Sophie REPINGON

Master Traitement Automatique des Langues

Ingénierie linguistique

RAPPORT d'ALTERNANCE

[*Contrat d'apprentissage effectué du 14 septembre 2020 au 13 septembre 2021*]

Cadic Services

146 rue Montmartre, 75002 Paris

Mise en place d'un portail de gestion de fonds documentaire

Tutrice entreprise : Jennifer HUBERT

Tutrice académique : Iris ESHKOL-TARAVELLA

Soutenu le 25 juin 2021

Université Paris Ouest Nanterre La Défense

200 avenue de la République 92001 Nanterre

Année universitaire 2020 - 2021

Avant-propos

Ce rapport d'alternance a été réalisé dans le cadre de mon Master 2 de Traitement Automatique des Langues à l'Université de Paris Nanterre. J'ai débuté mon contrat d'apprentissage en septembre 2020 chez Cadic Services.

Je tiens à remercier ma tutrice Jennifer HUBERT ainsi que tous mes autres collègues de Cadic. Je souhaite également remercier Iris TARAVELLA et l'ensemble de l'équipe pédagogique du Master pour leurs enseignements et leur disponibilité au cours de cette année universitaire.

Enfin, je remercie la Succession Saint Exupéry – d'Agay de m'avoir permis de citer leur nom dans ce rapport. Pour des raisons de confidentialité, les visuels de leur fonds ne seront pas affichés sur les captures d'écran et seront remplacés par un visuel par défaut.

Table des matières

Table des illustrations

I. Introduction

Dans le cadre du parcours d'Ingénierie Linguistique du Master 2 de Traitement Automatique des Langues de l'Université Paris Ouest Nanterre, j'effectue depuis septembre 2020 une alternance chez Cadic Services. J'ai découvert durant cette année le domaine de l'édition de logiciels, mais également celui de la documentation et de la gestion de fonds documentaires.

Durant mon alternance, ma mission de consultante était de participer à la mise en place de la *Solution Cadic Intégrale* chez les clients. Pour cela, je réalisais les paramétrages fonctionnels répondant à l'offre commerciale acceptée par le client et correspondant aux spécifications fonctionnelles réalisées et validées dans le cadre d'ateliers organisés avec le client. Ceux-ci se faisaient sur le logiciel développé par Cadic, mais nécessitaient également d'utiliser des langages tels que le SQL, le HTML, la CSS, le PHP, le TinyButStrong ou le WebScript (langage propre à Cadic). Au sein de l'entreprise, j'ai développé des connaissances sur le monde de la documentation, mais aussi sur des domaines comme la gestion de fonds multimédia, les archives ou bien la norme UNIMARC.

J'ai également acquis des connaissances dans la gestion de projet et notamment la méthode AGILE, utilisée chez Cadic. Lors de cette année, j'ai eu la chance de pouvoir participer à un projet du début à la fin, en équipe avec une collègue plus expérimentée. Cela m'a permis de découvrir comment se déroulait un projet, depuis les ateliers avec le client jusqu'à la livraison, et d'y participer activement. En parallèle, j'ai travaillé sur la mise en place d'un module de TAL dans la solution de Cadic. En effet, le TAL ne faisant pas partie du cœur de métier de l'entreprise, j'ai proposé la conception d'un module permettant d'allier les connaissances que j'ai acquises durant mon master et les besoins des clients de Cadic. J'ai donc pu mener ce projet en parallèle de ma mission principale qui était celle de consultante fonctionnelle.

Ce rapport d'alternance va donc décrire les missions que j'ai effectuées dans le cadre de ces deux projets : la mise en place de la *Solution Cadic Intégrale* chez un client, la Succession Saint Exupéry – d'Agay (ci-après "la Succession"), et l'intégration d'un module de Traitement Automatique des Langues dans la solution.

Mes problématiques dans ce rapport seront les suivantes : comment se déroule un projet entre la maîtrise d'ouvrage (MOA) du côté des documentalistes et la maîtrise d'œuvre (MOE) du côté de la réalisation informatique ? Et comment créer et mettre en place un module de traitement automatique des langues à partir d'un corpus amené à évoluer ?

II. Objet du stage

II.1. Présentation de l'entreprise

Cadic Services est une entreprise d'édition de logiciels informatiques, spécialisée dans la gestion de fonds documentaires. Elle propose également un module d'archives. Le produit vendu est la *Solution Cadic Intégrale*, sous la version actuelle nommée *Zéphyr*, un logiciel de gestion de l'information et de documents physiques, numériques et multimédia. Il permet de gérer une base de données documentaires et de proposer une interface de recherche, tout en gérant les utilisateurs de la plateforme. L'interface est personnalisable et paramétrable par le professionnel à l'aide des outils fournis avec la solution. Dans une commande, est comprise la formation du client sur les différents outils de Cadic.

Parmi les références clients de l'entreprise, on trouve des écoles, des conseils départementaux, des archives, des musées ainsi que des bibliothèques. Un nouveau projet est démarré lorsqu'un organisme a besoin de gérer son fonds documentaire ou de changer le fonctionnement de sa gestion. Il est possible que les métadonnées des fonds soient au format papier (peu de cas) ou répertoriées dans des bases de données stockées dans d'autres logiciels spécialisés ou dans des logiciels bureautiques tels que Excel. Il devient alors nécessaire pour l'organisme de pouvoir les gérer et effectuer des recherches de façon plus ergonomique. Il arrive également que le client bénéficie déjà d'une solution de gestion documentaire, mais qu'il souhaite la changer. Dans tous les cas, Cadic réalise des modèles permettant d'importer facilement les documents, les métadonnées et les vocabulaires associés (liste d'autorité, plan de classement ou thésaurus structuré) dans la base de données de l'application Cadic.

La solution logicielle s'installe sur un serveur (OS Linux, Windows, etc.) et est ensuite accessible à ses utilisateurs via l'adresse URL de l'application accessible depuis un navigateur (Firefox, Chrome, etc.). Il est possible, si cela est souhaité, d'héberger le site chez Cadic, ou le client peut décider de l'héberger sur un serveur qu'il possède ailleurs. La face visible de l'application est le portail documentaire, appelé également *front-office*, qui peut être mis en ligne pour ses utilisateurs qui y accèdent depuis internet ou depuis le réseau interne d'un client (accès intranet). Il est possible pour les utilisateurs externes de détenir un compte sur le portail, avec un certain niveau d'accès et une certaine catégorie (par exemple *abonné, lecteur simple*, etc.). Avec son compte l'utilisateur peut ainsi gérer ses emprunts, ses commandes et faire ses propres sélections de notices. La face cachée de l'application est accessible aux professionnels, on parle alors du *back-office*. Il est possible pour l'administrateur de paramétrer l'application et de la personnaliser totalement. Cela peut être au niveau esthétique comme fonctionnel, en déterminant par exemple les champs sur lesquels la recherche sera possible pour les utilisateurs, ou bien les informations qui seront mises à disposition de telles ou telles catégories d'utilisateurs. Les documentalistes qui ne sont pas administrateurs peuvent quant à eux créer de nouvelles notices ou en modifier des existantes. Ils ont un rôle de gestion, mais sans pouvoir

modifier le portail profondément dans son fonctionnement comme il est possible pour les administrateurs.

Chez Cadic Services, il y a un service de recherche et développement et un service opérationnel. J'ai, pour ma part, travaillé dans ce dernier. Notre rôle est de gérer le projet, depuis les spécifications avec le client jusqu'à la livraison du site, en passant par le paramétrage fonctionnel du portail. Nous devons suivre un planning afin que la livraison soit faite à temps, nous gérons la relation avec le client, nous paramétrons l'application et nous travaillons en équipe avec les développeurs afin de leur demander certaines fonctionnalités nouvelles ou bien l'amélioration de fonctionnalités déjà existantes.

"*Cadic Intégrale c'est la mise à disposition aux publics utilisateurs, de tous contenus internes ou externes à l'organisation, voire publiés directement par eux et accessibles depuis une interface Web personnalisée, le Portail documentaire.*" (Cadic Services, 2018)

D'un point de vue technique, l'application et tous ses modules sont développés en PHP, pour des raisons de sécurité et de puissance.

II.2. Présentation du travail à réaliser

Mon poste de consultante consiste à connaître le fonctionnement et le paramétrage de tous les aspects du logiciel. Celui-ci étant assez technique, j'ai été formée tout au long de mon alternance, petit à petit, au fur et à mesure des missions et réalisations qui m'étaient confiées. Mon alternance s'est axée autour de deux missions principales : le projet avec la Succession Saint Exupéry – d'Agay, avec notamment la mise en place d'un nouveau module multimédia au sein de l'application ; et la création d'un module de TAL. J'ai ainsi réalisé mes missions tout en étant formée sur les différents outils de Cadic. Le Traitement Automatique des Langues n'étant pas le cœur de métier de l'entreprise, j'ai conçu le module de TAL du début à la fin, en commençant par proposer des idées en échangeant avec le responsable du service de recherche et développement, puis en les réalisant et enfin en les adaptant en vue de l'intégration au produit. J'ai pu mener à bien ces tâches en m'appuyant sur les différents cours que j'ai suivis pendant mes deux années de master, à l'INALCO, Nanterre et Paris III.

Afin de réaliser les missions qui m'ont été confiées au sein de l'entreprise, j'ai utilisé les langages de programmation suivants : SQL, HTML, CSS, TinyButStrong, WebScript (langage interne) et PHP. J'ai également utilisé les outils d'administration de Cadic suivants : *WebAdmin, WebAdmin 2, eXtenso Designer* et *SQLAdmin*. Ces outils sont les principaux développés par l'entreprise afin de réaliser le paramétrage des différentes pages du site ainsi que la gestion de la base de données.

Pour le développement de mon module de TAL, j'ai utilisé les langages Python et Bash ainsi que les outils *Talismane* et *TreeTagger*.

II.3. Utilité du stage pour l'entreprise

Pour l'entreprise, le but était de former une nouvelle consultante technico-fonctionnelle afin de réaliser les paramétrages pour le client et de gérer la relation avec lui. Une consultante doit être à l'aise avec le produit de Cadic Services et avec ses outils d'administration. La réalisation d'un site étant complexe, cela nécessite du temps de formation et beaucoup de pratique afin d'être à l'aise dans cette mission. A terme, nous avons donc convenu que je continuerai à travailler chez Cadic, sous la forme d'un emploi salarié, à la suite de cette alternance.

De par ma formation en traitement automatique des langues, j'ai également pu apporter à l'entreprise les moyens de créer un nouveau module de TAL. Celui-ci est une plus-value pour les clients souhaitant avoir un aperçu statistique de leur fonds documentaire d'un point de vue textuel ou linguistique.

II.4. Limites du sujet et problèmes connexes qui ne seront pas traités

Un projet devant respecter des dates de livraison, je n'étais pas seule à travailler sur celui développé pour la Succession Saint Exupéry – d'Agay. J'ai travaillé en binôme avec une consultante plus expérimentée, ma tutrice, Jennifer HUBERT. Nous avions des échéances à respecter, ainsi je ne pouvais libérer que certaines journées ou demi-journées pour mon projet de TAL. J'ai donc pu établir des statistiques sur les mots les plus fréquents dans un fonds documentaire selon les catégories grammaticales ainsi qu'une carte de proximité sémantique à partir d'un fonds. Malheureusement, étant contraints par les échéances de livraison à court terme des divers projets en cours menés par les équipes de Cadic ainsi que la disponibilité des développeurs du service de recherche et développement, ces modules n'ont pas encore pu être intégrés dans l'application, mais il est prévu qu'ils le soient prochainement.

III. Présentation du contexte et du produit

III.1. La gestion de fonds documentaires

L'application de Cadic Services s'adresse aux professionnels de la documentation. Le métier de documentaliste consiste à "*rechercher une adresse dans une base de données, sélectionner des articles pour un dossier de presse, commander des ouvrages et modifier les plans de classement (...) enrichir son fonds documentaire. (...) Les rapports avec les utilisateurs et les partenaires représentent une large part de l'activité.*" (ONISEP)

Une notice bibliographique "*inclut l'ensemble des éléments présentant la description bibliographique, les accès et la cote d'un document (auteur, titre, éditeur, date d'édition, etc.). La notice bibliographique permet de décrire un livre, un article de périodique, une thèse ou tout autre support documentaire.*" (Université de Poitiers, 2019)

III.2. Le produit de Cadic Services

La *Solution Cadic Intégrale* propose un logiciel de gestion de fonds documentaires personnalisable et un choix de modules supplémentaires (bibliothéconomie, archives, etc.). Son fonctionnement repose sur une base de données SQL et le site est construit en PHP. Le logiciel met à disposition de l'administrateur des outils de construction des interfaces ainsi que des outils d'administration.

III.2.A. La base de données

L'application de Cadic permet la gestion d'une base de données. Celle-ci est composée de plusieurs tables permettant de gérer différentes données tels que les utilisateurs, le fonds documentaire, le fond multimédia, les commandes, la gestion des prêts de documents, etc.

"*Une table est définie par un ensemble de champs (ou colonnes) ; elle est peuplée par des enregistrements (ou lignes). Chaque ligne correspond à une notice ; chaque notice peut être liée à un fichier texte extérieur et à une image. Table, base et collection sont synonymes pour le moteur de données Cadic Intégrale*" (Cadic Services, 2020)

Il existe un certain nombre de tables standard qui sont utilisées dans la plupart des applications Cadic : la table des documents (ILS_DOC), la table des commandes d'ouvrages (ILS_CMD), la table des collections de périodiques (ILS_PER), la table des exemplaires (ILS_EXP), la table des lecteurs (ILS_LEC), la table des catégories de lecteurs (ILS_CAT), etc. Il existe également des tables servant à gérer les listes d'autorité, ILS_TBR (listes standard) et ILS_AUTO (listes spécifiques à un client) par exemple.

Dans mon projet pour la Succession Saint Exupéry – d'Agay, j'ai utilisé la table PHO_DOC et non la table ILS_DOC car il s'agissait d'un fonds multimédia alors que

PHO_DOC est adaptée à un fonds photothèque. Cette table contient donc des champs liés à la gestion multimédia qui ne sont pas dans ILS_DOC, plutôt destinée aux fonds documentaires.

Un champ est un "*terme documentaire utilisé en informatique qui désigne un des éléments de la notice bibliographique et qui sert à la saisie et à l'interrogation. Champ auteur, champ titre, etc.*" (J-P. Accart et M-P. Réthy, 1999)

Lorsqu'on crée un champ, on définit le type de données qu'il va contenir :

- des caractères alphanumériques : pour des mots ou textes ;
- des caractères numériques : pour les nombres ;
- une date (au format JJ/MM/AAAA) ;
- un petit entier : pour les champs booléens (0/1 ; vrai/faux ; oui/non).

Un nombre maximum de caractères est également défini en même temps.

