



Projet – Gestionnaire de photos

Départements : TIC

Unité d'enseignement BDR

Auteurs : **Camille Koestli**
Guilherme Pinto Da Cunha Da Mata

Professeur : **Marc Dikötter**
Assistant : **Cédric Campos Carvalho**

Classe : **A**

Date : **22.11.23**

Tables des matières

Phase 1 : Cahier des charges

1	Introduction.....	2
2	Besoins des données	2
3	Fonctionnalités.....	2
	Utilisateur.....	2
	Artiste.....	3
	Photo	3
	Flux d'actualité	3
	Recherche par tag.....	3
	Réaction	4
	Dossier	4
	Commentaire.....	4
	Badge	4

Phase 2 : schéma UML

Phase 3 : schéma relationnel et création de la base de données

1.1	Utilisateur	8
1.2	Photo.....	8
1.3	Badge.....	8
1.4	BadgeUtilisateur	8
1.5	Tag.....	8
1.6	Statut.....	9
1.7	Dossier.....	9
1.8	DossierPhoto	9
1.9	TagPhoto	9
1.10	Reaction	9
1.11	Commentaire.....	9
1.12	TagUtilisateurs.....	10

Phase 4 : Requêtes, vues et triggers

1	Requêtes.....	11
	1.1 Utilisateur	11
	1.2 Artiste	11
2	Triggers.....	11
3	Vues	11

Phase 5 : Evolution et Conclusion

	Fonctionnalités implémentées	12
	Fonctionnalités non implémentées ou partiellement	13

Phase 1 : cahier des charges

1 Introduction

Nous avons choisi comme projet de créer un gestionnaire de photos. Ce système nous permettrait d'organiser nos photos en utilisant des tags ou en les triant par date, facilitant ainsi la recherche de photos spécifiques, telles que celles contenant des chiens, par exemple.

Cette application serait conçue pour deux types de personnes : les artistes et les utilisateurs. Les artistes auraient la possibilité d'importer des photos dans la base de données, tandis que les utilisateurs auraient un accès restreint à certaines photos, en fonction de leurs autorisations. Il est important de préciser que l'artiste aussi est un utilisateur.

2 Besoins des données

- Une photo a une date, des tags, un propriétaire, bibliothèque privée ou publique.
- Un utilisateur a un nom, un prénom, une date de naissance, un sexe, une adresse mail et un pseudo.
- Un artiste est une sous-classe d'utilisateur. Il peut avoir plusieurs suiveurs et plusieurs abonnés. Il peut également suivre ou être abonné à d'autres artistes.
- Un tag est une catégorie à laquelle une photo peut être associée. La photo peut contenir plusieurs tags que l'artiste écrira.
- Un suivi permet à un utilisateur d'avoir accès aux photos publiques de l'artiste qu'il suit et ainsi les avoir dans son flux d'actualité.
- Un abonnement permet à un utilisateur d'avoir accès aux photos publiques et privées d'un artiste auxquelles il est abonné. Pour être abonné, il faut que le photographe accepte la demande d'abonnement.

3 Fonctionnalités

Utilisateur

L'utilisateur est associé à un pseudo et a accès à toutes les bibliothèques d'images qui sont publiques grâce aux tags ajoutés aux images. De plus, il peut affiner ses recherches pour trouver des images plus spécifiques. Si l'utilisateur choisi de suivre un artiste, les photos publics sont accessibles directement dans son flux d'actualité. En s'abonnant à ces artistes, il peut accéder au contenu des photos qui sont classées privées.

Il existe trois niveaux d'utilisateurs :

1. Utilisateur sans suivi et sans abonnement : Il peut consulter le contenu public d'un artiste sans nécessairement le suivre ou être abonné. Les images de cet artiste ne seront pas visibles dans son flux d'actualités.
2. Utilisateur avec suivi mais sans abonnement : Il suit le contenu public de l'artiste qu'il suit, ce qui signifie qu'il voit ces images dans son flux d'actualités. Cependant, il n'a pas accès aux photos privées de ce même artiste.

