

---

# ZOOM

---

## CAHIER DES CHARGES

Version 1 du 10/04/21

Destinataires : équipe projet

# Sommaire

Définition du projet.....	3
Enjeux.....	3
Priorités.....	3
Planning.....	3
Moyens.....	3
Budget.....	3
Spécifications fonctionnelles.....	4
Périmètre fonctionnel.....	4
Énoncé du besoin.....	4
Administration des contenus et gestion de l’application.....	4
Existant.....	4
Contrainte.....	4
Interface web.....	4
Aperçu.....	4
Spécifications techniques.....	5
Administration des contenus et gestion de l’application (Back Office).....	5
Interface web (Front Office).....	6
Tests unitaires.....	8

# Définition du projet

---

Présentation de galeries de photos au format web alimentées par différentes applications tierces.

## Enjeux

- 1.Mise en avant d'un sujet photographique
- 2.Administration automatisée

## Priorités

- 1.Limiter les coûts de gestion
- 2.Renforcer l'expérience utilisateur (qualité du vécu)
- 3.Modularité

## Planning

- Rétroplanning
- Date butoir

## Moyens

- Équipe projet
- Intervenants

## Budget

# Spécifications fonctionnelles

---

## Périmètre fonctionnel

Diagramme de cas d'utilisation :

- 1.Administration
- 2.Interface

## Énoncé du besoin

Il traduit de manière formelle la demande client et le contexte.

### Administration des contenus et gestion de l'application

- 1.Les contenus photographiques doivent pouvoir être mis à disposition par différentes applications tierces
- 2.La gestion des contenus doit être entièrement automatisée
- 3.Les photos doivent pouvoir être classées dans au moins une catégorie (galeries, albums)
- 4.L'interface devra permettre de présenter une ou plusieurs galeries sur la même page
- 5.Possibilité de choisir un thème graphique à l'initialisation

#### *Existant*

Une application tierce interne permet d'ores et déjà de gérer les contenus et offre la possibilité de déposer une série de photos dans un dossier de destination paramétrable.

#### *Contrainte*

L'application tierce interne ne permet aucun accès à une base de données.

### Interface web

- 1.Un système de navigation permet à l'utilisateur de choisir une thématique
- 2.Chaque galerie permet une lecture d'ensemble
- 3.L'utilisateur peut accéder à une version agrandie de la photo afin d'y trouver des informations complémentaires
- 4.L'interface s'adapte à différents médias de restitution
- 5.L'interface offre une expérience utilisateur de qualité

### Aperçu

- 1.Arborescence
- 2.Zoning & wireframe pour chaque page
- 3.Ambiance graphique

# Spécifications techniques

## Administration des contenus et gestion de l'application (Back Office)

Besoins	Contraintes	Solutions	Limites & évolutions possibles
La gestion des contenus doit être entièrement automatisée Les photos doivent pouvoir être classées dans au moins une catégorie (galeries, albums)	L'application tierce interne ne permet aucun accès à une base de données.	Dépôt des photos dans un dossier de l'application Le nom du dossier correspond à la catégorie Un script PHP serveur permet de lire le contenu d'un dossier  Le script devra contrôler le type de fichier déposé	Le nom de la catégorie correspond au nom donné à un dossier devant respecter la convention de nommage des fichiers numériques  Sécurité : version PHP 7.4 minimum
Les contenus photographiques doivent pouvoir être mis à disposition par différentes applications tierces		Module javascript/ajax permettant d'interroger une API  Mise au point du module pour l'API REST PIXABAY	Le module doit être paramétré pour chaque application  La solution proposée s'applique à un module
Possibilité de choisir un thème graphique à l'initialisation		L'application sera conforme au modèle MVC Chaque thème graphique s'appuiera sur un moteur unique  Un fichier de configuration permettra de rendre paramétrable les chemins d'accès au répertoire de personnalisation sans interface utilisateur	Évolution possibles  Possibilité de paramétrage depuis une interface utilisateur protégée par droit d'accès  Possibilité du choix du thème par le visiteur par enregistrement des préférences sur le navigateur client (cookies)

## Interface web (Front Office)

Besoins	Contraintes	Solutions	Limites
Un système de navigation permet à l'utilisateur de choisir une thématique		<p>Photos administrées par simple dépôt :</p> <p>Un script serveur permet de parcourir un dossier regroupant les dossiers de chaque galerie et de construire en dynamique un menu de navigation dont l'intitulé est le nom du dossier</p> <p>Photos administrées via une API</p> <p>La taxonomie dépend de chaque API</p>	
Chaque galerie permet une lecture d'ensemble		Chaque galerie présente l'ensemble des photos sous forme de miniature	En cas de nombre important de photos, un système de pagination permettra de limiter l'affichage à un nombre paramétrable de miniatures afin d'optimiser les temps de chargement et la navigation dans une galerie
L'utilisateur peut accéder à une version agrandie de la photo afin d'y trouver des informations complémentaires		<p>Un widget javascript permet d'accéder à la version détaillée et agrandie en cliquant ou en focusant la miniature</p> <p>Il est possible de prévoir plusieurs widgets paramétrable pour chaque thème</p>	<p>Photos administrées par simple dépôt :</p> <p>Les informations de chaque photo se limitent au nom de la photo</p> <p>Photos administrées via une API</p> <p>Les informations dépendent de chaque API</p>

Besoins	Contraintes	Solutions	Limites
L'interface s'adapte à différents médias de restitution		<p>L'interface sera compatible avec les principaux navigateurs, y compris mobile, à l'exception des versions non maintenues d'Internet Explorer</p> <p>Les modèles de page suivront les recommandations de bonnes pratiques en matière d'accessibilité web</p> <p>Référentiels :  <a href="http://www.accessiweb.org/index.php/accessiweb_2.2_liste_generale.html">http://www.accessiweb.org/index.php/accessiweb_2.2_liste_generale.html</a>  <a href="https://checklists.opquast.com/fr/assurance-qualite-web/">https://checklists.opquast.com/fr/assurance-qualite-web/</a></p> <p>La mise en page sera adaptée à différentes tailles d'écran par l'usage de medias queries en CSS</p>	Versions dont l'usage relatif est de + de 2 %
L'interface offre une expérience utilisateur de qualité		<p>Une animation, paramétrable pour chaque galerie, viendra conforter l'apparition de la photo agrandie</p> <p>Le chargement des pages sera optimisée par la création automatisée d'une miniature pour chaque photo déposée s'appuyant sur la librairie GD de PHP</p>	Les photos déposées doivent être optimisée au format web (à préciser)
L'interface devra permettre de présenter une ou plusieurs galeries sur la même page		Les widgets javascript seront développés sous forme de module indépendant et paramétrable depuis le code HTML de chaque galerie	

## Tests unitaires

Module	Tests	Résultat du test
Medias Lazy Lecture d'un dossier de galerie Langage php	Seuls les formats autorisés sont pris en compte  Tester avec différents types de fichiers : pdf, odt, docx + formats web autorisés	
Create Thumbnails Création d'une miniature Langage php	Le script ne doit traiter que les types de fichiers disposant de fonctions de traitement GD  Tester avec webm, svg + autres formats autorisés	
Zoom Apparition d'une image agrandie Langage javascript	Test avec plusieurs galeries sur une seule page  Le module fonctionne indépendamment pour chaque galerie	
Etc.		