC]. INSA de Lyon, 2010.
Disponible sur : <https://tel.archives-ouvertes.fr/tel-0049953>
- [13] : Manuel d'encodage TEI – Renaissance et temps modernes 2012 ; BVH L'équipe des Bibliothèques Virtuelles Humanistes
Disponible sur : <http://www.tei-c.org/>

- [14] : Massimo BRERO *Septembre 2013*, Système de visualisation, d'annotation et de transcription des manuscrits numérisés de Ferdinand de Saussure, université de Genève.
Disponible sur :
https://cui.unige.ch/~nerima/saussure/master_thesis_brero_2013.pdf .
- [15] : Sarra Ben Lagha2002, Sarra Ben Lagha Inforge, Ecole des HEC, Université de Lausanne – Document numérique. Volume 6 – n°1-2/2002 -"Les dossiers numériques" - Publications Hermes sciences – Octobre 2002.
- [16] : Site officiel d'Automatiser la constitution des catalogues-Biblio TIC.
[http://bibliotic.fr/sites/default/files/supports/catalogue bordeaux/co/automatiser.html](http://bibliotic.fr/sites/default/files/supports/catalogue_bordeaux/co/automatiser.html)
- [17] : http://www1.univ-ag.fr/buag/cours/LS5-web/co/LS5_ULcg03.html
- [18] : Site officiel de Bibliothèque nationale de France , catalogage et indexation.
http://www.bnf.fr/fr/professionnels/catalogage_indexation.html , juillet 2016
- [19] : Mémoire de Rima /Hanane Saouchi/ Boukerzaza, Conception et réalisation d'un site web dynamique pour un magazine en ligne, Université Mentouri Constantine- licence en informatique option académique 2011.
Disponible sur : http://www.memoireonline.com/06/12/5976/m_Conception-et-realisation-d-un-site-web-dynamique-pour-un-magazine-en-ligne7.html
- [20] : Site officiel de Stadium de Toulouse.
<http://www.map.toulouse.archi.fr/>.
- [21] : Site officiel de : Annotation-W3C.
<https://www.w3.org/Amaya/User/Annotations.html.fr>.
- [22] : Pdf Initiation aux techniques documentaires, Catalogage : Introduction-catalogage des ressources ici consultables.
Disponible sur : <http://combot.univ-tln.fr/lea/b.html>.
- [23] : Site officiel d'Enseigner avec le numérique-éduscol , Indexation de ressources (Métadonnées, normes et standards).
<http://eduscol.education.fr/numerique/dossier/archives/metadata/metadonnees>.
- [24] : Site office de la Bibliothèque nationale de France ,Document numérique et métadonnées , Février 2016

http://www.bnf.fr/fr/professionnels/numerisation_boite_outils/a.metadonnees_doc_numerique.html

- [25] : Philippe Bessières —Adeline Nazarenko — Claire Nédellec , Apport de l'apprentissage à l'extraction d'information : le problème de l'identification d'interactions géniques , Mathématique, Informatique et Génome (MIG) INRA ; Laboratoire d'Informatique de Paris-Nord, UPRESA 7030 CNRS Université Paris Nord ; Equipe Inférence et Apprentissage LRI UMR 8623 CNRS Université Paris-Sud.

Disponible sur :<http://cide-caderige.pdf>.

- [26] : Guillaume Joutel 2009, Analyse multi résolution des images de documents manuscrits, Institut National des Sciences Appliquées de Lyon Thèse en vue de l'obtention du grade de docteur Ecole Doctorale Informatique et Mathématiques (INFOMATHS) Spécialité : Informatique.

Disponible sur :<http://these-h.pdf> .

- [27] : Nancy IDEa et Jean VÉRONISb, Présentation de la TEI : Text Encoding Initiative, Department of computer science, vassar college Poughkeepsie. Laboratoire Parole et Langage, Université de Provence et CNRS, juin 1996.

Disponible sur :<http://sites.univ-provence.fr/~veronis/pdf/1996gut-presentation.pdf>

- [28] : Christophe Rey 2000, Informatisation des dictionnaires anciens : l'exemple de métalangage grammatical dans le dictionnaire François de César-Pierre-Rechelet, Université de provenance.

Disponible sur :https://www.u-picardie.fr/LESCLaP/reylreyc_dea.pdf

- [29] : Site officiel de : openclassroom-Introduction aux définitions et aux DTD, 2016.

