

# Dossier de Spécifications

## Projet Fil Rouge

Conception d'un moteur de recherche



**SeekFox**

UPSSITECH SRI 1A

2019-2020

Gaël Gamba – Etienne Combelles

Raphaël Bizet – Oualid El Abdaoui – Clément Truillet

Client : Isabelle Ferrané

# Dossier de Spécifications

1

I – Introduction	3
I.1 Objet du document	3
II - Cadre	4
II.1 Contexte & Objectif	4
II.2 Résumé	4
II.3 Livrables	5
II.4 Planning	6
II.5 Présentation de l'équipe	8
II.6 Identité	8
III - Spécifications Fonctionnelles	8
III.1 Fonctionnement Général	9
III.2 Indexation	12
III.2.1 Mode Manuel	12
III.2.2 Mode Automatique	12
III.3 Recherche	13
III.3.1 Recherche d'un texte	13
III.3.2 Recherche image	14
III.3.3 Recherche son	14
III.4 Comparaison	14
III.4.1 Comparaison de son	14
III.5 Interface	15
IV – Annexes	16
IV.1 Tâches	16
IV.2 Scénarios de test	17

## I – Introduction

### I.1 Objet du document

L'objet de ce document est de définir les spécifications fonctionnelles détaillées de la partie 1 du Projet Fil Rouge proposé à la promotion de 1<sup>ère</sup> année de la spécialité Systèmes Robotiques et Intelligents de l'UPSSITECH.

Les spécifications fonctionnelles détaillées ont pour but de décrire :

- L'ensemble des fonctionnalités de l'application.
- Les objets manipulés par l'application.
- Le fonctionnement interne de l'application.

## II - Cadre

### II.1 Contexte & Objectif

L'accès à l'information est un enjeu essentiel. Avec internet et les ordinateurs, de grandes bases de données sont accessibles à tout le monde. Pour exploiter ces bases de données, contenant des fichiers de diverse nature (fichiers textes, audio, images), des outils adaptés comme des moteurs de recherche sont indispensables. Notre objectif consiste en la création d'un tel outil, utilisable par tous, gérant à la fois l'indexation, la comparaison et la recherche de documents.

### II.2 Résumé

Notre moteur de recherche aura donc plusieurs tâches à remplir : indexation, comparaison, recherche, pour n'en citer que les principales. Il est à noter que seule la recherche sera disponible pour un utilisateur, le reste n'étant accessible qu'à l'administrateur.

Les documents traités étant de nature diverse (fichiers textes, audio, images) de formats prédéfinis (*XML*, *TXT*, *JPEG*, *WAV*...), nous devons d'abord faire une description synthétique de chacun qui sera utilisée pour son traitement par le moteur de recherche : il s'agit là de la première fonctionnalité de celui-ci, l'indexation automatique. Ces descripteurs comprendront les informations essentielles du document ainsi que des mots clefs, chaque document ne pouvant avoir qu'un seul descripteur dans l'index.

Le moteur de recherche utilisera ces descripteurs pour comparer les documents, afin de définir des taux de similarité entre différents documents. Cette similarité se base sur le nombre de points communs entre deux descripteurs : si ces descripteurs sont identiques, alors les documents sont les mêmes, sinon le nombre de caractéristiques en commun définira un taux de similarité qui permettra de fixer un seuil à partir duquel on considèrera que deux documents sont proches.

La fonction de recherche du moteur sera le résultat visible et utilisable des fonctionnalités décrites précédemment. Dans cette première phase du projet, il consistera visuellement en une entrée clavier et un affichage à l'écran des résultats de la recherche.

L'aspect graphique sera alors développé dans la seconde partie du projet. Le moteur de recherche aura deux modes d'utilisation, un mode administrateur qui

permettra de configurer et de lancer l'indexation et de visualiser les descripteurs, et un mode utilisateur qui sera l'aspect tout public du moteur, à savoir la recherche de documents dans la base de données. La recherche fonctionnera en utilisant les fichiers de descripteurs créés à la phase d'indexation, et leurs comparaisons ; elle affichera une liste de documents répondants à la requête ordonnés selon leur taux de similarité.