Le type *smallint* (petit entier) est utilisé par exemple pour le champ nommé DOC_AFFICHE qui est à 1 lorsque le professionnel décide que la notice est diffusable et à 0 sinon. Le champ peut se présenter sous forme de case à cocher :

Illustration 1 : Exemple de petits entiers avec les champs DOC_ANALYSE, DOC_AFFICHE et ASE_CONFID sur une vue de production

Lorsque le professionnel effectue une recherche depuis son espace, il peut choisir d'afficher les notices selon ces critères ou de ne pas en tenir compte en cochant la case souhaitée :

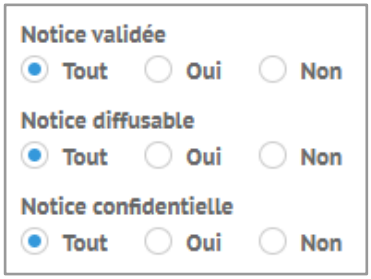


Illustration 2 : Exemple de petits entiers en recherche côté documentaliste

Du côté public, on retrouve d'autres champs booléens, comme par exemple les suivants :

Illustration 3 : Exemple de petits entiers en recherche côté utilisateur

Enfin, sur les notices documentaires, on retrouve ces mêmes informations :

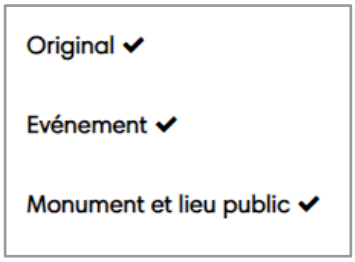


Illustration 4 : Exemple de petits entiers en notice côté utilisateur

La table standard ILS_DOC contient par exemple les champs DOC_TITRE ou DOC_AUTEUR qui sont de type alphanumérique. Les fonds des clients étant tous différents, il est parfois nécessaire de créer des champs spécifiques dans la structure de table définie pour un client. Ces nouveaux champs sont alors préfixés par un acronyme de trois caractères identifiant le client. Ceci permet à la lecture d'un script de table de les retrouver plus facilement. Par exemple, nous avons créé le champ ASE_ACTEUR pour le projet de la Succession d'Antoine de Saint Exupéry.

III.2.B. Les outils de construction des interfaces utilisés durant mon alternance

III.2.B.a) WebAdmin

A l'origine, toutes les interfaces étaient construites à l'aide d'un unique outil d'administration nommé *WebAdmin*. Cette interface servait à paramétrer les différentes vues composant le portail. Une vue est une page, elle est composée de trois écrans (même s'ils ne sont pas systématiquement tous utilisés) : l'écran de recherche, l'écran de résultat et l'écran de notice (ou saisie). Ils correspondent à la page de recherche documentaire, le premier écran étant celui sur lequel l'utilisateur va indiquer ses critères de recherche. Le second écran va afficher toutes les notices du fonds documentaire correspondant à la recherche effectuée. Enfin, le troisième écran va afficher le détail de la notice sur laquelle l'utilisateur aura cliqué sur l'écran de résultat. Parfois, on ne paramètre que l'écran de recherche, dans le cas d'une page d'accueil par exemple. D'autres fois, notamment dans les vues de panier, on ne paramètre que l'écran de résultat.

WebAdmin sert donc à paramétrer les interfaces publiques et de production. Les interfaces publiques sont celles qui vont être visibles par les utilisateurs du portail, sur lesquelles ils vont pouvoir notamment faire des recherches, comme décrit précédemment. Les vues de production vont quant à elles servir aux documentalistes et administrateurs qui vont les utiliser afin de gérer le fonds : ajout, modification, suppression de notices, gestion des utilisateurs, des dossiers thématiques, etc.

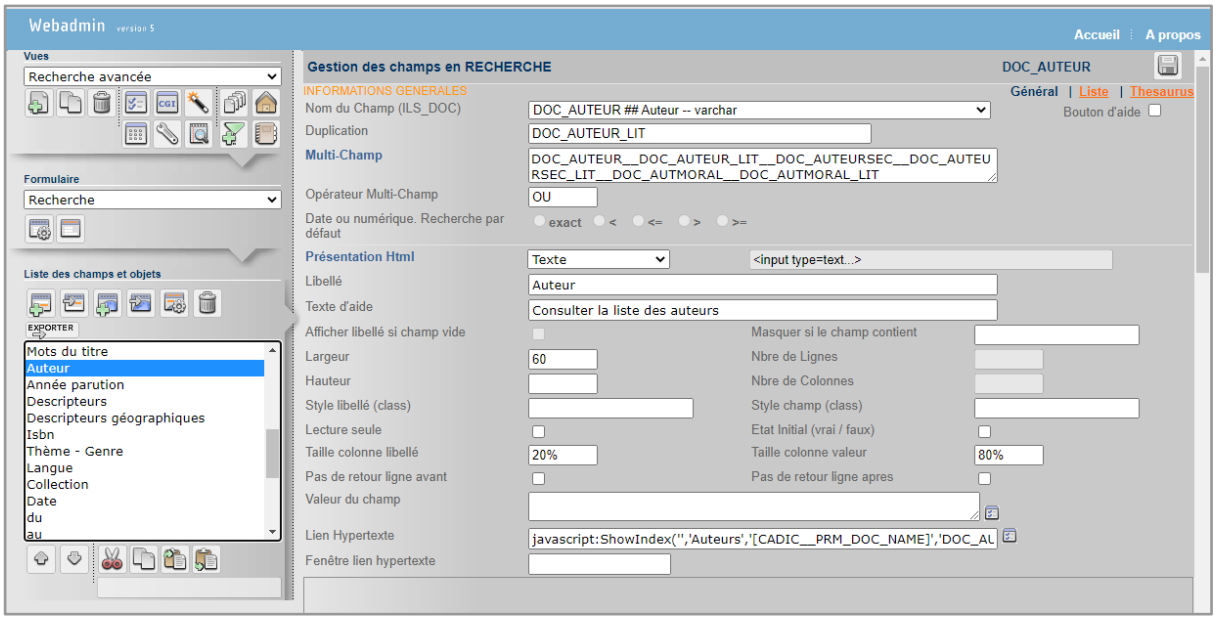


Illustration 5 : Paramétrage de l'écran de recherche de la vue de Recherche avancée sur WebAdmin

L'interface d'administration se présente en deux colonnes. En haut de la première colonne, on peut choisir la vue et modifier ses paramètres généraux. En dessous, on peut choisir l'écran à paramétrer parmi : recherche - résultat - notice. En bas, on trouve un menu déroulant

contenant chacun des éléments qui compose l'écran, dans leur ordre d'apparition. Lorsque l'on clique sur un élément, son paramétrage s'affiche sur la partie droite de l'écran, il est alors possible de le modifier. Il existe différents types d'éléments : les zones de texte (pour la recherche), l'affichage des valeurs des champs (pour les résultats et notice), les inclusions (pour insérer un morceau de code HTML par exemple), etc.

III.2.B.b) WebAdmin 2

Récemment, les interfaces de l'application ont évolué afin de revêtir une apparence plus moderne et ergonomique. Il est maintenant possible d'adapter la structure des pages selon la taille d'écran, ce qui rend le portail facilement utilisable sur smartphone et tablette. On parle alors d'interfaces *responsives*. Cette nouvelle version du produit se nomme *Zéphyr*, elle possède un socle technique plus performant et sécurisé par rapport aux versions antérieures.

Le paramétrage des interfaces publiques et de production est maintenant séparé. *WebAdmin* n'est quasiment plus utilisée et c'est *WebAdmin 2* qui est utilisée pour les vues publiques et *eXtenso Designer* pour les vues de production et d'administration.

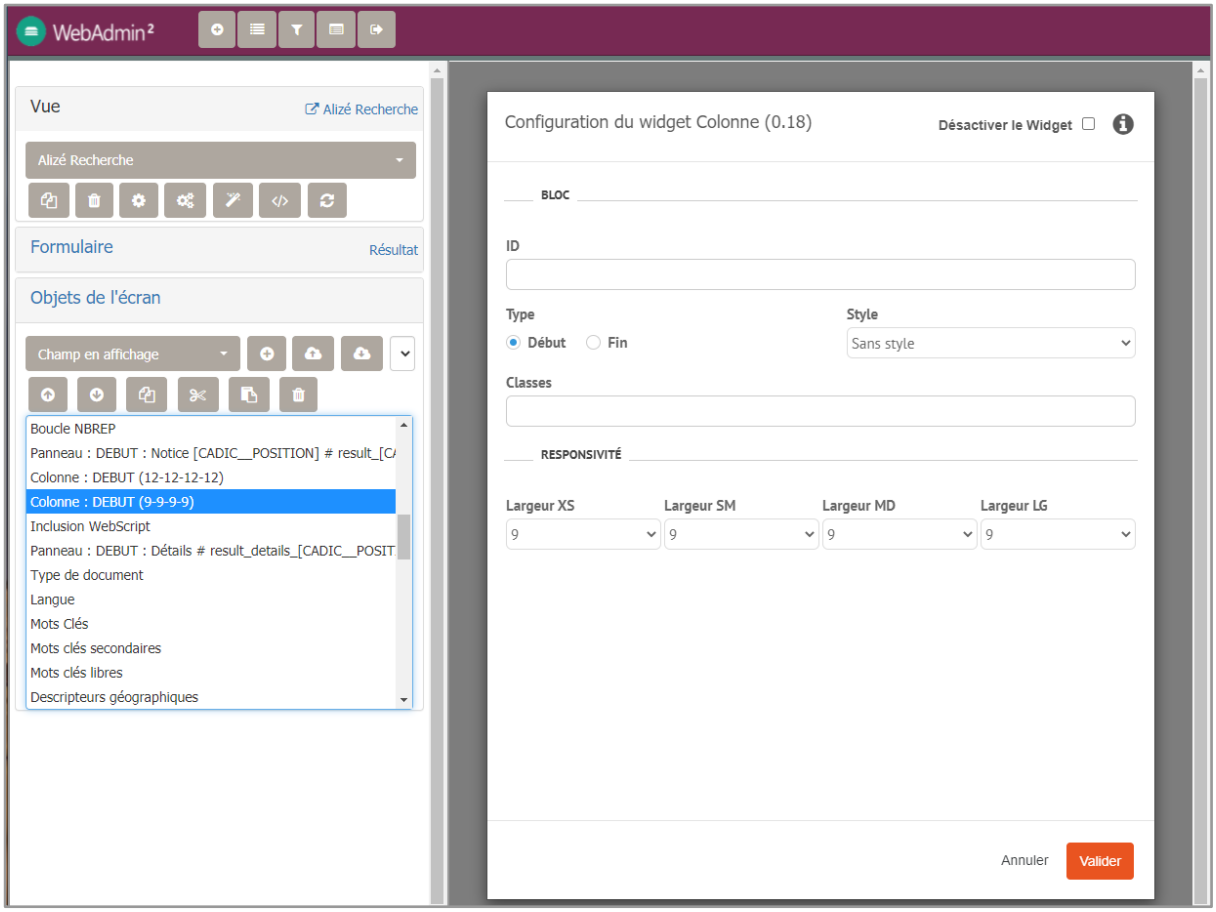


Illustration 6 : Paramétrage de l'écran de résultat de la vue Alizé Recherche sur WebAdmin 2

Le fonctionnement de *WebAdmin 2* est assez similaire à celui de *WebAdmin*. Il est possible d'assembler les éléments d'une vue comme on le souhaite en les faisant glisser dans

le menu déroulant. La grande différence est qu'il est maintenant possible d'organiser une page en colonnes et en panneaux. Cela simplifie la structuration de la page. Les colonnes prennent en compte les différentes tailles d'écran. En sachant qu'un écran a une largeur totale de 12, il est possible d'intégrer par exemple une colonne de quatre et une colonne de huit, les possibilités sont nombreuses. Pour chaque colonne, il est également possible de définir quatre largeurs selon la taille de l'écran : XL (grand écran), MD (écran moyen), SM (tablette) et XS (smartphone). Ainsi, on peut définir par exemple deux colonnes qui auront une largeur de six en XL et MD et qui seront donc côtes à côtes, mais qui en SM et XS auront une largeur de 12 et seront donc l'une au-dessus de l'autre. Cela permet de mieux optimiser l'espace et l'ergonomie de la page web afin qu'elle s'adapte à tous les supports.

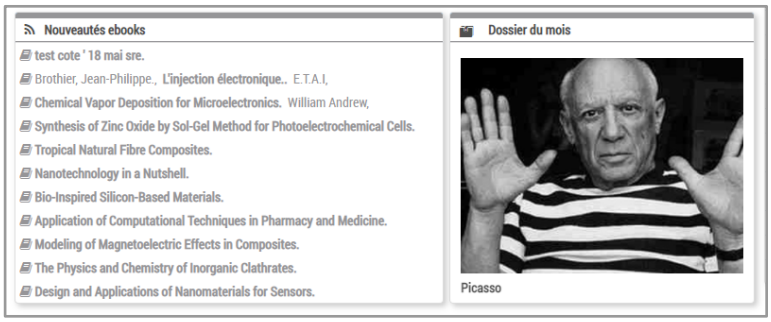


Illustration 7 : Exemple de deux colonnes de tailles 7 et 5 sur des écrans moyens et grands. Elles apparaissent côte à côte.

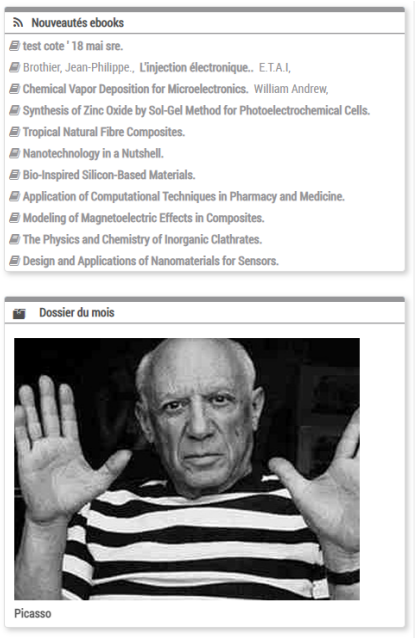


Illustration 8 : Ces mêmes colonnes sont de tailles 12 sur des écrans petits et de smartphones. Elles apparaissent l'une en dessous de l'autre.

III.2.B.c) eXtenso Designer

eXtenso Designer permet de paramétrer les vues de production, qui sont au format *eXtenso*. Ces vues étant réservées aux professionnels et ne faisant pas partie de l'interface publique, leur apparence est moins personnalisable.

Prenons l'exemple d'une vue de production documentaire. Sur l'écran de recherche il est possible de paramétrer comme sur *WebAdmin 2* des champs de recherche, tels que le titre, l'auteur, l'éditeur, etc. L'écran de résultat va se présenter différemment de celui d'une vue publique de recherche, il va être sous forme de tableau. Cela permet au professionnel de voir beaucoup de notices d'un simple coup d'œil et d'ainsi visualiser et gérer son fonds plus facilement. L'écran de saisie est le plus important. Il va contenir toutes les zones de saisie correspondant à chaque champ afin que le documentaliste les remplisse pour chaque notice.