3. Utilisateur avec abonnement : Il peut voir à la fois le contenu public et privé de l'artiste qu'il suit dans son flux d'actualités. De plus, l'utilisateur a la possibilité de mettre des photos en favori.

Artiste

L'artiste est une sous-classe de l'utilisateur. En tant qu'artiste, il a la possibilité de publier des photos et d'autres contenus sur son profil. De plus, il bénéficie d'un accès à toutes les bibliothèques de photos, de la même manière qu'un utilisateur standard, et peut également choisir de suivre et de s'abonner à d'autres artistes de son choix.

Les autres utilisateurs auront accès aux photos de l'artiste s'il les publie en tant que photos publiques. L'artiste a le contrôle sur la gestion des photos qu'il publie, notamment en ce qui concerne :

- La date de publication de la photo sur l'application.
- Les tags associés à l'image pour faciliter la recherche.
- Le paramètre de confidentialité de l'image, qui peut être public ou privé, déterminant ainsi si d'autres utilisateurs auront ou non accès à ces photos (via l'abonnement).

Photo

Il s'agit des images qu'il mettra en ligne sur son profil. Chaque image sera accompagnée d'un ou plusieurs tags pour faciliter la recherche, d'un titre ainsi que de la date à laquelle elle a été publiée.

Chaque artiste aura deux portfolios distincts :

1. Le contenu public : Tous les utilisateurs auront accès à ce contenu sans nécessairement s'abonner. Ces images seront accessibles grâce aux tags et à la date. En d'autres termes, n'importe quel utilisateur pourra voir ces images en utilisant les balises ou en effectuant une recherche basée sur la date.
2. Le contenu privé : Seuls les abonnés de l'artiste pourront accéder à ce contenu plus spécifique. Ces images ne seront visibles que par les utilisateurs qui ont choisi de s'abonner à cet artiste en particulier.

Chaque image inclura également un compteur de favoris, permettant aux utilisateurs de marquer leurs images préférées. Les images pourront également être classées en fonction des tags associés.

Flux d'actualité

Le flux d'actualité de chaque utilisateur sera personnalisé en fonction de ses propres préférences. Il affichera les photos les plus récentes (selon leur date de publication) des artistes qu'il suit ou auxquels il est abonné.

Recherche par tag

La recherche par tag doit permettre de retrouver rapidement une photo contenant un élément spécifique. Par exemple, si vous effectuez une recherche avec le tag "lune", vous verrez d'abord toutes les photos privées des artistes auxquels vous êtes abonné et qui contiennent ce même tag. Ensuite,

vous verrez toutes les photos publiques comportant également ce tag. Chaque groupe de photos sera trié par ordre décroissant en fonction du nombre de favoris qu'elles ont accumulés, vous montrant ainsi les images les plus populaires en premier.

Réaction

Sous chaque photo, vous trouverez un compteur en lien avec les réactions des utilisateurs. Ils auront la possibilité d'attribuer un "plus" ou un "moins" en fonction de leur appréciation de la photo. Les "plus" augmenteront le compteur, tandis que les "moins" le diminueront. Cette fonction permettra de classer les photos en fonction du nombre de réactions positives et négatives attribuées par les utilisateurs, offrant ainsi un moyen de trier les images en fonction de leur popularité et de la satisfaction des utilisateurs.

Dossier

Chaque utilisateur aura la possibilité de créer des dossiers personnels dans lesquels il pourra stocker les photos de son choix. Ces dossiers seront de nature individuelle, ce qui signifie qu'ils ne seront pas partagés avec d'autres utilisateurs. L'utilisateur pourra organiser et retrouver toutes les photos qu'il a sauvegardées dans ces dossiers.

