



---

# Spécification des exigences

---

Turing Gallery



ESI Digital Corporation

Version	Modifications apportées	Date de publication
1.0	-	16/11/2019

# 1. Introduction

## 1.1 Objet

L'objet de ce document est de décrire l'environnement, les fonctions et les contraintes, la caractéristique des utilisateurs finaux ainsi que les exigences fonctionnelles et non fonctionnelles, du système de générateur de catalogue automatique « Turing Gallery » assisté par Machine Learning. Ce document est destiné principalement aux développeurs du logiciel mais aussi à toute autre personne voulant découvrir en détail le fonctionnement du logiciel.

## 1.2 Portée

Le logiciel « Turing Gallery » permettra aux photographes, aux artistes et aux vendeurs de stocker leurs photos et œuvres sur le Cloud. Ensuite, à partir d'un ensemble d'images et d'un modèle d'affichage choisis par l'utilisateur, un catalogue est généré où ces images seront organisées selon le modèle prédéfini. Une intelligence artificielle s'occupera de mettre à jour la liste des modèles disponibles selon les préférences de l'utilisateur, ce dernier pourra aussi apporter des modifications manuelles sur le catalogue généré. Enfin, l'utilisateur aura la possibilité de télécharger ses propres catalogues sous différents formats.

## 1.3 Glossaire

Le tableau ci-dessous expliquera les différents vocabulaires techniques pouvant être susceptible de freiner la compréhension de ce document :

Acronyme	Définition
Cloud	Consiste à utiliser des serveurs informatiques distants par l'intermédiaire d'un réseau, généralement Internet, pour stocker des données ou les exploiter.
Windows	Système d'exploitation utilisant des interfaces graphiques, produit par Microsoft.
Linux	Système d'exploitation utilisant des interfaces graphiques et en ligne de commande, fondé sur le noyau Linux.
Scikit-Learn	Bibliothèque libre Python destinée à l'apprentissage automatique.
HTTP	Hypertext Transfer Protocol, protocole de communication client-serveur développé pour le World Wide Web.
API	Application Programming Interface, un ensemble normalisé de classes, de méthodes, de fonctions et de constantes qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels.
RGPD	Règlement général sur la protection des données, un règlement de l'Union européenne qui constitue le texte de

	référence en matière de protection des données à caractère personnel.
Modal	Une boîte de dialogue qui s'exécute à l'intérieur d'une page Web.
SQL	Langage informatique normalisé servant à exploiter des bases de données relationnelles.
Java	Langage informatique très populaire, doté d'un écosystème très complet et important.
Python	Python est un langage de programmation interprété, multiparadigme et multiplateformes.
Git	Logiciel de gestion de versions décentralisé.
Framaboard	Logiciel de gestion de tâches visuel. Il permet de gérer des projets de manière collaborative selon la méthode Kanban.
Microsoft Word	Logiciel de traitement de texte publié par Microsoft.
Google Drive	Service de stockage et de partage de fichiers dans le Cloud lancé par la société Google.
Draw	Logiciel libre pour effectuer des représentations par diagrammes.
Mo	Mégaoctet, unité de mesure de quantité d'information numérique, valant 1 048 576 octets.

*Figure 1 : Acronymes et définitions*

## 1.4 Vue d'ensemble

Ce document présentera, en début, une description générale des différentes interfaces, fonctions, caractéristique des utilisateurs, contraintes et hypothèses du logiciel « Turing Gallery ». Ensuite, il s'occupera de bien détailler les exigences fonctionnelles et non fonctionnelles ainsi que tout autre type d'exigence liée au bon fonctionnement du système.

## 2. Description générale

### 2.1 Interfaces du logiciel

#### 2.1.1 Interfaces système

Le logiciel « Turing Gallery » est indépendant de tout autre logiciel, par conséquent, aucune interface système n'est nécessaire.

#### 2.1.2 Interfaces utilisateurs

Le logiciel « Turing Gallery » proposera des interfaces similaires selon la plateforme de l'utilisateur. Il devra tourner dans un navigateur Web, dans

un système d'exploitation Windows et Linux. Les utilisateurs auront accès aux interfaces selon leur statut, défini comme suit :

- Utilisateur non authentifié : Il aura accès à l'interface de connexion, d'inscription et celle permettant la visualisation d'un exemple des catalogues disponibles.
- Utilisateur authentifié : Il aura accès à toutes les interfaces proposées par le logiciel sauf celle de connexion et d'inscription.

### 2.1.3 Interfaces matérielles

Le logiciel « Turing Gallery » n'établit aucune relation avec le matériel, par conséquent, aucune interface matérielle n'est nécessaire.

### 2.1.4 Interfaces logicielles

Le logiciel « Turing Gallery » utilisera le système de Machine Learning pour gérer le concept d'organisation automatique des images dans un catalogue. Il devra être compatible avec les fonctionnalités proposées par la librairie « Scikit-Learn ».

