REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

**UNIVERSITE SAAD DAHLAB DE BLIDA**

**FACULTE DES SCIENCES**

**DEPARTEMENT D'INFORMATIQUE**

**Thème :**

# Alimentation automatique d'un corpus

Pour l'obtention du diplôme de MASTER

Spécialité : Informatique

Option : Ingénierie des logiciels

**Présentée par :**

MAHMOUDI Lakhdar

Président : BALA. M

Examinateur : FERFERA. S

**Promoteur :**

Mr. NEHAL Djilali

2015/2016

## ملخص

يأتي هذا المشروع في إطار استخدام تكنولوجيا المعلومات والاتصالات الجديدة في بناء مدونة نصية باللغة العربية. ويمكن استخدام المدونات في العديد من المجالات بما في ذلك اللسانيات ، تقنيات الترجمة والتدريس. والهدف هو بناء مدونة باستغلال البيانات النصية المتاحة في شبكة الإنترنت. الجديد في هذا العمل هو ان عملية الانشاء تصبح آلية. من خلال مصادر مختلفة للبيانات النصية المتاحة مجانا على شبكة الإنترنت، اعتمادا على روبوت فهرسة يسترجع البيانات بطريقة آلية ويشفرها، بعد ذلك تخزن في مدونة أساسية معدة مسبقا كقاعدة بيانات على شكل لغة التوصيف الموسعة (xml).



**كلمات مفتاحية**

مدونة، الويب، الترميز، لغة التوصيف الموسعة، آلية فهرسة، المحلل .



## Résumé

Ce projet rentre dans le cadre de l'utilisation des nouvelles technologies de l'information et de la communication dans la construction de corpus textuels en arabe. Ces corpus peuvent alors être utilisés dans plusieurs domaines dont la linguistique de corpus, la traductique et l'enseignement. L'objectif est de construire un corpus en profitant des données textuelles disponibles au niveau du web. Ce qui est nouveau dans ce travail est l'automatisation du processus. À partir de diverses sources de données disponibles en libre accès sur le web, un robot d'indexation récupère et encode d'une manière automatique du contenu textuel puis l'injecte dans un corpus noyau préalablement préparé sous la forme d'une base de données XML.



**Mots clés :**

Corpus , web, encodage, XML, robot d'indexation, parseur.



## Abstract

This project is about using new information and communication technologies in order to create Arabic textual corpora. These corpora can be used in lot of domains, linguistics, translation and especially teaching. The main purpose is to create corpora using text data which are available on web sites. What's new in our project is the automating of the process. From so many free access data resources on the web, a web crawler is able to encode automatically the text and then throw it into a core corpora prepared in a data base of XML



## Keywords:

corpora, web, encoding,XML,web crawler,parser.

# *Dédicace*

*Merci Allah (mon Dieu) de m'avoir donné la capacité d'écrire et de réfléchir, la force d'y croire, La patience d'aller jusqu'au bout du rêve et le bonheur de lever mes mains vers le ciel et de dire " Ya Kayoum "*

*Je dédie ce travail à :*

*Mes parents : pour tous les sacrifices qu'ils ont consacrés pour mes éducations, mes études et mon égard.*

*Tous les membres de ma famille : je vous souhaite plein de succès et beaucoup de bonheur dans votre vie.*

*Particulièrement, Mes frères et sœurs qui n'ont cessé d'être pour moi des exemples de persévérance, de courage et de générosité.*

*Mes professeurs de primaire jusqu' a l Université: veillez trouver dans ce travail l'expression de ma profonde reconnaissance et ma grande estime.*

*Finalement, tous mes amis et à touts personnes qui m'a porté d'aide d'une manière directe ou soit elle indirecte, le long de ma vie.*

# *Remerciement*

*Au terme de ce travail, je tiens à remercier tout d'abord mon promoteur Mr NEHAL Djilali et mon encadreur Mr MAMMERI Mahmoud Fawzi de ma avoir guidés tout au long de ce travail, pour ses conseils, patience et aide.*

*Mes vifs remerciements vont à tous les Profs au niveau de département de l'informatique pour leurs soutiens, leurs disponibilités tout au long de nos cinq années de spécialité ainsi que tout le staff de département.*

*Je remercie tous ceux qui m'ont aidés de près ou de loin à faire ce travail.*

*Mon profonds remerciements et mon gratitude vont aussi aux membres du jury, pour leur accord de juger mon travail.*

# Sommaire

## Introduction Générale



## Chapitre1 : Le corpus



## Chapitre 2 : Corpus de Latifa AL-sulaiti

# Sommaire

## Introduction Générale



## Chapitre1 : Le corpus



## Chapitre 2 : Corpus de Latifa AL-sulaiti

## Chapitre 3 : Conception

## Chapitre 4 : implémentation



## Conclusion Générale

# Sommaire de Figures



## Sommaire de Tableaux

# Introduction Générale

L'objet de notre travail est le corpus. Les corpus sont des collections de textes regroupés sur la base d'hypothèses de travail en vue de les interroger par la suite. Le corpus est devenu un moyen de travail indispensable dans le développement d'outils modernes pour l'enseignement, dans le développement de lexiques et de dictionnaires et dans la recherche. Pour mettre en valeur l'objet que nous manipulerons tout au long de ce texte, nous citons trois exemples d'utilisation des corpus. C'est à partir de corpus que les lexicographes, par exemple, confectionnent les exemples que nous retrouvons dans les dictionnaires usuels. Désormais, les élèves et les enseignants ne se contentent plus d'exemples artificiels combien même bien conçus par des spécialistes. Ces exemples artificiels cèdent de plus en plus de la place aux données authentiques extraites de textes réels. Les corpus sont aussi utilisés par les linguistes pour analyser des textes, émettre et confirmer (ou infirmer) des hypothèses linguistiques, étudier et comparer des textes de différentes époques, … Enfin, des corpus parallèles sont utilisés en traduction.

En langue arabe, peu de corpus existent parmi lesquels peu sont disponibles en libre accès, et la communauté arabophone souffre encore d'un manque terrible en matière d'outils linguistiques soit pour une utilisation personnel soit pour une utilisation dans le domaine de recherche et développement. Ainsi, la construction de corpus en arabe est un domaine de développement très porteur mais reste toujours très long, lourd et restreint à une communauté scientifique très limitée.

En effet, le développement d'un corpus est une tâche rude qui nécessite temps, personnes et compétences. Pour construire un corpus, il faut suivre plusieurs étapes que nous résumons en : (i) un travail linguistique où il s'agit de choisir les textes à inclure dans le corpus, cette étape doit être validée par un expert, suivit par (ii) un travail de collection, digitalisation (numérisation) et stockage des textes appropriés, et enfin (iii) un travail de classification et d'annotation du corpus.

Ainsi, la nécessité de s'approprier d'un corpus robuste d'une part et l'impossibilité d'avoir le temps, les personnes et par conséquent les compétences pluridisciplinaires nécessaires à sa construction d'autre part, nous ont poussé à nous intéresser uniquement à certaines des étapes du processus de développement du corpus que nous avons jugé à notre portée.

L'objectif de notre travail est donc la construction d'un corpus pour la langue arabe, mais qui ne suit pas toutes les étapes du processus de développement du corpus. En d'autres termes, nous ne démarrons pas à partir de zéro. En effet, la disponibilité d'un corpus « noyau » en libre accès d'une part et la « disponibilité » du Web seraient deux moyens très

puissants et donc suffisants pour mettre en place un dispositif de construction automatique et pérenne d'un corpus aussi large que l'on voudra et dans peu de temps. Ce qui est très attirant dans ce procédé de réalisation est le minimum de personnes et de compétences linguistiques dont nous aurons besoin pour construire et alimenter notre corpus. En effet, la construction « linguistique » réelle du corpus serait assuré au sein du corpus « noyau » que nous nous approprions dès le départ du projet (il s'agit là d'une hypothèse). D'autre part, le Web nous assurera les textes nécessaires que nous aurons besoin pour mettre notre corpus à disposition des utilisateurs en un temps « record ».

Dans une étude préalable, nous nous sommes intéressés donc aux corpus annotés et en libre accès. Cette étude exploratoire nous a menés vers un corpus ayant les caractéristiques du « corpus noyau » que nous nous sommes fixé dans nos objectifs suscités. Il s'agit du corpus de Latifa Al-Solaiti. C'est un corpus arabe développé à l'université de Leeds, annoté en XML et disponible en libre accès. Ce corpus a été construit manuellement. Cette procédure est sans aucun doute nécessaire pour la maitrise du processus de construction mais reste tout de même insuffisante pour des extensions futures.

Par conséquent, le travail réel à réaliser est une extension quantitative du corpus de Latifa Al-Solaiti avec des textes provenant du Web. Ce corpus étant réalisé manuellement, nous nous intéressons alors à le doter de nouveaux textes pour les genres déjà existants. Ainsi, nous nous restreignons aux deux dernières étapes du processus de construction du corpus qui se résument en la collection et le stockage des textes avec une certaine annotation.

En résumé, ce que nous suggérons est une extension d'un corpus « noyau » qui permettra à partir du « Web » d'automatiser les taches de construction à savoir la récupération des textes, leur nettoyage, stockage, classification et annotation. Cette automatisation requière notamment des tâches secondaires mais non moins importantes qui se résument en la veille (notre application aurait pour tâche d'être à l'écoute des nouveaux textes disponibles sur le Web) et la cohérence (notre application serait en mesure bien sur d'éviter des textes redondants).

L'objectif souhaité serait d'arriver à une extension (i) organisée, (ii) automatisé et (iii) programmée pour un corpus arabe qui sera (i) basée sur les besoins des utilisateurs, (ii) librement accessible et (iii) qui reflète l'état de la langue arabe à l'heure actuelle.

# Chapitre 1

# Corpus

# 1. Introduction

Les corpus sont reconnus en tant que ressources importantes dans plusieurs domaines tels que l'enseignement des langues, la traductique, la lexicologie et la recherche. Ainsi, ce chapitre s'intéresse à la notion de corpus du point de vue de sa définition, construction, consistance et domaine d'utilisation. Il s'intéresse aussi à énumérer les corpus existants sur le marché ou dans le domaine de la recherche pour (i) l'anglais, qui est la première langue ayant bénéficiée des premières et sérieuses recherches en linguistique de corpus et pour (ii) l'arabe, qui est la langue de notre corpus. L'objectif de notre projet étant la construction par alimentation automatique d'un corpus arabe, nous réservons la dernière section de ce chapitre à la présentation du corpus de Latifa Al-Solaiti qui sera l'un des deux objets principaux du projet : la cible des données textuelles récupérées à partir du web ; le second objet étant le web lui-même à travers les sites sources de données textuelles.

# 2. Corpus

## 2.1. Définitions

Un corpus est :

- « un ensemble fini d'énoncés écrits ou enregistrés, constitués en vue de leur analyse linguistique. » (définition du dictionnaire Larousse) [1]
- « une collection de données langagières qui sont sélectionnées et organisées selon des critères linguistiques et extra-linguistiques explicites pour servir d'échantillon d'emplois déterminés d'une langue » Habert (2000, p:1) [3]

Sinclair (1996a, p:5) définit le corpus électronique comme :

« a corpus which is encoded in a standardized and homogenous way for open-ended retrieval tasks. » [2]

Ainsi un corpus est une collection de morceaux de textes dans un langage sous forme électronique écrite ou parlée artistique ou non (textes, images, vidéos, etc.). [4] Les textes du corpus doivent être sélectionnés selon des critères externes, et on peut l'utiliser dans plusieurs domaines : études littéraires, linguistique, informatique recherche, etc.

## 2.2. Linguistique de corpus

La linguistique de corpus peut être définie comme l'étude de la langue grâce à l'utilisation de grandes collections de textes lisibles par machine. La linguistique de corpus n'est pas une branche de la linguistique, mais plutôt une méthodologie et des techniques qui

peuvent être utilisées pour étudier tous les aspects de la langue tels que la syntaxe, sémantique, pragmatique, parole, études lexicographiques, etc.

## 2.3. Construction de corpus

Le développement d'un corpus est une tâche rude qui nécessite temps, personnes et compétences. Pour construire un corpus, il faut suivre plusieurs étapes que nous résumons en :

a) **Un travail linguistique :** Où il s'agit de choisir les textes à mettre dans le corpus. Pour cela, il faut préalablement choisir les genres à y inclure (genre journalistique, scientifique, nouvelle, divertissement, éducation, culture, politique, etc.). Cette étape doit être validée par un expert. Dans une perspective d'utilisation dans l'enseignement, par exemple, il serait nécessaire de demander conseils et avis des enseignants et étudiants sur la meilleure structure pour le corpus de sorte que le choix des textes soit le plus adéquat et reflète les besoins des utilisateurs finaux qui sont les enseignants et les étudiants eux-mêmes.

b) **Un travail de collection :** Dans cette étape, il s'agir de digitaliser (numériser) et stocker les textes appropriés. Le traitement dépendra du type de la source (textes, images, vidéos, etc..) et de l'outil de stockage (format de fichier). Le choix de la méthode de stockage se fait selon plusieurs critères ainsi que la capacité des périphériques informatiques utilisés.

c) **Un travail de classification et d'annotation :** La classification et l'annotation du corpus se fait selon la stratégie de construction du corpus, mais pour le moment, il n'y a pas une norme d'or pour l'annotation.

## 2.4. Types de corpus

Que le corpus soit écrit ou parlé, deux types sont identifiés :

### 2.4.1. Corpus bruts

Le corpus brut est principalement le texte lui-même, sans autres informations supplémentaires sur son contenu. En d'autres termes, il n'est pas annoté.

### 2.4.2. Corpus annotés (ou balisé)

L'annotation est la pratique consistant à ajouter des informations linguistiques interprétatives à un corpus. Par exemple, un type commun d'annotation est l'ajout d'étiquettes indiquant chacune la partie du discours (part of speech) à

laquelle appartiennent chacun des mots du corpus. Donc, il s'agit de textes annotés et enrichi avec une variété d'informations.

Si un corpus a été annoté à l'avance, cela serait bien évidemment d'une grande utilité pour de nombreux types de traitements ou d'analyses automatiques.

## 2.5. But du corpus

Le but d'un corpus est non seulement de recueillir un gros fichier de textes différents et le stocker sur ordinateur, mais aussi de préparer les textes et les mettre dans un certain format afin qu'ils puissent être utilisés par les outils de recherche. Ainsi, les résultats des recherches dans le corpus peuvent être affichés d'une manière significative et utile pour les utilisateurs (linguiste, enseignant, apprenant). Par exemple, dans un niveau avancé, enseignants et apprenants peuvent explorer les différentes utilisations d'un mot donné dans différents types de textes, sa fréquence, ses différentes significations, ses environnements syntaxiques, et voir même si le mot a la même fréquence d'occurrence dans les différents types de textes contenus dans le corpus.

## 2.6. Corpus et enseignement

Les corpus ont longtemps été utilisés pour la recherche, et uniquement en 1992 que certaines idées ont été proposées pour utiliser les corpus dans l'enseignement. Une fois la disponibilité croissante d'ordinateurs rapides et munis de larges espaces de stockage, les corpus sont vite devenus un outil indispensable dans beaucoup de domaines, en particulier l'enseignement. Aujourd'hui, les corpus sont utilisés dans de nombreuses universités et institutions d'enseignement des langues. Ils ont été expérimentés dans l'enseignement de la grammaire, la traduction, le vocabulaire et dans beaucoup d'autres cours.

## 2.7. Domaines d'application

- **Études littéraires**

  Le corpus regroupe un ensemble de textes ayant une visée commune. Un corpus peut être constitué de documents différents (tableaux, extraits de textes, ...). Ces documents divers ont un point en commun. En général, c'est le thème qui fait figure de leur ressemblance et il faut avoir une technique particulière pour le déchiffrer.