## II.3 Livrables

En plus du présent dossier de spécifications, le produit final de cette première partie sera fourni sous forme d'archive compressée, comprenant tout le code nécessaire au fonctionnement du moteur de recherche, ainsi qu'un mode d'emploi pour générer un exécutable.

Sera aussi fourni un rapport détaillant tout le processus de création de l'outil ainsi que son fonctionnement.

De plus, les fonctionnalités du produit final seront présentées lors d'un entretien à la date convenue du 24 janvier 2020.

## II.4 Planning

- Remise du dossier de spécifications : 13/11/2019
- Conception et développement du moteur de recherche : entre le 13/11/2019 et le 12/01/2020
- Tests et intégration : du 12/01/2020 au 20/01/2020
- Livraison du moteur de recherche : 24/01/2020
- Remise du rapport : 31/01/2020

Vous pourrez suivre l'avancée de notre travail ici : <https://trello.com/b/efDdpilV/projet-fil-rouge-1a-sri-seekfox> .

Afin d'avancer de manière efficace, nous avons décidé de prévoir les tâches et d'en estimer les durées de réalisation.

De manière graphique, ce planning se présente par un diagramme de Gantt (cf. figure 1).

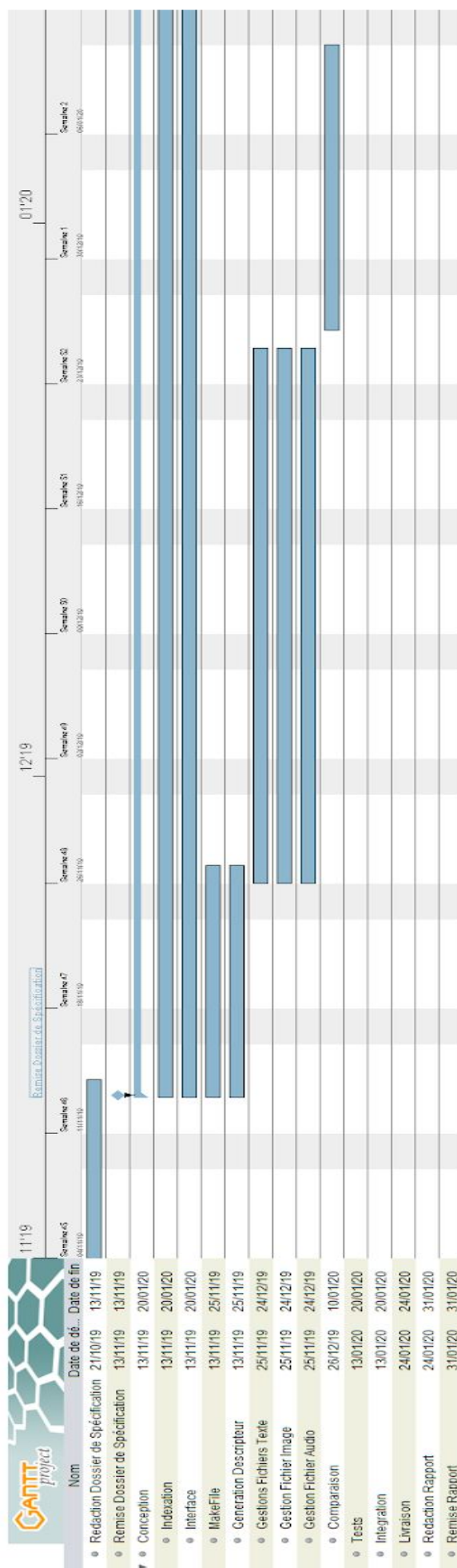


Figure SEQ Figure \ ARABIC 1 :  
Diagramme de Gantt

## II.5 Présentation de l'équipe

Afin de mener à bien ce projet, notre équipe est constituée de 5 personnes réunies sous une équipe nommée SeekFox.

- Clément TRUILLET
- Etienne COMBELLES
- Gaël GAMBA
- Oualid EL ABDAOUI
- Raphaël BIZET

## II.6 Identité

Dans un objectif de démarcation, nous avons voulu créer une identité propre représentant l'esprit de notre projet, un moteur de recherche, et celui de notre école par sa mascotte, le renard.