### 2.1.5 Interfaces de communication

Le logiciel « Turing Gallery » devra s'échanger des données entre le client et le serveur à travers des requêtes HTTP. Il devra également gérer des appels API avec la librairie « Scikit-Learn ».

### 2.1.6 Contraintes de mémoires

Le logiciel « Turing Gallery » est amené à recevoir un grand nombre d'images avec des tailles différentes. Un espace de stockage assez important est à prévoir pour garantir la persistance des données.

## 2.2 Fonctions du logiciel

Les principales fonctionnalités du logiciel « Turing Gallery » seront citées dans cette partie, puis une description plus détaillée sera faite dans la partie des exigences spécifiques. On distinguera 2 types d'utilisateurs, les utilisateurs authentifiés et ceux non authentifiés.

Le logiciel devra permettre un utilisateur non authentifié de :

- Créer son compte membre.
- Se connecter à son compte membre.
- Avoir accès à son compte membre en cas de perte de mot de passe.
- Visualiser les catalogues disponibles.

Le logiciel devra permettre un utilisateur authentifié de :

- Modifier les paramètres de son compte membre.
- Se déconnecter de son compte membre.
- Supprimer son compte membre.
- Visualiser les catalogues disponibles dont ses propres catalogues.

- Importer et visualiser sur son espace ses propres images.
- Générer un catalogue à partir des images choisies et du modèle défini.
- Modifier ses propres catalogues existants.
- Télécharger un catalogue existant sous différents formats.

Pour bien schématiser le fonctionnement du système, un diagramme d'activité est disponible cf ANNEXE 1.

## 2.3 Caractéristiques des utilisateurs

Le logiciel « Turing Gallery » vise principalement le public des photographes, artistes et vendeurs qui travaillent au quotidien avec des images numériques. On supposera qu'ils ont déjà les connaissances basiques sur la manipulation des sites Web et des logiciels informatiques.

## 2.4 Contraintes

### 2.4.1 Politiques réglementaires

Le logiciel « Turing Gallery » récupérera quelques données privées sur ses utilisateurs, par conséquent, il devra respecter la norme du RGPD.

### 2.4.2 Limites imposées par le matériel

Le logiciel « Turing Gallery » dans sa version Web est fortement lié à une connexion Internet permanente pour assurer son bon fonctionnement. En revanche, les versions Windows et Linux, permettront à l'utilisateur d'avoir accès à quelques fonctionnalités en hors connexion.

### 2.4.3 Exploitation en parallèle

Le logiciel « Turing Gallery » est prévu pour être utilisé par des professionnels dans l'événementiel, il faut donc s'attendre à une utilisation pouvant être intensive avec un grand nombre de connexions.

## 2.5 Hypothèses et dépendances

Le bon fonctionnement du logiciel « Turing Gallery » dépendra d'un ordinateur pour sa version Windows ou Linux. Il dépendra d'un écran et une connexion Internet pour sa version Web. Il dépendra également des serveurs sur lesquels il tournera ses fonctions et stockera ses données.

# 3. Exigences spécifiques

## 3.1 Exigences fonctionnelles

### 3.1.1 Fonctionnalités accessible au public (mode non authentifié)

#### 3.1.1.1 Consulter les catalogues disponibles

3.1.1.1.1 Le système doit afficher tous les catalogues générés par les utilisateurs.

3.1.1.1.2 Le système doit afficher au moins 2 catalogues comme modèle.

3.1.1.1.3 Le système doit afficher la liste des catalogues organisés par thème.

3.1.1.1.4 Le système doit afficher au maximum 10 catalogues.

3.1.1.1.5 Le système doit afficher un bouton permettant de charger plus de catalogue s'il existe plus de 10 catalogues disponibles.

3.1.1.1.6 Le système doit afficher l'auteur de chaque catalogue.

3.1.1.1.7 Le système doit afficher la date de publication de chaque catalogue.

3.1.1.1.8 Le système doit afficher le nombre de consultation de chaque catalogue.

3.1.1.1.9 Le système doit afficher le nombre d'appréciation reçu par chaque catalogue.

3.1.1.1.10 L'utilisateur doit pouvoir trier les catalogues par popularité.

3.1.1.1.11 L'utilisateur doit pouvoir trier les catalogues par thème.

3.1.1.1.12 Le système doit afficher une barre de recherche permettant de saisir des mots clés pour effectuer une recherche spécifique.

3.1.1.1.13 L'utilisateur doit pouvoir saisir n'importe quelle chaîne de caractère dans la barre de recherche cf 3.1.1.1.12.

3.1.1.1.14 Le système doit vérifier préalablement que les caractères saisis par l'utilisateur dans la barre de recherche cf 3.1.1.1.12 n'est pas un code JavaScript qui peut être dangereux.

3.1.1.1.15 Le système ne doit pas exécuter un code JavaScript externe au code source du système.

3.1.1.1.16 Le système doit vérifier préalablement que les caractères saisis par l'utilisateur dans la barre de recherche cf 3.1.1.1.12 n'est pas une requête SQL qui peut être dangereux.