- **Études Linguistiques**

  La branche de la linguistique qui se préoccupe plus spécifiquement des corpus s'appelle logiquement la linguistique de corpus. Elle est liée au développement des systèmes informatiques, en particulier à la constitution de bases de données textuelles.

- **Études scientifiques**

  Les corpus sont des outils précieux et indispensables en traitement automatique du langage naturel. En effet, ils permettent d'extraire un ensemble d'informations utiles pour des traitements statistiques. Il est possible de s'appuyer sur des corpus (à condition qu'ils soient bien formés) pour formuler et vérifier des hypothèses scientifiques.

## 3. Caractéristiques d'un Corpus bien formé

Plusieurs caractéristiques sont à prendre en compte pour la création d'un corpus bien formé :

### 3.1. Taille

Le corpus doit évidemment atteindre une taille critique pour permettre des traitements statistiques fiables. Il est impossible d'extraire des informations fiables à partir d'un corpus trop petit.

### 3.2. Langue du corpus

Un corpus bien formé doit nécessairement couvrir une seule langue et une seule déclinaison de cette langue. Il existe par exemple de subtiles différences entre le français parlé en France et ceux parlés en Belgique ou au Québec. Il ne sera donc pas possible de tirer des conclusions fiables à partir d'un corpus franco-belge ou franco-québécois sur le français de France, ni sur le français de Belgique ou u Québec.

### 3.3. Période couverte par les textes du corpus

Le temps joue un rôle important dans l'évolution de la langue. Le français parlé aujourd'hui ne ressemble pas au français parlé il y a 200 ans ni, de façon plus subtile, au français parlé il y a 10 ans, à cause notamment des néologismes et à l'évolution du vocabulaire et aux autres composantes de la langue. C'est un phénomène à prendre en compte

pour toutes les langues vivantes. Un corpus ne doit donc pas contenir de textes rédigés à des intervalles de temps trop larges, sous pine de les dater (pour un usage pour des historiens de la langue ou des concepts).

### 3.4. Registre de langue

IL ne faut pas non plus mélanger des registres différents et le scientifique ne peut s'autoriser à extraire des informations d'un corpus destiné à un certain registre en les appliquant à un autre. Un corpus construit à partir de textes scientifiques ne peut être utilisé pour extraire des informations sur les textes vulgarisés, et un corpus mélangeant des textes scientifiques et vulgarisés ne permettra de tirer aucune conclusion quant à ces deux registres.

## 4. Corpus Anglais

Le premier corpus moderne lisible électroniquement est le Brown Corpus de l'anglais américain standard. Il est constitué d'un million de mots de textes anglais américains imprimés en 1961. Pour que ce corpus soit un bon représentant de la langue, ses textes ont été échantillonnés dans des proportions différentes de 15 catégories différentes de texte: presse, compétences et loisirs, religion, fiction, etc. Dans la norme moderne de corpus, le Brown Corpus est considéré comme faible. Cependant, il est encore utilisé dans l'enseignement et pris comme un modèle pour le développement d'autres corpus comme :

- Lancaster-Oslo-Bergen (LOB) (Johansson et al. 1986).
- Corpus de l'anglais britannique.
- British National Corpus (BNC) (Leech 1993).
- L'ANC (American National Corpus) (Ide 2003)

Outre ces corpus généraux - qui constituent les principales variétés de l'anglais britannique et américain - qui peuvent être utilisés pour la recherche dans divers domaines linguistiques, il y a d'autres corpus qui sont spécialisés.

Compte tenu de la grande valeur des corpus dans la recherche et l'enseignement, de nombreux autres corpus ont été produits pour d'autres langues telles que le français, l'espagnol, l'allemand, le néerlandais, l'arabe, etc.

## 5. Corpus Arabes

L'Arabe est une langue internationale et langue sacrée du Coran. Elle est de ce fait utilisée par plus d'un milliard de musulmans pour pratiquer de leur religion. L'arabe connaît une grande stabilité et c'est l'arabe standard ou littéraire, universellement partagé par les lettrés de tous les pays arabes, qui est utilisé dans la littérature classique, les milieux de l'enseignement, la culture officielle et la presse. Parallèlement à cette lignée, il existe de nombreuses branches s'écartant plus ou moins de la norme. De manière générale, il ya une ignorance généralisée de la langue arabe dans les universités, en raison non seulement de la séparation historique et culturel mais aussi de la complexité de la structure de la langue arabe et son scripte unique. En outre, les progrès ont été entravés par le manque d'outils efficaces tels que les analyseurs morphologiques et la lecture optique de caractères, qui ont un apport considérables dans la construction de corpus.

Des chercheurs arabes et européens qui se sont intéressés à l'étude de l'arabe ont développé plusieurs corpus, qui peuvent être une ressource de recherche importante puisque les recherches sur l'arabe, dans différents domaines, ont besoin d'une enquête solide basée sur de grandes quantités de documents authentiques.

À l'heure actuelle, la recherche sur corpus en arabe est loin derrière celle des langues européennes modernes. Pour autant que nous le sachions, la plupart des études sur l'arabe, jusqu'à présent, ont été basées sur des données plutôt limitées.

Le tableau ci-dessous résume le résultat d'une enquête sur les corpus arabe dans l'ordre dans lequel ils sont apparus [11].

| Nom du Corpus | Source | Moyen | Taille | Objectif | Matériel |
|---|---|---|---|---|---|
| CALLHOME Corpus (1997) | University de Pennsylvania LDC | convers ation | 120 conversations téléphoniques | La reconnaissance vocale produite à partir de lignes téléphoniques | locuteurs natifs égyptiens |
| CLARA (1997) | Charles University, Prague | Écrit | 50M mots | fins lexicographiques | Périodiques, livres, sources Internet i |

| | | | | | |
|---|---|---|---|---|---|
| Egypt (1999) | John Hopkins University | Écrit | Unknown | MT | Un corpus parallèle du Coran en anglais et en arabe |
| Broadcast News Speech (2000) | University of Pennsylvania LDC | parlé | Plus de 110 émissions | Reconnaissance de la parole | Nouvelles diffusion de la radio de la voix de l'Amérique. |
| DIINAR Corpus (2000) | Nijmegen Univ., SOTE TEL-IT, | Écrit | 10M mots | Lexicographie, la recherche générale, | Unknown |
| An-Nahar Corpus (2002) | ELRA | Écrit | 140M mots | la recherche générale | Annahar journal (Liban) |
| Al-Hayat Corpus (2002) | ELRA | Écrit | 18.6M mots | Ingénierie de la Langue et de l'information | Le journal Al-Hayat (Liban) |

**Tableau 1.1 : Corpus Arabe selon leur ordre d'apparition**

## 6. Utilisation des corpus

Le corpus est une ressource à utiliser pour l'étude des phénomènes linguistiques ainsi que pour l'enseignement de la langue. Pour ce faire, de plus en plus d'études récentes sur la langue ont adopté une approche fondée sur des corpus qui peut être décrit comme étant quantitatives ainsi que qualitatives.

La technique quantitative peut être considérée comme une partie essentielle de l'analyse sur le corpus. Lorsque l'on compare l'utilisation de deux mots ou deux structures, il ne suffit pas d'affirmer leurs caractéristiques contextuelles, mais aussi de calculer le nombre de leurs occurrences ou de leur co-occurrente avec d'autres mots.

La méthode quantitative nécessite le calcul des statistiques pour évaluer l'importance de la fréquence. Cela peut paraître trop compliqué, mais puisque nous avons les données sur ordinateur, nous pouvons utiliser des outils pour nous aider à obtenir le résultat dont nous avons besoin.

Certains programmes appelés **concordanciers** qui font la tâche de rechercher, trier et classer ont été conçus pour nous aider à manipuler les données. Les concordanciers sont des outils

utilisés pour rechercher dans un corpus tout type d'informations linguistiques tels que le sens des mots ou des phrases lexicales et grammaticales. Aujourd'hui, plusieurs concordanciers sont disponibles. Certain sont commerciaux tels que WordSmith, MonoConc et ParaConc. D'autres sont libres comme ConcApp, Wconcord et AntConc. Ces outils fonctionne parfaitement bien sur l'anglais et d'autres langues avec l'alphabet latin mais jusqu'à maintenant, il n'y a aucun outil efficace disponible pour le traitement de l'arabe.

L'utilisation de corpus et de concordanciers dans l'enseignement de l'anglais et d'autres langues latin a été démontrée. On pense que l'utilisation de ces ressources et outils est très importante pour l'enseignement de l'arabe à des apprenants étrangers mais malheureusement, l'utilisation de ces ressources dans l'enseignement de l'arabe est encore très limites.

## 7. Cible des données

### 7.1. Corpus de Latifa AL-Sulaiti

Le corpus que nous envisageons de construire dans le cadre de notre projet se base sur un corpus arabe « noyau » créé par Latifa Al-Sulaiti et Eric Atwell (2003) à l'université de Leeds. [1] Ce corpus, comme le souligne ses concepteur, est un corpus de l'arabe contemporain librement accessible, qui inclue non seulement des textes et des extraits d'enregistrements oraux de l'arabe standard mais aussi des échantillons de variétés familières. Ses textes ont été recueillis auprès de quatre sources principales : magazines, journaux, sites web et la radio.

Le but de ce corpus a été de développer un prototype pour un corpus de l'arabe contemporain. Les utilisateurs cibles de ce corpus sont des professeurs de langues, des ingénieurs de la langue, des apprenants étrangers de la langue arabe et des écrivains.

Le corpus contient a peut près un million de mots. Il a été compilé en couvrant les principales catégories d'utilisateurs. Ce corpus contient 84,268,40 mots et 415 textes dans certaines catégories identifiées par les professeurs de langues et les ingénieurs de langue.

## 7.2. Sources du corpus

Aujourd'hui, dans la plupart des pays arabes, les sociétés d'édition produisent une grande quantité de matériel sur le web. Ainsi, il existe un nombre croissant de textes disponibles sous forme lisible par machine sur un large éventail de sujets.

Le corpus de Latifa Al-Solaiti utilise des textes tirés principalement à partir de sites web. Un site web est considéré comme une source après avoir eu l'autorisation d'utiliser des échantillons de ses textes. Voici la liste des sites qui ont accordé des autorisations d'utilisation dans le cadre du projet de Latifa Al-Solaiti :

| sites web | Description |
|---|---|
| http://www.alarabimag.com | Majallat al-Arabi (Kuwait) |
| http://www.ofouq.com | Majallat Ofouq (Saudi Arabia) |
| http://www.arabicstory.net | Al-qissa Al-Arabiya site |
| http://www.pcmag-arabic.com | Majallat PC Al-Arabiyya (UAE) |
| http://www.BBCArabic.com | Arabic BBC site (UK) |
| http://www.sayidaty.net | Majallat Sayyidaty (UK) |
| http://www.ecoworld-mag.com | Majallat Arabia 'aalam Al-'iqtisaad ( Saudi) |
| http://www.nizwa.com/ | Majallat Nizwa (Oman) |
| http://arabmedmag.com | Al-Dawriyya Al-Tibbiyya Al-Arabiyya (Syria) |
| http://aklaat.com/ | Aklaat site (UAE) |
| http://www.islamonline.net | Islam on line site (Qatar) |
| http://www.alraialaam.com | Al-Raay Al-'aam newspaper (Kuwait) |
| http://www.almarefah.com | Majallat Al-Ma'rifa: (Saudi Arabia) |
| http://www.akhbarelyom.org/akhersaa | Majallat Akhir Saa'a (Egypt) |
| http://www.arabcomputing.com | Al-Kumpyuter fi Al-'aalam Al-Arabi (UK) |
| http://www.alamalcomputer.com | aalam Al-kumpyuter (Egypt) |
| http://www.raya.com | Al-Raya Newspaper (Qatar) |
| http://www.kisr.edu.kw/science/ | uluum wa tuknologia (Kuwait) |

**Tableau 1.2 : Sites web donnant certains droits d'auteur à Latifa Al-Solaiti**

Les textes écrits étant obtenus à partir de ces sites, il ya en outre des fichiers oraux qui sont obtenus à partir de Radio Qatar. Mais le nombre de ces fichiers est très faible, car la saisie de ces types de fichiers prend du temps et de la compétence.

### 7.3. Encodage du corpus

Après obtention des autorisations des droits d'auteur appropriée, les textes ont ensuite été inclus dans le corpus et annotés manuellement en utilisant l'éditeur Unicode UNIRED. Le corpus a été codé avec une sorte de mark-up langue pour permettre à l'utilisateur d'extraire cette information. Les informations qui sont codées incluent des fonctionnalités linguistiques et non linguistiques. Ils sont tels que:

- Paragraphes, sections, rubriques, des phrases.
- Limite de la partie du discours de chaque mot.
- Tournure du discours, pause.
- caractéristiques paralinguistiques tels que le rire et l'hésitation.
- informations méta-textuelle comme la source du texte, l'auteur, la maison d'édition, etc.

Un balisage XML a été utilisé dans le corpus. Pour atteindre cet objectif, il a eu un déploiement d'un en-tête avec les éléments suivants:

- Description du fichier
- Description de l'encodage
- Description du profil.

### 7.4. Procédure d'encodage du corpus

Les annotations d'un grand corpus se fait généralement au moyen de certains programmes d'ordinateur afin que ces nombreux textes peuvent être annotées dans un laps de temps court. Il est normalement effectué automatiquement et s'il y a des erreurs elles seront corrigées manuellement. Il n'y a aucun moyen pour élaborer un programme pour encoder le corpus automatiquement, par conséquent, l'encodage a été effectué manuellement.

## 7.5. Classification du texte

Pendant l'encodage d'un type de texte il est parfois difficile de décider a quel catégorie ce texte appartient et a quel domaine. beaucoup de classification de texte est basé sur le sujet tel qu'il est représenté dans la source, le corpus de latifa al-soliti divisé en deux partie (ecrit , parlé ) et seize catégorie

| Catégorie | Nombre fichier | Nombre de mots |
|---|---|---|
| Autobiography | 73 | 153,459 |
| Short Stories | 31 | 45,460 |
| Children's Stories | 27 | 21,958 |
| Economics | 29 | 67,478 |
| Education | 10 | 25,574 |
| Health and Medicine | 32 | 40,480 |
| Interviews | 24 | 58,408 |
| Politics | 9 | 46,291 |
| Recipes | 9 | 4,973 |
| Religion | 19 | 111,199 |
| Sociology | 30 | 85,688 |
| Science | 45 | 50,219 |
| Sports | 3 | 8,290 |
| Tourist and Travel | 61 | 46,093 |
| Spoken (Sports, entertainment, education) | 7 | 5,605 |

**Tableau 1.3 : Les catégories de corpus Latifa Al-Solaiti**

## 8. Conclusion

Dans ce chapitre nous avons définis la notion de corpus. En partant d'une définition très générale couvrant plusieurs aspects nous avons donné une définition des corpus s'appliquant aux sources web. Nous avons ensuite décrit les différents typologies et caractéristiques des corpus. Enfin, nous avons identifié la cible des données, le corpus sur lequel porte notre projet (le corpus de Latifa AL-Sulaiti).

Cette première partie de l'état de l'art nous a permit de décrire les outils de notre travail et sur lequel notre conception sera implémentée. Dans le chapitre suivant, nous nous intéressons à la source des données et les sites web source.

# Chapitre 2

# Web

## 1. Introduction

Nous avons consacré ce deuxième chapitre au Web comme étant la source des données textuelles avec lesquelles nous comptons alimenter notre corpus d'une façon automatique. Le Web, mais aussi Internet d'une manière générale offre une source inépuisable de données. Ces données sont emmagasinées de différentes façons. Les sites web, qui sont constitués à leur tour d'un ensemble de pages web, sont les endroits les plus appropriés pour proposer et communiquer ces données. Ainsi, faudrait-il connaître comment est organisé le Web dans ses moindre détails pour pouvoir récupérer ce dont nous avons besoin. Par conséquent, le texte de ce chapitre passera en revue tous les outils dont nous aurons besoin pour comprendre l'environnement où sont stockées les données textuelles sur le Web, en particulier les notions de sites et de pages web qu'il est primordial de maitriser. L'objectif étant de maitriser la construction, mas surtout la déconstruction, d'un site ou d'une page web pour pouvoir en récupérer les données dont nous avons besoin. Ça sera aussi l'occasion de donner des exemples de sites web qui seront utilisés dans notre projet comme source de données et qu'il serait nécessaire d'analyser.