<https://openclassrooms.com/courses/structurez-vos-donnees-avec-xml/introduction-aux-definitions-et-aux-dtd,Aout>

ANNEXE

A.1. Présentation d'UML

A.1.1. Définition

UML, Unified Modeling Language, est le langage de modélisation d'objet, conçu pour fournir une méthode normalisée afin de visualiser la conception d'un système. il est couramment utilisé en développement logiciel et en conception orientée objet.

L'UML est le résultat de la fusion de précédents langages de modélisation objet : Booch, OMT, OOSE⁷.

A.1.2. utilisation

UML est utilisé pour spécifier, visualiser, modifier et construire les documents nécessaires au bon développement d'un logiciel orienté objet. UML offre un standard de modélisation, pour représenter l'architecture logicielle. Les différents éléments représentables sont :

- Activité d'un objet/logiciel
- Acteurs
- Processus
- Schéma de base de données
- Composants logiciels
- Réutilisation de composants

Grâce aux outils de modélisation UML, il est également possible de générer automatiquement une partie de code, par exemple en langage JAVA, à partir des divers documents réalisés.

A.1.3. Formalisme d'UML :

UML se décompose en plusieurs sous-ensembles :

- ❖ Les **vues** : Les vues sont les observables du système. Elles décrivent le système d'un point de vue donné, qui peut être organisationnel, dynamique, temporel, architectural, géographique, logique, etc. En combinant toutes ces vues, il est possible de définir (ou retrouver) le système complet.
- ❖ Les **diagrammes** : Les diagrammes sont des éléments graphiques. Ceux-ci décrivent le contenu des vues, qui sont des notions abstraites. Les diagrammes peuvent faire partie de plusieurs vues.

⁷La méthode OOSE est avec OMT et Booch, l'une des méthode d'analyse et de conception orientée objet à l'origine d'UML.

- ❖ Les **modèles d'élément** : Les modèles d'élément sont les briques des diagrammes UML, ces modèles sont utilisés dans plusieurs types de diagrammes. Exemple d'élément : cas d'utilisation (CU ou cadut'), classe, association, etc.

A.1.3.1 Les vues d'UML :

Une façon de mettre en œuvre UML est de considérer différentes vues qui peuvent se superposer pour collaborer à la définition du système :

- ❖ Vue des **cas d'utilisation** : c'est la description du modèle vu par les acteurs du système. Elle correspond aux besoins attendus par chaque acteur (c'est le QUOI et le QUI).
- ❖ Vue **logique** : c'est la définition du système vu de l'intérieur. Elle explique comment peuvent être satisfaits les besoins des acteurs (c'est le COMMENT).
- ❖ Vue **d'implémentation** : cette vue définit les dépendances entre les modules.
- ❖ Vue des **processus** : c'est la vue temporelle et technique, qui met en œuvre les notions de tâches concurrentes, stimuli, contrôle, synchronisation, etc.
- ❖ Vue de **déploiement** : cette vue décrit la position géographique et l'architecture physique de chaque élément du système (c'est le OÙ).

A.1.3.2 Les diagrammes :

UML 2.3 propose 14 types de diagrammes :

Diagrammes structurels ou statiques :

Les diagrammes structurels ou statiques (Structure Diagram) rassemblent :

- ✓ Diagramme de classes (Class diagram).
- ✓ Diagramme d'objets (Object diagram).
- ✓ Diagramme de composants (Component diagram).
- ✓ Diagramme de déploiements (Deployment diagram).
- ✓ Diagramme de paquetages (Package diagram).
- ✓ Diagramme de structure composite.
- ✓ Diagramme de profils (Profile diagram).

Diagrammes comportementaux

Les diagrammes comportementaux (Behavior Diagram) rassemblent :

- ✓ Diagramme de cas d'utilisation (use-cases ou Use Case Diagram).
- ✓ Diagramme états-transitions (State Machine Diagram).
- ✓ Diagramme d'activité (Activity Diagram).

Diagrammes d'interaction ou dynamiques

Les diagrammes d'interaction ou dynamiques (Interaction Diagram) rassemblent :

- ✓ Diagramme de séquence (Sequence Diagram).
- ✓ Diagramme de communication (Communication Diagram).
- ✓ Diagramme global d'interaction (Interaction Overview Diagram).
- ✓ Diagramme de temps (Timing Diagram).