3.1.1.1.17 Le système ne doit pas exécuter une requête SQL externe au système.

3.1.1.1.18 L'utilisateur doit pouvoir rechercher les catalogues par titre.

3.1.1.1.19 L'utilisateur doit pouvoir rechercher les catalogues par auteur.

3.1.1.1.20 Si aucun catalogue ne correspond à la recherche effectuée par l'utilisateur, le système doit afficher le message : Aucun catalogue ne correspond à la recherche « la chaîne de caractère saisi par l'utilisateur dans la barre de recherche cf 3.1.1.1.12 ».

3.1.1.1.21 L'utilisateur ne doit pas pouvoir modifier les catalogues affichés.

### 3.1.1.2 Créer un compte membre

3.1.1.2.1 Le système doit détecter si l'utilisateur a déjà visité l'application ou pas.

3.1.1.2.2 Le système doit afficher un modal proposant à l'utilisateur de créer un compte membre après exactement 1 minute si l'utilisateur visite pour la première fois l'application cf 3.1.1.2.1.

3.1.1.2.3 L'utilisateur doit pouvoir créer un compte membre en utilisant le modal cf 3.1.1.2.2.

3.1.1.2.4 L'utilisateur doit pouvoir créer un compte membre à n'importe quel moment.

3.1.1.2.5 Chaque utilisateur membre doit être identifié par un pseudonyme, un e-mail et un mot de passe.

3.1.1.2.6 L'utilisateur doit pouvoir saisir un pseudonyme

3.1.1.2.7 L'utilisateur ne doit pas pouvoir saisir un pseudonyme contenant des chiffres.

3.1.1.2.8 Le système doit afficher un message d'alerte si l'utilisateur a saisi un chiffre dans son pseudonyme.

3.1.1.2.9 L'utilisateur ne doit pas pouvoir saisir un pseudonyme contenant des caractères spéciaux.

3.1.1.2.10 Le système doit afficher un message d'alerte si l'utilisateur a saisi un caractère spécial dans son pseudonyme.

3.1.1.2.11 L'utilisateur ne doit pas pouvoir saisir un pseudonyme de moins de 3 caractères.

3.1.1.2.12 Le système doit afficher un message d'alerte si l'utilisateur a saisi moins de 3 caractères dans son pseudonyme.



3.1.1.2.13 L'utilisateur doit pouvoir saisir un mot de passe.

3.1.1.2.14 L'utilisateur ne doit pas pouvoir saisir un mot de passe de moins de 6 caractères.

3.1.1.2.15 Le système doit afficher un message d'alerte si l'utilisateur a saisi moins de 6 caractères.

3.1.1.2.16 Le système doit masquer le mot de passe saisi par l'utilisateur.

3.1.1.2.17 Le système doit vérifier que le mot de passe saisi par l'utilisateur contient au moins un chiffre.

3.1.1.2.18 Le système doit afficher un message d'alerte si l'utilisateur n'a pas saisi un chiffre dans son mot de passe.

3.1.1.2.19 Le système doit vérifier que le mot de passe saisi par l'utilisateur contient au moins un caractère spécial.

3.1.1.2.20 Le système doit afficher un message d'alerte si l'utilisateur n'a pas saisi un caractère spécial dans son mot de passe.

3.1.1.2.21 Le système doit vérifier que le mot de passe saisi par l'utilisateur contient au moins une lettre en majuscule.

3.1.1.2.22 Le système doit afficher un message d'alerte si l'utilisateur n'a pas saisi une lettre en majuscule dans son mot de passe.

3.1.1.2.23 Le système doit vérifier que le mot de passe saisi par l'utilisateur contient au moins une lettre en minuscule.

3.1.1.2.24 Le système doit afficher un message d'alerte si l'utilisateur n'a pas saisi une lettre en majuscule dans son mot de passe.

3.1.1.2.25 L'utilisateur doit pouvoir ressaisir le mot de passe pour confirmer la validité du mot de passe qu'il a saisi cf 3.1.1.2.13.

3.1.1.2.26 Le système doit vérifier que les mots de passe saisis par l'utilisateur cf 3.1.1.2.13 et cf 3.1.1.2.25 sont équivalents.

3.1.1.2.27 Le système doit afficher un message d'alerte si les mots de passe cf 3.1.1.2.13 et cf 3.1.1.2.25 ne sont pas équivalents.

3.1.1.2.28 L'utilisateur doit pouvoir saisir un e-mail.

3.1.1.2.29 Le système doit vérifier que le format de l'e-mail saisi par l'utilisateur est valide cf ANNEXE 2.

3.1.1.2.30 L'utilisateur doit pouvoir réinitialiser les informations saisis cf 3.1.1.2.6, cf 3.1.1.2.13, cf 3.1.1.2.25 et cf 3.1.1.2.28.

3.1.1.2.31 L'utilisateur doit pouvoir envoyer les informations saisis cf 3.1.1.2.6, cf 3.1.1.2.13 et cf 3.1.1.2.28 au système.