Alors inspiré par la visite de l'IOT Valley par notre promo en début d'année, et plus spécialement par celle de l'entreprise SigFox, un trait d'humour a alors lancé l'idée de nous nommer **SeekFox**.

Bien que la traduction littérale de ce nom corresponde à « **Chercher Renard** », nous aimons l'associer à « **Le Renard qui Cherche** ».

L'identité visuelle de notre équipe est alors représentée par un renard (*Figure 2*) accompagné d'une loupe sur l'œil droit rappelant le caractère de recherche de notre objectif.



*Figure 2 : Logo de l'équipe SeekFox*

La couleur dominante, orange, n'est pas sans rappeler les couleurs de l'UPSSITECH. Elle est, cependant, présente en deux teintes (**#BE4A23** et **#D1532B**) symbolisant les origines de filière différentes de notre équipe.



## III - Spécifications Fonctionnelles

### III.1 Fonctionnement Général

Le programme, un moteur de recherche, retourne les  $n$  fichiers les plus proches d'un fichier (texte, audio ou image) ou d'un mot-clé entré par l'utilisateur. Pour cela, il disposera d'un répertoire contenant les descripteurs<sup>1</sup> de chacun des fichiers présents dans le répertoire de recherche, ceux-ci seront créés par le programme lors de l'indexation des fichiers du répertoire.

Deux types d'utilisateurs pourront utiliser ce programme, à savoir les utilisateurs classiques et un administrateur<sup>2</sup>. Les deux auront accès à la recherche classique mais l'administrateur pourra faire des choses supplémentaires (voir la Figure 3).

Utilisateur Classique	Recherche	Texte	Par mot-clé
			Fichiers
		Audio	Fichier
Administrateur	Image		Fichier
			Couleur Dominante
	Indexation manuelle		
	Visualisation de descripteur		
	Configuration		

Figure 3 : Visualisation des choix possibles suivant le type d'utilisateur

Le menu de configuration permettra d'afficher la liste de paramètres du moteur de recherche qu'il sera possible de modifier et d'actions, comprenant :

- ☐ Le chemin du répertoire de recherche
- ☐ Le chemin du répertoire de descripteurs
- ☐ Le nombre de fichiers affichés par page de recherche
- ☐ Le pourcentage de ressemblance minimum requis pour qu'un fichier soit affiché
- ☐ Lancer une indexation manuelle
- ☐ Afficher la liste des descripteurs ainsi que leur contenu

<sup>1</sup> Descripteur : fichier texte associé à un autre fichier (texte, image ou audio) et contenant les informations essentielles à la comparaison de ce fichier avec d'autres fichiers du même type.

<sup>2</sup> Administrateur : compte utilisateur protégé avec un mot de passe et ayant accès à des fonctionnalités supplémentaires (menu de configuration, lancer l'indexation et visualiser des descripteurs)

Les changements seront enregistrés dans un fichier ce qui permettra de les conserver à chaque redémarrage du programme.

**Scénario :**

1. Le programme demande à l'utilisateur si il est administrateur ou non
2. Le programme propose la liste d'actions disponible par l'utilisateur choisi (recherche ou options administrateur)
  1. Si l'administrateur a choisi les actions administrateur, un menu s'affiche lui permettant de faire toutes les actions qu'il veut, une option lui permet également de revenir en arrière au point 2
3. Le programme demande à l'utilisateur quel type de recherche l'utilisateur veut mener.
4. Le programme demande à l'utilisateur les informations nécessaire au déroulement de la recherche (Quel mot-clé rechercher? Quelle couleur? Quel fichier comparer?)
5. Le programme affiche un nombre  $n$  de résultat et si il a trouvé plus de  $n$  fichiers, propose à l'utilisateur d'en afficher  $n$  autres. Les résultats seront classés par ordre décroissant de similarité avec la recherche.
6. Le fichier le plus ressemblant s'ouvrira.