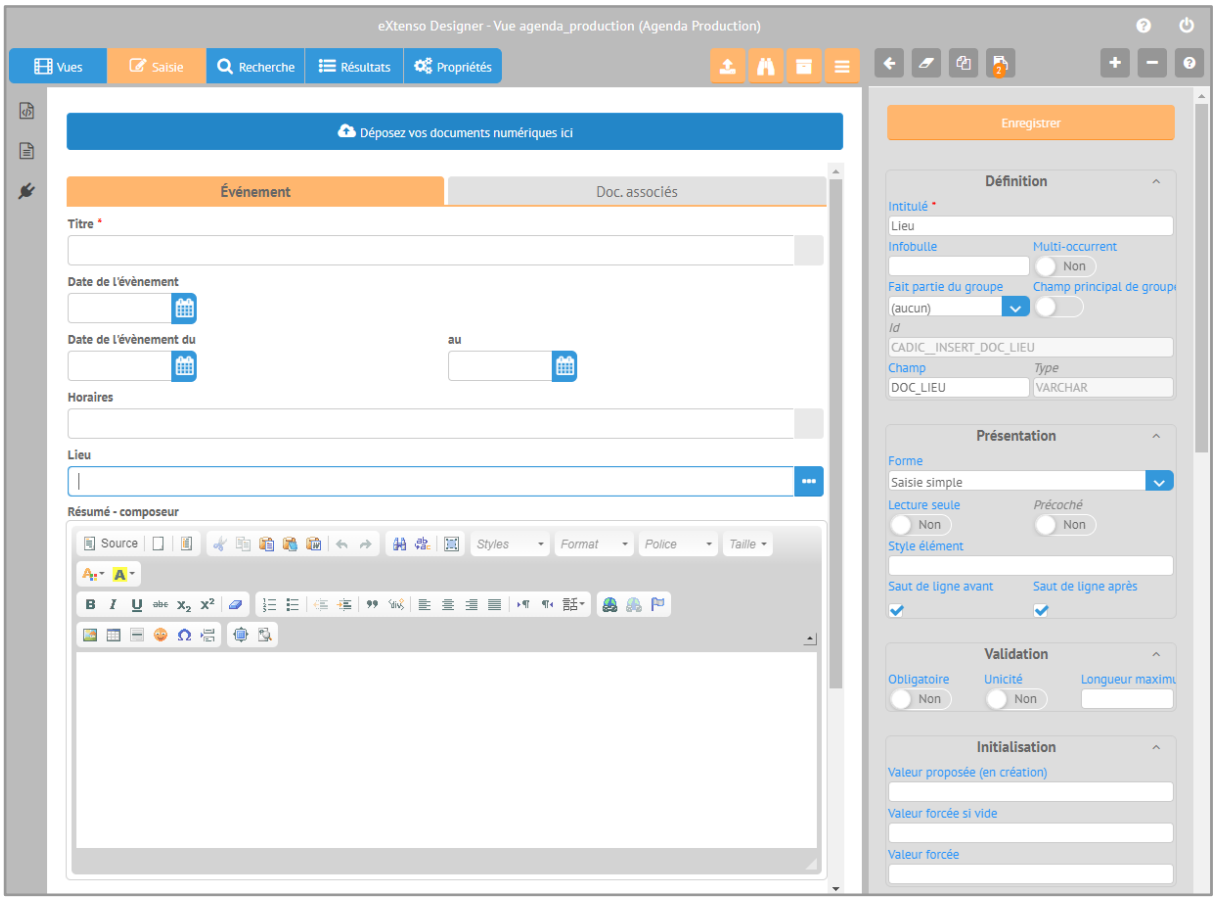
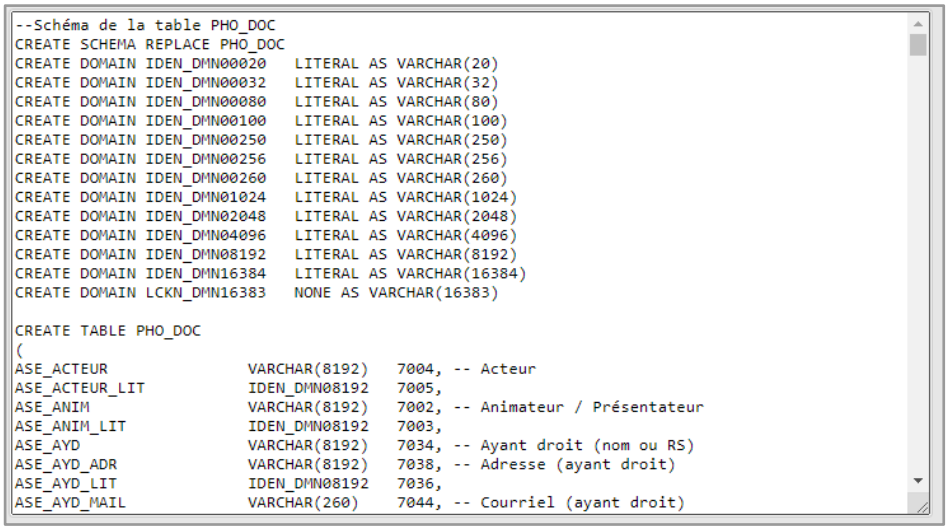


Illustration 9 : Paramétrage de l'écran de saisie de la vue de Agenda production sur eXtenso Designer

III.2.C. Les outils d'administration

SQLAdmin est un outil de gestion de bases de données, il permet de construire les différentes tables qui composent l'application ainsi que de les visualiser et les modifier. J'ai utilisé cet outil lorsque j'ai dû ajouter les champs spécifiques au client à la table PHO_DOC par exemple. Il est possible de cliquer sur "schémas des tables" puis sélectionner une table afin d'afficher son script de création / mise à jour. Pour PHO_DOC :



```
--Schéma de la table PHO_DOC
CREATE SCHEMA REPLACE PHO_DOC
CREATE DOMAIN IDEN_DMN00020   LITERAL AS VARCHAR(20)
CREATE DOMAIN IDEN_DMN00032   LITERAL AS VARCHAR(32)
CREATE DOMAIN IDEN_DMN00080   LITERAL AS VARCHAR(80)
CREATE DOMAIN IDEN_DMN00100   LITERAL AS VARCHAR(100)
CREATE DOMAIN IDEN_DMN00250   LITERAL AS VARCHAR(250)
CREATE DOMAIN IDEN_DMN00256   LITERAL AS VARCHAR(256)
CREATE DOMAIN IDEN_DMN00260   LITERAL AS VARCHAR(260)
CREATE DOMAIN IDEN_DMN01024   LITERAL AS VARCHAR(1024)
CREATE DOMAIN IDEN_DMN02048   LITERAL AS VARCHAR(2048)
CREATE DOMAIN IDEN_DMN04096   LITERAL AS VARCHAR(4096)
CREATE DOMAIN IDEN_DMN08192   LITERAL AS VARCHAR(8192)
CREATE DOMAIN IDEN_DMN16384   LITERAL AS VARCHAR(16384)
CREATE DOMAIN LCKN_DMN16383   NONE AS VARCHAR(16383)

CREATE TABLE PHO_DOC
(
ASE_ACTEUR              VARCHAR(8192)   7004, -- Acteur
ASE_ACTEUR_LIT          IDEN_DMN08192   7005,
ASE_ANIM                VARCHAR(8192)   7002, -- Animateur / Présentateur
ASE_ANIM_LIT            IDEN_DMN08192   7003,
ASE_AYD                 VARCHAR(8192)   7034, -- Ayant droit (nom ou RS)
ASE_AYD_ADR             VARCHAR(8192)   7038, -- Adresse (ayant droit)
ASE_AYD_LIT             IDEN_DMN08192   7036,
ASE_AYD_MAIL            VARCHAR(260)    7044, -- Courriel (ayant droit)
```

Illustration 10 : Début du script de création de la table PHO_DOC

S'affichent d'abord les différentes tailles utilisées par les champs ainsi que par les champs littéraux[1] correspondant. Ensuite, vient la liste de tous les champs de la table, avec leur taille, leur référence unique (c'est cette référence qui contient réellement les données, ainsi si l'on modifie le mnémonique d'un champ[2] les données ne seront pas perdues), et enfin leur libellé.

Lors de la réalisation du portail, j'ai souvent lancé des requêtes afin d'étudier les données. Par exemple, on peut lancer une recherche sur le fonds documentaire pour connaître les titres, auteur et éditeur des notices dont le champ *auteur* contient "Hugo" :

[1] Un champ littéral est associé à un champ de type alphanumérique afin de pouvoir rechercher par mot ou par expression dans les index. Par exemple, l'entrée "Victor Hugo" dans un champ alphanumérique sera indexée sous deux entrées : "Victor" et "Hugo". En revanche, l'entrée "Victor Hugo" dans un champ littéral sera indexée sous une seule entrée : "Victor Hugo".

[2] Le mnémonique d'un champ est son nom : DOC_TITRE ou DOC_AUTEUR par exemple.

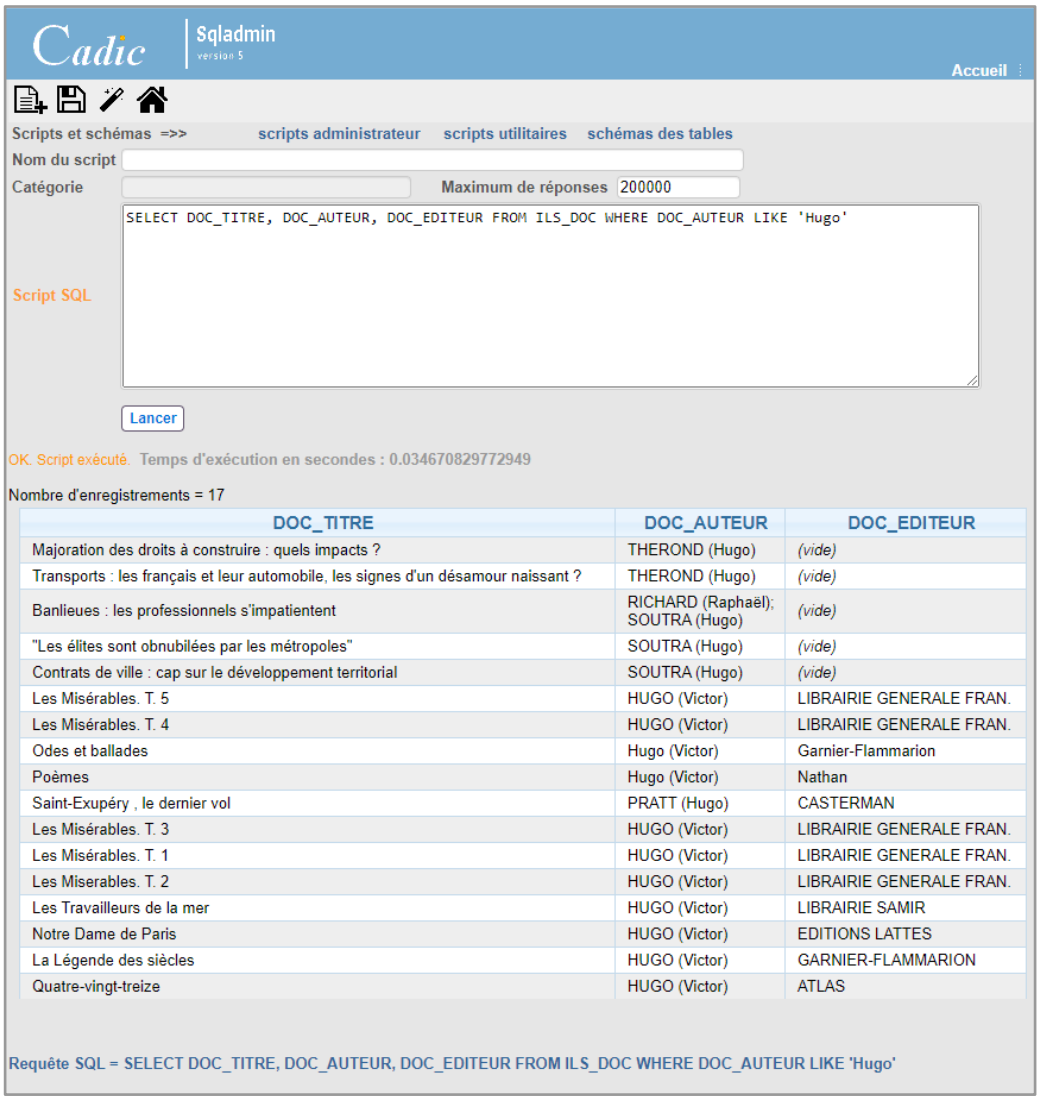


Illustration 11: Exemple de requête SQL et ses résultats

On obtient 17 résultats et les informations des trois champs demandés sont bien affichées.

Il existe d'autres outils d'administration permettant par exemple d'effectuer une même modification sur un grand nombre de notices à la fois, ou bien d'afficher des statistiques relatives à la fréquentation du site, ou encore de tester l'installation d'un site. Lors des diverses missions qui m'ont été confiées, j'ai manipulé ces outils ainsi que des fichiers de paramétrage, de HTML ou de CSS. L'ensemble m'a permis de comprendre comment fonctionnait l'application Cadic en profondeur, avec la gestion de base de données, mais aussi en surface, avec la gestion de l'affichage de l'interface.

IV. Travail réalisé

IV.1. Le projet Succession Saint Exupéry – d'Agay

IV.1.A. Étapes et méthode AGILE de gestion de projet

Chez Cadic Services, la méthode de gestion de projet suivie est une adaptation de la méthode AGILE. Cette démarche consiste à découper le projet en *sprints*[3], avec des objectifs à remplir et un livrable à la fin de chaque *sprint* qui peut durer par exemple deux ou trois semaines. Ainsi, le projet dans sa globalité peut toujours être réajusté et les priorités revues. Chez Cadic, les spécifications sont tout de même définies entièrement en début de projet, et les paramétrages réalisés ensuite, pour éviter trop d'allers-retours. Les fonctionnalités du portail sont testées par le client lors de la phase de recette, puis les éventuelles corrections sont effectuées avant d'être contre-testées par le client. Cette méthode permet plus de communication et plus de visibilité sur les réalisations. Les ajustements sont effectués rapidement et le produit répond ainsi vraiment aux besoins du client.

Au début de chaque journée de travail, tout le service opérationnel se réunit en visioconférence avec le responsable du service de recherche et développement ainsi que le directeur de Cadic. Lors de cette réunion, chacun présente rapidement ce qu'il a fait la veille et ce qui est prévu pour la journée. Nous profitons de ce créneau pour communiquer sur des nouvelles importantes, comme des évolutions dans l'application ainsi que pour demander de l'aide si nous sommes bloqués sur certains problèmes. Cette mise en commun permet de fluidifier la communication et de répandre l'information. Ainsi, lorsqu'une correction est effectuée, tout le monde est au courant. Il nous arrive de ne pas assister à la réunion, par exemple lorsque nous sommes en intervention ou avec un client.

Pour le projet Succession Saint Exupéry – d'Agay, la responsable du planning en a élaboré un en détaillant les avancées à effectuer selon les dates. Les trois acteurs principaux du projet sont, côté MOA, notre contact à la Succession et, côté MOE, ma tutrice et moi-même. L'équipe de R&D de Cadic a également été mobilisée pour la réalisation de ce projet, notamment afin de développer le module multimédia.

La première étape du projet à laquelle j'ai assistée est celle des spécifications. Celles-ci se sont déroulées en présentiel sur quatre journées entre le 11 et le 24 février 2021. Ensuite, nous avons rédigé les documents de spécifications et fait des visioconférences avec le client afin de confirmer certains points. Ces documents ont été envoyés à la Succession afin de valider les demandes et passer à la phase de paramétrage. Celle-ci a duré de fin mars à fin avril. Durant ce mois, nous avons, avec Jennifer, utilisé les outils d'administration de Cadic afin de paramétrer le portail et les différentes pages qui le composent : la page d'accueil et la vue de recherche notamment. Nous avons également dû faire une reprise des données. Cela consiste à

[3] Un *sprint* est un cycle de développement au cours duquel des tâches vont être réalisées.

créer un gabarit Excel contenant des titres de colonnes correspondant au mnémonique de chaque champ (par exemple ASE_ACTEUR pour le champ “Acteur”) qui sera ensuite rempli par les métadonnées du client. Un champ clef (l’identifiant) va permettre de relier chaque notice à chaque document (image, vidéo, audio ou PDF) également fourni par le client. Cette reprise permet d’intégrer les données à la base de données du site.