## 2. Internet

Internet est un réseau informatique mondial constitué d'un ensemble de réseaux nationaux, régionaux et privés. L'ensemble utilise un même protocole de communication : TCP/IP (Transmission Control Protocol/Internet Protocol). [URL 3].

Internet propose trois types de services fondamentaux :

- le courrier électronique (e-mail).
- le Web (les pages avec des liens et contenus multimédia de ses sites Web).
- l'échange de fichiers par FTP (File Transfer Protocol).

## 3. World Wide Web


**Figure 2.1: Logo du World Wide Web**

## 3.1. Web

Le World Wide Web (ou WWW), littéralement la « toile (d'araignée) mondiale », communément appelé le Web (et parfois la Toile) est un système hypertexte publique fonctionnant sur Internet. Le Web permet de consulter, avec un navigateur, ou bien un protocole http, des pages accessibles sur des sites. L'image de la toile d'araignée vient des hyperliens qui lient les pages web entre elles [URL 1].

### 3.2. Hypertexte

Un système hypertexte est un système contenant des nœuds liés entre eux par des hyperliens permettant de passer automatiquement d'un nœud à un autre.

### 3.3. Hyperlien

C'est un élément d'une page web qui, lorsque l'internaute clique dessus, dirige celui-ci vers une autre page web. Il est associé à une URL. Seule la ressource à la source contient les données définissant l'hyperlien [URL 1]. Les liens sont à la base du web : c'est grâce à eux que les internautes peuvent naviguer d'une page à une autre, et ainsi explorer le World Wide Web.

### 3.4. HTTP (HyperText Transfer Protocol)

Le HTTP est le protocole de communication communément utilisé pour transférer les ressources du Web. Le HTTPS en est une variante muni d'une authentification et d'un chiffrement.

### 3.5. URL (Uniform Resource Locator)

URL signifie littéralement « localisateur de ressource uniforme ». Il s'agit d'une chaîne de caractères décrivant l'emplacement d'une ressource. Elle prend la syntaxe suivant :

a. Une indication du protocole de communication, le plus souvent *http://* pour les serveurs web.
b. Un sous domaine cible, si applicable. Par exemple : *www*.
c. Un nom de domaine.
d. Le chemin de la ressource, par exemple : /mapage
e. D'éventuels paramètres ou données supplémentaires, utilisés par exemple pour effectuer une recherche ou pointer vers un signet d'une page web.

Exemple d'url : http://www.unsiteweb.com/mapage?s=3.

### 3.6. HTML (HyperText Markup Language)

HTML et XHTML (Extensible HyperText Markup Language) sont des langages informatiques permettant de décrire le contenu d'un document (titres, paragraphes, disposition des images, etc.) et d'y inclure des hyperliens. Un document HTML est un document décrit avec le langage HTML.

### 3.7. Client-serveur

Dans un mode de communication client-serveur, un serveur est un hôte sur lequel fonctionne un logiciel serveur auquel peuvent se connecter des logiciels clients fonctionnant sur des hôtes clients, un seul hôte peut contenir les deux.

## 4. Site web

### 4.1.Definition

Un site web - ou simplement site - est un ensemble de pages web et de ressources liées accessible à travers une URL (ou adresse web). Un site est hébergé sur un serveur web accessible via le réseau mondial Internet ou un intranet local [URL 4]. L'ensemble des sites web constituent le World Wide Web.

### 4.2.Page web

Une page web est une ressource informatique. Elle est conçue pour être consultée à l'aide d'un navigateur web et possède une adresse web. Techniquement, une page web est souvent constituée d'un document en HTML (Hypertext Markup Language) formé de textes et d'images. D'une manière générale, tout type de ressources ou d'assemblage de ressources textuelles, visuelles, sonores ou logicielles peuvent constituer une page web.

### 4.3. Type de sites web

On distingue deux types de sites web : les sites web statiques et les sites web dynamiques.

#### 4.3.1. Sites web statiques :

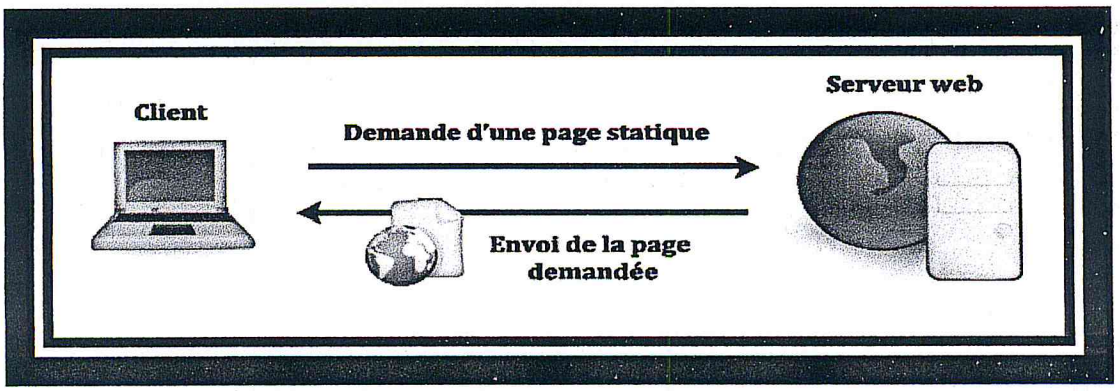


**Figure 2.2: Site web statique**

Ces sites ont un mécanisme de fonctionnement très plus simple : une URL qui correspond à un fichier est envoyée par un navigateur à un serveur web qui renvoie à son tour le fichier demandé. Le contenu des pages d'un site statique ne dépend donc pas de variables telles qu'une date ou une base de données. Pour changer le contenu d'une page, il est nécessaire de changer le contenu du fichier.

En outre, les visiteurs peuvent seulement voir le contenu du site mais sans y participer. Pour réaliser des pages, des langages dits d'interface utilisateur sont nécessaires, à savoir HTML, CSS et JavaScript. Dans certain cas, le HTML peut être suffisant pour réaliser des pages web simples.

### 4.3.2. Sites web dynamiques :

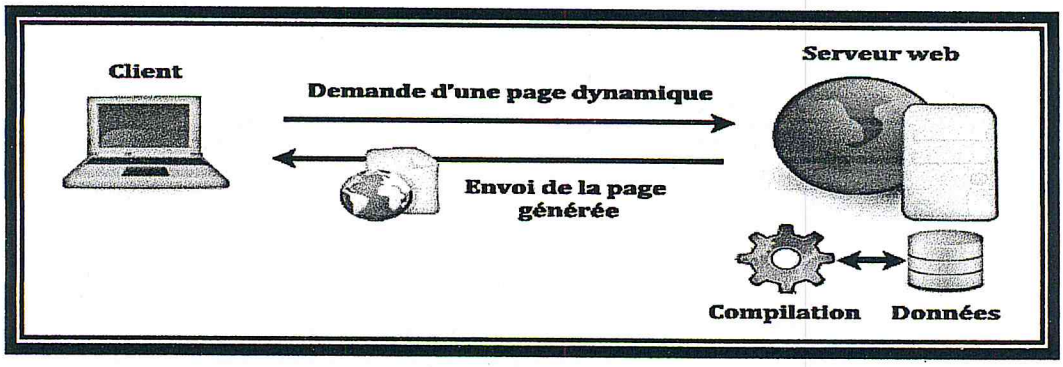


**Figure 2.3: Site web dynamique**

Ces sites offrent un contenu qui peut évoluer dans le temps. Des programmes tournent du côté des serveurs, en arrière plan, pour générer les pages du site. Ces programmes peuvent se servir de bases de données ou autres sources de données pour composer les pages qui seront affichées dans le navigateur.

Ce dynamisme apporte des fonctionnalités que ne peuvent offrir les sites web dits statiques. Par exemple les visiteurs peuvent y participer (commentaires sur un blog, changement du contenu des pages, etc.). Ces sites ont ainsi pratiquement supplanté les sites statiques dès le début des années 2000.

Pour réaliser ce type de sites, nous avons également besoin de HTML, de CSS et de JavaScript. Les programmes qui tournent du côté serveur utilisent aussi d'autres langages qui peuvent créer dynamiquement des pages, en analysant les requêtes des visiteurs pour ensuite fabriquer une réponse adaptée.

Il existe plusieurs langages pour créer ce type de pages : PHP, Java, C#, Ruby, C++, Python, Visual Basic, etc.

## 4.4.Création d'un site web

La création d'un site web est un projet à part entière comprenant un grand nombre de phases que nous pouvons résumer en :

- Une conception, représentant la formalisation de l'idée,
- Une réalisation, correspondant au développement du site web (consiste à créer des fichiers HTML),
- Un hébergement, se rapportant à la mise en ligne du site, de manière permanente (chez un serveur web connecté).

**Faut-il connaître le HTML pour faire un site web ?**

Les CMS modernes permettent de faire des pages web en utilisant des éditeurs de texte simplifiés (dits WYSIWYG). Il est donc possible de rédiger des pages sans passer par la phase html. Par contre, pour celui qui souhaite maîtriser la structure de ses pages, il est indispensable de savoir lire et écrire du html et de corriger les éventuelles lacunes des éditeurs WYSIWYG, qui offrent toujours la possibilité de visualiser un code source html. [URL 4].

### 4.5. Code source

On appelle code source d'une page web le code informatique envoyé au navigateur pour qu'il affiche la page.

Pour visualiser le code source d'une page avec un navigateur (on prend chrome comme exemple), on doit faire un clic droit à l'intérieur d'une page web (en dehors d'un paragraphe, d'une image ou d'une vidéo), ensuite choisir « Afficher le code source de la page ». Nous pouvons également choisir dans le même menu « Inspecter l'élément » pour ouvrir l'outil d'exploration intégré au navigateur.

#### 4.5.1. Contenu du code source

Vous trouverez ci-dessous quelques exemples d'informations disponibles en consultant le code source d'une page web.

- **Avec quoi est fait le site ?**

  En recherchant dans le texte du code source la balise <meta> generator, vous verrez parfois avec quel outil (par exemple quel CMS) le site web que vous analysez a été fait, tout comme la version du CMS utilisé.

- **Quel outil de tracking est utilisé ?**

  En faisant une recherche sur le terme « analytics » dans le code source de la page, vous pouvez savoir si ce site utilise par exemple le système de tracking statistique de Google.

- **Où sont stockées les images du site ?**

  Peut être utile pour, par exemple, pour récupérer des images dans leur résolution d'origine, qui sont bien souvent redimensionnées dans le navigateur.

- **Quel est le niveau d'optimisation du référencement naturel ?**

  L'analyse des balises Title, Description et Keywords sont utiles pour évaluer le niveau d'optimisation (SEO) du site ainsi que les expressions clefs sur lesquels votre concurrent (ou futur client) essaye de se positionner.

# 5. HTML

## 5.1.Définition

HTML est un langage de description de format de document qui se présente sous la forme d'un langage de balisage dont la syntaxe vient du Standard Generalized Markup Language (SGML).

## 5.2.Encodage du HTML (jeux de caractères)

Pour afficher une page HTML correctement, un navigateur Web doit connaître le jeu de caractères (ou l'encodage de caractères) à utiliser. Ceci est spécifié dans la balise <meta>. Par exemple : <meta charset="UTF-8">

## 5.3. Catégories de contenus des éléments HTML

Chaque élément HTML doit respecter un certain nombre de règles définissant le type de contenu qu'il peut avoir. Ces règles sont regroupées dans des modèles de contenus, communs à plusieurs éléments. Chaque élément HTML appartient à zéro, un, ou plusieurs modèles de contenus. Chacun de ces modèles définit un ensemble de règles devant être respectées par le contenu de l'élément pour que le document HTML soit conforme.

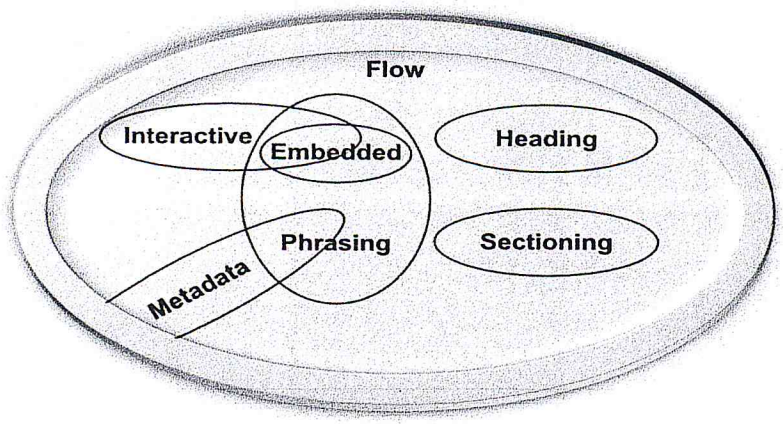


**Figure 2.4: Catégories de contenus des éléments HTML**

### 5.3.1. Contenu de méta-données

Les éléments appartenant à cette catégorie modifient la présentation ou le comportement du reste du document. Ils insèrent des liens vers d'autres documents ou comportent des informations sur la structure même des données.

### 5.3.2. Contenu de flux

Les éléments appartenant à la catégorie de contenu de flux contiennent généralement du texte ou du contenu intégré.

### 5.3.3. Contenu sectionnant

Les éléments appartenant à cette catégorie sont ceux créant une nouvelle section dans le plan du document qui définit la portée des éléments <header>, des éléments <footer> et du contenu de titre.

### 5.3.4. Contenu de titre

Le contenu de titre définit le titre d'une section, qu'elle soit marquée par un contenu sectionnant de manière explicite ou qu'elle soit définie de manière implicite par le contenu de titre lui-même.

### 5.3.5. Contenu de phrasé

Le contenu de phrasé définit le texte et le balisage qu'il contient. Des séquences de contenu de phrasé constituent des paragraphes.

### 5.3.6. Contenu intégré

Le contenu intégré importe une autre ressource ou intègre du contenu provenant d'un autre langage de balisage ou d'un autre espace de noms dans le document.

### 5.3.7. Contenu interactif

Le contenu interactif regroupe des éléments spécialement conçus pour une interaction avec l'utilisateur.

### 5.3.8. Contenu tangible

Un contenu peut être dit tangible lorsqu'il n'est ni vide ni caché. Les éléments dont le modèle de contenu est celui de flux ou de phrasé devraient toujours avoir au moins un nœud dont le contenu est tangible. Cette recommandation n'est cependant pas obligatoire.

### 5.3.9. Contenu associé aux formulaires

Le contenu associé aux formulaires contient des éléments possédés par un formulaire, exposé avec un attribut **form**. Être possédé par un formulaire signifie être descendant d'un élément <form> ou d'un élément dont l'identifiant est référencé par la valeur de l'attribut **form**.

## 6. Web scraping

Le web scraping est une technique d'extraction de contenu de sites Web, via un script ou un programme, dans le but de le transformer pour permettre son utilisation dans un autre contexte. Cette opération se pratique le plus souvent de façon automatique, ce qui permet de constituer des pages à bon compte. Le web scraping est le processus de collecte automatiquement des informations sur le World Wide Web. Il s'agit d'un domaine avec des développements actifs partageant un objectif commun avec le web sémantique.

Parfois, même la meilleure technologie de web scraping ne peut pas remplacer l'examen manuel d'un être humain et des copier-coller, et parfois cela peut être la seule solution fiable lorsque les sites Web mettent en place des barrières pour empêcher l'automatisation des machines.