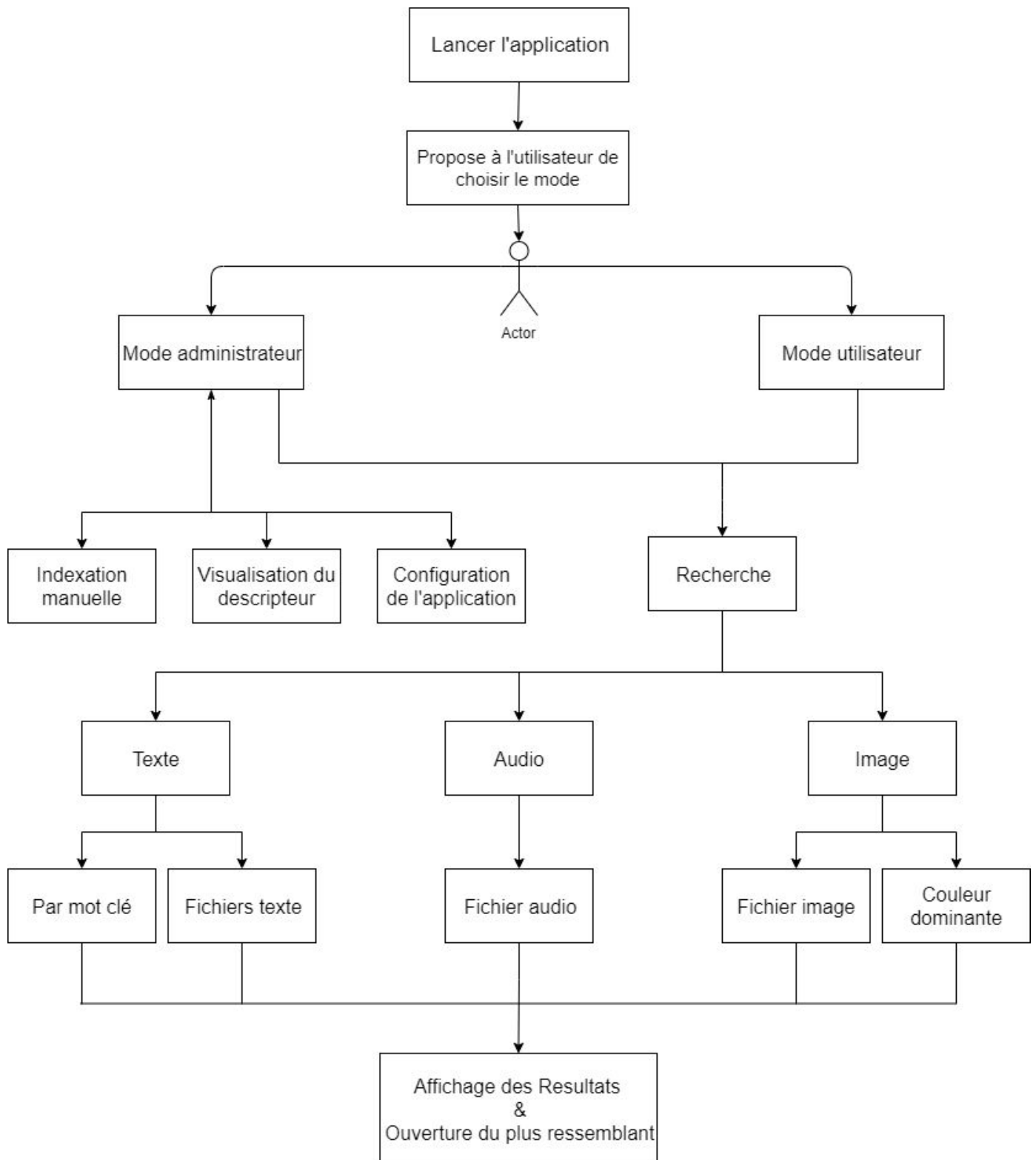


Figure 4: Schéma de fonctionnement

## III.2 Indexation

L'indexation consiste en la création des descripteurs pour pouvoir faire une recherche. Ces descripteurs seront sous forme de texte et pourront représenter des fichiers de différents types : texte, image, audio dont le processus de création est détaillé plus bas. L'indexation sera effectuable selon deux modes, un automatique et un manuel.