Afin de commencer le paramétrage, nous avons installé un site standard sur le serveur dédié au projet. Pour cela, nous suivons une procédure consistant à remplir des fichiers de paramètres et à lancer un programme exécutable. Nous avons également effectué une duplication du site afin de disposer d’un environnement dédié aux développements, sur lequel l’équipe de recherche et développement a pu travailler sans avoir d’incidence sur notre paramétrage.

Ensuite, il a fallu définir les besoins afin de faire évoluer le module multimédia que nous allions développer dans le cadre de ce projet. Pour cela, nous nous sommes appuyés sur les demandes du client ainsi que sur les notions de sécurité et de droits concernant les médias. Nous avons fait plusieurs réunions avec l’équipe de recherche et développement afin de définir concrètement ce dont nous avions besoin et ce qui allait être développé.

Afin d’ajouter les champs spécifiques au client (par exemple *acteur, réalisateur, scénariste*) nous avons mis à jour le script des tables en SQL. Nous avons également ajouté les champs anglais correspondant à des champs déjà existants ainsi que des champs d’autorité. Ces derniers sont utilisés dans les listes d’autorité, qui sont des aides à la saisie pour le documentaliste. Il existe des listes fermées, elles contiennent une liste de valeurs, et le champ correspondant ne peut pas contenir une valeur qui n’est pas dans la liste. Une liste fermée sera utilisée par exemple pour le champ *Type de document*. Dans une liste ouverte, il est possible d’entrer dans le champ une nouvelle valeur et celle-ci sera alors ajoutée à la liste d’autorité. Elle sera utilisée par exemple pour le champ *Propriétaire*. Il existe un troisième type de liste d’autorité, la liste d’aide. Dans celle-ci, il est également possible d’entrer dans le champ une valeur qui n’est pas dans la liste, mais celle-ci ne sera pas ajoutée à la liste. Une même liste d’autorité peut être utilisée pour plusieurs champs. Par exemple, nous avons créé la liste *Personnes* qui aide à la saisie dans les champs *Propriétaire, Licencié* et *Ayant-droit*.

Nous avons ensuite effectué les paramétrages des différentes vues sur *WebAdmin 2* et *eXtenso Designer*. Tout au long des réalisations, nous avons communiqué avec le client afin de définir plus précisément certains points, en le faisant choisir entre deux captures d’écran par exemple, ou lorsqu’il voulait modifier un libellé ou une traduction. Il nous a également fourni des listes pour certains critères de recherche.

Lorsque les paramétrages et le module multimédia ont été finalisés fin avril, nous avons présenté le site au client lors d’une journée de prise en main. Ensuite, il a disposé d’une période de recette de trois semaines durant laquelle il a pu tester l’ensemble de l’application, en s’appuyant sur un cahier de tests que nous lui avions préparé au préalable. Le cahier de tests

est un document comprenant chaque action possible sur le portail, il permet d'indiquer s'il y a des anomalies. Ces anomalies sont déclarées sur une plateforme spécifique qui permet au client de déposer des tickets que nous pouvons ensuite traiter, en répondant à ses questions ou en indiquant lorsque nous avons corrigé une anomalie.

A la fin du projet, une fois que toutes les anomalies ont été corrigées, nous sommes passés à la mise en production. Ce site étant hébergé sur nos serveurs, nous n'avons pas eu à le copier d'un serveur à un autre. Nous avons suivi une procédure d'installation consistant à indiquer des paramètres tels que l'URL et à lancer un programme exécutable d'installation, qui va se charger d'installer le contenu de l'environnement de test sur l'environnement de production. Enfin, nous avons vérifié la bonne installation et le bon fonctionnement du site en production.

IV.1.B. Paramétrage des différentes vues de l'application

La première vue à créer était une vue de production générique, afin de pouvoir déjà insérer des données de test sur le portail qui allaient permettre de tester les différentes fonctionnalités paramétrées avant la reprise du fonds. Plus tard, une fois cette vue finalisée et optimisée, nous l'avons dupliquée afin de créer les vues de production par type de documents : *audiovisuel, imprimés, manuscrits* et *défaut*. Pour cette vue, nous nous sommes d'abord concentrées sur l'écran de saisie, qui nous a permis d'entrer toutes les informations concernant les données d'exemples :

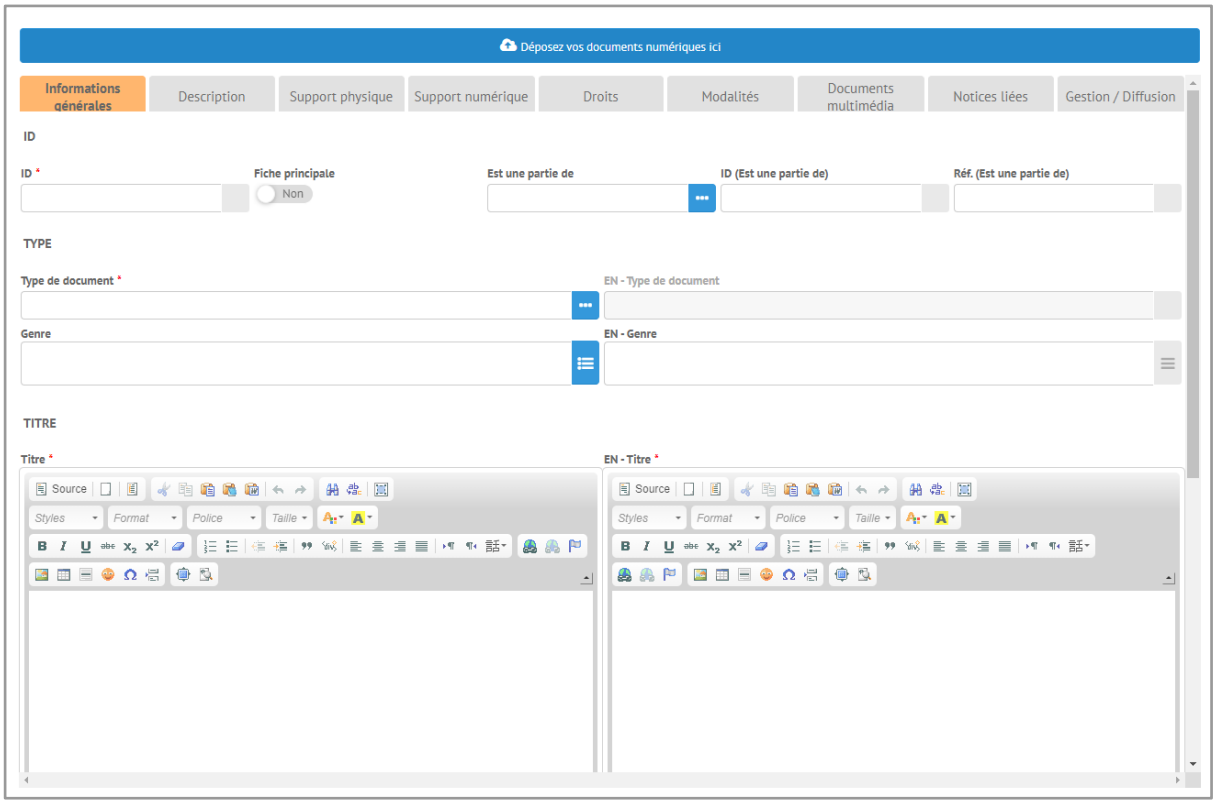


Illustration 12 : Exemples de champs dans la vue de production générique eXtenso

Les écrans de recherche et de résultat de cette vue sont également importants car ils seront utilisés par le client lorsqu'il voudra effectuer des recherches sur l'ensemble de son fonds.

IV.1.B.a) Vues publiques

Pour le paramétrage des vues publiques, nous avons utilisé *WebAdmin 2*. L'écran de recherche de la vue de recherche se compose d'une colonne contenant les conseils de recherche et le lien vers l'historique, et d'une colonne contenant les champs de recherche. La barre de navigation permet de lancer la recherche, d'effacer les critères et d'afficher le panier. Les champs de recherche peuvent être de différents types :

Champ texte :

Illustration 13 : Exemple d'un champ texte

Menu déroulant :

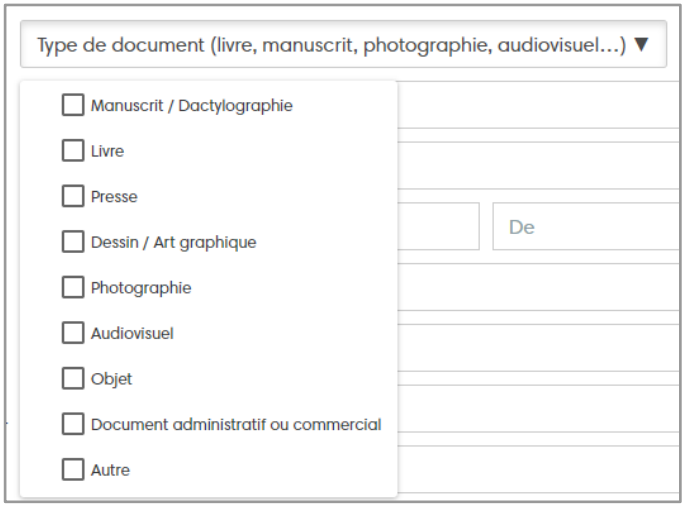


Illustration 14 : Exemple d'un menu déroulant

Champ booléen :

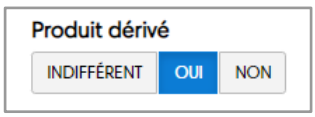


Illustration 15 : Exemple d'un champ booléen

Pour certains champs textes, il est possible d'accéder à un index pour faciliter la recherche :

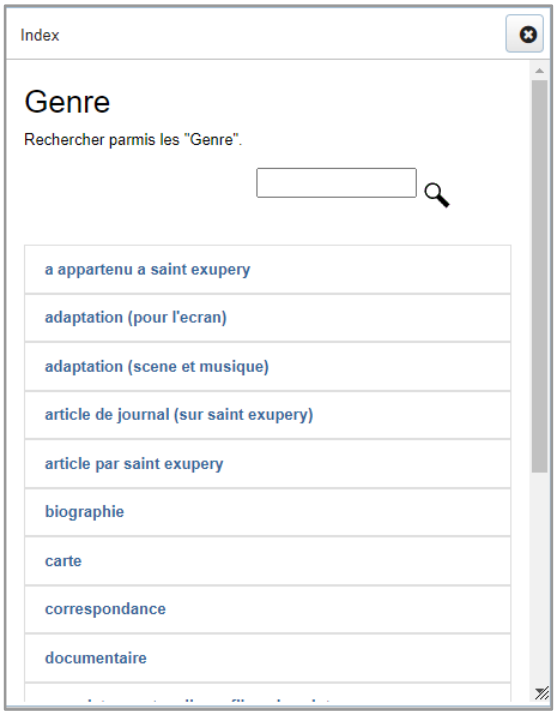


Illustration 16 : Exemple d'un index sur un champ

L'autocomplétion permet de proposer à l'utilisateur les différentes valeurs possibles (présentes dans l'index) à partir de ce qu'il commence à écrire dans le champ :

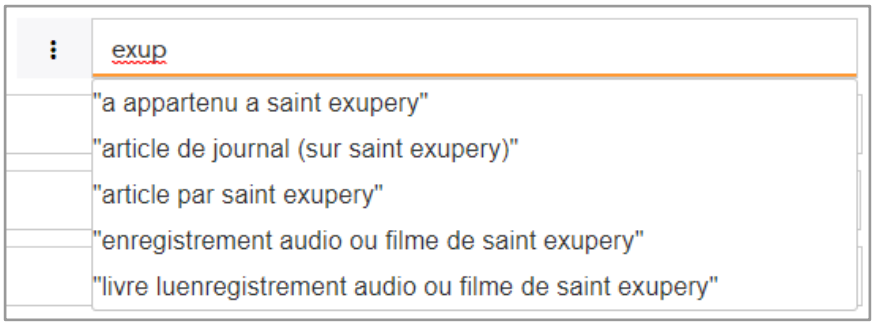


Illustration 17 : Exemple d'une autocomplétion de champ

Enfin, sont proposées des options de recherche, telles que les opérateurs entre les champs ou le nombre de caractères maximum entre les mots dans les expressions exactes :

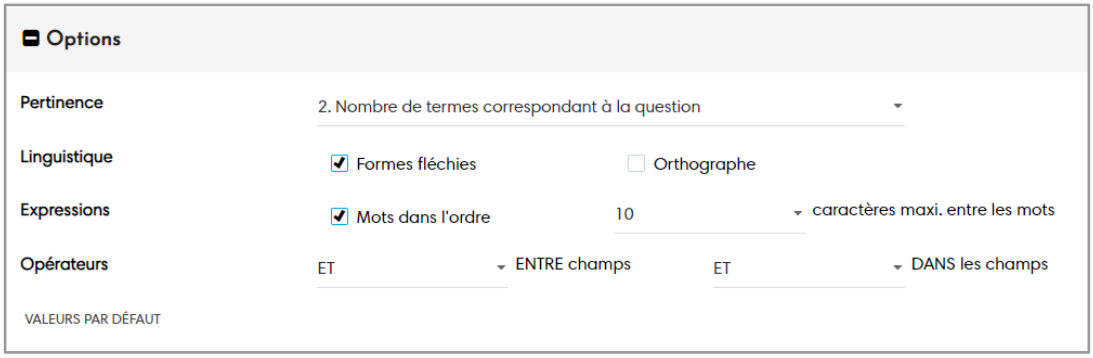


Illustration 18 : Options de recherche

Si l'utilisateur choisit l'opérateur *"OU" entre les champs* et qu'il effectue une recherche sur "Année = 2020" et "Lieu = France", alors il obtiendra en résultat les notices qui ont pour année 2020 et les notices qui ont pour lieu la France. Si à l'inverse il choisit *"ET"*, alors il obtiendra uniquement les notices qui ont pour année 2020 et pour lieu la France. Si l'utilisateur choisir *"ET" dans les champs* et qu'il entre les mots-clefs "avion" et "témoignage", alors il obtiendra les notices contenant ces deux mots-clefs, alors qu'avec *"OU"*, il obtiendra également les notices qui ne contiennent qu'un seul des deux mots-clefs.

A ce stade du projet, nous pouvions déjà tester le bon fonctionnement de ces différents champs de recherche grâce à nos données d'exemple.

L'écran de résultat de la vue de recherche se compose également de deux colonnes. Sur celle de gauche, on trouve les facettes qui servent à affiner les résultats. Sur celle de droite, on retrouve la liste de résultats, présentée sous forme de mosaïque d'images ou bien de liste.