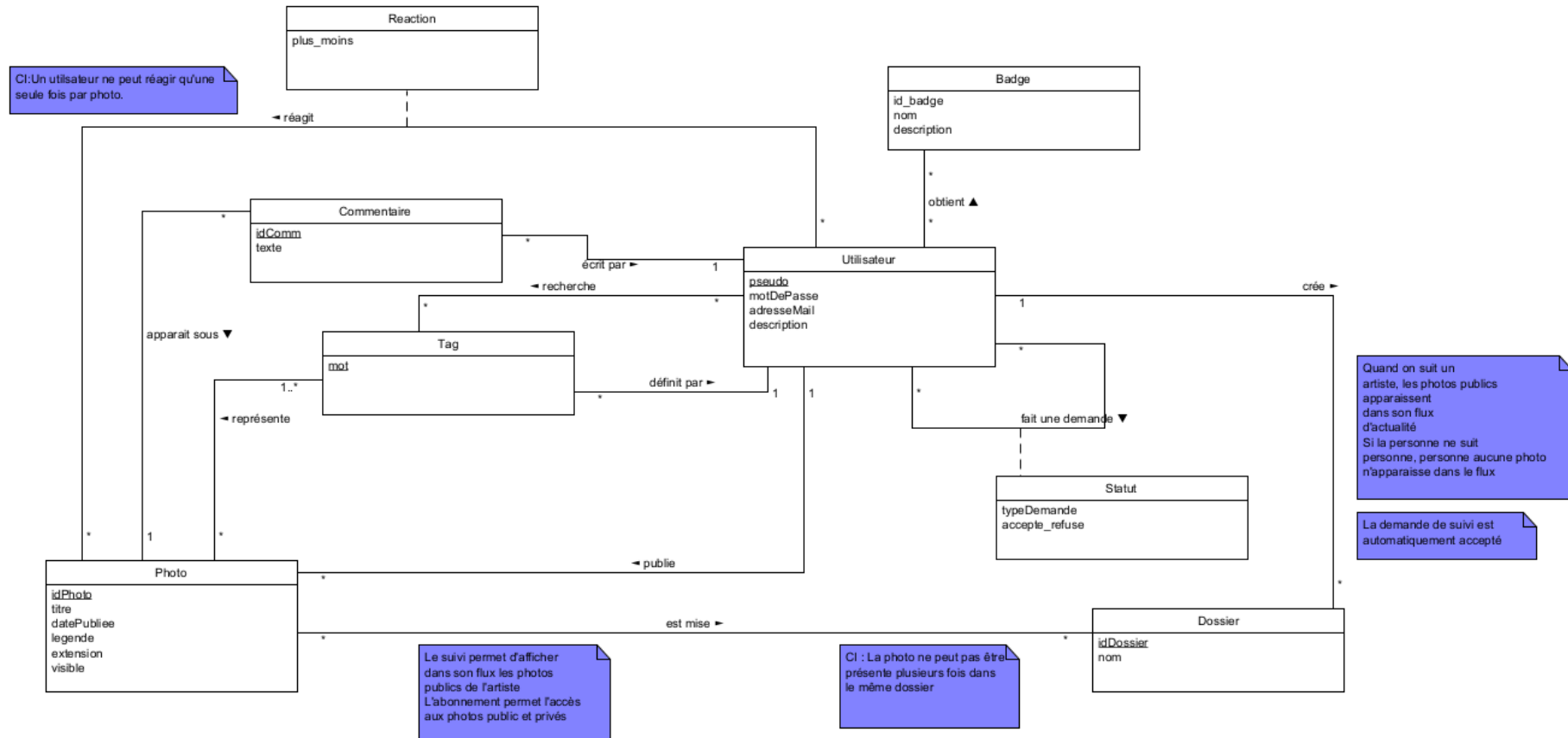
Commentaire

La dernière fonctionnalité que nous avons décidé d'ajouter est la possibilité de laisser des commentaires sur les photos. Chaque utilisateur peut ajouter des commentaires à une photo autant de fois qu'il le souhaite. Les commentaires seront affichés sous la photo dans l'ordre chronologique de leur date d'ajout, ce qui permettra aux utilisateurs de partager leurs réactions en lien avec les images, ce qui permettra une interaction entre les utilisateurs mais aussi l'artiste.