A.1.3.3 Les éléments de modélisation :

- ❖ Le stéréotype est une marque de généralisation notée par des guillemets, cela montre que l'objet est une variété d'un modèle.
- ❖ Le classeur est une annotation qui permet de regrouper des unités ayant le même comportement ou structure. Un classeur se représente par un rectangle conteneur, en traits pleins.
- ❖ Un paquetage regroupe des diagrammes ou des unités.
- ❖ Chaque classe ou objet se définit précisément avec le signe « :: », ainsi l'identification d'une Classe X en dehors de son package ou de son classeur sera définie par « Package A::Classeur B::Classe X ».

II.1.4 : Avantages d'UML :

- **UML est un langage formel et normalisé**
 - Gain de précision.
 - Gage de stabilité.

- Encourage l'utilisation d'outils.
- Un langage sans ambiguïté.

• **UML est un support de communication performant**

- Il cadre l'analyse.
- Il facilite la compréhension de représentation abstraite complexe.
- Un langage universel pouvant servir de support pour tout langage orienté objet.
- Une notation graphique simple compréhensible même par des non informaticien.

A.1.5. Inconvénients d'UML :

- La mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation.
- Son point faible est sans contestation possible, la lourdeur (relative) de sa mise en place au sein de n'importe quel processus.
- Son apprentissage assez long et rigoureux peut également être un frein à son utilisation.

A.2. Présentation de XML

A.2.1. Définition du XML

L'*Extensible Markup Language* (XML, « langage de balisage extensible » en français) est un métalangage informatique de balisage générique qui dérive du SGML. Cette syntaxe est dite « extensible » car elle permet de définir différents espaces de nom c'est-à-dire des langages avec chacun leur vocabulaire et leur grammaire. Elle est reconnaissable par son usage des chevrons (< >) encadrant les balises. L'objectif initial est de faciliter l'échange automatisé de contenus complexes (arbres, texte riche...) entre systèmes d'informations hétérogènes.

Enfin, on peut dire que *le langage XML est un langage qui permet de décrire des données à l'aide de balises et de règles que l'on peut personnaliser.*

A.2.2. Les objectifs du XML

Comme nous l'avons vu, l'objectif du XML est de faciliter les échanges de données entre les machines. A cela s'ajoute un autre objectif important : décrire les données de manière aussi bien compréhensible par les hommes qui écrivent les documents XML que par les machines qui les exploitent.

Le XML se veut également compatible avec le web afin que les échanges de données puissent se faire facilement à travers le réseau Internet.

Le XML se veut donc standardisé, simple, mais surtout extensible et configurable afin que n'importe quel type de données puisse être décrit.

A.2.3. Les éléments de base de xml

a. les balises

Comme nous l'avons vu précédemment, XML est défini comme un langage informatique de balisage. En effet, les balises sont les éléments de base d'un document XML. Une balise porte un nom qui est entouré de chevrons. Une balise commence donc par un < et se termine par un >. Par exemple : <balise> définit une balise qui s'appelle "balise".

En XML, on distingue 2 types de balises : les **balises par paires** et les **balises uniques**.

➤ les balises par paires

Les balise par paires sont composées en réalité de 2 balise que l'on appelle ouvrantes et fermantes. la balise ouvrante commence par < et se termine par > tandis que la balise fermante commence par </ et se termine par >.

Bien évidemment, on peut mettre "des choses" entre ces balises. On parle alors de **contenu**.

Par exemple :

<balise> contenu de la balise </balise>

On parle d'arborescence, lorsqu'une balise par paires peut contenir d'autres balises.

➤ les balises uniques

Une **balise unique** est en réalité une balise par paires qui n'a pas de contenu. Cependant elle commence par < et se termine par />.

Par exemple :

<balise/>

➤ Les règles de nommage des balises

Ce qui rend le XML générique, c'est la possibilité de créer notre propre langage balisé. Ce **langage balisé**, comme son nom l'indique, est un langage composé de balises sauf qu'en XML, c'est au programmeur de choisir leurs noms.

Il y a cependant quelques règles de nommage à respecter pour les balises de votre langage balisé :

- Les noms peuvent contenir des lettres, des chiffres ou des caractères spéciaux.
- Les noms ne peuvent pas débiter par un nombre ou un caractère de ponctuation.
- Les noms ne peuvent pas commencer par les lettres XML (quelle que soit la casse).
- Les noms ne peuvent pas contenir d'espaces.
- On évitera les caractères - , ; . < et > qui peuvent être mal interprétés dans vos programmes.

b. Les attributs

Il est possible d'ajouter à nos balises ce qu'on appelle des **attributs**. Tout comme pour les balises, c'est à nous d'en choisir le nom.