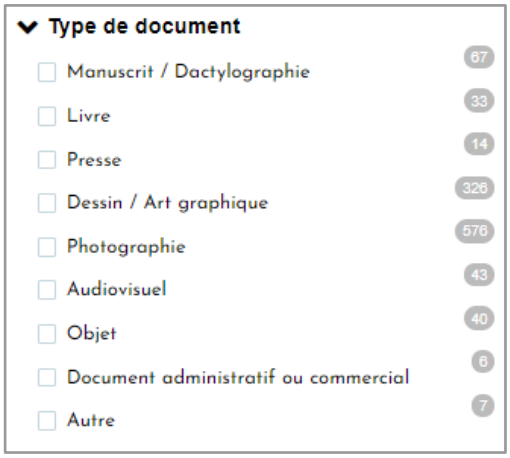


Illustration 19 : Exemple de facettes sur les types de documents en français

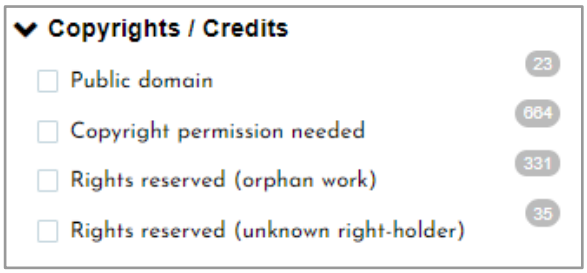


Illustration 20 : Exemple de facettes sur les droits et crédits en anglais

Si l'on effectue une recherche sur un terme, celui-ci sera mis en valeur en écran de résultat. J'ai paramétré cela dans la valeur du champ dans *WebAdmin 2* et j'ai déterminé la couleur et le style que celui-ci allait prendre en CSS.

Illustration 21 : Exemple d'un résultat en liste pour une recherche sur genre = "biographie"

Illustration 22 : Exemple d'un résultat en mosaïque pour une recherche sur mots du titre = "prince"

Au-dessus des résultats, j'ai paramétré une barre de navigation, permettant d'effectuer des actions comme revenir à la recherche, ajouter les notices au panier, afficher un diaporama ou imprimer les résultats. Ensuite, sont indiqués le nombre de résultats, les critères de recherche ainsi qu'un menu déroulant permettant de choisir comment trier les résultats (par date, par titre, etc.).

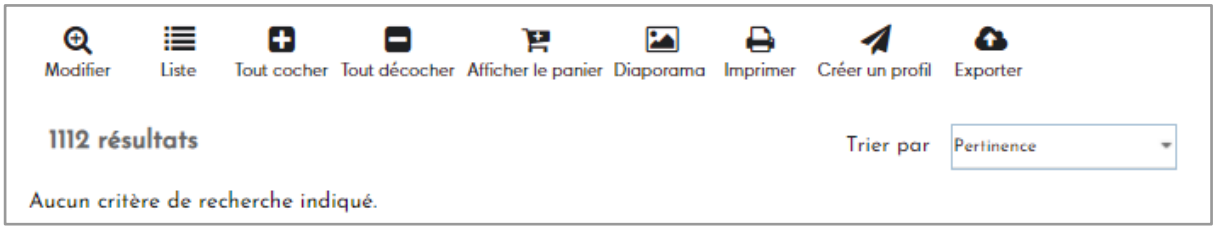


Illustration 23 : Barre de navigation, nombre de résultats, critères de recherches et options de tri

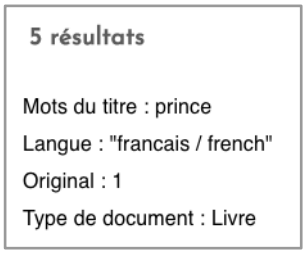


Illustration 24 : Exemple de critères de résultats affichés sur un écran de résultats

J'ai également paramétré des rebonds[4] qui permettent de relancer la recherche sur un terme, tel que le type de document ou le genre.

Illustration 25 : Exemple de rebond en liste de résultats. Au survol de "Dessin / Art graphique" le texte devient gris et il est possible de cliquer dessus afin de relancer une recherche sur toutes les notices ayant pour type de document "Dessin / Art graphique".

[4] Un rebond est un texte cliquable permettant de relancer une recherche.

Sur chaque notice, nous avons paramétré des boutons sous le visuel, permettant d'ouvrir la notice, de l'ajouter au panier et d'ouvrir le document PDF lorsqu'il y en a un. Pour les utilisateurs de catégories "administrateur", "professionnel" et "interne à la Succession", nous avons mis un bouton permettant de télécharger le média, et pour les "partenaire" et "extérieur" un bouton permettant de commander le média.

Illustration 26 : En mosaïque, les boutons apparaissent au survol de l'image

Au clic sur le titre d'une notice ou sur le bouton d'accès à la notice, la notice complète s'affiche.

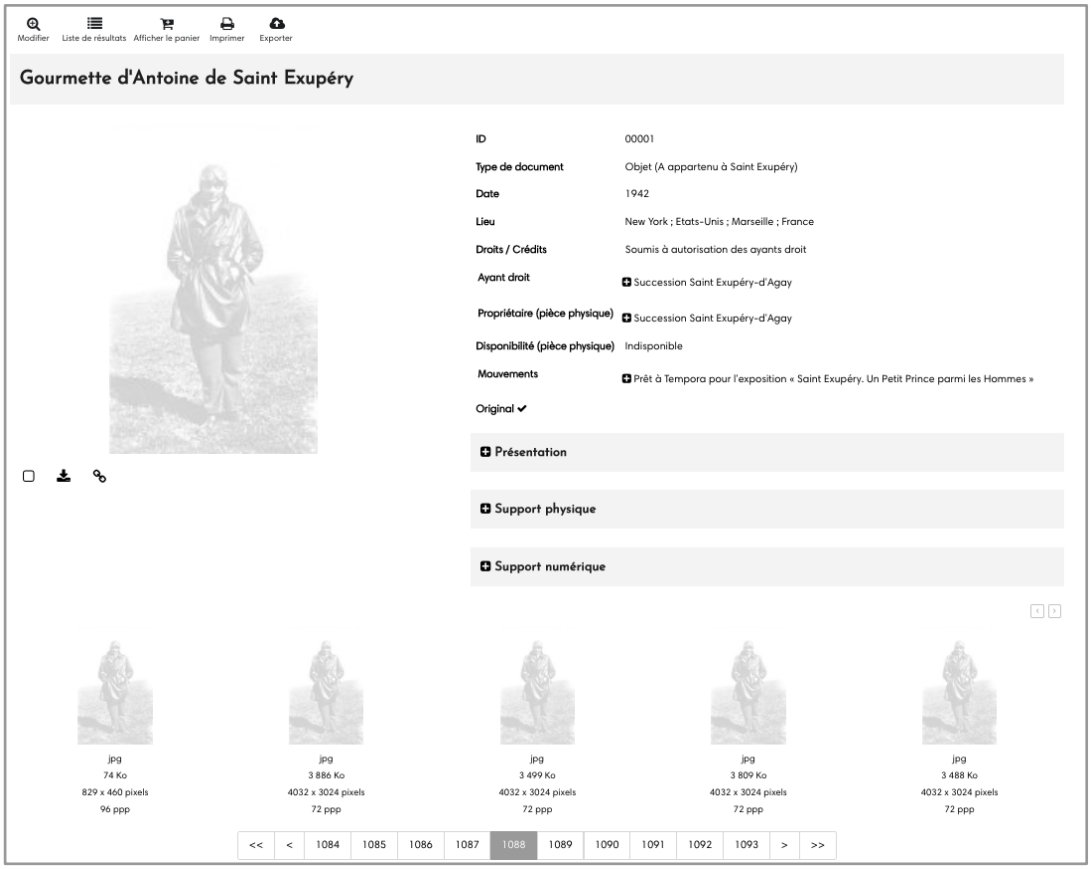


Illustration 27 : Exemple d'un écran de notice

Sur l'écran de notice, nous avons fait le choix de prendre toute la largeur des 12 colonnes pour afficher le titre. Ensuite, nous avons affiché le média en grand sur la gauche afin de mettre celui-ci en valeur. En dessous, on retrouve les boutons déjà présents en écran de résultat ainsi que le permalien, qui est un lien permanent permettant d'accéder à la notice. A droite, nous avons affiché la plupart des métadonnées liées au document. Pour certaines, nous les avons mises dans des panneaux repliés qu'il est possible d'ouvrir en cliquant sur le "+", comme les informations concernant le support numérique ou la description du document.

En dessous de la notice, nous avons paramétré un carrousel. Celui-ci permet d'afficher les notices secondaires qui correspondent en général à d'autres visuels de l'objet ou du document. J'ai pour cela écrit un modèle d'affichage en utilisant les langages TinyButStrong et HTML, permettant d'afficher l'image associée à la notice et quelques informations en dessous. Au clic sur l'image, j'ai ajouté un lien ouvrant une fenêtre popup contenant la notice complète.

Nous avons également créé une page d'accueil. Sur celle-ci, j'ai paramétré des widgets tels qu'un carrousel proposant une sélection de notices ou un éditeur présentant un texte de présentation du portail. J'ai également paramétré quelques champs de recherche afin de pouvoir lancer une recherche rapide directement à partir de la page d'accueil.

Une autre vue de recherche paramétrée pour ce projet est la recherche thématique. Celle-ci se base sur un référentiel contenant des thématiques et sous-thématiques, et le professionnel choisi d'associer ses notices à la thématique souhaitée. Au clic sur une thématique, un écran de résultat contenant les notices correspondantes s'ouvre. Nous avons également mis en place l'autopostage : au clic sur une branche, toutes les notices appartenant à une branche inférieure à celle-ci s'affichent.

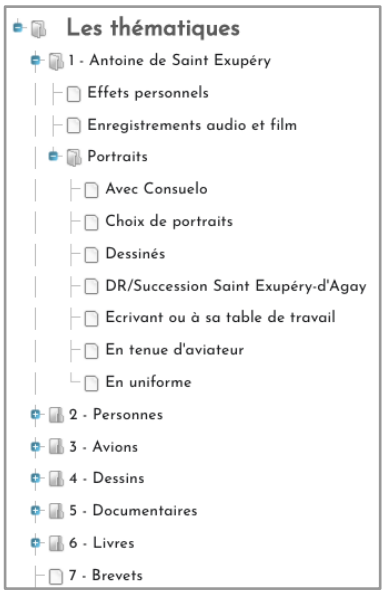


Illustration 28 : Exemples de thématiques et sous-thématiques

IV.1.B.b) Vues professionnelles

Le professionnel alimente son fonds à partir de l'espace numérique. Il peut effectuer des recherches sur son fonds, modifier ses notices existantes ou en créer de nouvelles dans cet espace. Nous avons créé des sous-dossiers pour chaque type de document, chacun associé à une vue de production créée au préalable sur *eXtenso Designer*. Nous avons ajouté d'autres dossiers, comme celui permettant d'afficher les notices de la sélection affichée en page d'accueil ou celui contenant les vidéos à traiter. En effet, ce projet avait comme particularité d'inclure des vidéos parfois très lourdes, de plusieurs gigaoctets. Afin de créer les notices correspondantes, il n'était pas possible de téléverser directement ces vidéos dans une vue de production. Nous avons donc demandé au client de les déposer dans un répertoire sur le serveur et nous avons fait tourner un *batch*[5] de nuit traitant les vidéos, en créant une version *streamable* pour l'affichage sur le portail ainsi que des vignettes. Le lendemain, le professionnel peut, à partir de ce dossier "vidéos à traiter", retrouver les notices pré-remplies contenant les vidéos, afin de les compléter et les publier.

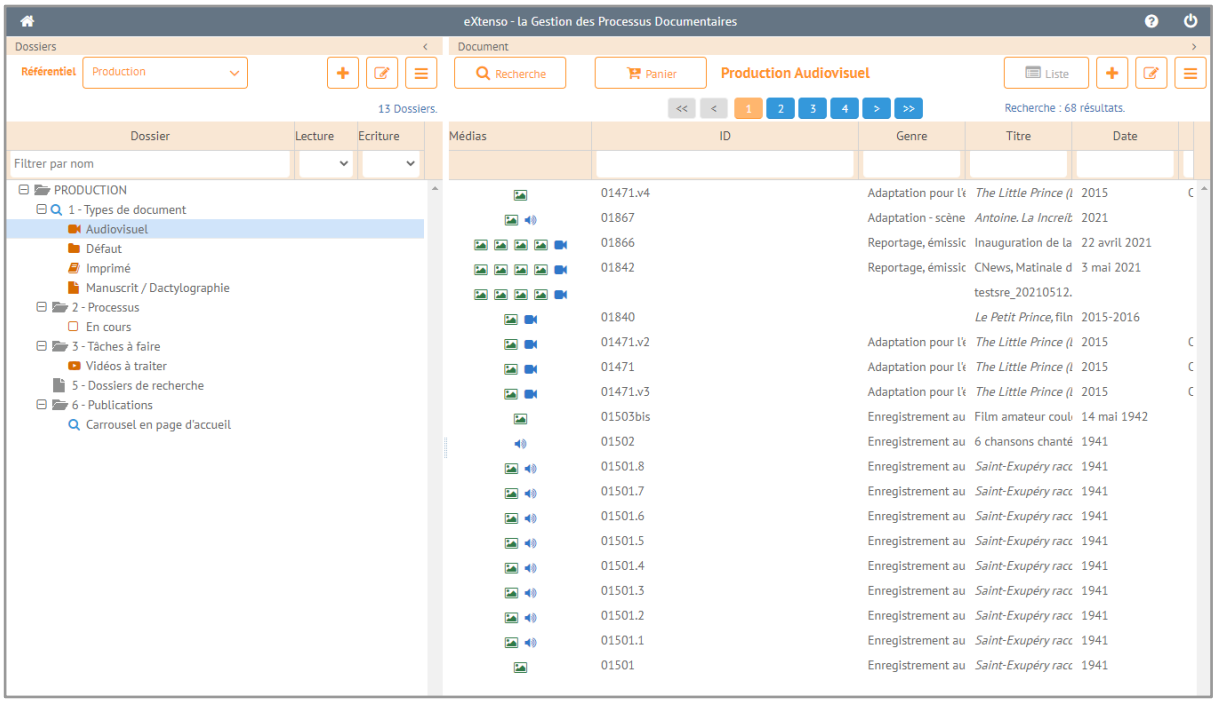


Illustration 29 : Espace numérique du professionnel

[5] Un *batch* est un programme permettant d'effectuer des actions automatisées.

Afin de gérer les commandes de médias, nous avons créé dans l'espace du professionnel des dossiers représentant le *workflow*[6] des commandes. Lorsqu'un utilisateur fait une demande, celle-ci se retrouve dans le dossier "en attente". Le professionnel peut alors afficher la demande et les différentes informations concernant le demandeur et l'utilisation prévue des médias que celui-ci a renseignée. Il peut ensuite décider de modifier le statut en "acceptée" ou "refusée". Si la commande est acceptée, une archive contenant le média demandé et une licence de téléchargement est générée pendant la nuit.