3.1.1.2.32 Le système ne doit pas pouvoir traiter les informations envoyées par l'utilisateur cf 3.1.1.2.31 à moins que cf 3.1.1.2.7, cf 3.1.1.2.9, cf 3.1.1.2.11, cf 3.1.1.2.14, cf 3.1.1.2.17, cf 3.1.1.2.19, cf 3.1.1.2.21, cf 3.1.1.2.23, cf 3.1.1.2.26 et cf 3.1.1.2.29 ne soient satisfaites.

3.1.1.2.33 Le système doit afficher un message d'erreur s'il ne peut pas traiter les informations cf 3.1.1.2.32.

3.1.1.2.34 Le système doit vérifier que le pseudonyme n'existe pas encore dans sa base de données.

3.1.1.2.35 Le système doit afficher un message d'alerte si le pseudonyme existe déjà dans sa base de données.

3.1.1.2.36 Le système doit vérifier que l'e-mail de l'utilisateur n'existe pas encore dans sa base de données.

3.1.1.2.37 Le système doit afficher un message d'alerte si l'e-mail de l'utilisateur existe déjà dans sa base de données.

3.1.1.2.38 Le système doit proposer à l'utilisateur de s'authentifier si l'e-mail de l'utilisateur existe déjà dans sa base de données.

3.1.1.2.39 L'utilisateur doit pouvoir récupérer ces informations d'authentification si son e-mail existe déjà dans la base de données du système.

3.1.1.2.40 Le système doit crypter le mot de passe envoyé par l'utilisateur vers le système.

3.1.1.2.41 Le système doit pouvoir créer un compte membre pour l'utilisateur à partir de son e-mail, son pseudonyme et son mot de passe crypté.

3.1.1.2.42 Le système doit afficher un message d'alerte informant que le compte a été créé avec succès si aucune erreur n'est survenu pendant la création du compte de l'utilisateur cf 3.1.1.2.41.

3.1.1.2.43 Le système doit pouvoir vérifier que l'e-mail que l'utilisateur a utilisé pour créer son compte membre correspond à une boîte mail existante.

3.1.1.2.44 Le système doit afficher un message d'alerte si une erreur est survenu pendant la création du compte de l'utilisateur cf 3.1.1.2.41.

### 3.1.1.3 S'authentifier à son compte membre

3.1.1.3.1 L'utilisateur doit pouvoir s'authentifier à n'importe quel moment.

3.1.1.3.2 L'utilisateur doit pouvoir saisir son pseudonyme ou son e-mail pendant l'authentification.

3.1.1.3.2 L'utilisateur doit pouvoir saisir son mot de passe pendant l'authentification.

3.1.1.3.4 L'utilisateur doit pouvoir enregistrer ses données d'authentification.

3.1.1.3.5 Le système doit pouvoir récupérer les données d'authentification que l'utilisateur a enregistré cf 3.1.1.3.4.

3.1.1.3.6 Le système doit d'être capable d'authentifier un utilisateur avec son pseudonyme ou e-mail cf 3.1.1.3.2 et son mot de passe cf 3.1.1.3.2.

3.1.1.3.7 Le système doit vérifier si le couple (pseudonyme, mot de passe) ou (e-mail, mot de passe) correspond à un compte existant.

3.1.1.3.8 Le système doit afficher un message d'erreur si aucun compte existant correspond aux données d'authentification.

3.1.1.3.9 Le système doit proposer à l'utilisateur de récupérer son mot de passe si l'e-mail saisis par l'utilisateur correspond à un e-mail d'un compte existant.

3.1.1.3.10 L'utilisateur ne doit pas pouvoir accéder à son compte membre si les données d'authentification ne correspondent pas à un compte existant.

3.1.1.3.11 L'utilisateur doit pouvoir accéder à son compte membre si les données d'authentification correspondent à un compte existant.

### 3.1.1.4 Récupérer le mot de passe de son compte membre

3.1.1.4.1 L'utilisateur doit pouvoir récupérer le mot de passe de son compte membre au moment de l'authentification cf 3.1.1.3.1.

3.1.1.4.2 L'utilisateur doit pouvoir saisir un e-mail.

3.1.1.4.3 Le système doit pouvoir vérifier si l'e-mail saisi par l'utilisateur cf 3.1.1.4.2 correspond à un e-mail d'un compte membre.

3.1.1.4.4 L'utilisateur doit pouvoir recevoir un nouveau mot de passe temporaire sur sa boîte mail pour accéder à son compte membre.

3.1.1.4.5 L'utilisateur doit pouvoir modifier par la suite de cf 3.1.1.4.4 son mot de passe de compte membre.

### 3.1.2 Fonctionnalités accessible au membre (mode authentifié)

#### 3.1.2.1 Créer un catalogue à partir d'un modèle disponible

3.1.2.1.1 Le système doit fournir au moins 2 modèles de catalogue basique pour permettre à l'utilisateur membre de générer un catalogue à partir d'un ensemble d'images.