### III.2.1 Mode Manuel

L'indexation manuelle sera disponible uniquement pour l'administrateur : il pourra la lancer depuis son menu, où elle sera présentée sous la forme d'une option "mettre à jour la base de données".

Cette indexation commencera par vérifier si un index de descripteurs est déjà présent. Si c'est le cas, l'indexeur continuera en vérifiant quels descripteurs sont "orphelins", c'est-à-dire quand leur fichier d'origine n'existe plus ou a été modifié ; ces descripteurs seront supprimés.

Ensuite, il passera en revue l'ensemble des fichiers présents dans la base de données, et créera des descripteurs pour ceux qui n'en ont pas déjà.

Dans le cas où il n'y a pas d'index de descripteurs, l'indexeur va simplement en créer un avec tous les fichiers présents dans la base de données.

Une fois ces opérations faites, le moteur renverra le message "Base de données à jour".

Si l'indexeur a rencontré un obstacle l'empêchant de correctement s'exécuter, il renverra un message d'erreur décrivant au mieux l'erreur rencontrée (ou "Erreur inconnue" le cas échéant), et proposant une solution pour réparer l'erreur dans les cas les plus courants (par exemple lorsque la base de données comprend des fichiers dans des formats non pris en compte).

### III.2.2 Mode Automatique

Le mode d'indexation automatique s'exécutera à chaque lancement du moteur de recherche en vérifiant la présence d'un index de descripteurs. Pour ne pas ralentir le lancement du moteur de recherche à chaque fois, il ne fera l'indexation que s'il voit qu'il n'y a pas d'index de descripteurs (donc au premier lancement, ou si l'index a été supprimé manuellement).

En cas d'indexation, elle fonctionnera comme pour le mode manuel dans le cas où il n'y a pas d'index de descripteurs, renvoyant les mêmes messages de réussite ou d'erreur. Lorsqu'il y a déjà un index de descripteurs, l'indexation ne se fera tout simplement pas, elle devra être faite manuellement par l'administrateur si la base de données est modifiée.

### III.3 Recherche

Chaque recherche par l'utilisateur ou l'administrateur se déroule selon un scénario global décrit ci-dessous :

Scénario :

1. L'utilisateur saisit le chemin ou choisit le fichier (stocké dans le répertoire requête) dans une liste affichée. (Note : Une recherche par mot-clé et par couleur dominante est aussi possible.)
2. Le système crée un descripteur au fichier à chercher et effectue une comparaison avec les descripteurs du même type déjà indexés.
3. Le système affiche la liste des fichiers les plus proches du fichier d'entrée et ouvre le fichier le plus pertinent avec un éditeur de texte, lecteur d'image ou audio suivant le type du fichier.

Exceptions :

1. Le fichier d'entrée est inexploitable.
  - Le fichier n'est pas un fichier texte, image ou audio
  - Le fichier n'est pas lisible par le système
  - Le fichier est dans un format incompatible avec le système
  - Le chemin vers le fichier est non valide
2. Il n'y a aucun fichier indexé.

Il est important de noter que lors d'une recherche à partir d'un fichier, les fichiers affichés le seront selon un taux de similarité défini par la comparaison (voir chapitre suivant). Le seuil d'affichage (soit le taux minimal à partir duquel on considère deux fichiers comme similaires) ainsi que le nombre de résultats pourra être choisi par l'administrateur.

Le type de recherche le plus commun est la recherche par fichier, celle-ci a pour but de trouver des fichiers ressemblants le plus possible au fichier que l'on fournit et le programme affichera tous les fichiers qui correspondent à la recherche et ouvrira le fichier le plus ressemblant. La manière dont nous comparerons ces fichiers est expliquée plus bas.

Cependant, chaque recherche d'un fichier de format différent a ses spécificités, elles sont décrites dans les sous-parties suivantes.

#### III.3.1 Recherche d'un texte

En plus de la recherche par fichier, nous pouvons également faire une recherche par mot-clé, dont le but est de trouver les fichiers ayant le plus d'occurrence du mot-clé en question, cela va se faire à partir du descripteur généré pour chaque fichier.