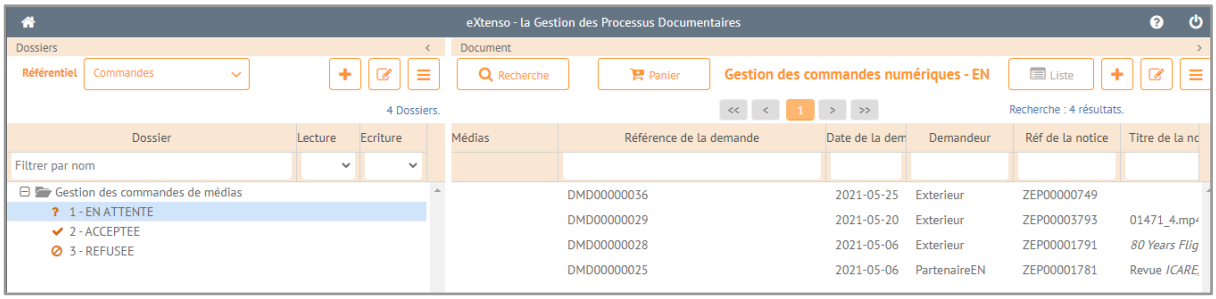


Illustration 30 : Espace de gestion des commandes de médias pour le professionnel

Le demandeur peut visualiser ses demandes et leur statut dans une vue. Lorsqu'une archive est prête, c'est ici qu'il trouve le lien permettant de la télécharger.

Illustration 31 : Affichage des commandes de médias pour le demandeur

[6] Un *workflow* est une suite de tâches effectuées par une personne ou par un groupe de personnes.

IV.1.C. Autres paramétrages

J'ai effectué de nombreux paramétrages pour la réalisation de ce portail internet. Parfois, il a fallu directement manipuler des fichiers plutôt que de passer par un outil d'administration. C'est le cas par exemple pour le style des différents éléments du site, défini dans un fichier CSS, mais aussi pour le bandeau, présent en haut de toutes les pages du site, et le pied de page, présent en page d'accueil.

Pour ces derniers, j'ai dû éditer des fichiers en HTML. Dans le bandeau, j'ai inséré le logo du site, permettant également de revenir à l'accueil lorsqu'on clique dessus. Il est également possible de dérouler le menu "mon compte" qui contient des liens vers l'historique de recherche, les paniers sauvegardés, les commandes de médias ou les espaces pour les professionnels par exemple. Le bandeau contient également un menu permettant de passer de la version française du site à la version anglaise et vice-versa. Dans le pied de page, j'ai intégré des liens vers les autres sites de la Succession Saint Exupéry – d'Agay ainsi que vers leurs différents réseaux sociaux.

IV.1.D. Développement du module multimédia

Nous avons développé le module multimédia, afin de le rendre compatible avec la dernière version du logiciel. Nous avions pour cela différentes contraintes à respecter. Nous avons travaillé en collaboration avec l'équipe de recherche et développement, qui a développé de nouveaux outils et widgets, que nous nous sommes ensuite chargées de mettre en place. Nous leur avons donné des retours et demandé des améliorations lorsque cela était nécessaire.

Notre client possédant un fonds multimédia très varié, il fallait harmoniser l'utilisation de l'application afin que le fonctionnement soit le même pour chaque type de média. La particularité de ce fonds est d'être très visuel, il fallait mettre cela en valeur.

IV.1.D.a) Types de médias

Le fonds est composé de photographies, de dessins, de vidéos, d'audios, de livres ainsi que de manuscrits. Ce sont des œuvres réalisées par Antoine de Saint Exupéry, des médias sur lesquels il apparaît ou bien à son sujet. Le fonds est très varié et représentatif de la vie de Saint Exupéry, entre ses écrits, notamment *Le Petit Prince*, ses dessins et son métier d'aviateur.

Le cas le plus simple est celui où une image est associée à la notice. Nous avons fait le choix d'associer à chaque notice un seul média. S'il y a plusieurs images ou médias associés à une même notice, le professionnel crée autant de notices que de médias et en fait des notices secondaires associées à une notice principale. Ainsi, seule la notice principale apparaîtra dans les résultats et les notices secondaires seront accessibles à partir du carrousel présent sur l'écran de notice, qui permettra au clic d'afficher les notices secondaires dans une fenêtre popup.

Il est possible d'associer une vidéo à une notice directement dans le formulaire de saisie si elle n'est pas trop lourde. Dans le cas contraire, le professionnel la dépose dans un répertoire spécifique sur le serveur. Un *batch* de nuit se lance afin que le lendemain une notice vide soit disponible dans l'espace du professionnel. Celle-ci ne contiendra qu'un titre et un média, le reste doit être complété par le professionnel avant d'être publié. Lorsqu'une vidéo est intégrée à l'application, quatre vignettes sont générées automatiquement à partir de quatre moments aléatoires dans la vidéo. Le professionnel peut ensuite choisir d'en supprimer trois afin de garder uniquement la vignette qu'il souhaite conserver. Sur le portail, c'est cette vignette qui s'affichera et il sera possible de cliquer dessus afin de lancer la lecture de la vidéo.

Un système similaire est utilisé pour les fichiers PDFs, qui sont notamment des manuscrits dans notre cas. Une image est générée automatiquement à partir de la première page du PDF et est associée à la notice.

Enfin, pour les fichiers audios, aucune image n'est associée aux notices. Nous gérons cela au niveau de l'affichage public. Lorsqu'une notice contient un audio, nous affichons un visuel par défaut au-dessus d'une barre de lecture du fichier audio.

IV.1.D.b) Notions de sécurité

Pour chaque notice, le professionnel décide quelles catégories d'utilisateurs ont le droit d'y accéder ainsi que celles qui ont le droit d'accéder aux médias. Par exemple, la description d'un document peut être visible par tous, mais la visibilité du document en lui-même peut être limitée aux utilisateurs de catégories "professionnel" et "interne". Un premier contrôle est donc effectué à ce niveau. Les widgets de lecture vidéo, lecture audio et visualisation des images intègrent ce contrôle. Ainsi, le média ne s'affiche pas si l'utilisateur ne fait pas partie d'une catégorie autorisée, nous avons fait en sorte d'afficher à la place un visuel par défaut, fourni par le client.

La contrainte suivante était d'empêcher les téléchargements directs des médias. Il n'est donc pas possible de faire un clic-droit sur les différents médias. Dans le lecteur de PDFs, il n'est pas non plus possible de les télécharger directement. Pour les catégories "administrateur", "professionnel" et "interne à la Succession", nous avons ajouté à chaque notice un bouton spécifique permettant de télécharger le média. Celui-ci affiche d'abord des conditions de téléchargement, puis génère une archive contenant le média sélectionné ainsi que la licence de téléchargement contenant le nom des fichiers et les droits qui y sont liés. Pour les catégories "partenaire" et "extérieur", il est possible d'effectuer une commande de médias. A la place du bouton de téléchargement, c'est un bouton de commande qui est affiché pour eux. En cliquant dessus, un formulaire pré-rempli s'affiche, contenant leurs informations d'utilisateur ainsi que les informations concernant le média. Ils doivent ensuite renseigner dans un champ l'utilisation prévue des médias, et cocher une case pour accepter les conditions d'exploitation des documents. L'administrateur et les professionnels ont accès à un espace où ils peuvent consulter toutes les demandes et les accepter ou les refuser ainsi qu'y ajouter un commentaire.

L'utilisateur "partenaire" ou "extérieur" a également un espace pour consulter l'état de ses commandes. Ensuite, un *batch* se lance chaque nuit et génère des archives pour chaque commande de médias acceptée. L'utilisateur peut récupérer cette archive en cliquant sur un lien dans son espace.

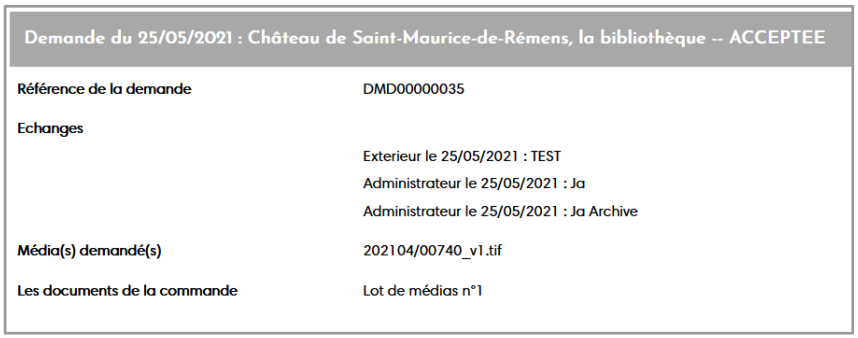


Illustration 32 : Exemple d'affichage d'une commande de média acceptée, avec en bas le lien pour télécharger l'archive : "Lot de médias n°1"

Lorsqu'une image est intégrée à l'application, que ce soit en tant que document associé ou vignette, elle est stockée dans les serveurs sous son format original, mais également sous trois résolutions : haute définition, écran et vignette. Nous n'affichons jamais l'image originale ou la haute définition directement sur le portail public, afin de rester cohérent avec les notions de sécurité, mais aussi pour que le chargement des pages ne soit pas trop long. Nous affichons en général la résolution écran. La vignette est celle qui est utilisée par exemple lors de l'envoi d'un panier par mail.

IV.1.D.c) Rendu visuel de l'affichage public

Afin de mettre en valeur ce fonds multimédia, nous avons proposé sur l'écran de résultats deux visualisations possibles : une liste et une mosaïque. La liste est l'affichage classique des résultats, avec les notices les unes sous les autres, le titre au-dessus, l'image associée à la notice sur la gauche, et les métadonnées principales sur la droite. L'affichage en mosaïque permet de mettre l'image en valeur. Les images ainsi que les lecteurs vidéos et audios se combinent de façon à occuper toute la place sur la page. Sous chaque visuel, nous avons affiché le titre, dont la fin est remplacée par "..." s'il dépasse une ligne, les noms des contributeurs ainsi que la date. Les métadonnées sont moins mises en valeur que dans le mode liste, au profit du visuel. Les boutons permettant d'effectuer les actions d'accès à la notice, d'ajout au panier, de téléchargement et de commande apparaissent au survol d'une image.

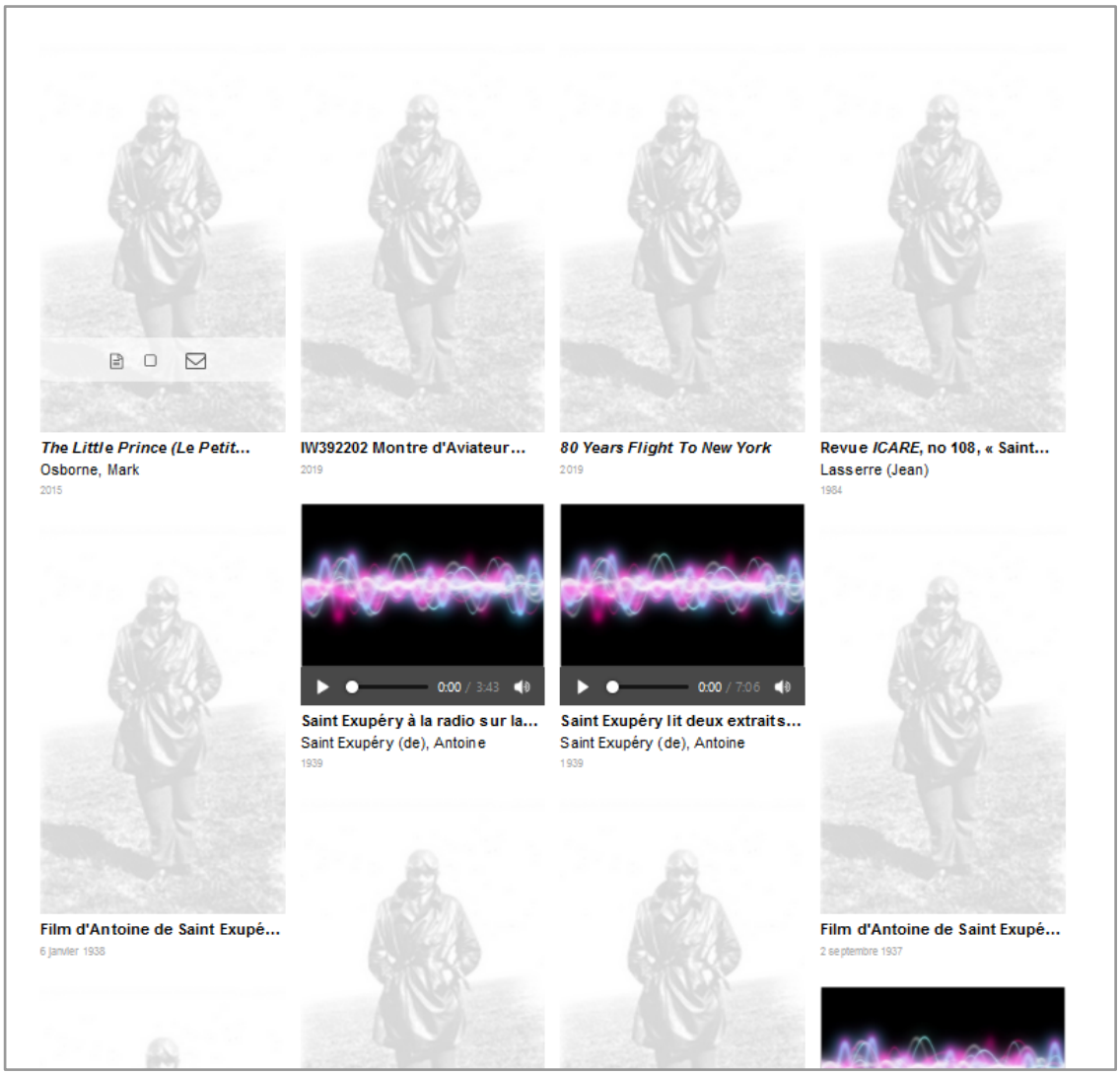


Illustration 33 : Exemple de mosaïque avec les boutons apparaissant au survol sur la première notice