3.1.2.1.2 L'utilisateur doit pouvoir sélectionner un seul modèle parmi les modèles de catalogue à sa disposition.

3.1.2.1.3 L'utilisateur doit pouvoir voir un exemple de catalogue générer à partir du modèle sélectionné par l'utilisateur cf 3.1.2.1.2.

3.1.2.1.4 L'utilisateur doit pouvoir importer un ensemble d'images à partir de sa machine.

3.1.2.1.5 L'utilisateur doit pouvoir voir un aperçu des images importés.

3.1.2.1.6 Le système ne doit pas accepter les fichiers autre que des fichiers images sous format SVG, TIF/TIFF, PNG, JPEG/JPG pendant l'importation cf 3.1.2.1.4.

3.1.2.1.7 Le système doit afficher un message d'alerte si un fichier importé ne respecte pas les formats cf 3.1.2.1.6.

3.1.2.1.8 Le système ne doit pas accepter un fichier d'image de plus de 100 Mo.

3.1.2.1.9 Le système doit afficher un message d'alerte si un fichier importé dépasse 100 Mo cf 3.1.2.1.8.

3.1.2.1.10 L'utilisateur doit pouvoir voir sur l'aperçu cf 3.1.2.1.5 les fichiers des images qui ne sont pas acceptés par le système cf 3.1.2.1.8.

3.1.2.1.11 L'utilisateur doit pouvoir supprimer chaque fichier image importé via l'aperçu cf 3.1.2.1.5.

3.1.2.1.12 Le système doit demander une confirmation à l'utilisateur pour supprimer un fichier image cf 3.1.2.1.11.

3.1.2.1.13 L'utilisateur doit pouvoir annuler une suppression d'un fichier image cf 3.1.2.1.11.

3.1.2.1.14 L'utilisateur doit pouvoir confirmer une suppression d'un fichier d'image cf 3.1.2.1.12.

3.1.2.1.15 L'utilisateur doit pouvoir ajouter d'autres fichiers d'image à partir de sa machine à importer.

3.1.2.1.16 Le système doit proposer le format de papier A4 comme format par défaut des pages du catalogue à générer.

3.1.2.1.17 L'utilisateur doit pouvoir choisir d'autres formats de papier comme format des pages du catalogue à générer.

3.1.2.1.18 L'utilisateur doit pouvoir préciser la largeur et la hauteur des pages du catalogue à générer.

3.1.2.1.19 L'utilisateur ne doit pas pouvoir choisir une largeur de page inférieur à 105 mm et supérieur à 1682 mm.

3.1.2.1.20 L'utilisateur ne doit pas pouvoir choisir une hauteur de page inférieur à 148 mm et supérieur à 2378 mm.

3.1.2.1.21 L'utilisateur doit pouvoir choisir l'ordre d'insertion des images dans le catalogue.

3.1.2.1.22 Le système doit insérer les images dans le catalogue en respectant l'ordre établi par l'utilisateur cf 3.1.2.1.21.

3.1.2.1.23 Le système doit insérer les images sur chaque page en respectant le modèle choisi par l'utilisateur cf 3.1.2.1.2.

3.1.2.1.24 Le système doit pouvoir redimensionner les images sans réduire la qualité de l'image.

3.1.2.1.25 Le système doit gérer automatiquement la pagination du catalogue.

3.1.2.1.26 Le système doit respecter la taille et la police des numéros de page défini par le modèle de catalogue choisi par l'utilisateur cf 3.1.2.1.2.

3.1.2.1.27 Le système doit respecter le nombre minimum et maximum d'images à insérer par page selon ce qui est défini par le modèle de catalogue choisi par l'utilisateur cf 3.1.2.1.2.

3.1.2.1.28 Le système doit respecter la taille et la police du modèle de catalogue choisi par l'utilisateur cf 3.1.2.1.2.

3.1.2.1.29 L'utilisateur doit pouvoir ajouter des éléments de texte au catalogue.

3.1.2.1.30 L'utilisateur doit pouvoir choisir le style des éléments de texte cf 3.1.2.1.29.

3.1.2.1.31 Le système doit proposer plusieurs choix pour styler les éléments de texte cf 3.1.2.1.30.

3.1.2.1.32 Le système doit ordonner les éléments de texte en fonction des images correspondantes.

### 3.1.2.2 Modifier un catalogue existant

3.1.2.2.1 Le système doit empêcher un utilisateur de modifier un catalogue qui n'est pas son créateur.

3.1.2.2.2 L'utilisateur doit pouvoir sauvegarder les modifications apportées à un catalogue.

3.1.2.2.3 L'utilisateur doit pouvoir déplacer les éléments qui composent un catalogue manuellement.

3.1.2.2.4 L'utilisateur doit pouvoir redimensionner les éléments qui composent un catalogue manuellement.

3.1.2.2.5 L'utilisateur doit pouvoir changer le style des éléments qui composent un catalogue manuellement.

3.1.2.2.6 L'utilisateur doit pouvoir supprimer les éléments qui composent un catalogue manuellement.

3.1.2.2.7 L'utilisateur doit pouvoir supprimer tout le contenu du catalogue.

3.1.2.2.8 L'utilisateur doit pouvoir ajouter de nouvelles images au catalogue en respectant la cf 3.1.2.1.6 et cf 3.1.2.1.8.