Badge

Les badges peuvent être gagné après avoir fait certaines actions, comme publier une première photo. Ces badges sont attribués aux artistes et ils symbolisent leur parcours en tant qu'artiste contribuant au réseau.

Phase 2 : schéma UML



Phase 3 : schéma relationnel et création de la base de données

Utilisateur (pseudo, motDePasse, adresseMail)

Statut (typeDemande, accepte_refus, pseudo, pseudoArt)

Statut.pseudo references Utilisateur.pseudo

Statut.pseudoArt references Artiste.pseudoArt

Commentaire (ID_comm, texte, pseudo, ID_photo)

Commentaire.pseudo references Utilisateur.pseudo

Commentaire.ID_photo references Photo.ID_photo

Commentaire.pseudo NOT NULL

Commentaire.ID_photo NOT NULL

Photo (ID_photo, titre, datePubliee, legende, extension, pseudo)

Photo.pseudo references Artiste.pseudo

Photo.pseudo NOT NULL

Tag (mot, pseudo)

Tag.pseudo references Artiste.pseudo

Tag.pseudo NOT NULL

TagUtilisateur (pseudo, mot)

TagUtilisateur.pseudo references Utilisateur.pseudo

TagUtilisateur.mot references Tag.mot

TagPhoto (mot, ID_photo)

TagPhoto.ID_photo references Photo.ID_photo

TagPhoto.mot references Tag.mot

Dossier (ID_dossier, nom, pseudo)

Dossier.pseudo references Utilisateur.pseudo

Dossier.pseudo NOT NULL

Reaction (plus_moins, ID_photo, pseudo)

Reaction.ID_photo references Photo.ID_photo

Reaction.pseudo references Utilisateur.pseudo

DossierPhoto (ID_dossier, ID_photo)

DossierPhoto.ID_dossier references Dossier.ID_dossier

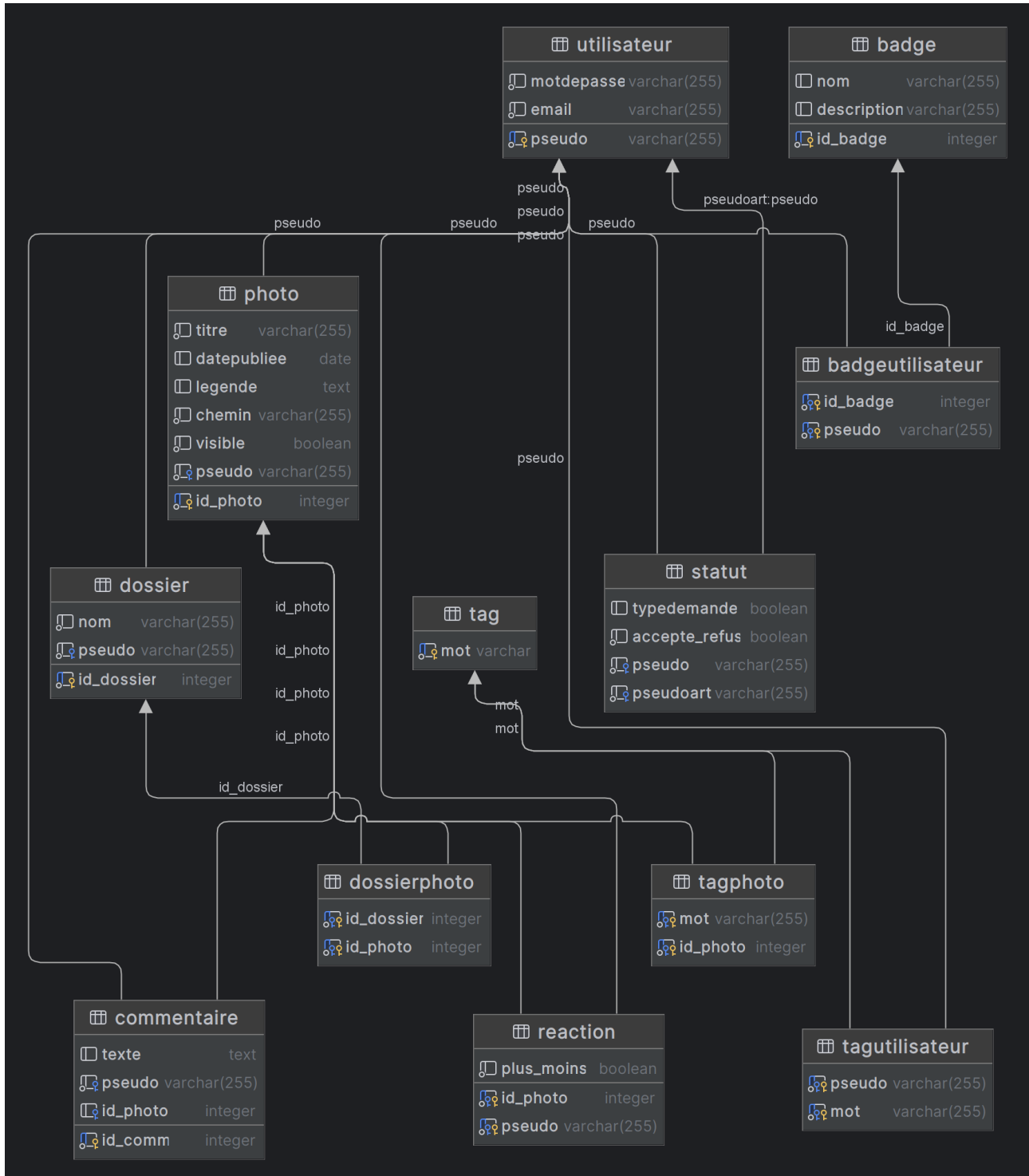
DossierPhoto.ID_photo references Photo.ID_photo