Un **attribut** peut se décrire comme une option ou une donnée cachée. Ce n'est pas l'information principale que souhaite transmettre la balise, mais il donne des renseignements supplémentaires sur son contenu.

Exemple:

```
<prix unité="dinar">250</prix>
```

c. Les commentaires :

Un commentaire est un texte qui permet de donner une indication sur ce que l'on fait. Il vous permet d'annoter votre fichier et d'expliquer une partie de celui-ci.

En XML, les commentaires ont une syntaxe particulière. C'est une balise unique qui commence par `<!--` et qui se termine par `-->`.

Exemple

```
<!-- ceci est un commentaire -->.
```

A.2.4. Structure d'un document XML

Un document XML peut être découpé en 2 parties : le **prologue** et le **corps**.

a. Le prologue

Le prologue correspond à la première ligne du document XML. Il donne des informations de traitement.

Voici à quoi ressemble le prologue :

```
<?xml version = "1.0" encoding="UTF-8" standalone="yes" ?>
```

- **La version** : Dans le prologue, on commence généralement par indiquer la version de XML que l'on utilise pour décrire nos données.
- **Le jeu de caractères** : La seconde information de notre prologue est `encoding="UTF-8"`. Il s'agit du jeu de caractères utilisé dans le document XML. Par défaut, l'encodage de XML est l'UTF-8
- **Un document autonome** : La dernière information présente dans le prologue est `standalone="yes"`. Cette information permet de savoir si votre document XML est autonome ou si un autre document lui est rattaché.

b. Le corps

Le **corps** d'un document XML est constitué de l'ensemble des balises qui décrivent les données. Il y a cependant une règle très importante à respecter dans la constitution du corps: *une balise en paires unique doit contenir toutes les autres*. Cette balise est appelée **élément racine** du corps.

c. Document bien formé

On sous entend par un document XML bien formé à un document XML avec une syntaxe correcte, c'est-à-dire :

- S'il s'agit d'un document utilisant la version 1.1 du XML, le prologue est bien renseigné.
- Le document XML ne possède qu'une seule balise racine.
- Le nom des balises et des attributs est conforme aux règles de nommage.
- Toutes les balises en paires sont correctement fermées.
- Toutes les valeurs des attributs sont entre guillemets simples ou doubles.
- Les balises de votre document XML ne se chevauchent pas, il existe une arborescence dans votre document.

En résumé

- Le XML a été créé pour faciliter les échanges de données entre les machines et les logiciels.
- Le XML est un langage qui s'écrit à l'aide de **balises**.

- Le XML est une recommandation du **W3C**, il s'agit donc d'une technologie avec des règles strictes à respecter.
- Le XML se veut compréhensible par tous : les hommes comme les machines.
- Le XML nous permet de créer notre propre vocabulaire grâce à un ensemble de règles et de balises personnalisables.
- un document XML doit être bien formé pour être exploitable.

A.3. Présentation du langage de programmation:

A.3.1. Langage JAVA

Java est le langage phare de la société Sun Microsystems. Il a été créé en 1991, dans le but d'intégrer des appareils domestiques pour un projet de domotique, dont le nom de code était « Green ». Il fallait donc que le langage soit léger et portable sur toute configuration. Baptisé dans un premier temps « Oak », il changea rapidement de nom, Oak étant déjà utilisé, pour devenir « Java », terme de l'argot en anglais qui signifie café. Les versions suivantes, avec l'ajout notamment de Swing permettant de créer des interfaces graphiques stables, rendront le langage de plus en plus populaire pour la création de nombreux types d'application, avec comme atout principal : sa portabilité.

Java se décline en de nombreuses versions, pour ordinateurs (avec une version de la machine virtuelle par système d'exploitation), pour téléphones mobiles, pour la programmation d'applications commerciales... Chacune des versions possède certaines bibliothèques en commun et certaines autres spécifiques, écrites différemment ou absentes. Certains systèmes d'exploitation possèdent la machine virtuelle dans leur configuration de base (Mac OS X...), tandis qu'elle doit être installée séparément sur d'autres (Windows, Mac OS 9...)