### 3.1.2.3 Télécharger un catalogue existant

3.1.2.3.1 Le système doit empêcher un utilisateur de télécharger un catalogue qui n'est pas son créateur.

3.1.2.3.2 L'utilisateur doit pouvoir choisir un seul catalogue à télécharger à la fois.

3.1.2.3.3 Le système à travers un modal doit proposer les formats disponibles pour télécharger un catalogue à la suite de cf 3.1.2.3.2.

3.1.2.3.4 Le système doit lancer le téléchargement dès la validation de cf 3.1.2.3.3 par l'utilisateur.

#### 3.1.2.4 Importer des images sur son espace membre

3.1.2.4.1 L'utilisateur doit pouvoir importer des fichiers image dans son espace membre.

3.1.2.4.2 Le système ne doit pas accepter les fichiers autre que des fichiers images sous format SVG, TIF/TIFF, PNG, JPEG/JPG pendant l'importation cf 3.1.2.4.1.

3.1.2.4.3 Le système ne doit pas accepter un fichier image qui dépasse 100 Mo.

3.1.2.4.4 L'utilisateur doit pouvoir importer un nombre illimité d'images dans son espace membre.

3.1.2.4.5 Le système doit filtrer les duplications d'images dans l'espace membre lors de l'importation des fichiers cf 3.1.2.4.1.

3.1.2.4.6 Le système doit être capable de récupérer les dimensions des images importées.

3.1.2.4.7 Le système doit être capable de récupérer les résolutions des images importées.

3.1.2.4.8 Le système doit pouvoir détecter si 2 images sont identiques ou pas.

3.1.2.4.9 Le système ne doit pas filtrer les images identiques mais qui n'ont pas les même dimensions ou résolutions cf 3.1.2.4.6, cf 3.1.2.4.7, cf 3.1.2.4.8.

3.1.2.4.10 Le système doit enregistrer la date d'importation de chaque image.

3.1.2.4.11 Le système doit enregistrer les images importées dans la base de données du système.

#### 3.1.2.5 Visualiser des images sur son espaces membre

3.1.2.5.1 L'utilisateur doit pouvoir voir toutes les images qu'il les a importées sur son espace membre.

3.1.2.5.2 L'utilisateur ne doit pas voir les images dans l'espace membre d'un autre utilisateur.

3.1.2.5.3 L'utilisateur doit pouvoir voir la date d'importation de chaque image cf 3.1.2.4.10.

3.1.2.5.4 L'utilisateur doit pouvoir voir les dimensions de chaque image cf 3.1.2.4.6.

3.1.2.5.5 L'utilisateur doit pouvoir voir la résolution de chaque image cf 3.1.2.4.7.

3.1.2.5.6 L'utilisateur doit pouvoir supprimer chaque image dans son espace membre.

3.1.2.5.7 L'utilisateur doit pouvoir sélectionner un ensemble d'images dans son espace membre.

3.1.2.5.8 L'utilisateur doit pouvoir supprimer les images sélectionnées dans son espace membre cf 3.1.2.4.6.7.

3.1.2.5.9 Le système doit demander la confirmation de l'utilisateur via un modal pour une suppression d'image cf 3.1.2.5.8.

3.1.2.5.10 Le système doit demander la confirmation de l'utilisateur via un modal pour une suppression des images sélectionnées cf 3.1.2.5.6.

3.1.2.5.11 L'utilisateur doit pouvoir visualiser toutes les images qui seront supprimées cf 3.1.2.5.10.

3.1.2.5.12 L'utilisateur doit pouvoir sélectionner un ensemble d'images dans son espace membre pour créer ou modifier un catalogue.

## 3.2 Exigences non fonctionnelles

### 3.2.1 Temps de réponse

3.2.1.1 Le système doit être capable de répondre à une demande d'un utilisateur en moins de 10 secondes quelque en soit la complexité de celle-ci.

### 3.2.2 Taille des images

3.2.2.1 Le poids d'une image (sous tous les formats possibles : JPEG, PNG, etc.) doit être au maximum 100 Mo.

### 3.2.3 Performance

3.2.3.1 Le système doit être capable de supporter au moins 100 utilisateurs connectés simultanément sur l'application.

3.2.3.2 L'application doit pouvoir être lancée en moins de 5 secondes lors du démarrage.

3.2.3.3 Le système doit pouvoir traiter 180 transactions à l'heure quand il est soumis à des temps de réponse de 3 secondes.

3.2.3.4 Lors des heures de pointe (9h-midi et 14h-16h) 95% des transactions doivent être traitées en moins de 10 secondes.