Badge (ID_badge, nom, description)

BadgeUtilisateur (pseudo, ID_badge)

BadgeUtilisateur.pseudo references Utilisateur.pseudo

BadgeUtilisateur.ID_badge references Badge.ID_badge



1.1 Utilisateur

Attribut	Type	Null	Signification
<u>pseudo</u>	Varchar(255)	Non	Pseudo de l'utilisateur
email	Varchar(255)	Non	Email de l'utilisateur
motdepasse	Varchar(255)	Non	Mot de passe de l'utilisateur

1.2 Photo

Attribut	Type	Null	Signification
<u>id_photo</u>	integer	Non	Identifiant de la photo
titre	Varchar(255)	Non	Email de l'utilisateur
datepubliee	date	Non	Date de publication
legende	text	Oui	Legende de la photo
chemin	Varchar(255)	Non	Chemin du fichier photo
visible	boolean	Non	Photo public si true, Photo privé si false
pseudo	Varchar(255)	Non	Pseudo de l'utilisateur

1.3 Badge

Attribut	Type	Null	Signification
<u>id_badge</u>	integer	Non	Identifiant du badge
nom	Varchar(255)	Non	Nom du badge
description	Varchar(255)	Non	Description

1.4 BadgeUtilisateur

Attribut	Type	Null	Signification
<u>id_badge</u>	integer	Non	Identifiant du badge
<u>pseudo</u>	Varchar(255)	Non	Pseudo de l'utilisateur

1.5 Tag

Attribut	Type	Null	Signification
<u>mot</u>	varchar	Non	Mot utilisé comme tag

1.6 Statut

Attribut	Type	Null	Signification
<u>pseudoart</u>	Varchar(255)	Non	Pseudo de l'artiste
<u>pseudo</u>	Varchar(255)	Non	Pseudo de l'utilisateur
accepte_refus	boolean	Non	Si true accepte, si false refuse
typedemande	boolean	Non	Si true suivre, si false abonnement