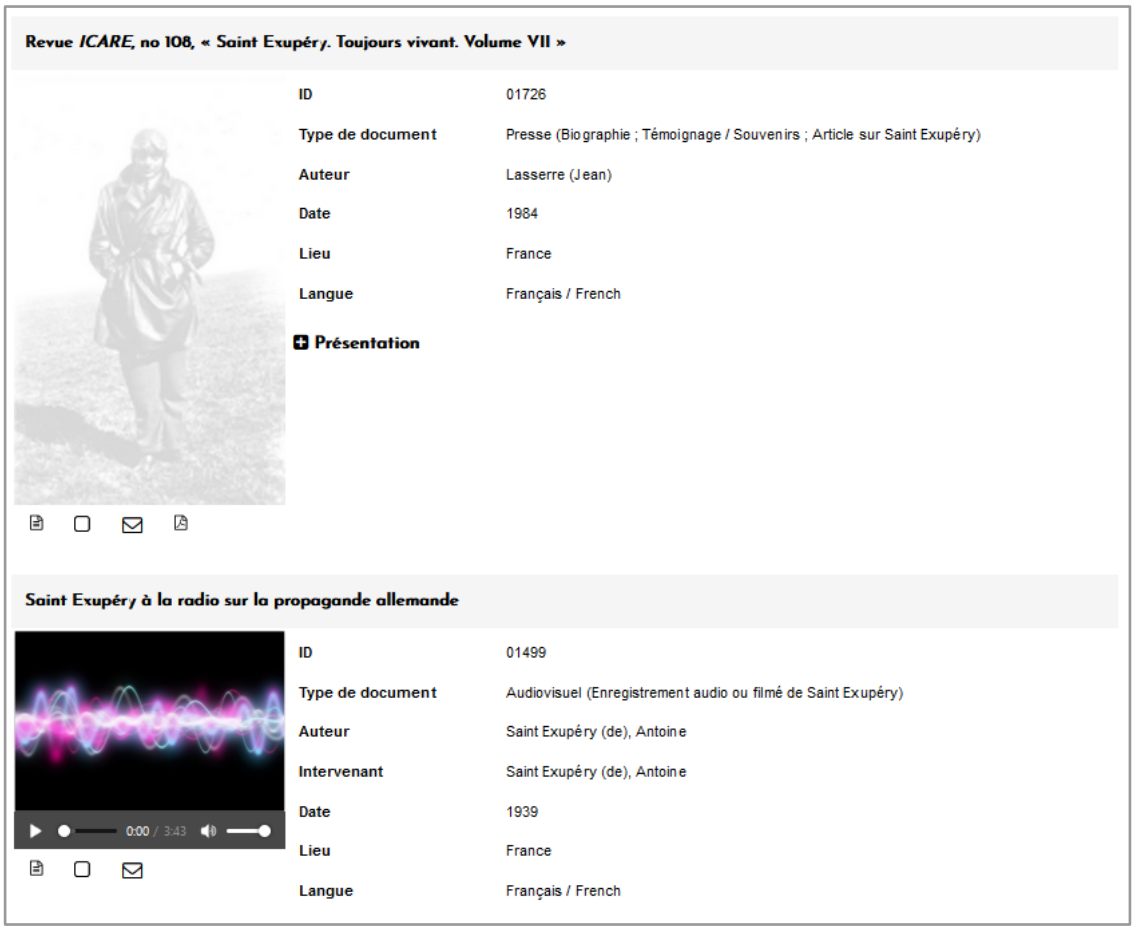


Illustration 34 : Exemple d'affichage des résultats en mode liste

La mise en place de ce module multimédia compatible avec la dernière version de la solution de Cadic a été une expérience très enrichissante durant mon alternance. En effet, nous avons dû définir des besoins précis, suivre de près les développements effectués, les tester puis faire un retour à leur propos. Cela s'est présenté sous la forme d'un projet dans le projet. Il a fallu communiquer régulièrement avec l'équipe de développement et vérifier que le fonctionnement obtenu était celui que nous souhaitions, en testant les fonctionnalités, en regardant les effets de certaines actions sur les fichiers ainsi que sur les tables SQL. Nous avons dû faire des choix et rédiger un cahier des charges précis pour pouvoir développer ce module de façon optimale, afin qu'il soit utilisable dans plusieurs projets. Cela a apporté une réelle valeur ajoutée à notre outil qui en plus de faire de la gestion de fonds documentaires, fait maintenant de la gestion de fonds multimédias.

Durant mon alternance, j'ai ainsi appris à réaliser un projet concret dans le monde professionnel et ai acquis des connaissances pragmatiques. Je me sens maintenant plus à l'aise pour réaliser mon projet de traitement automatique des langues, afin d'exploiter ce domaine que j'ai étudié durant mon master.

IV.2. Module TAL

IV.2.A. Premières idées

Chez Cadic Services, le traitement automatique des langues n'est pas un domaine exploité. Je me suis basée sur les différents cours suivis durant mon master afin de concevoir un module susceptible de faire appel aux ressources du TAL tout en apportant une réelle plus-value au produit proposé par Cadic. La première difficulté a été de trouver des idées afin de réaliser un module de TAL sur lequel je travaillerai avec l'équipe de développement, mais pour lequel j'allais être la seule à apporter une expertise en TAL. De plus, je devais trouver des idées qui permettraient à l'entreprise de disposer d'un module valorisant qu'elle pourrait proposer à ses clients et qui leur apporterait une réelle valeur ajoutée. Enfin, les réalisations ne devaient pas prendre trop de temps car je continuais en même temps à travailler sur les projets auxquels j'étais associée et le planning restait chargé pour moi.

En respectant ces trois contraintes, je suis arrivée à créer une ébauche d'un module qui permettra, à terme, de réaliser des statistiques textuelles ainsi que de calculer des proximités sémantiques sur un fonds documentaire. J'ai réalisé cela sur un corpus de test, et l'intégration de ce module dans l'application n'étant pas de mon ressort, c'est l'équipe de recherche et développement qui s'en chargera à terme.

IV.2.B. Statistiques textuelles

IV.2.B.a) Etape 1 : L'idée

J'ai réfléchi à une innovation susceptible d'intéresser la clientèle de Cadic Services et qui pourrait lui apporter des informations utiles à la gestion d'un fonds documentaire afin de l'optimiser. Je me suis aperçue que le produit habituellement proposé ne comportait pas de statistiques relatives au contenu textuel des fonds documentaires. Or, il m'est apparu que, pour un gestionnaire d'un tel fonds, il pouvait être utile d'avoir connaissance des mots les plus utilisés.

J'ai décidé de commencer par compter le nombre d'occurrences de chaque terme (hormis les mots vides[7]) dans un corpus de textes représentant un fonds documentaire. Pour cela, j'ai rassemblé dans un fichier texte 10 articles de journaux issus de la rubrique "politique" du *Monde* et de *Libération*. Ensuite, j'ai fait de même selon les catégories grammaticales. Pour cela, il a fallu lemmatiser les tokens[8], pour que les mêmes mots soient bien comptés ensemble

[7] Un mot vide est un mot non significatif dans un texte, qui ne porte pas de sens lexical. Par exemple, les auxiliaires ou les déterminants.

[8] Lemmatiser des tokens (forme d'un mot tel qu'il apparaît dans un texte) consiste à les remplacer par leur forme neutre, en retirant toute trace de variation morphologique telle que le genre, le nombre ou le temps. Par exemple, le token *mangera* sera lemmatisé en *manger* et le token *amies* sera lemmatisé en *ami*.

même s'ils avaient une variation morphologique de temps, de genre ou de nombre par exemple. Puis il a fallu étiqueter en parties du discours[9] ces lemmes.

IV.2.B.b) Etape 2 : Comparaison des outils

Les étapes du prétraitement, qui sont la lemmatisation et l'étiquetage en parties du discours, sont donc très importantes car toute la suite de l'analyse reposent sur elles. J'ai voulu comparer *Stanford CoreNLP, TreeTagger, Talismane, SpaCy* et *NLTK*. De par mes travaux réalisés lors de mon master, je savais déjà que *SpaCy* n'avait pas une bonne précision sur le français. J'ai décidé de comparer *TreeTagger* et *Talismane* car ce sont les outils que je maîtrise le mieux et avec lesquels j'ai le plus travaillé dans le passé. De plus, ils obtiennent de bons scores pour le français. Il était envisageable de poursuivre le travail en comparant avec les autres outils, mais chacun produisant des résultats de formes différentes, il aurait fallu ensuite adapter le code à chacun. Par rapport au temps dont je disposais, cela me paraissait plus raisonnable de commencer par en comparer deux.

IV.2.B.c) Etape 3 : Ecriture du code

J'ai d'abord récupéré la liste des *stopwords*[10] issue de *NLTK* (cf. annexe 1) et une liste de ponctuations (cf. annexe 2).

Pour les statistiques avec *TreeTagger*, j'ai d'abord mis chaque token sur une ligne et isolé les caractères de ponctuation chacun sur une ligne également.

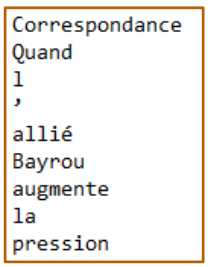


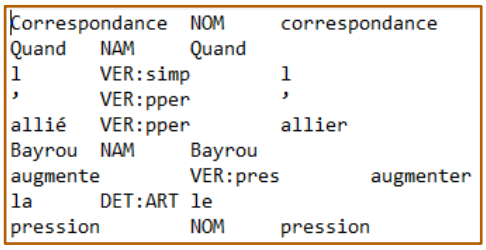
Illustration 35 : Extrait du fichier contenant un token par ligne

Ensuite, j'ai lancé grâce au module Python "Os" une commande Bash permettant de tagger le fichier de tokens avec *TreeTagger* :

```
"tree-tagger -token -lemma -no-unknown
C:\\Users\\SRepingon\\Documents\\TAL\\treetagger\\models\\fr.p
ar article_tokens.txt > article_sortie_treetagger.txt"
```

[9] Les parties du discours correspondent aux catégories grammaticales : nom, verbe, adjectif, etc.
[10] Un *stopword* est un mot vide.

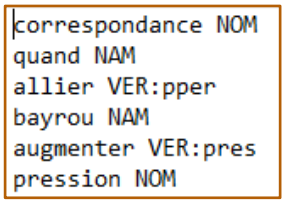
Les arguments “token” et “lemma” permettent de garder dans le fichier de sortie les tokens et les lemmes, et l’argument “no-unknown” permet de ne pas avoir en sortie l'affichage “<unknown>” si le lemme n'est pas connu.



```
Correspondance  NOM     correspondance
Quand   NAM     Quand
l       VER:simp        l
’       VER:pper        ’
allié   VER:pper        allier
Bayrou  NAM     Bayrou
augmente        VER:pres        augmenter
la      DET:ART le
pression        NOM     pression
```

Illustration 36 : Extrait du fichier contenant un token, sa partie du discours et son lemme par ligne

Ensuite, j’ai décidé de supprimer les lignes contenant des mots vides, des caractères de ponctuation ou des mots de deux caractères ou moins et de garder uniquement le lemme et sa catégorie grammaticale.



```
correspondance NOM
quand NAM
allier VER:pper
bayrou NAM
augmenter VER:pres
pression NOM
```

Illustration 37 : Extrait du fichier contenant uniquement les mots lexicaux

L’étape suivante consiste à simplifier les catégories grammaticales. En effet, il existe des parties du discours et des sous-parties du discours. Par exemple :

PRO = pronom
PRO:DEM = pronom démonstratif
PRO:IND = pronom indéfini
PRO:PER = pronom personnel
PRO:POS = pronom possessif
PRO:REL = pronom relatif

Pour les verbes, les temps sont également distingués :

VER:cond = verbe au conditionnel
VER:futu = verbe au futur

VER:impe = verbe à l'impératif
VER:impf = verbe à l'imparfait
VER:infi = verbe à l'infinitif
VER:pper = verbe au participe passé
VER:ppre = verbe au participe présent
VER:pres = verbe au présent
VER:simp = verbe au passé simple
VER:subi = verbe à l'imparfait du subjonctif
VER:subp = verbe au subjonctif présent

Ces sous-catégories n'étant pas pertinentes pour mes statistiques qui s'appuient sur les lemmes, j'ai décidé de toutes les rassembler en une seule partie du discours, ici "VER", en supprimant tout ce qu'il y a après les ":".

J'ai également construit une liste rassemblant toutes les catégories grammaticales, afin de pouvoir ensuite effectuer le classement en fonction de celles-ci.

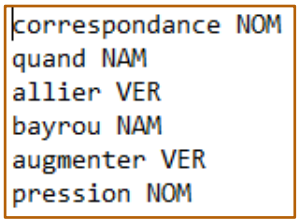


Illustration 38 : Extrait du fichier contenant les catégories grammaticales simplifiées

J'ai ensuite lancé une nouvelle commande Bash afin d'effectuer un tri par nombre d'occurrences de chaque paire de lemme et partie du discours.

```
"sort -f article_treetagger_lemmas_pos_2.txt | uniq -ci | sort -gr > article_treetagger_trie.txt"
```

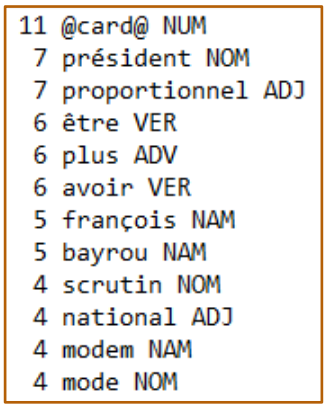


Illustration 39 : Extrait du fichier contenant tous les lemmes du corpus triés par nombre d'occurrences

Enfin, pour chacune des parties du discours de la liste précédemment constituée, j'ai créé un document contenant les occurrences des lemmes de cette catégorie.

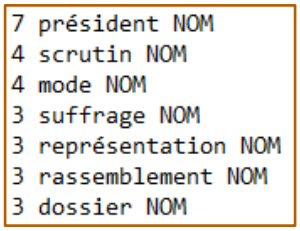


Illustration 40 : Extrait du fichier contenant les noms triés par nombre d'occurrences avec TreeTagger

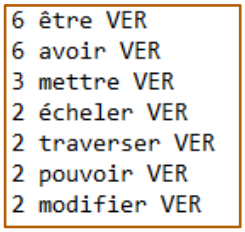


Illustration 41 : Extrait du fichier contenant les verbes triés par nombre d'occurrences avec TreeTagger

J'ai ensuite effectué les mêmes statistiques avec *Talismane* afin de pouvoir comparer les deux outils. J'ai également utilisé une commande Bash pour lancer la lemmatisation et l'étiquetage en partie du discours sur le texte. Cet outil ne nécessitait pas une tokenisation[11] au préalable.

```
"java -Xmx1G -Dconfig.file=talismane-fr-6.1.0.conf -jar
talismane-core-6.1.2-shaded.jar --analyse --sessionId=fr --
```

[11] La tokenisation est la séparation de chaque token (mot) en une série de tokens individuels.

```
encoding=UTF8 --inFile=article.txt --
outFile=article_talismane.txt"
```

```
1       Correspondance  correspondance  NC      NC      n=s|g=f 0       _       0       _

1       Quand   quand   CS      CS              0       _       0       _
2       l'      le      DET     DET     n=s     3       det     3       det
3       allié   allié   NC      NC      n=s|g=m 5       suj     5       suj
4       Bayrou  _       NPP     NPP             3       mod     3       mod
5       augmente        augmenter       V       V       n=s|t=P,S|p=1,3 1       sub     1       sub
6       la      la      DET     DET     n=s|g=f 7       det     7       det
7       pression        pression        NC      NC      n=s|g=f 5       obj     5       obj
8       sur     sur     P       P               5       mod     5       mod
9       le      le      DET     DET     n=s|g=m 10      det     10      det
10      président       président       NC      NC      n=s|g=m 8       prep    8       prep
11      Macron  _       NPP     NPP             10      mod     10      mod
```

Illustration 42 : Extrait du fichier au format Conllu obtenu en sortie de Talismane

Après cela, j'ai conservé les lemmes et catégories grammaticales des mots qui ne sont ni vides de la ponctuation ni de deux caractères ou moins, et lorsque le lemme était inconnu (indiqué "_"), alors j'ai conservé à la place le token. Enfin, j'ai effectué les mêmes traitements que pour *TreeTagger* afin d'obtenir les statistiques.

```
6 président NC
6 proportionnelle NC
4 scrutin NC
4 mode NC
3 suffrage NC
3 représentation NC
3 rassemblement NC
```

Illustration 43 : Extrait du fichier contenant les noms triés par nombre d'occurrences avec Talismane

```
6 être V
6 avoir V
3 mettre V
2 traverser V
2 pouvoir V
2 modifier V
2 exprimer V
```

Illustration 44 : Extrait du fichier contenant les verbes triés par nombre d'occurrences avec Talismane

IV.2.B.d) Etape 4 : Intégration dans l'application

Ces premières statistiques ont permis de comparer les deux outils afin de déterminer lequel sera utilisé pour la future intégration dans l'application Cadic. J'ai remarqué quelques

erreurs de lemmatisation ou d'étiquetage en parties du discours dans les fichiers de sortie des deux outils, mais *TreeTagger* semble dans l'ensemble meilleur que *Talismane*. L'intégration dans l'application Cadic devant être faite en liaison avec l'équipe de recherche et développement, celle-ci n'a pas encore eu lieu, mais il est prévu qu'elle soit faite prochainement. Le fonctionnement sera similaire : le répertoire contenant mon corpus représente le répertoire contenant le fonds documentaire du client. Afin de suivre l'évolution du fonds, il faudra automatiser le lancement de mon code régulièrement, par exemple chaque semaine.