3.2.3.5 Le temps de rafraîchissement de l'écran doit être inférieur à 1 seconde.



3.2.3.6 Le système doit produire un catalogue en moins de 10 secondes dans 95% des situations.

#### 3.2.4 Fiabilité

3.2.4.1 Le taux d'erreurs du système doit être de moins de 100 erreurs par 7 jours d'activité.

3.2.4.2 La probabilité d'inaccessibilité au système doit être inférieure à 3%.

3.2.4.3 La durée moyenne de rétablissement du système après défaillance doit être moins de 24h.

3.2.4.4 Le temps moyen entre 2 incidents du système doit être moins de 1h.

3.2.4.5 En cas de défaillance mineure le système doit pouvoir être redémarré automatiquement en moins de 10 secondes.

#### 3.2.5 Disponibilité

3.2.5.1 Le système doit être disponible au minimum 90% du temps.

3.2.5.2 Le système ne doit pas être indisponible plus 24h par 7 jours d'activité.

#### 3.2.6 Utilisabilité

3.2.6.1 Les nouveaux utilisateurs du système doivent être capables de créer des catalogues en moins de 15 minutes sur le site après visionnage d'une vidéo d'aide de 2 minutes.

3.2.6.2 Les utilisateurs basiques du système doivent être capables de créer des catalogues en moins de 2 minutes sur le site.

3.2.6.3 Après visionnage de cette vidéo cf 3.2.6.1, le nombre moyen d'erreurs effectuées par les utilisateurs du système ne doit pas excéder 2 par heure.

3.2.6.4 Le système doit comporter au moins 1 bouton d'aide pour chaque fonctionnalité du système.

3.2.6.5 Le niveau de satisfaction envers le système doit être très bon ou meilleur pour au moins 90% des clients ayant utilisé le système pendant 3 mois consécutifs.

#### 3.2.7 Maintenabilité

3.2.7.1 Le système doit garantir une maintenabilité de 100% des modules composants le logiciel.

3.2.7.2 Au moins 70% des modules du logiciel doivent avoir la côte « hautement maintenable ».

3.2.7.3 L'installation d'une nouvelle version du logiciel doit laisser 100% des préférences du catalogue des utilisateurs inchangées.

3.2.7.4 100% du code écrit doit être bien commenté et documenté pour faciliter les opérations de maintenance.

3.2.7.5 Aucune méthode d'une classe ne doit avoir plus de 300 lignes de code.

### 3.2.8 Portabilité

3.2.8.1 100% des fonctionnalités du logiciel doit fonctionner correctement sur les systèmes d'exploitation : Windows et Linux.

3.2.8.2 100% des fonctionnalités du logiciel doit fonctionner correctement sur les navigateurs : Chrome, Firefox, Opera, Safari, Edge.

3.2.8.3 100% des modules développés séparément doivent pouvoir fonctionner parfaitement et portable sur d'autres plateformes.

3.2.8.4 La taille du logiciel pour les versions Windows et Linux ne doit pas dépasser les 100 Mo.

### 3.2.9 Sécurité

3.2.9.1 Le système doit détecter au moins 99.9% des tentatives d'intrusion en moins de 10 secondes.

3.2.9.2 Après 10 tentatives erronées de connexion avec un mot de passe, le compte concerné est bloqué temporairement.

3.2.9.3 La durée de vie d'une session inactive doit être moins de 30 minutes, puis l'utilisateur sera déconnecté automatiquement.

3.2.9.4 Le pourcentage d'attaque réussie doit être moins de 1%.

3.2.9.5 La probabilité de trouver un mot de passe crypté d'un utilisateur doit être moins de 0.1%.

### 3.2.10 Ethique

3.2.10.1 Le système doit respecter le RGPD entré en vigueur le 25 mai 2018.

3.2.10.2 Après un signalement, l'administrateur doit supprimer les contenus illicites (images pornographiques, images de propagande du terrorisme, etc.) en moins de 10 minutes.

## 3.3 Exigences relatives à la base de données

Les données personnelles des utilisateurs récupérées, devront être stocker dans la base de données de façon sécurisée. De plus, l'utilisateur doit autoriser ce stockage lors de son inscription.