Il est souvent utile de récupérer automatiquement des données à partir d'une page web, en analysant son code HTML pour extraire des informations qui nous intéressent. Si nous n'utilisons pas les outils adéquats, l'écriture du code pour faire ce genre de chose peut vite devenir fastidieuse.

## 7. Source principale des données

### 7.1.ArabicStory

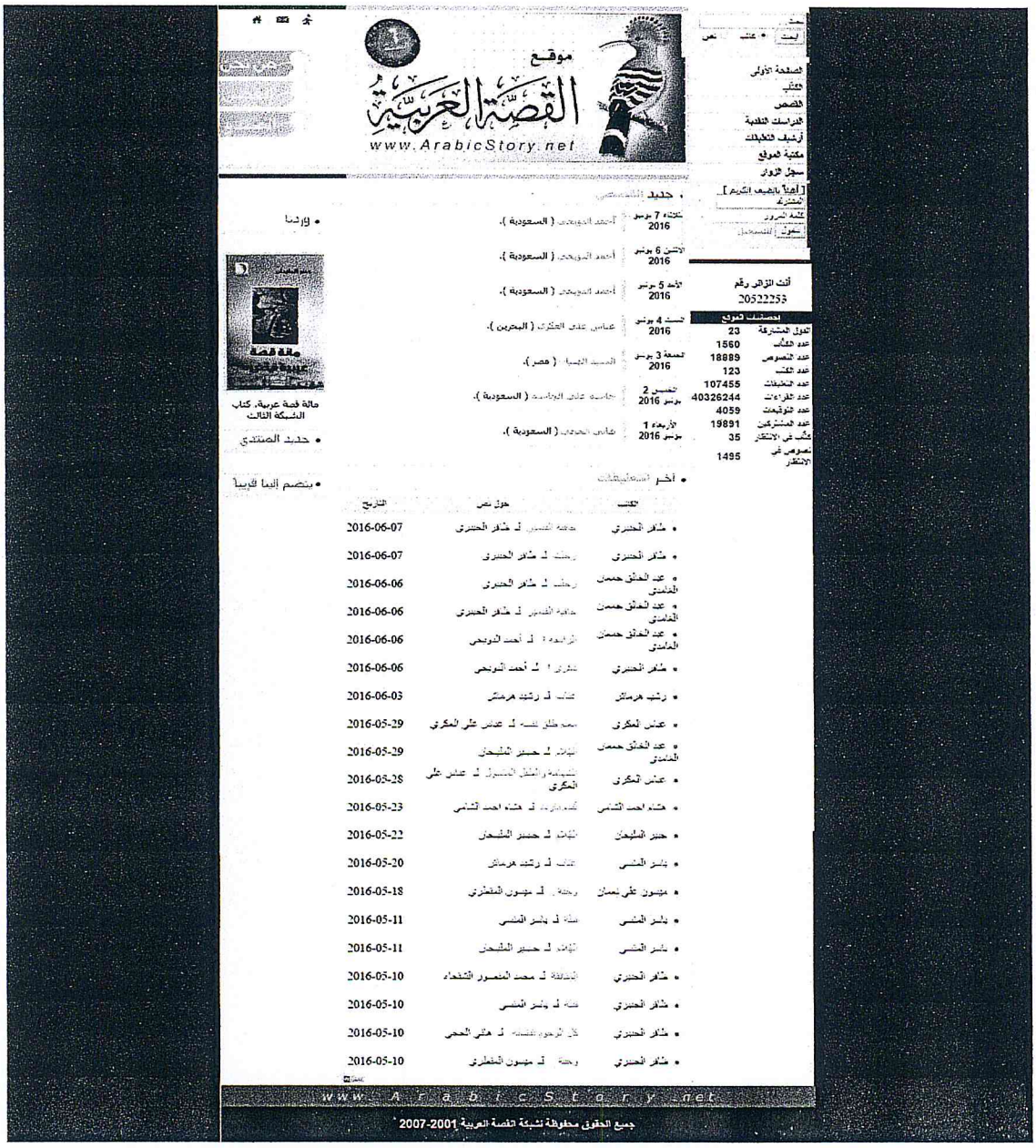


**Figure 2.5: Le site web ArabicStory**

**URL :** http://www.arabicstory.net/?p=home

**Propriétaire :** Jubair almelaihan

**Création** : 9 mai 2000

**Type du site :** Le site est spécialisé principalement dans l'édition d'histoires courtes et très courtes en arabe (Short Stories) et de faire connaître sa lettre.

| Statistiques du site | |
|---|---|
| Pays participants | 23 |
| Nombre de livres | 1560 |
| Nombre de textes | 18889 |
| Nombre de livres | 123 |
| Nombre de commentaire | 107455 |
| Nombre de Lectures | 40326276 |
| Le nombre de signatures | 4059 |
| Le nombre d'abonnés | 19891 |
| Livre en attente | 35 |
| Textes en attente | 1495 |

**Tableau 2.1 : Statistiques sur ArabicStory**

### 7.1.1. Structure du site

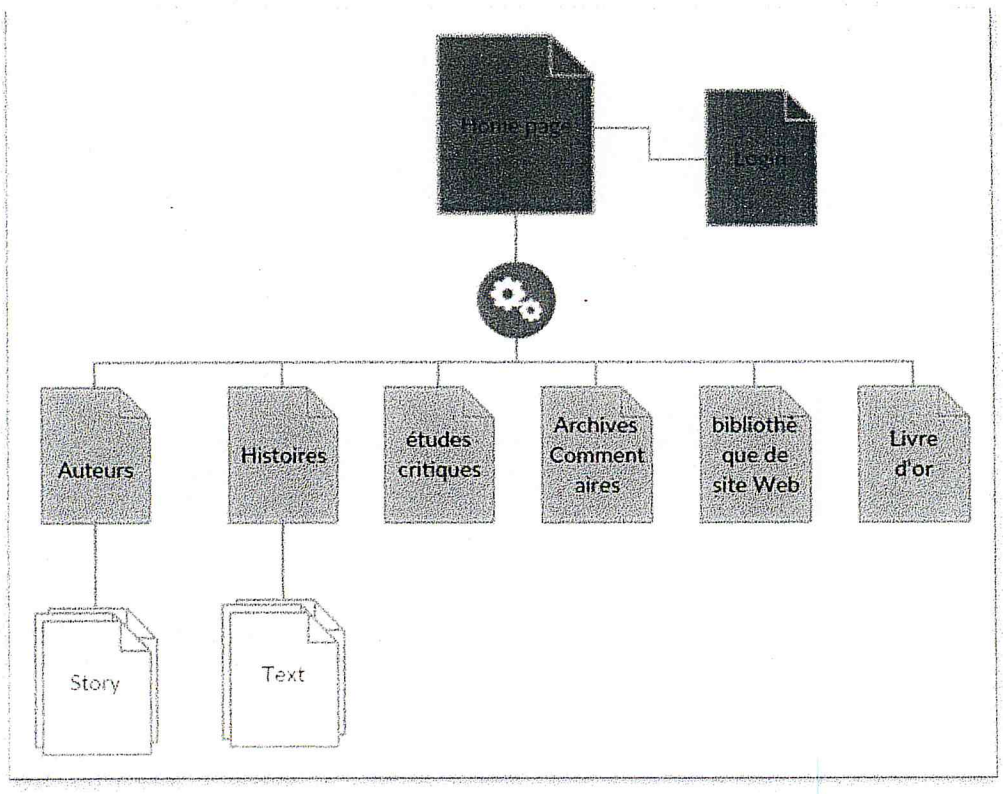


**Figure 2.6: Structure de ArabicStory**

➢ **Page d'un auteur**

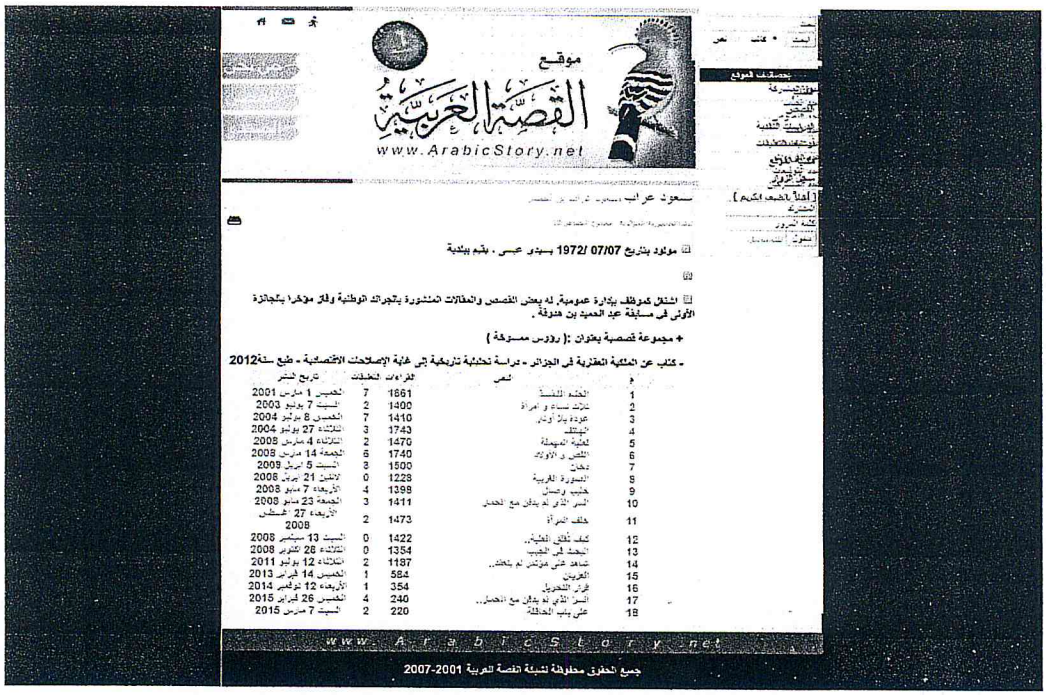


**Figure 2.7: La page d'un auteur de ArabicStory**

Elle contient les information personnel de l'auteur tel que le nom, prénom, date et lieu de naissance nationalité et résidence. En plus, un CV de l'auteur est parfois ajouté. Finalement et principalement, cette page contient l'ensemble des histoires disponibles pour cet auteur fournies avec leurs dates de publication respectives.

➢ **Page d'une histoire**

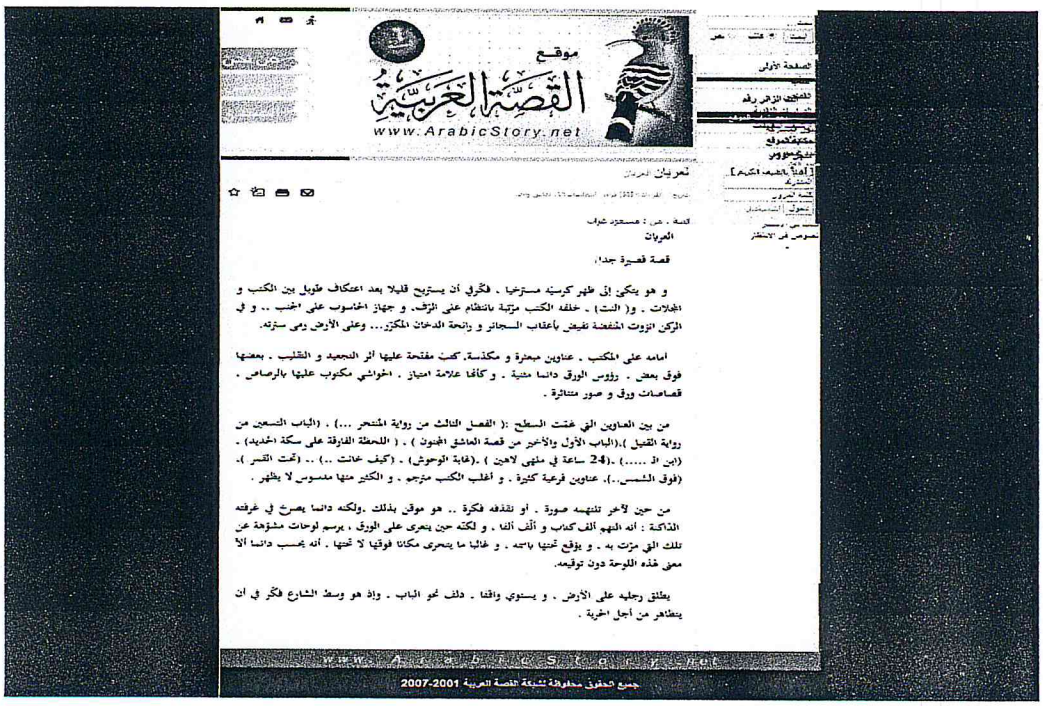


**Figure 2.8 : Page d'une histoire dans ArabicStory**

Cette page contient principalement le texte d'une histoire plus un certain nombre d'informations concernant l'histoire telles que le titre et le nom de l'auteur.

## 8. Conclusion

Pour extraire un texte du Web, il faut être en mesure de connaitre où et par quel moyen le trouver ensuite comment et par quel moyen le récupérer. Ce chapitre a couvert plusieurs aspects du web qui nous ont permit de répondre à ces questions fondamentales quant à la réalisation de notre projet. En outre, nous avons présenté d'une manière sommaire notre principale source de donnés : le site ArabicStory.

# Chapitre 3

# Conception

## 1. Introduction

Afin de garantir le développement efficace des applications de qualité, de bonnes pratiques doivent impérativement être adoptées. Le processus unifié semble être la solution idéale pour les concepteurs et les développeurs. Il regroupe les activités à mener pour transformer les besoins d'un utilisateur en un système logiciel quelque soit la complexité du projet, la taille et le domaine d'application du nouveau système.

Dans ce chapitre, nous présentons la spécification et l'analyse des besoins ainsi que la conception de l'application. La réalisation sera présentée dans le chapitre suivant.