Java ressemble en plusieurs points au C++, Java est donc un langage objet ; toutefois, étant prévu pour tourner sur des petites configurations (souvenons-nous qu'il est prévu à la base pour les appareils ménagers), il est aussi utilisé à la manière d'un langage procédural. Il possède plusieurs autres caractéristiques, dont le fait d'être fortement typé (le type des variables en détermine strictement le type de contenu).

Il est aussi, et c'est la une de ses grandes forces, portable (quasiment) sans modifier une seule ligne du code.

A.3.2. Les avantages du JAVA:

- Le Byte-code, qui assure à Java une Portabilité complète vers de très nombreux systèmes.
- L'importance de l'api de base qui offre tous les services de base, notamment pour la construction des interfaces graphiques.
- La 3^{ème} force de Java, c'est son adaptabilité dans de nombreux domaines, autant pour le web que pour les systèmes embarqués.

Les caractéristiques du langage Java :

- Orienté-objet, en intégrant
 - l'encapsulation (masquage d'information, séparer fonction et représentation)
 - l'héritage (déclarer une nouvelle classe comme extension d'une classe existante),
 - la liaison dynamique (les appels à une opération ou à n'importe laquelle de ses redéfinitions dans les classes dérivées sont "résolus" au moment de l'exécution, en fonction du type de l'objet concerné).
 - une gestion automatique de la mémoire
 - Une bibliothèque de classes standard

- Indépendant de la machine

C'est peut-être un des plus gros points forts de Java. Un source Java compilé donne des "bytes-codes", sorte de pseudo-assembleur qui s'adresse à une machine virtuelle Java. Cela signifie que pour exécuter ces bytes-codes, il faut un interpréteur Java qui simule cette machine virtuelle. Les logiciels de navigation compatibles Java intègrent un tel interpréteur. Les applications ou applets Java sous leur forme "bytes-codes" sont donc indépendantes de la machine physique.

- Multi-thread

Une application Java peut lancer plusieurs tâches ou processus indépendants qui s'exécutent simultanément. Java gère un mécanisme de moniteurs, qui permet de synchroniser ces processus ou "threads "

- Sécurisé

Assurer la sécurité des programmes qui circulent sur le réseau, est un problème majeur. Java a intégré, dès la conception, plusieurs mécanismes de sécurité visant à rendre les programmes fiables et à éliminer les risques de virus (vérification du "bytes codes", pas de manipulation pointeurs).

- Simple

Plus simple que le C++, Java n'a pas de pointeurs - arithmétique sur les pointeurs - , pas de fonctions. Il n'intègre ni l'héritage multiple, ni la surcharge d'opérateurs.

- Conçu pour les réseaux, paradigme client-serveur.

A.4. Présentation de DOM :

DOM ou **Document Object Model**, son nom complet, est ce qu'on appelle un **parseur XML**, c'est-à-dire, une technologie grâce à laquelle il est possible de lire un document XML et d'en extraire différentes informations (éléments, attributs, commentaires, etc...) afin de les exploiter.

DOM est un standard du W3C, Il est très important de noter qu'il est une recommandation complètement *indépendante* de toute plate-forme et langage de programmation. Au travers de DOM, le W3C fournit une recommandation, c'est-à-dire une manière d'exploiter les documents XML.

Le modèle DOM est non seulement une spécification multiplateformes, mais aussi multi-langages. Aujourd'hui, la plupart des langages de programmation propose leur implémentation de DOM :

- C.
- C ++.
- Java.
- C#.
- Perl.
- PHP.
- etc.

❖ Manipulation de fichier XML grâce à l'API DOM

L'API DOM permet de manipuler facilement des fichiers XML en tirant parti de leur structure d'arbre.

Le modèle d'objet de document fourni tout une panoplie d'outils destinés à construire et manipuler un document XML. Pour cela, le DOM met à disposition des interfaces, des méthodes et des propriétés permettant de gérer l'ensemble des composants présents dans un document XML.

Le DOM spécifie diverses méthodes et propriétés permettant notamment, de créer (createNode...), modifier (replaceChild...), supprimer (remove...) ou d'extraire des données (get...) de n'importe quel élément ou contenu d'un document XML.

De même, le DOM définit les types de relation entre chaque nœud, et des directions de déplacement dans une arborescence XML. Les propriétés parentNode, childNodes, firstChild, lastChild, previousSibling et nextSibling permettent de retourner respectivement le père, les

enfants, le premier enfant, le dernier enfant, le frère précédent et le frère suivant du nœud courant.

Le modèle d'objet de document offre donc au programmeur les moyens de traiter un document XML dans sa totalité