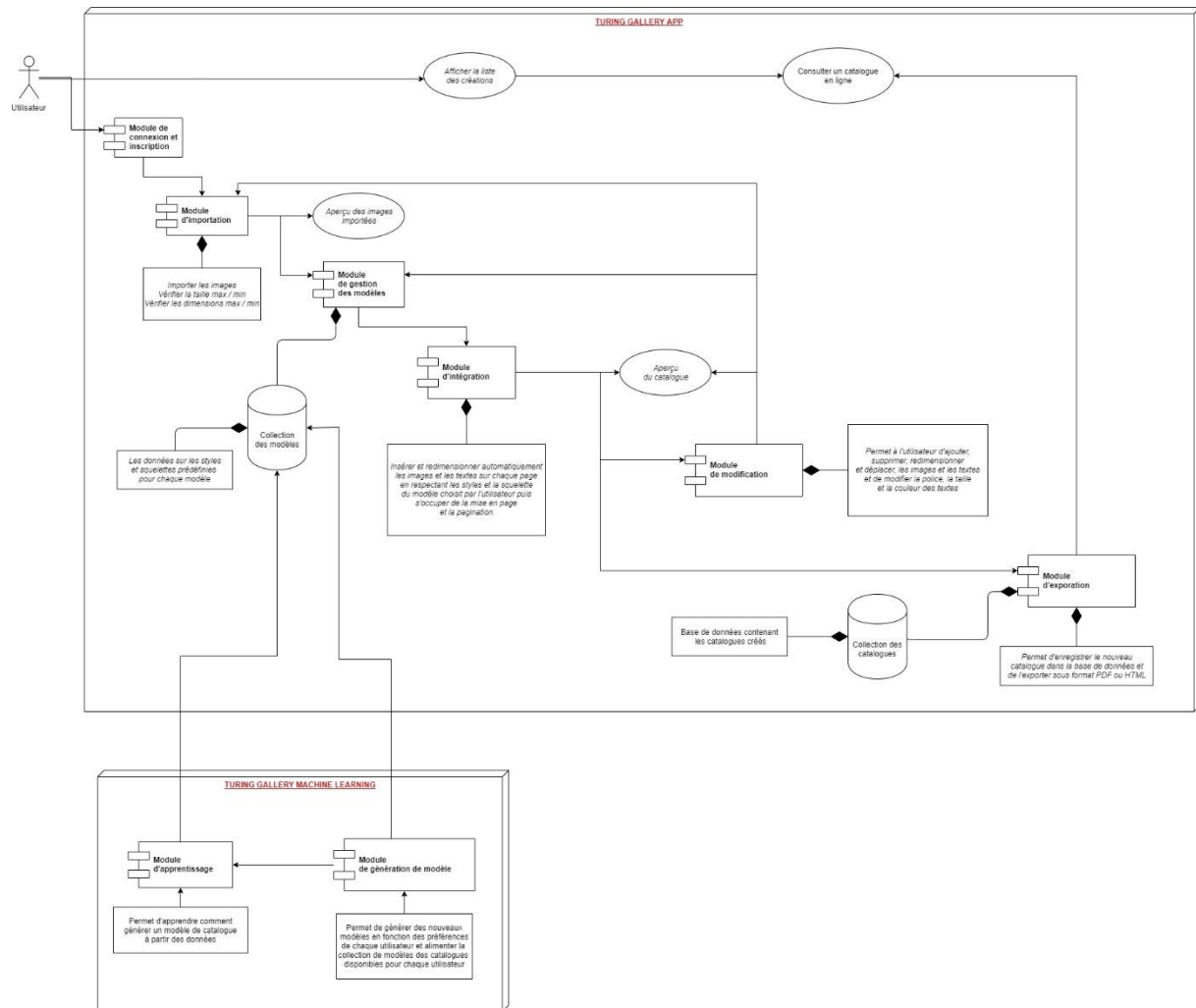
La base de données doit être capable de supporter un grand nombre de requêtes SQL simultanées et elle doit être assez large pour assurer le stockage d'un grand nombre de données.

### 3.4 Contraintes de conception

- Le langage utilisé pour le développement doit être Java.
- Le langage utilisé pour la partie du Machine Learning doit être Python.
- Le gestionnaire de versioning doit être Git.
- Le suivi du projet doit être fait avec Framaboard.
- Les documents envoyés doivent être sous format PDF.
- La rédaction des différents documents doit être faite avec Microsoft Word.
- La contribution sur les différents documents doit être faite avec Google Drive.
- La contribution sur les diagrammes doit être faite avec Draw.

## 4. Annexes

### ANNEXE 1 : Diagramme d'activité



## ANNEXE 2 : Format d'un email valide

### Valid email address format

A valid email address is required to sign up for an xMatters trial user account. A valid email address consists of an email prefix and an email domain, both in acceptable formats.

The **prefix** appears to the **left** of the @ symbol.

The **domain** appears to the **right** of the @ symbol.

For example, in the address *example@mail.com*, "example" is the email prefix, and "mail.com" is the email domain.

#### Acceptable email prefix formats

- Allowed characters: letters (a-z), numbers, underscores, periods, and dashes.
- An underscore, period, or dash must be followed by one or more letter or number.

Invalid email prefixes:	Valid email prefixes:
abc-@mail.com	abc-d@mail.com
abc..def@mail.com	abc.def@mail.com
.abc@mail.com	abc@mail.com
abc#def@mail.com	abc_def@mail.com

#### Acceptable email domain formats

- Allowed characters: letters, numbers, dashes.
- The last portion of the domain must be at least two characters, for example: .com, .org, .cc

Invalid email domains:	Valid email domains:
abc.def@mail.c	abc.def@mail.cc
abc.def@mail#archive.com	abc.def@mail-archive.com
abc.def@mail	abc.def@mail.org
abc.def@mail..com	abc.def@mail.com