## 2. Architecture de l'application

Nous présentons l'architecture globale de l'application, la stratégie de travail et la manière d'interaction entre les différentes parties du système afin de produire les fonctionnalités attendues de l'application. Cette architecture peut être synthétisée selon la figure suivante :

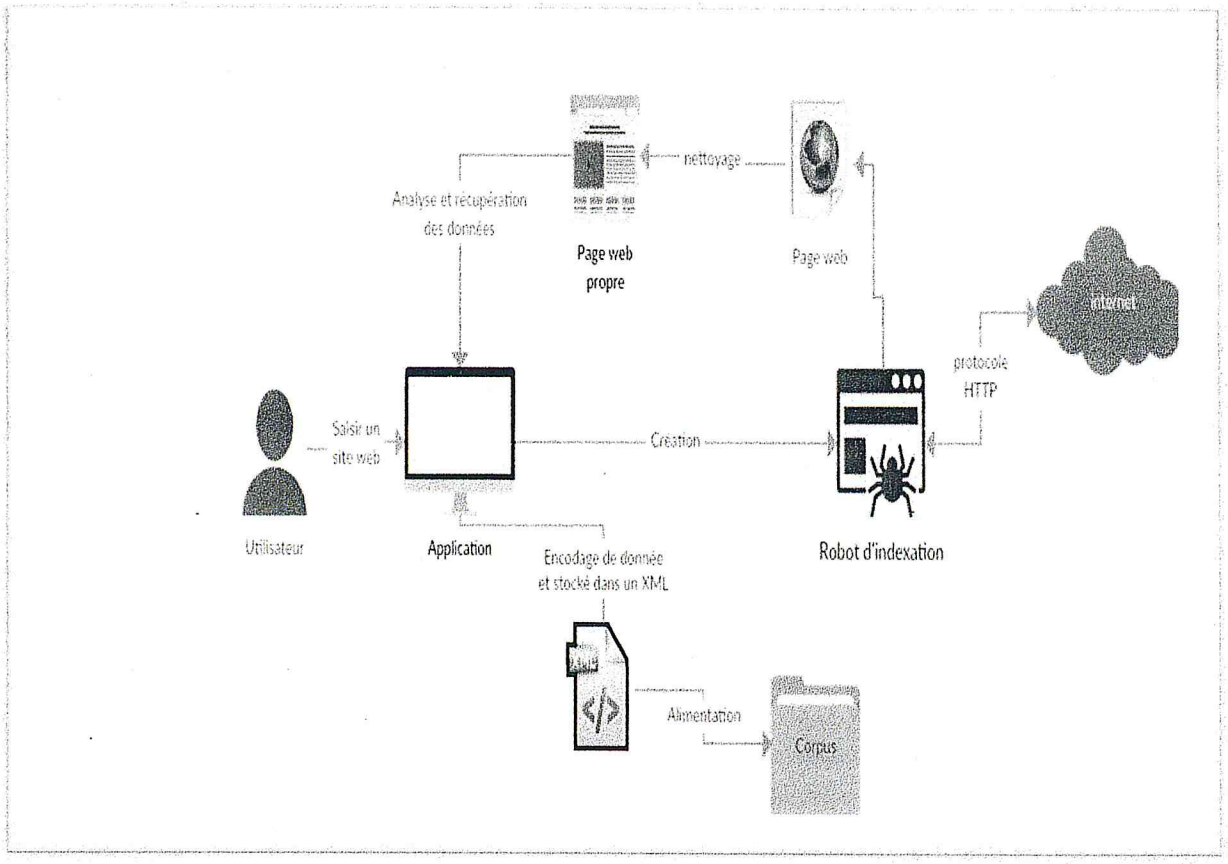


**Figure 3.1 : Architecture globale de l'application**

Notre stratégie est basée sur cinq grandes étapes :

- considérer le web comme une source de donnée (sélectionner la bonne source),
- utiliser des robots d'indexation (ou crawlers) pour faire le parsing,
- nettoyer les pages collectées par le crawler,
- encoder,
- stocker les différentes données (sous le format XML).

## 2.1.Source du corpus : le World Wilde Web

L'utilisation du Web comme base pour la constitution de ressources textuelles est très récente. Ces dernières années sont le témoin sur les travaux tentant d'exploiter ce type de données. Avec le développement d'Internet et de ses services, le Web est devenu une immense source de documents de toutes natures (textes, images, sons et vidéo). Ce qui fait que les contenus disponibles sur Internet soient actuellement le réservoir textuel le plus important pour l'humanité. Ce qui facilite une construction rapide de corpus de tous genres.

Il existe deux moyens pour constituer un corpus à partir du Web [8] :

- l'utilisation d'un aspirateur Web. En d'autres termes, un outil qui permet de récupérer un ensemble de pages à partir d'une adresse. Cette méthode est très rapide mais n'est utilisée que si les données qui constitueront le corpus sont regroupées sur un ensemble de sites connus au préalable.
- l'interrogation automatique de moteurs de recherche pour effectuer la sélection d'un certain nombre d'adresses. Il s'agit ensuite de récupérer manuellement ou automatiquement les pages correspondantes.

## 2.2.Robot d'indexation

Un robot d'indexation (ou littéralement araignée du Web, en anglais web crawler ou web spider) est un logiciel ou programme qui explore automatiquement le Web. Il est capable de télécharger en totalité la partie visible d'un site Internet, mais il est généralement conçu pour collecter des ressources spécifiques (pages web, images, vidéos, documents Word, PDF ou PostScript, etc.) afin de permettre à un moteur de recherche de les indexer[URL1]. Ces indexes peuvent ensuite être utilisés par un moteur de recherche pour trouver rapidement toutes les pages Web parlant contenant un mot précis. Pour comprendre le fonctionnement d'un index, il suffit de penser à l'index d'un livre que l'on y trouve souvent à la fin. C'est une liste de mots classés par ordre alphabétique. À côté de chaque mot sont écris les numéros des

pages du livre où il se trouve. Il s'agit donc d'un moyen très rapide pour trouver un contenu rapidement et d'une manière précise parmi une grande quantité d'information.

**Comment les moteurs de recherches font-ils pour indexer les pages des sites web ?**

Les moteurs de recherche agissent comme les humains : ils se rendent sur les pages, les lisent et suivent les liens hypertextes. Ils procèdent alors en suivant récursivement les hyperliens trouvés à partir d'une page pivot. Ceci leur permet, de fil en aiguille, de visiter des milliers, des millions puis des milliards de pages web. Par la suite, il est avantageux de mémoriser l'URL de chaque ressource récupérée et d'adapter la fréquence des visites à la fréquence observée de mise à jour de la ressource. Toutefois, de nombreuses ressources échappent à cette exploration récursive, les hyperliens créés à la demande sont introuvables par un robot. Cet ensemble de ressources inexploré est parfois appelé le web profond.

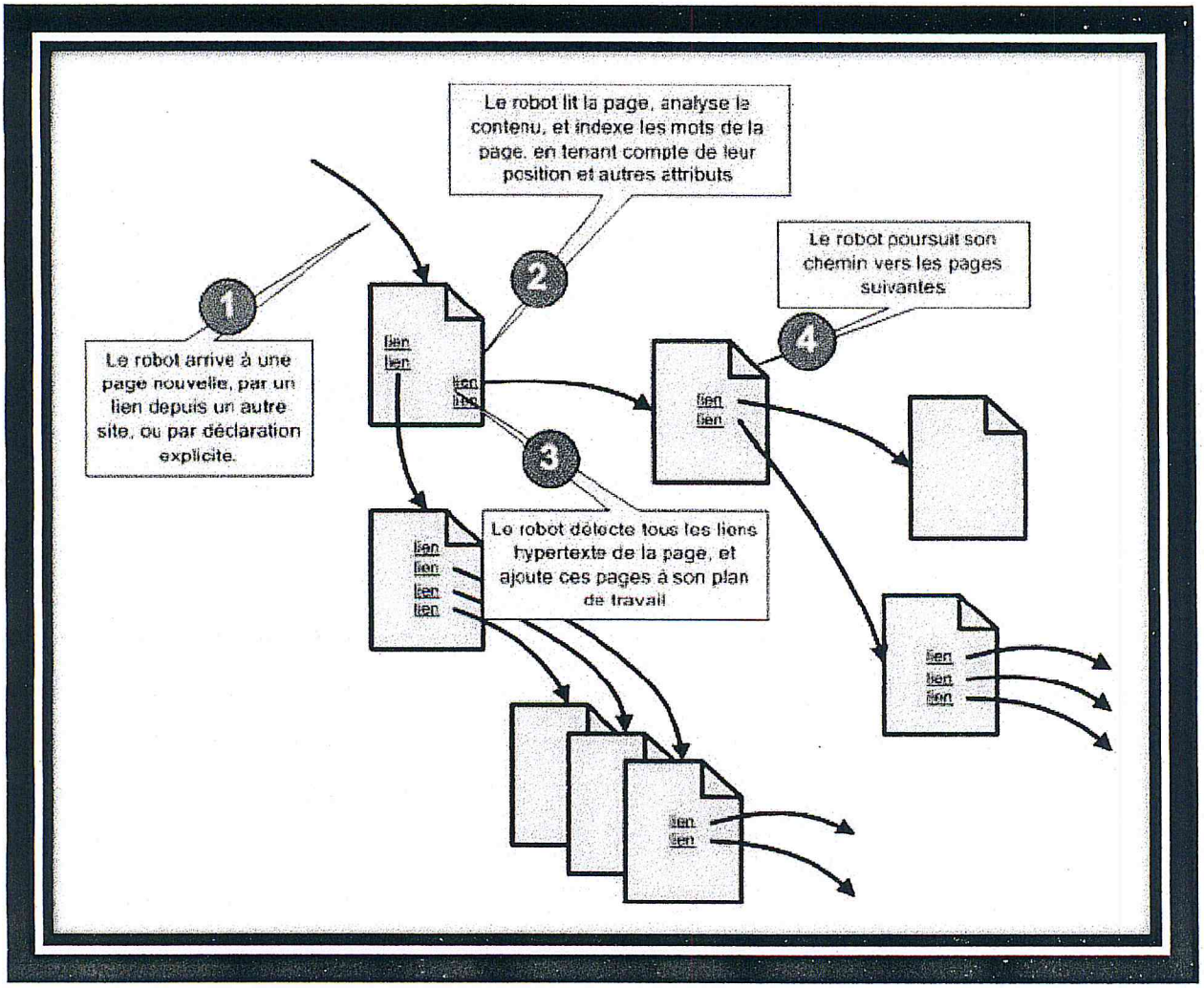


**Figure 3.2 : Principe de fonctionnement d'un robot d'indexation**

**La différence avec nous ?**

Les moteurs de recherche ont une capacité de mémorisation infinie et une vitesse de lecture très importantes ; comme indice, ils lisent un texte de 2000 mots en quelques millisecondes. Leur « lecture » consiste à collecter les informations des pages sous forme de métadonnées, puis les stocker dans une base de données. C'est cette base de données qui sera ensuite consultée pour chaque recherche effectuée par un internaute (utilisateur) sur un moteur.

## 2.3.Nettoyage

Pour obtenir un corpus propre et exploitable, il est nécessaire de nettoyer les pages récupérées. Ce nettoyage est une opération informatique qui demande un paramétrage souvent poussé. Les pages collectées par le crawler (ou robot d'indexation) ne peuvent pas être immédiatement soumises à une analyse linguistique. Il faut d'abord filtrer les informations qui sont inutiles et qui, surtout, risquent de fausser une telle analyse. En effet, la plupart des pages contiennent des quantités d'informations sans intérêt par rapport à une problématique donnée. Celles-ci peuvent polluer les résultats si elles ne sont pas éliminées (publicités, liens vers d'autres pages, menus actifs, liens sponsorisés, ...). Ce filtrage ne peut pas s'opérer manuellement (trop grandes quantités de données) et doit être paramétré différemment en fonction des types de sites (en fonction des types de pages, on n'aura pas le même positionnement des encarts publicitaires, etc.). Pour chaque type de site, il est possible de définir un paramétrage particulier. Il existe des robots ayant des fonctionnalités de filtrage des pages web.

Pour garantir un bon nettoyage, il faut répondre à un certain nombre de questions :

- Comment détecter et se débarrasser des barres de navigation, en-têtes, pieds de pages et autres données textuelles n'ayant pas d'intérêt linguistique ?
- Comment identifier les paragraphes ?
- Comment standardiser le contenu pour que le texte soit exploitable pour une analyse linguistique ?

## 2.4.Encodage

Nous avons étudié deux types d'encodage : l'Unicode et l'UTF-8. L'UF-8 n'est autre qu'une version simplifiée et la plus populaire de l'Unicode.

### 2.4.1. Codage universel : Unicode

Il existe un encodage qui essaye de regrouper toutes les langues du monde. Ce codage est connu sous la dénomination Unicode. Unicode est un tableau gigantesque qui contient des combinaisons de 1 et de 0 d'un côté, et les lettres ou syllabes, chiffres ou nombres, symboles divers, signes diacritiques et signes de ponctuation et les caractères de toutes les langues possibles de l'autre : arabe, latin, chinois, français, espagnol, russe… mais aussi toutes les langues non naturelles telles que les langues de signes.

Certes, Unicode ne contient pas encore absolument tous les caractères possibles et imaginables, mais il couvre suffisamment de terrain pour éliminer 99.99% des problèmes de communications de texte entre machines dans le monde actuel.

Tous les langages, services et logiciels les plus importants gèrent de l'Unicode. Pour créer un texte Unicode sous Python, par exemple, il faut l'écrire entre guillemets précédés de la lettre U : *U"ceci est une chaine Unicode"* ou bien *U'Encore un texte Unicode'*.

Aujourd'hui, Unicode peut être utilisé presque partout. L'inconvénient d'Unicode est qu'il est plus lent et prend plus de place que d'autres représentations du même texte. C'est pour cette raison qu'il existe sous le standard Unicode plusieurs implémentations (sous forme de transformations), la plus célèbre et donc la plus utilisée est la norme "UTF-8".

### 2.4.2. UTF-8 (Unicode Transformation Format - 8 bits)

UTF-8 est un codage de caractères informatiques conçu pour coder l'ensemble des caractères du « répertoire universel de caractères codés ». Initialement développé par l'ISO dans la norme internationale ISO/CEI 10646, il est aujourd'hui totalement compatible avec le standard Unicode, en restant compatible avec la norme ASCII limitée à l'anglais de base (et quelques autres langues beaucoup moins fréquentes), mais très largement répandue depuis des décennies déjà [URL1].

L'UTF-8 est utilisé par 82,2 % des sites web en décembre 2014, puis par 86% en 2016. Par sa nature, UTF-8 est d'un usage de plus en plus courant sur Internet, et dans les systèmes devant échanger de l'information. Il s'agit également du codage le plus utilisé dans les systèmes GNU, Linux et compatibles pour gérer le plus simplement possible des textes et leurs traductions dans tous les systèmes d'écritures et tous les alphabets du monde.

## 2.5. Stockage de données (Composants de document XML)

### 2.5.1. Description du fichier <fileDesc> :

C'est un élément obligatoire dans la tête. Il comprend des informations bibliographiques sur le texte tel que le titre de l'œuvre, le nom de l'entreprise de l'auteur et de l'édition. Ci-dessous est un exemple :

```
<teiHeader>
        <fileDesc>
                <titleStmt>
                        <title>  Data Mining: practical machine learning </title>
        <author> Ian H. Witten, Eibe Frank </author>
        </titleStmt>
        <publicationStmt>
        <publisher>Morgan Kaufmann</publisher>
        <pubPlace>San Francisco</pubPlace>
        <date>2000</date>
        </publicationStmt>
        <sourceDesc>
                        <bibl> Data Mining: practical machine learning tools and
                        techniques
                        with Java implementations by Ian H. Witten and Eibe Frank
                        (San Francisco, 2000)  </bibl>
</sourceDesc>
</teiHeader>
```

**Figure 3.3 : Description du fichier <fileDesc>**

### 2.5.2. Description de l'encodage <encodingDesc>

Cette description précise la relation entre le texte et sa source, et elle contient neuf subdivisions en option. Pour le corpus de Latifa AL-Sulaiti deux éléments ont été choisis parmi cette catégorie. Il s'agit de la description du projet et la déclaration de l'échantillon. En

fait, il est indispensable de préciser le but du projet et les informations détaillées sur l'échantillon. Voici un exemple de la façon dont ils sont énoncés (Baker et al, 2003):

```
<encodingDesc>

    <ProjectDesc> Texte collectée pour être utilisée dans le projet </ProjectDesc>

    <sampleDesc> Texte écrit simple seulement a été transcrite. Des diagrammes, des
    images et des tableaux ont été omis et leur place marquée avec un élément de
    l'écart </sampleDesc>

</encodingDesc>
```

**Figure 3.4 : Description de l'encodage <encodingDesc>**

### 2.5.3. Description du profil <profileDesc>

Cette description fournit des informations non-bibliographiques sur le texte et sur les participants. Elle contient les éléments suivants: la création, l'utilisation de la langue, la classe du texte, une description du texte, la description du participant et la mise en description. Comme ces éléments sont optionnels, il a été décidé que ceux qui seront relatifs au corpus seront ajoutés. Les éléments qui ont été choisis sont les suivants:

#### 2.5.3.1. Description du texte <textDesc>

Cet élément fournit des informations sur le moyen par lequel le texte est livré. Que ce soit par impression, reçu par e-mail, en face-à-face, à partir de la télévision, etc. Également s'il est écrit ou parlé, parlé pour écrire ou écrit pour être parlé. Sa dérivation doit être indiquée si elle est originale ou traduite. Il est important de garder un état de son domaine (art, religion, histoire, etc.). En général, tout ce qui est disponible sur le texte doit être inclus. Cependant, dans le cas où l'information ne pouvait pas être trouvée, l'entrée serait indiquée « inconnue », et dans le cas où elle n'est pas concernée, elle serait indiquée « inapplicable ». Ci-dessous, un exemple :

```
<textDesc n='novel'>    <channel mode=w>print; part issues</channel>
                        <constitution type=single>
                        <derivation type=original>
                        <domain type=art>
                        <factuality type=fiction>
                        <interaction type=none>
                        <preparedness type=prepared>
                        <purpose type=entertain degree=high>
                        <purpose type=inform degree=medium>
</textDesc>
```

**Figure 3.5 : Description du texte < textDesc >**

#### 2.5.3.2. Description du Participant <participantDesc>:

Les informations sur les participants dans le texte sont très importantes. Les participants peuvent être des auteurs, haut-parleurs dans un dialogue, des intervieweurs … L'information qui doit être fournie se compose de sexe, âge, nationalité, date de naissance, lieu de résidence, langues parlées, éducation et occupation. En outre, il est nécessaire que chaque participant ai un identificateur pour rendre facile lors du traitement des textes la recherche de certaines personnes qui appartiennent, par exemple à une certaine culture sociale. Voici un échantillon :

```
<person id=PT sex=F age='mid'>
        <birth date='1950-01-12'> <date>12 Jan 1950</date>
                                  <name type=place>Shropshire, UK</name> </birth>
        <firstLang>English</firstLang>
        <langKnown>French</langKnown>
        <residence>Long term resident of Hull</residence>
        <education>University postgraduate</education>
        <occupation>Unknown</occupation>
        <socecstatus source=PEP code=B2>
</person>
```

**Figure 3.6 : Description du Participant <participantDesc>.**

### 2.5.4. Description de révision <revisionDesc>

Elle donne un résumé de l'histoire du texte ou du livre. Ainsi, si elle est mise à jour à un moment donné, il devrait y avoir une mention de la date et de l'état de sa modification. Il s'agit d'un élément très important dans un corpus. Cependant, il est difficile de passer en revue l'histoire de chaque document obtenu.