IV.2.B. Proximité sémantique

IV.2.C.a) Etape 1 : L'idée

La seconde idée que nous avons eue est de construire un système de mots-clefs permettant de proposer, à partir d'un mot, des mots sémantiquement proches dans le contexte d'un fonds documentaire. J'ai donc effectué une carte de proximité sémantique se basant sur un corpus. Selon le corpus, un mot tel que "java" pourra se trouver proche de mots comme "danse" ou plutôt de mots comme "programmation". C'est pour cela que la proximité sémantique dépend de chaque corpus, et qu'il était intéressant de la calculer pour un fonds documentaire en particulier. Je suis donc repartie du même corpus d'articles que celui utilisé pour les statistiques textuelles.

IV.2.C.b) Etape 2 : Prétraitement

J'ai effectué un prétraitement en écrivant un programme en Python permettant de tokeniser mon texte, de supprimer les mots vides et la ponctuation, puis de le mettre en minuscules.

```
1  correspondance
2  quand allié bayrou augmente pression président macron
3  haut commissaire plan a adressé courrier président
   république demander respecter promesse campagne introduire
   proportionnelle législatives 2022
4  françois bayrou tient vraiment proportionnelle
5  vraiment
6  après avoir entamé séquence lobbying semaine dernière
   président modem a envoyé lettre président république jeudi
   demandant nouveau prendre compte nécessité urgence loi
   électorale juste
7  comprendre enfin mettre place mode scrutin tant désiré haut-
   commissaire plan promis emmanuel macron allié centriste lors
   campagne présidentielle 2017
```

Illustration 45 : Extrait du fichier obtenu

IV.2.C.c) Etape 3 : Construction du graphe

Afin de construire le graphe, j'ai écrit un programme en Python utilisant les librairies *Gensim, Matplotlib, Numpy* et *Sklearn*. J'ai créé à partir du corpus prétraité un modèle vectoriel. Ensuite, j'ai créé une liste contenant le vocabulaire du corpus à partir de ce modèle, puis j'ai visualisé les mots du vocabulaire et le modèle vectoriel ainsi créés.

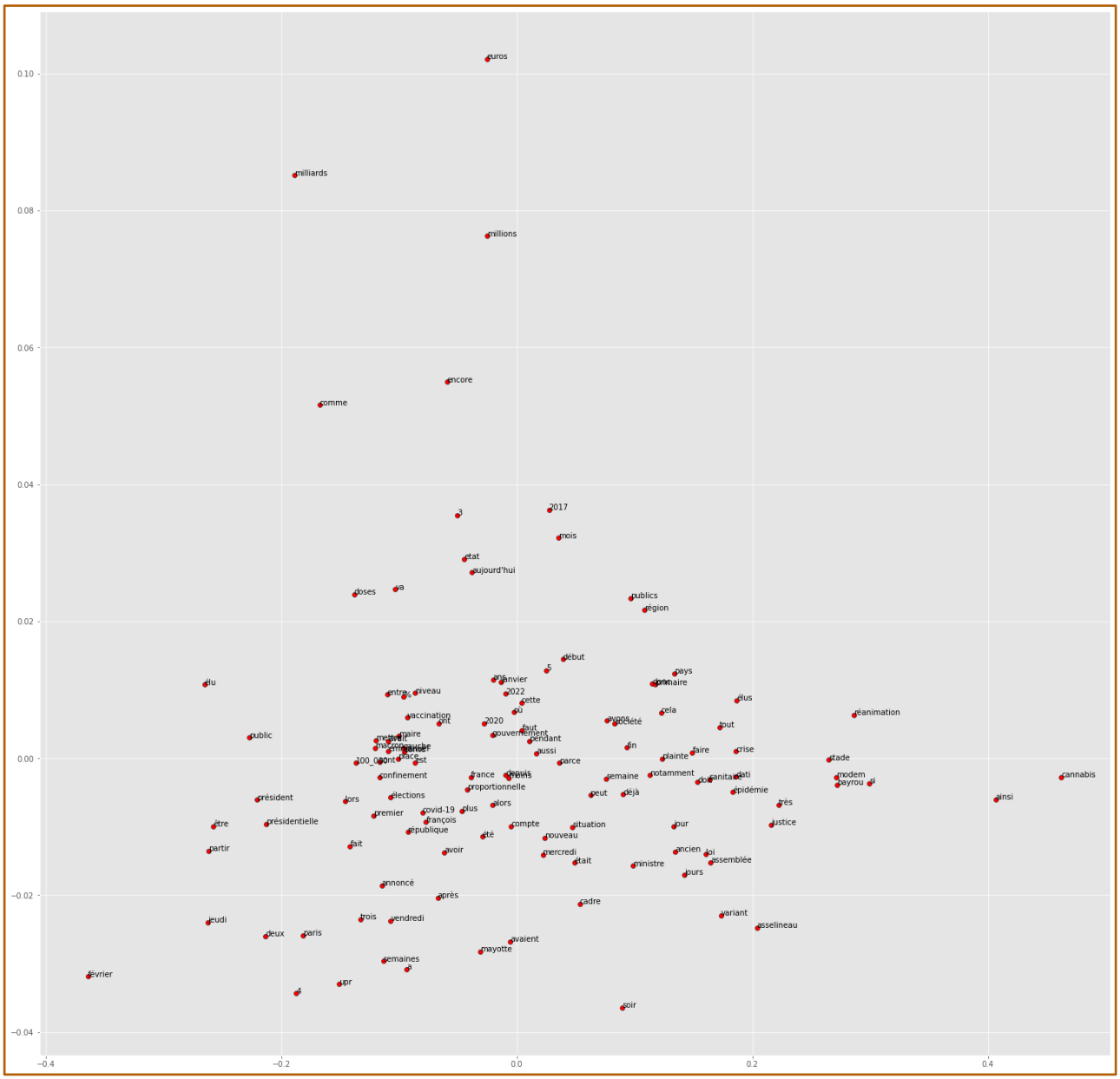
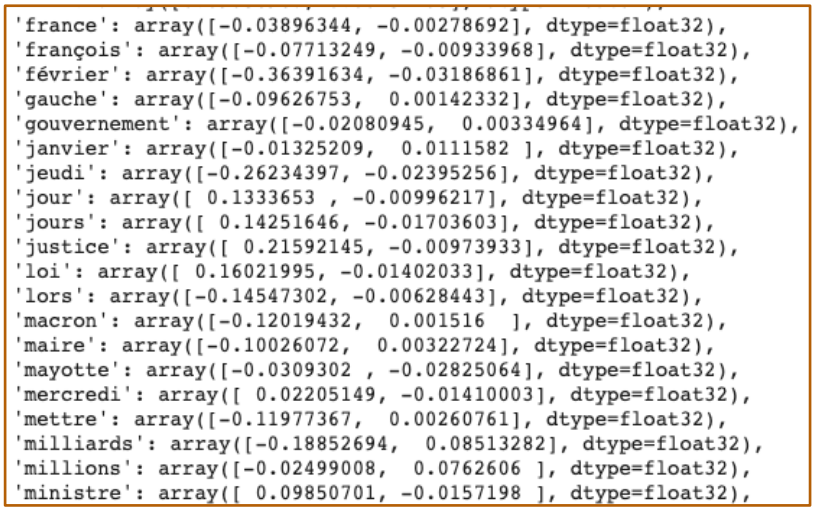


Illustration 46 : Chaque mot du vocabulaire est représenté sur ce graphe en fonction de sa similarité sémantique avec les autres mots

Le corpus utilisé n'est pas très grand, mais l'on peut déjà voir des mots sémantiquement proches se rapprocher également sur le graphe :

Illustration 47 : Exemples plus précis de mots dans le graphe

J’ai ensuite écrit une fonction permettant d’extraire les coordonnées de chaque point.



```
'france': array([-0.03896344, -0.00278692], dtype=float32),
'françois': array([-0.07713249, -0.00933968], dtype=float32),
'février': array([-0.36391634, -0.03186861], dtype=float32),
'gauche': array([-0.09626753,  0.00142332], dtype=float32),
'gouvernement': array([-0.02080945,  0.00334964], dtype=float32),
'janvier': array([-0.01325209,  0.0111582 ], dtype=float32),
'jeudi': array([-0.26234397, -0.02395256], dtype=float32),
'jour': array([ 0.1333653 , -0.00996217], dtype=float32),
'jours': array([ 0.14251646, -0.01703603], dtype=float32),
'justice': array([ 0.21592145, -0.00973933], dtype=float32),
'loi': array([ 0.16021995, -0.01402033], dtype=float32),
'lors': array([-0.14547302, -0.00628443], dtype=float32),
'macron': array([-0.12019432,  0.001516  ], dtype=float32),
'maire': array([-0.10026072,  0.00322724], dtype=float32),
'mayotte': array([-0.0309302 , -0.02825064], dtype=float32),
'mercredi': array([ 0.02205149, -0.01410003], dtype=float32),
'mettre': array([-0.11977367,  0.00260761], dtype=float32),
'milliards': array([-0.18852694,  0.08513282], dtype=float32),
'millions': array([-0.02499008,  0.0762606 ], dtype=float32),
'ministre': array([ 0.09850701, -0.0157198 ], dtype=float32),
```

Illustration 48 : Extrait des coordonnées de certains points

Il est également possible d’extraire les coordonnées pour un mot en particulier.

```
print(coordonnees['société'])
```

```
[0.08305386 0.00504323]
```

Illustration 49 : Exemple des coordonnées pour le point correspondant au mot “société”

IV.2.C.d) Etape 4 : Intégration dans l’application

Afin de reproduire au mieux un fonds documentaire comme celui de nos clients, j’ai adapté les scripts afin qu’ils fonctionnent sur plusieurs fichiers textes dans un répertoire. L’étape suivante sera d’intégrer ces calculs de proximité sémantique à l’application Cadic afin de proposer, pour un fonds documentaire, des termes similaires, par exemple lorsqu’une recherche sera lancée sur un terme ou au clic sur un mot-clef. Nous pourrions également imaginer, côté documentaliste, en production, une suggestion de mots-clefs où lorsqu’un terme serait ajouté, d’autres similaires seraient proposés ; ou bien une suggestion de mots-clefs directement à partir du contenu sémantique du document associé à la notice. Comme pour les statistiques, il faudra automatiser le processus afin qu’il suive l’évolution du fonds.

V. Conclusion

Cette alternance chez Cadic Services a été pour moi une bonne façon de rentrer dans la vie active et de découvrir le monde de l'entreprise. J'ai pu voir comment la méthode de gestion de projet AGILE était appliquée en entreprise, après l'avoir étudiée lors de mon master. La mise en place du portail de la Succession Saint Exupéry – d'Agay, depuis les ateliers de spécifications jusqu'à la mise en production, a été une expérience très enrichissante pour moi. Je maîtrise maintenant les outils d'administration de l'application Cadic et je me sens à l'aise dans la relation avec les clients.

Les différentes missions qui m'ont été confiées m'ont permis de développer mon esprit d'analyse. En effet, lorsque je relevais un mauvais fonctionnement, je cherchais à le corriger, notamment en allant rechercher dans des fichiers d'où il pouvait provenir. J'ai donc développé au cours de cette année un esprit critique et d'analyse ainsi qu'une méthodologie rigoureuse de réalisation de projet.

Le développement de mon projet de module de Traitement Automatique des Langues m'a permis de travailler en autonomie tout en restant en contact régulier avec l'équipe de recherche et développement. Allier ce projet avec mes autres missions m'a appris à m'organiser de façon à gérer au mieux le temps dont je disposais. Les contraintes liées au fait que le module était nouveau par rapport au cœur de métier de l'entreprise ont été bénéfiques pour moi car j'ai pu proposer mes idées et juger par moi-même de ce qui serait réalisable et des moyens à déployer pour y arriver. Il ne reste maintenant plus que l'intégration à l'application afin de finaliser la création de ce module.

Pour conclure ce rapport, je dirais que les compétences et connaissances que j'ai acquises chez Cadic depuis septembre 2020 sont fondamentales pour ma future vie professionnelle et pour les futures missions qui me seront confiées.

VI. Références bibliographiques

Cadic Services, “Formation Administration Informations générales — Les points essentiels” (2020)

Cadic Services, “Plaquette Cadic Intégrale Zéphyr” (2018)

J-P. Accart et M-P. Réthy, “Le métier de documentaliste”, Editions du Cercle de la Librairie (1999)

ONISEP, “Documentaliste”, URL : https://www.onisep.fr/Ressources/Univers-Metier/Metiers/documentaliste

Université de Poitiers, Coopération des centres régionaux de formations aux carrières des bibliothèques, “Notice bibliographique” (2019), URL : https://blogs.univ-poitiers.fr/glossaire-mco/2012/06/11/notice-bibliographique/

VII. Annexes

Annexe 1 : Liste des *stopwords* issue de *NLTK*

['au', 'aux', 'avec', 'ce', 'ces', 'dans', 'de', 'des', 'du', 'elle', 'en', 'et', 'eux', 'il', 'ils', 'je', 'la', 'le', 'les', 'leur', 'lui', 'ma', 'mais', 'me', 'même', 'mes', 'moi', 'mon', 'ne', 'nos', 'notre', 'nous', 'on', 'ou', 'par','pas', 'pour', 'qu', 'que', 'qui', 'sa', 'se', 'ses', 'son', 'sur', 'ta', 'te','tes', 'toi', 'ton', 'tu', 'un', 'une', 'vos', 'votre', 'vous', 'c', 'd', 'j', 'l', 'à', 'm', 'n', 's', 't', 'y', 'été', 'étée', 'étées', 'étés', 'étant', 'étante', 'étants', 'étantes', 'suis', 'es', 'est', 'sommes', 'êtes', 'sont', 'serai', 'seras', 'sera', 'serons', 'serez', 'seront', 'serais', 'serait', 'serions', 'seriez', 'seraient', 'étais', 'était', 'étions', 'étiez', 'étaient', 'fus', 'fut', 'fûmes', 'fûtes', 'furent', 'sois', 'soit', 'soyons', 'soyez', 'soient', 'fusse', 'fusses', 'fût', 'fussions', 'fussiez', 'fussent', 'ayant', 'ayante', 'ayantes', 'ayants', 'eu', 'eue', 'eues', 'eus', 'ai', 'as', 'avons', 'avez', 'ont','aurai', 'auras', 'aura', 'aurons', 'aurez', 'auront', 'aurais', 'aurait', 'aurions', 'auriez', 'auraient', 'avais', 'avait', 'avions', 'aviez', 'avaient', 'eut', 'eûmes', 'eûtes', 'eurent', 'aie', 'aies', 'ait', 'ayons', 'ayez', 'aient', 'eusse', 'eusses', 'eût', 'eussions', 'eussiez', 'eussent']

Annexe 2 : Liste de ponctuations

"!\"#$%&''()*+,--./:;<=>?@[\]^_`{|}~\«\»"