### III.3.2 Recherche image

Il y a également une deuxième option pour la recherche d'image, il s'agit de la recherche par couleur dominante ayant pour but d'afficher les fichiers contenant le plus la couleur recherchée.

### III.3.3 Recherche son

La recherche audio est orientée sur la recherche de l'occurrence d'un jingle audio dans les fichiers sons. Ici, l'utilisateur va nous fournir un fichier sonore court et le programme va afficher les fichiers sonores contenant ce son ainsi que le time code de l'apparition dudit son.

## III.4 Comparaison

La comparaison entre deux fichiers est traitée de façon interne par l'application, sans que l'utilisateur n'ait à intervenir. Pour n'importe quelle comparaison, le scénario global est le même :

Scénario :

1. Le système reçoit un fichier à traiter.
2. Le système génère le descripteur associé.
3. Le système compare ce descripteur avec ceux déjà stockés en prenant en compte le type de comparaison.
4. Le système renvoie une valeur de correspondance

Exceptions :

1. Le fichier d'entrée est inexploitable.
  1. Le fichier n'est pas un fichier texte, image ou audio
  2. Le fichier n'est pas lisible par le système
  3. Le fichier est dans un format incompatible avec le système
2. Il n'y a aucun fichier du même type indexé.

### III.4.1 Comparaison de son

La comparaison entre fichiers audio est un peu différente étant donné qu'on ne cherche pas à comparer la ressemblance entre 2 descripteurs mais à savoir si l'un des descripteurs (qui est une représentation du fichier) est compris dans l'autre. La comparaison va donc retourner plus d'informations que pour les autres, à savoir le nombre d'occurrences du jingle ainsi que le time code lié à ces occurrences.

### III.5 Interface

L'interface sera pour l'instant seulement composée du terminal, toutes les interactions se feront à l'aide de messages textuels. Les utilisateurs devront faire des choix, comme par exemple quel type de recherche ils veulent faire. Pour cela le programme affichera des chiffres devant chaque choix et l'utilisateur devra écrire le chiffre correspondant à l'action voulue. Cela nous permettra en plus de tester la validité des entrées de l'utilisateur afin qu'il n'écrive pas des caractères indésirables ou hors sujets.

## IV – Annexes

### IV.1 Tâches

Dans une envie de réaliser ce projet de manière efficace, il est cohérent que chacun des membres de l'équipe SeekFox se répartissent les tâches de manière équitable.

L'équipe étant composée de 5 membres, nous avons décomposé les tâches en 5 grandes parties.

Nom	Description	Responsable
Texte	Comparaison et génération de descripteur	Raphaël Bizet
Image	Comparaison et génération de descripteur	El Abdaoui Oualid
Audio	Comparaison et génération de descripteur	Gaël Gamba
Indexation	Génération de la base de descripteurs	Etienne Combelles
Interface et Recherche	Option, mode, affichage, recherche	Clément Truillet



## IV.2 Scénarios de test

Afin de produire un moteur de recherche fonctionnel, il est primordial d'effectuer des tests.

Ces tests peuvent être unitaires (sur une fonction seulement), ou sur une fonctionnalité, nous allons alors utiliser ces scénarios de tests.

Indexation:

1. Ajouter un nouveau fichier (texte, image ou audio) dans le dossier des fichiers indexés
  2. Lancer le moteur de recherche
  3. Visualiser le nouveau descripteur (mode administrateur)
- 
1. Supprimer le descripteur d'un fichier
  2. Lancer le moteur de recherche
  3. Visualiser le nouveau descripteur (mode administrateur)

Recherche:

1. Lancer le moteur de recherche
  2. Effectuer une recherche à partir d'un fichier déjà indexé
  3. Constater que le fichier le plus pertinent est bien le fichier entré
- 
1. Lancer le moteur de recherche
  2. Effectuer une recherche à partir d'un fichier non indexé
  3. Constater la validité de la recherche (test subjectif)
- 
1. Lancer le moteur de recherche
  2. Effectuer une recherche à partir d'un fichier non pris en charge
  3. Constater l'affichage d'une erreur
- 
1. Lancer le moteur de recherche
  2. Effectuer une recherche en entrant un chemin invalide
  3. Constater l'affichage d'une erreur