## 3. Manière d'interaction

Le schéma suivant présente la manière d'interaction entre les différentes parties du système :

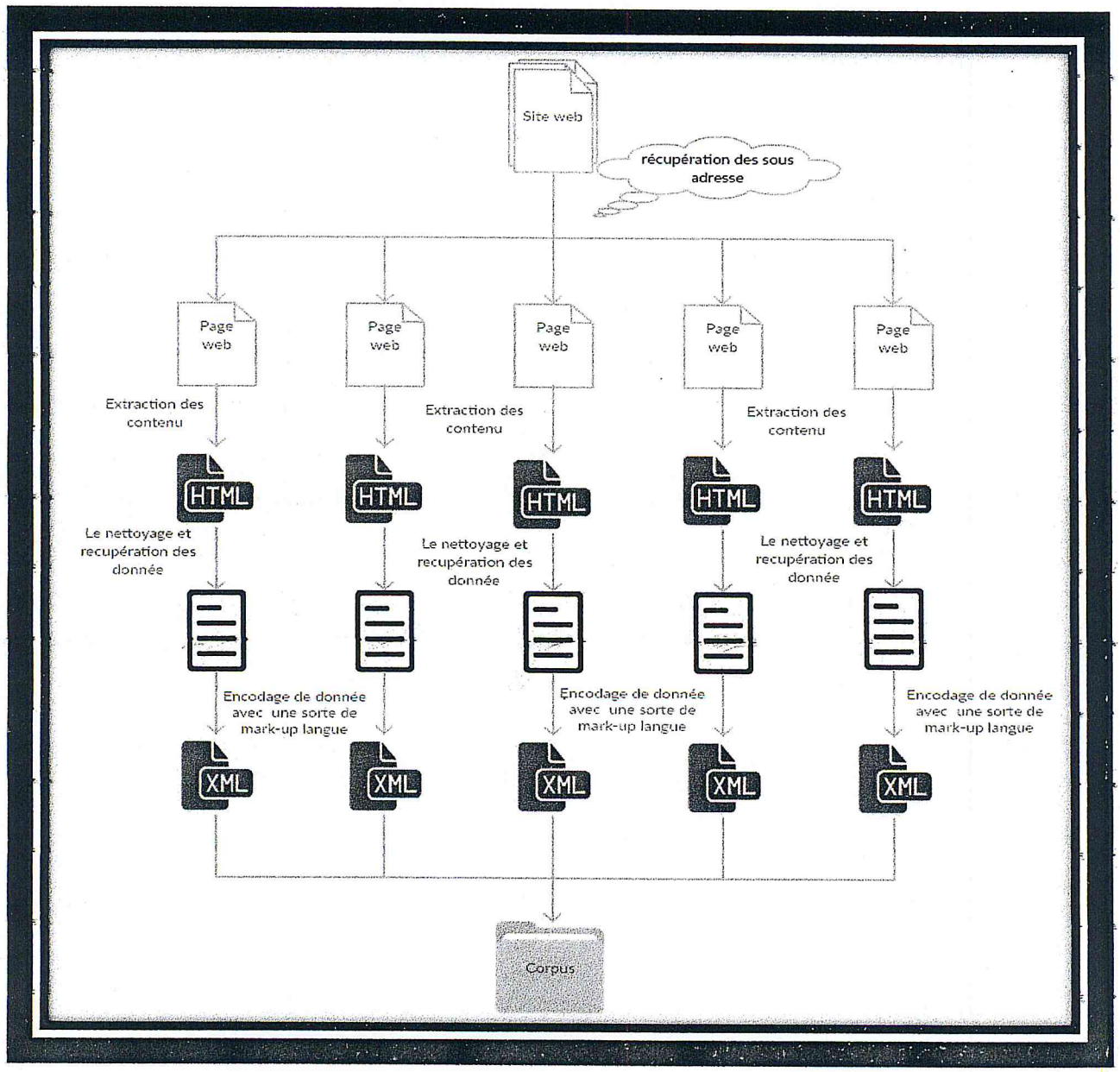


**Figure 3.7 : Manière d'interaction entre les différentes parties du système.**

## 4. Fonctionnalités offertes par le système

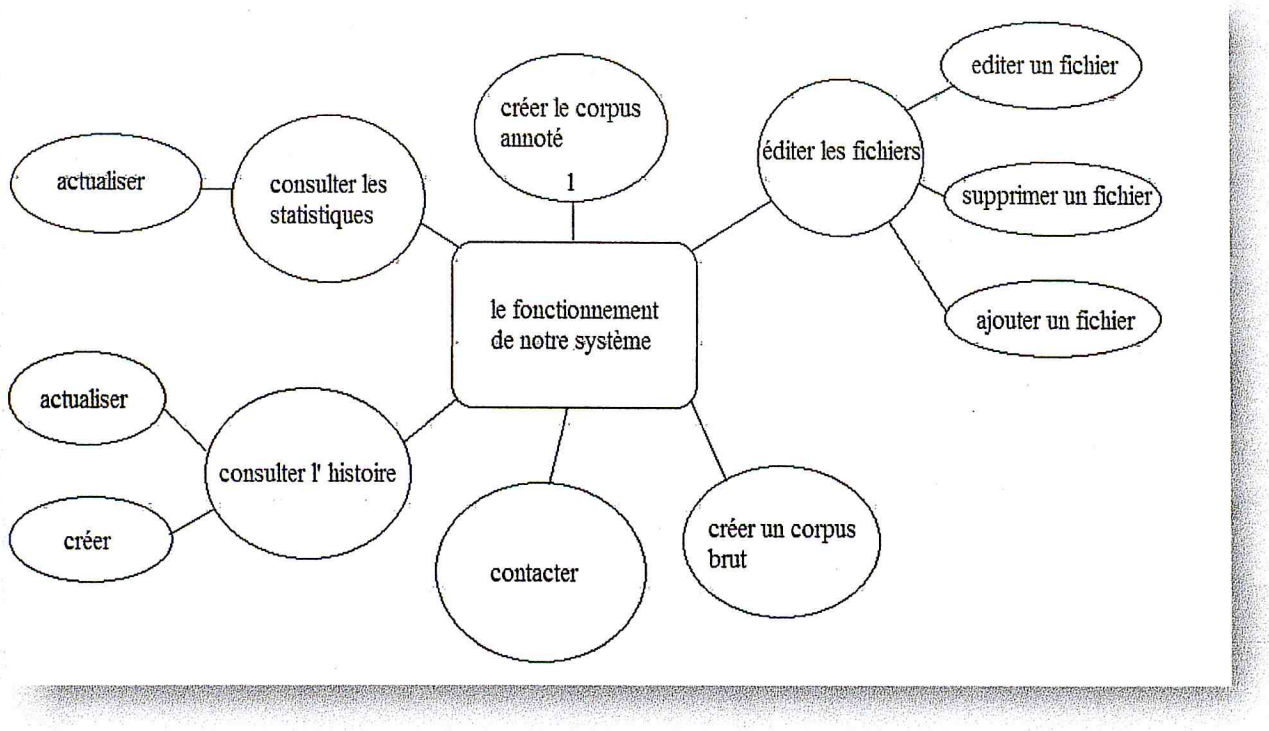


**Figure 3.8 : Schéma décrivant les fonctionnalités du système.**

### 4.1. Création du corpus

Si l'utilisateur veut alimenter le corpus annoté avec les sources web disponibles, il doit dans une première étape créer un corpus. Il doit ensuite suivre les étapes suivantes :

- L'utilisateur sélectionne un site web parmi la liste proposée.
- L'utilisateur démarre le crawler.
- L'application vérifie si le site web est déjà visité.
    - Sinon, l'application crée un dossier avec le nom de domaine.
- L'application fait une mise à jour de la liste des url-s existantes
    - Si L'url a été déjà visitée, l'application passe à l'url suivante.
- L'application crée un ficher XML avec les données fournies par l'url.
- Si la structure du site web (Template) est changée, le système notifie l'utilisateur pour une mise à jour du crawler.

- En cas d'un problème de connexion avec le site web, le système notifie l'utilisateur.

### 4.2.Création d'un corpus brut

- L'utilisateur saisi le site web.
- L'application vérifie le site web.
    - Si les données sont fausses, l'application notifie l'utilisateur.
- L'utilisateur démarre le crawler.
- L'application vérifie si le site web est déjà visité.
    - Sinon, l'application crée un dossier avec le nom de domaine.
- L'application fait une mise à jour de la liste d'url-s existantes.
    - Si L'url est déjà visitée, l'application passe à l'url suivante.
- L'application crée un ficher 'txt' avec les données fournies par l'url.
- Si la syntaxe du site web est erronée, le système notifie l'utilisateur pour porter les corrections nécessaires.
- En cas d'un problème de connexion avec le site web, le système notifie l'utilisateur.

### 4.3.Édition des fichiers

- L'utilisateur sélectionne un fichier.
- L'utilisateur sélectionne une action (ajouter, modifier, supprimer).
- L'application fait une mise à jour du fichier.
- Si le format du fichier est non supporté, le système notifie l'utilisateur pour changer le fichier.

### 4.4.Consultation des statistiques

- L'utilisateur accède à l'application.
- L'utilisateur fait une actualisation.
- Si le corpus a été supprimé ou bien le chemin n'existe plus, le système notifie l'utilisateur la situation.

### 4.5.Consultation de l'historique

- L'utilisateur accède à l'application.
- L'utilisateur peut faire une sélection par jour, par mois, par année.
- L'utilisateur procède à une actualisation.

- Si le corpus a été supprimé ou bien le chemin n'existe plus, le système notifie l'utilisateur la situation.

## Conclusion

Nous avons présenté dans ce chapitre la méthode suivi pour alimenter automatiquement notre corpus noyau d'une manière récursive. Notre démarche est déductive et contrastive : nous partons de l'architecture globale de notre système pour finir par découvrir et décrire ses différentes fonctionnalités.

La difficulté était de comment choisir et surtout comment combiner les techniques disponibles du modèle orienté objet afin de faire une conception qui répondre aux exigences des standards de développement et ainsi aboutir à un modèle qui soit stable aux futures exigences du système.

Nous présenterons dans le chapitre suivant la mise en œuvre de notre méthodologie, c'est-à-dire l'implémentation des différents critères. Cette mise en œuvre nous permettra d'apprendre l'encodage d'un corpus, mais aussi de tester la robustesse et la généricité de notre système.

# Chapitre 4

# Implémentation

# 1. Introduction

Ce chapitre décrit les différentes technologies adoptées et utilisées pour la réalisation de notre projet. Le texte qui suit est organisé de la manière suivante. Dans la section suivante, nous allons commencer par la description des outils et modèles que nous avons utilisé tout au long du développement de notre application. Dans la section 3, nous présenterons notre application à travers un sous-ensemble des interfaces que nous avons pu développer au cours de notre projet. La dernière section sera consacrée à la discussion des résultats.

# 2. Outil de Réalisation

## 2.1.Environnement matériel

- **Machine pour le développement :** Ordinateur portable DELL Intel Core i7-2670QM (8_CPU) @ 2.20GHz, 6 Go de RAM.
- **Système d'exploitation :** Windows 10 Professional 64 bit (10.0 version 10586)
- **Connexion internet :** Début de connexion 2 Mbps

## 2.2.Paradigme de programmation

Nous avons opté pour un modèle objet. La programmation orientée objet (POO), ou programmation par objet, est un paradigme de programmation informatique élaboré par les norvégiens Ole-Johan Dahl et Kristen Nygaard au début des années 1960 et poursuivi par les travaux d'Alan Kay dans les années 1970. Il consiste en la définition et l'interaction de briques logicielles appelées objets [9]. La programmation orientée objet est un style de programmation qui permet de regrouper au même endroit les comportements (les fonctions) et les données (les structures qui sont faites pour aller ensemble). Elle permet de créer des entités (objets) que l'on peut manipuler et elle impose des structures solides et claires. Les objets peuvent interagir entre eux, et cela facilite grandement la compréhension du code et sa maintenance. On oppose souvent la programmation objet à la programmation procédurale, la première étant plus "professionnelle" que l'autre car plus fiable et plus propre.

## 2.3.Python

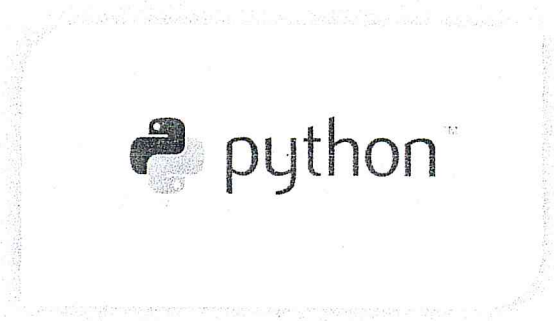


**Figure 4.1 : Logo du Python**

Python est un langage de programmation qui a été inventé par Guido van Rossum durant la période de 1985 à 1990. La première version de python n'est sortie qu'en 1991 [URL 6]. Python est un langage de programmation interprété. En d'autres termes, il n'est pas nécessaire de compiler du code Python pour l'exécuter. Python est à la fois simple, puissant, facile à apprendre et son code est très lisible. En Python, tout est objet : une chaîne, un entier, un dictionnaire, une liste, une fonction … D'ailleurs, en pratique, il est souvent le cas de manipuler des objets sans s'en rendre compte. Ainsi, Python permet d'écrire des scripts très simples mais qui peuvent par la suite être étendus grâce à ses nombreuses bibliothèques de tous genres. Ce qui fait qu'il est possible de travailler en Python sur des projets d'envergure.

## 2.4.XML (eXtensible Markup Language):

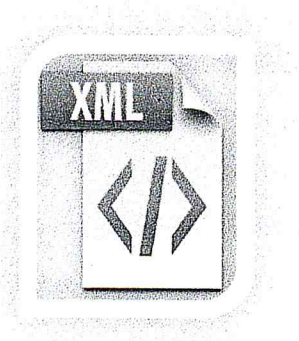


**Figure 4.2 : Logo du XML file**

XML est un outil logiciel indépendant du matériel pour le stockage et le transport de données entre machines et logiciels. Il s'agit d'un format de texte simple et très souple. Il est dérivé de SGML (ISO 8879). [URL 7]. C'est aussi un langage qui s'écrit à l'aide de balises.

Initialement, XML a été conçu pour relever les défis de l'édition électronique à grande échelle. Il joue également un rôle de plus en plus important dans l'échange d'une grande variété de données sur le Web et ailleurs. XML se veut donc un standard simple mais surtout extensible et configurable afin que n'importe quel type de données puisse être décrit avec. De plus, XML permet à l'utilisateur de créer son propre vocabulaire grâce à un ensemble de règles et de balises personnalisables.

Le XML est une recommandation du **W3C** [4], il s'agit donc d'une technologie très soutenue avec des règles strictes à respecter.

## 2.5.API (Application Programming Interface) :

Nous avons résumé dans le tableau suivant les différentes API utilisées par notre application.

| **Api** | **Description** |
|---|---|
| Urllib V3 | Fournit une interface de haut niveau pour l'extraction de données à travers le World Wide Web. |
| Sys | Permet d'accéder à certaines variables utilisées ou maintenues par l'interprète et aux fonctions qui interagissent fortement avec l'interprète. Elle est toujours disponible. |
| Re | Fournit l'expression régulière. Les fonctions de ce module nous permettent de vérifier si une chaîne particulière correspond à une expression régulière (ou si une expression régulière donnée correspond à une chaîne particulière, ce qui revient à la même chose). |
| Os | Fournit un moyen portable d'utiliser la fonctionnalité dépendante du système d'exploitation (lire ou écrire un fichier, manipuler des chemins, ...) |
| Time | Bien que ce module soit toujours disponible, toutes les fonctions sont disponibles sur toutes les plateformes. |

| | |
|---|---|
| Queue | Implémente les files d'attente multi-productrice, multi-consommateurs.<br>Il est particulièrement utile dans la programmation filetée lorsque les informations doivent être échangées en toute sécurité entre plusieurs threads. |
| Threading | Construit des interfaces de filetage de niveau supérieur sur le dessus du module de thread de niveau inférieur. |
| Requests<br>V2.10.0 | Permet d'utiliser le protocole http de façon ultra simple. Conçu pour les êtres humains. |
| Lxml<br>V3.4.4 | La bibliothèque la plus riche en fonctionnalités, facile à utiliser pour le traitement XML et HTML dans le langage Python. |
| Bs4<br>V 0.0.1 | Beautiful Soup est une bibliothèque Python pour tirer des données sur les fichiers HTML et XML. Il fonctionne avec un parseur (lxml, urllib,…) pour fournir des moyens idiomatiques de la navigation, la recherche, et de modifier l'arbre de parseur. Il sauve souvent les programmeurs heures ou des jours de travail. |
| PyQt4<br>V 4.11.4 | PyQt4 est un ensemble complet de liaisons Python pour Qt plate-forme. |

**Tableau 4.1 : Les API utilisées**

## 2.6.Qt Designer 5.6.0 :


**Figure 4.3 : Logo du Qt Designer**

Qt Designer est l'outil de communauté Qt (Qt plate-forme) pour la conception et la construction d'interfaces utilisateur graphiques (GUI) avec Qt Widgets. Vous pouvez composer et personnaliser vos fenêtres ou boîtes de dialogue dans une manière WYSIWYG (c'est-à-dire ce que vous voyez est ce que vous obtenez) [URL 9], et vous pouvez les tester en utilisant des styles et des résolutions différents.

Widgets et les formes créées avec Qt Designer s'intègrent de façon transparente avec le code programmé, en utilisant des signaux et slots mécanisme de Qt, de sorte que vous pouvez facilement affecter le comportement d'éléments graphiques.

Toutes les propriétés définies dans Qt Designer peuvent être modifiés dynamiquement dans le code. En outre, des fonctionnalités telles que la promotion des widgets et plugins personnalisés vous permettent d'utiliser vos propres composants avec Qt Designer.

Qt Designer n'est pas un programme magique qui va réfléchir à votre place. Il vous permet juste de gagner du temps et d'éviter les tâches répétitives d'écriture du code de génération de la fenêtre

## 2.7.JetBrains PyCharm

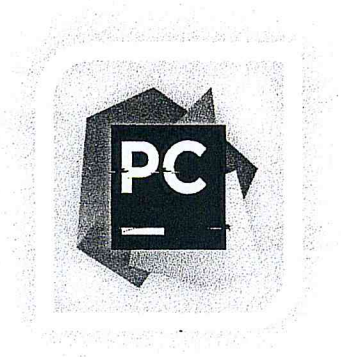


**Figure 4.4 : Logo du PyCharm**

PyCharm est un environnement de développement intégré (IDE) utilisé pour la programmation en Python. Il fournit une analyse de code, un débogueur graphique, un testeur d'unité intégrée, l'intégration avec les systèmes de contrôle de version et soutient le développement web avec Django. PyCharm est développé par la société tchèque JetBrains. Il est multiplateforme (Windows, Mac OS X et Linux). PyCharm possède une Professional

Edition, publiée sous une licence propriétaire et une communauté Edition publiée sous une licence Apache.

## 2.8.Architecture Model-View-Controller (MVC)

L'organisation globale d'une interface graphique est souvent délicate. Pour le développement de notre interface, nous avons appliqué l'architecture Model-View-Controller (MVC). Ce paradigme divise l'IHM (Interface Homme Machine) en un modèle, une vue et un contrôleur, chacun ayant un rôle précis dans l'interface. L'architecture MVC ne résout pas tous les problèmes. Elle fournit souvent une première approche qui peut ensuite être adaptée. Elle offre aussi un cadre pour structurer une application.

Ce modèle d'architecture impose la séparation entre les données, la présentation et les traitements, ce qui donne trois parties fondamentales dans l'application finale : le modèle, la vue et le contrôleur. En voici l'interprétation de chaque brin du modèle :

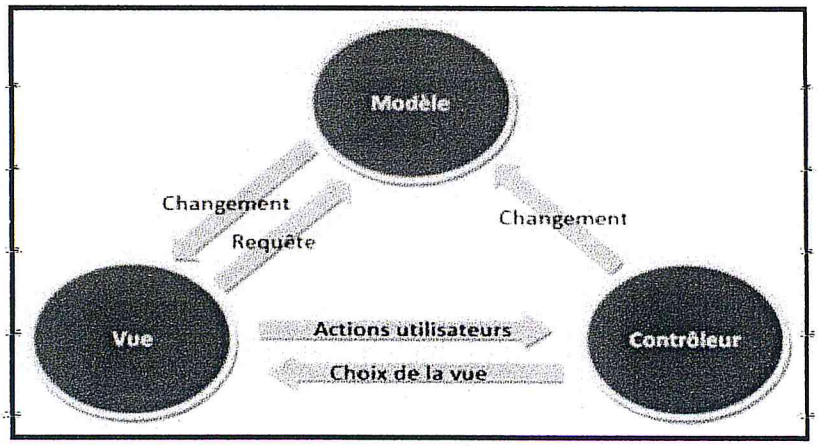


**Figure 4.5 : Schéma du modèle MVC**

- **Modèle :** Rassemble des données du domaine et des connaissances du système. Contient les classes dont les instances doivent être vues et manipulées. Le modèle représente le comportement de l'application : traitements des données, interactions avec la base de données, etc. Il décrit ou contient les données manipulées par l'application. Il assure la gestion de ces données et garantit leur intégrité. Dans le cas typique d'une base de données, c'est le modèle qui la contient. Le modèle offre des méthodes pour mettre à jour ces données (insertion, suppression, changement de valeur). Il offre aussi des méthodes pour récupérer ces données. Les résultats renvoyés par le modèle sont dénués de toute présentation.

- **Vue :** La vue correspond à l'interface avec laquelle l'utilisateur interagit. Sa première tâche est de présenter les résultats renvoyés par le modèle. Sa seconde tâche est de recevoir toutes les actions de l'utilisateur (clic de souris, sélection d'une entrée, boutons, etc.). Ces différents événements sont envoyés au contrôleur. En d'autres termes, la vue sert d'interface entre les actions de l'utilisateur et le contrôleur. La vue n'effectue aucun traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle et d'interagir avec l'utilisateur.

- **Contrôleur :** Le contrôleur prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle, et ce dernier notifie à la vue que les données ont changée pour qu'elle les mette à jour.

## 3. Interfaces graphiques

L'interface graphique est une partie très importante dans la réalisation d'une application convenable et conviviale. L'interface se doit d'offrir un certain plaisir à l'utilisateur lors de sa navigation. Ainsi ce critère peut faire la différence entre une application et une autre bien qu'elles aient les mêmes fonctionnalités. Dans les paragraphes subséquents, nous présenterons un ensemble de captures d'écrans montrant les principaux points d'entrées de l'application.

### 3.1. Interface de l'onglet « corpus annoté »

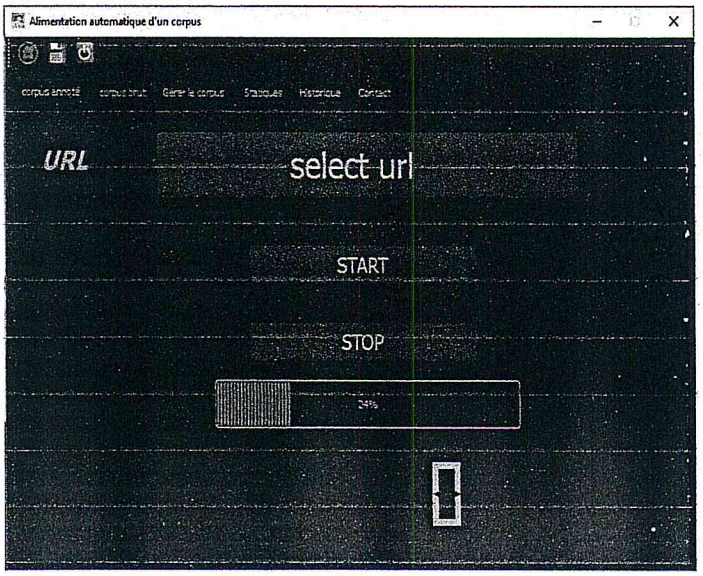


**Figure 4.6 : L'onglet « corpus annoté »**

L'utilisateur doit d'abord sélectionner un site web parmi une liste de sources web disponibles. Ensuite, il doit cliquer sur le bouton START pour démarrer l'alimentation. En retour, l'application présente l'état d'avancement du processus d'alimentation sous la forme d'une barre de progression traditionnelle et sous la forme d'un afficheur LCD. Ceci permettra à l'utilisateur de suivre l'état de progression et le nombre de donnée.

### 3.2. Interface de l'onglet « corpus brut »

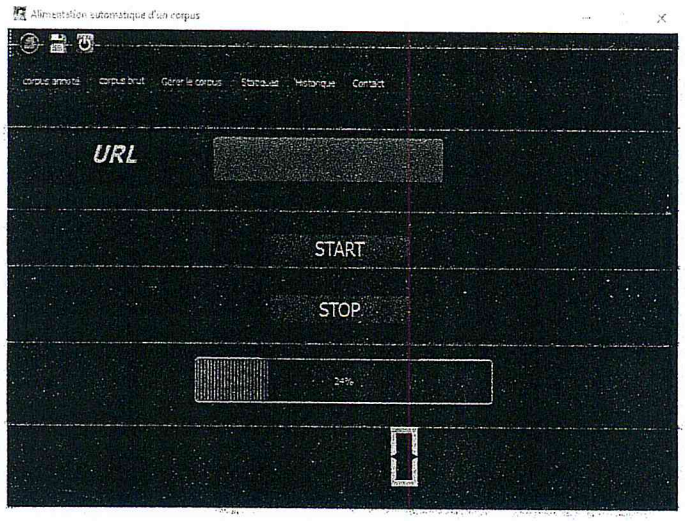


**Figure 4.7 : L'onglet « corpus brut »**

L'utilisateur doit d'abord saisir l'URL valide d'un site web. Ensuite, il doit Cliquer sur le bouton START pour démarrer l'alimentation du corpus à partir du site. L'application fournit dans ce cas aussi une barre de progression et un afficheur LCD pour le suivi.

### 3.3.Interface de l'onglet « gérer le corpus »

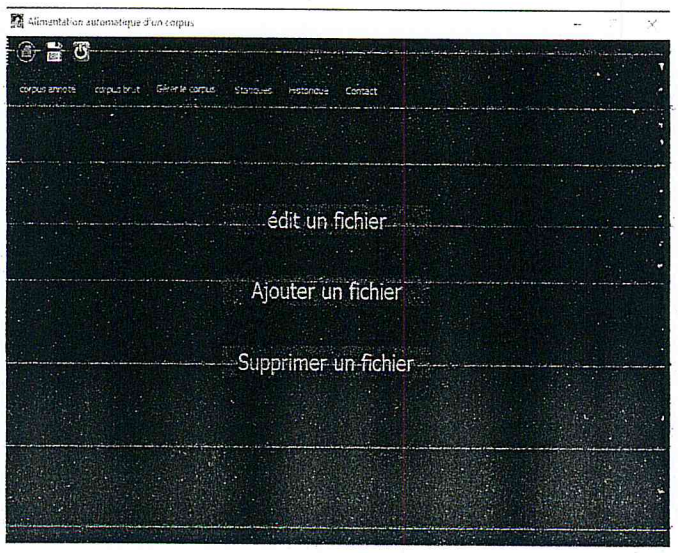


**Figure 4.8 : L'onglet « gérer le corpus »**

À partir de cette page, l'utilisateur est capable d'ajouter, modifier ou supprimer n'importe quel document dans le corpus. Un clic sur l'une des fonctionnalités de cette interface permet d'afficher la nouvelle interface suivante :

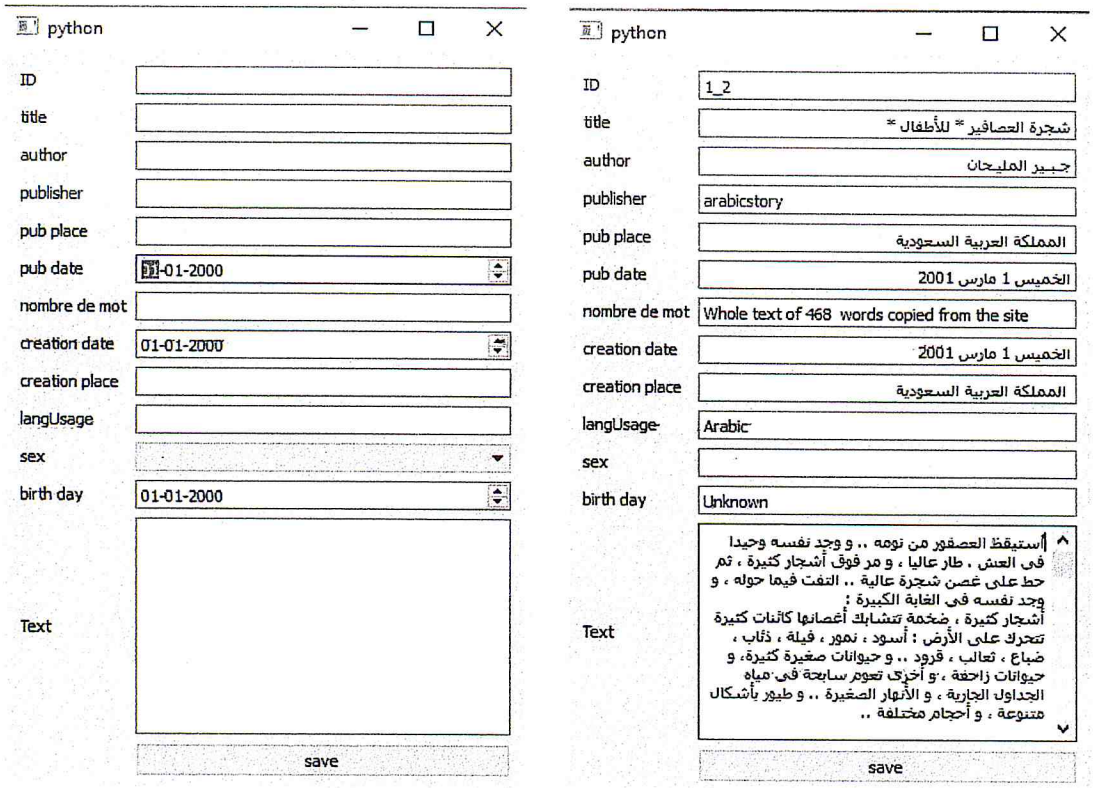


**Figure 4.9 : La fenêtre « éditer un fichier »**

## 3.4.Interface de l'onglet « statistiques »

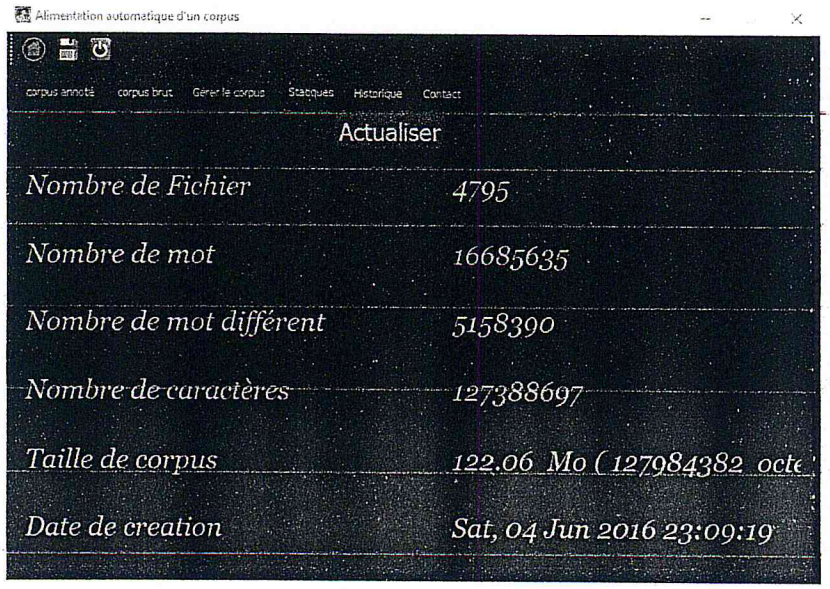


**Figure 4.10 : L'onglet « statistiques »**

Dans cet onglet apparaissent certaines statistiques utiles pour le gestionnaire du corpus : nombre de fichiers, nombre de mots, nombre de mots différents, nombre de caractères, taille réelle du corpus et la date de la dernière opération d'alimentation du corpus. En outre, il est possible pour l'utilisateur d'actualiser les statistiques à chaque fois qu'il lui serait nécessaire.

### 3.5.Interface de l'onglet « historique »

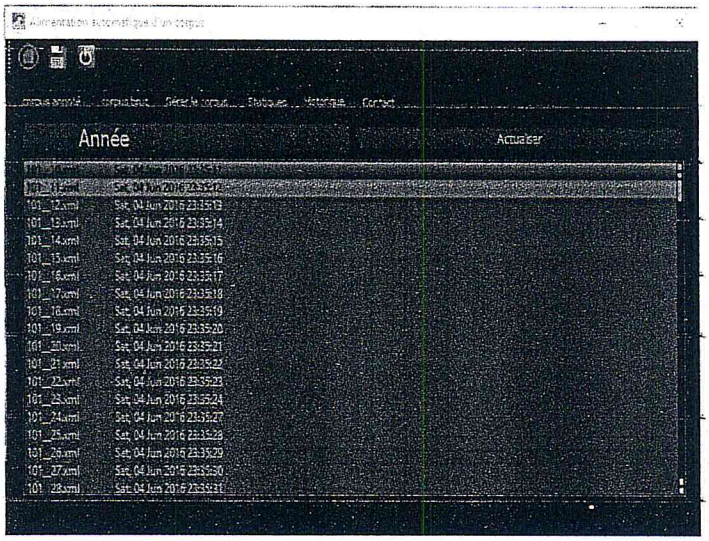


**Figure 4.11 : L'onglet « historique »**

À partir de cet onglet, l'utilisateur peut obtenir un historique trié par jour, semaine, mois et année. L'utilisateur est aussi capable de modifier le contenu d'un document. Il peut également accéder à un document à travers la fenêtre « éditer un fichier » de l'onglet « gérer le corpus » (*Figure 4.9*).

## 4. Évaluation et discussion des résultats

Pour des raisons de validation de l'application, du essentiellement au temps imparti à la réalisation du projet, nous nous sommes contenté d'alimenter une seule partie du corpus. Ainsi, nous nous sommes restreints au genre « Short stories ».

Vu que nous n'avons pas en notre possession d'autres applications similaires à la notre -nous pensons bien sur avoir réalisé la première application de ce genre- nous nous contentons d'évaluer notre application sur deux volets : le temps d'exécution du processus d'alimentation et la taille du corpus.

### 4.1. Temps d'exécution

Lors d'une collecte manuelle de texte, le temps pris par les différentes étapes du processus (voir chapitre 1 (7.4)) est évalué à environ 15 minutes. [11] Avec le nouveau système en place, le temps est corrélé avec la vitesse et le débit de la connexion. Avec un débit de 2 Mbps, le système a besoin d'uniquement 1,15 seconde.

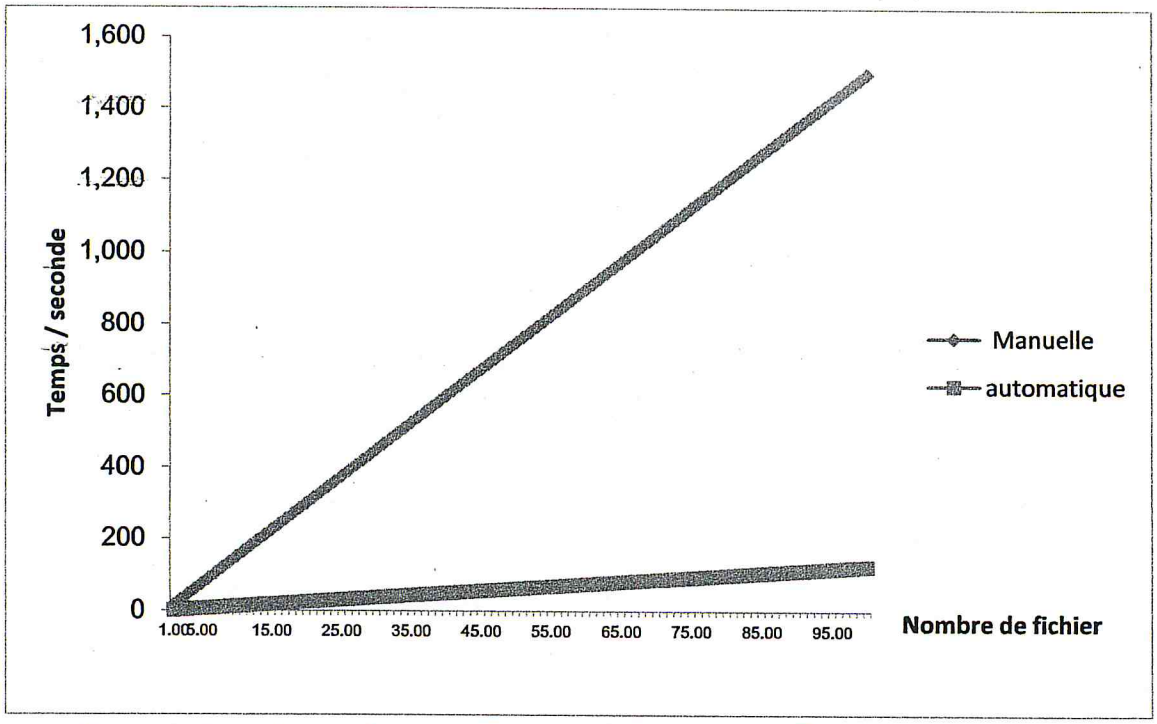


**Figure 4.12 : Temps d'exécution alimentation manuelle vs. Automatique**

### 4.2.Taille du corpus

Nous avons démarré avec un corpus « noyau » contenant 842.684 mots distribués sur 416 fichiers couvrant un certain nombre de catégories. [11] Notre sous-corpus cible, à savoir le corpus " Short stories ", contient lui 31 fichiers avec 45.460 mots. Jusqu'à notre dernière extraction sur les sites sources du Web, nous avons accumulé sur le genre " Short stories " 65.729.920 mots répartis sur 18.889 fichiers. Statistiquement, nous avons rapatrié 158.000 fois ce que contenait notre corpus.

## 5. Conclusion

Dans ce chapitre nous avons décrit les technologies utilisées pour la réalisation de notre projet ainsi que les fonctionnalités de base de l'application à travers un ensemble de captures d'écran. Nous avons par la suite donné un aperçu sur les apports quantitatifs du processus d'alimentation automatique au corpus. Il est clair à partir des expérimentations faites qu'une extraction quantitatives de textes à partir du Web est faisable malgré un certain nombre de problèmes que nous énumérerons ci-dessous.

# Conclusion Générale

Tout au long de ce mémoire, nous avons présenté les différentes étapes de la réalisation de notre projet de recherche. Nous avons tout d'abord énoncé notre problématique de récherche que nous avons résumé en la difficulté de construire, avec un minimum de moyens (scanners, OCR-s, vérificateurs d'orthographe, ...), de compétence (mesuré en hommes : linguistes, spécialistes en traitement automatique des langues, ...) et de temps (mesuré en heures : des centaines voir des milliers d'heures), un corpus consistant pour la langue arabe qui soit utilisable dans des domaines aussi variés que l'enseignement, la traduction et la recherche. L'idée derrière cette problématique est basée sur deux hypothèses principales. La première stipule qu'il existe, et en libre accès et en open source, un corpus de départ (ou noyau) qu'il serait possible d'élargir. La seconde hypothèse stipule que le web constitut une source de données textuelles énorme qu'il est possible d'utiliser dans des projets aussi important que la construction de corpus textuels. La question principale de recherche que nous avons alors proposée a été la suivante : étant donné un corpus « noyau » qui est le corpus libre de Latifa Al-Solaiti, d'une part, et une immense bibliothèque textuelle qui est le web, est-il possible d'alimenter le premier à partir du second et dans quel condition ? Pour concrétiser cette idée, nous avons proposé par la suite un état de l'art sous la forme de deux chapitres. Puisqu'il s'agit de manipuler des données textuelles qui vont d'une source vers une cible, le premier chapitre s'est intéressé à la cible qui est le corous noyau, alors que le second a été déstiné à la source de données. Dans ces deux chapitres, nous avons présenté les structures de la cible et de la source et nous avons aussi donné une idée sur la génèse de chacune. Par la suite, nous avons entamé un travail de conception dans lequel nous avons proposé notre nouveau système. Dans le dernier chapitre, nous avons décrit l'implémentation du système avec une illustration de ses différentes fonctionalité.

À l'issue de la réalisation de ce travail, nous avons pu confirmer qu'un projet d'une telle envergure est réalisable et à moidre coût et d'une grande utilité dans beaucoup de domaine où un corpus s'avère important. Comme exemple d'utilisation d'un tel corpus, il suffit d'immaginer un enseignant qui veut préparer des textes pour lectures, examens ou exercices d'application avec des données authentiques, modernes et d'une langue spécifique au niveau des ses apprenants. Pour s'en rendre compte, nous n'avons qu'à voir les textes utilisés dans les manuelles scolaires, les textes pour lecture et les textes des examens des élèves du primaires qui sont dans la plupart du temps hors de leur portée ou qui n'attirent même pas leur attention.

Ce projet nous a permis de nous familiariser avec plusieurs domaines tels que le web et ses technologies, la linguistique de corpus et le traitement automatique des langues. Ce qui a été validé par un certain degré de compétence dans ce domaine pluridisciplinaire.

Comme tout les sujets de recherche, on trouve toujours des obstacles théorique et téchnique. L'esprit de problème en utilisant le web comme source de donnés est qu'il est vraiment difficile de trouver des échantillons idéales et de taille considérables. La plupart du donnes sont des courts articles ou des groupe d'articles qui traitent un sujet spécifique. Et il est rare de trouver une page web ou un article web de la longueur d'un livre sur le web.ainsi que la majourité des fondateurs de sites web arabe ne respecter pas les conventions internationales structurellement pour construire leurs sites Web, voilà ce que forme d'un embarras majeur dans la sélection des sources des donnes.Même les sites web qui ont été sélectionnés ne sont pas libres de ce défaut , il y a un changement permanent dans les positions et dans la forme de données. La majorité de ces sites utilisent des système de codage déserte , ce qui rend difficile de faire la recuperation des données.Certaines sources ne se soucient pas pour la classification des contenus, et comme les données soient récupérées d'une large quantitées, cela rend le processus de classification automatique parmi les miracles de notre application. Notre travail peut être un sujet qui possede des extensions. Car on peut toujours l'améliorer et à cause des difficultés techniques de la part des site web et la désorganisation de ces code source et à cause du temps limité nous n'avons pas pu faire tout le travail que nous avons envisagé à faire, et nous voudrons bien mettre notre application sur des plateforme professionnelle, et nous comptons persévérer pour l'améliorer et la rendre plus pratique et plus utile pour les corpus.

# Bibliographie


[1] Dictionnaire Larousse http://www.larousse.fr/

[2] Sinclair, J. (1996). Preliminary recommendations on text typology. Eagles Document EAGTCWG-TTYP/P. http://www.ilc.cnr.it/EAGLES96/texttype/texttyp.html.

[3] Habert (2000) « Regroupements issus de dépendances syntaxiques sur un corpus de spécialité : catégorisation et confrontation à deux conceptualisations du domaine » http://perso.ens-lyon.fr/benoit.habert/Publications.php

[4] Leech, G. (1997). Teaching and Language Corpora: a Convergence. In Teaching and language corpora, Wichmann, A. et al,eds., Longman, pp. 1-23

[5] Gérard Swinnen(2000,2005) , Apprendre à programmer avec Python,P370

[6] Al-Sulaiti, L. & Knowles, G. (2002). A multimedia Arabic course. In Proceedings of the International Symposium on: The Processing of Arabic, 94-105.

[7] Sharoff 2004: 1745)

[8] Berland, constitution de corpus à partir du web pour l'acquisition terminologique : une expérience, mémoire de DEUA, 2000.

[9] De Hugues Bersini (2007). L'Orienté Objet, ISBN 978-2-212-12084-4.

[10] Mancer (2015), Style architecture

[11] Al-Sulaiti, L. & Atwell, E. (2003). The Design of a Corpus of Contemporary Arabic (CCA). School of Computing, Research Report Series, University of Leeds.


# Webographie


[URL1] - https://fr.wikipedia.org/wiki/Corpus

[URL 2] - http://www.comp.leeds.ac.uk/latifa/

[URL 3] - http://www.futura-sciences.com/

[URL 4] - http://www.dictionnaireduweb.com/

[URL 5] - http://www.arabicstory.net/?p=home

[URL 6] - http://apprendre-python.com/

[URL 7] - http://www.w3schools.com/default.asp

[URL 8] - https://docs.python.org/3/library/

[URL 9] - http://doc.qt.io/qt-4.8/

[URL 10]- https://www.jetbrains.com/pycharm/