1.7 Dossier

Attribut	Type	Null	Signification
<u>id_dossier</u>	integer	Non	Identifiant du dossier
<u>pseudo</u>	Varchar(255)	Non	Pseudo de l'utilisateur
<u>nom</u>	Varchar(255)	Non	Nom du dossier

1.8 DossierPhoto

Attribut	Type	Null	Signification
<u>id_dossier</u>	integer	Non	Identifiant du dossier
<u>id_photo</u>	integer	Non	Identifiant de la photo

1.9 TagPhoto

Attribut	Type	Null	Signification
<u>mot</u>	Varchar(255)	Non	Mot utilisé comme tag
<u>id_photo</u>	integer	Non	Identifiant de la photo

1.10 Reaction

Attribut	Type	Null	Signification
<u>pseudo</u>	Varchar(255)	Non	Pseudo de l'utilisateur
<u>id_photo</u>	integer	Non	Identifiant de la photo
plus_moins	boolean	Non	Réaction positive/négative

1.11 Commentaire

Attribut	Type	Null	Signification
<u>id_comm</u>	integer	Non	Identifiant du commentaire

id_photo	integer	Non	Identifiant de la photo
pseudo	Varchar(255)	Non	Pseudo de l'utilisateur
texte	text	Non	Texte du commentaire

1.12 TagUtilisateurs

Attribut	Type	Null	Signification
<u>pseudo</u>	Varchar(255)	Non	Pseudo de l'utilisateur
<u>mot</u>	Varchar(255)	Non	Mot utilisé comme tag

Phase 4 : Requêtes, vues et triggers

1 Requêtes

1.1 Utilisateur

- Consulter tous les utilisateurs
- Consulter, créer et modifier un utilisateur
- Suivre ou ne plus suivre à un artiste
- S'abonner ou se désabonner à un artiste
- Créer, modifier et supprimer un dossier
- Mettre ou enlever une photo d'un dossier
- Ajouter ou modifier un commentaire
- Ajouter ou modifier une réaction
- Consulter les badges d'un utilisateur

1.2 Artiste

- Consulter tous les artistes
- Consulter les photos publiques et privées d'un artiste
- Accepter ou refuser une demande d'abonnement
- Ajouter ou supprimer une photo
- Spécifier les attributs de sa photo et spécifier le statut d'une photo (public ou privé)
- Ajouter, modifier ou supprimer des tags

2 Triggers

- Mise à jour des badges en fonction des différentes actions : première photo

3 Vues

- L'accès aux photos, qu'elles soient publiques ou privées, dépend des abonnements et des suiveurs.
- Les tags facilitent la recherche et la catégorisation des photos.

Phase 5 : Evolution

Le projet PhotoZ a connu plusieurs évolutions, dans le but de simplifier le code tout en gardant un maximum de fonctionnalités.

1. Unification des Entités Artiste et Utilisateur :

- **Avant** : Au départ, le système gérait séparément les entités Artiste et Utilisateur, ce qui entraînait une complexité supplémentaire dans la gestion des données et des interactions.
- **Après** : Les entités Artiste et Utilisateur ont été fusionnées en une seule entité Utilisateur. Cela a simplifié la structure de la base de données, réduit la redondance et facilité la gestion des utilisateurs.

2. Gestion de la Visibilité des Photos :

- **Avant** : Les photos étaient classées comme sous-classe privées ou publiques, ce qui nécessitait une gestion complexe des permissions et de l'affichage.
- **Après** : La distinction entre photos privées et publiques a été remplacée par un attribut de visibilité dans l'entité Photo. Ce changement a permis une gestion plus simple de la visibilité des photos.

3. Ajout de la Fonctionnalité de Badges :

- **Avant** : L'application ne disposait pas de récompense ou de reconnaissance pour les actions des utilisateurs sur la plateforme.
- **Après** : La fonctionnalité de badges a été ajoutée, afin de féliciter les utilisateurs pour certaines actions, comme la publication de photos. Cette fonctionnalité comprend des triggers, comme l'attribution d'un badge lors de la première publication d'une photo.

Conclusion

Fonctionnalités implémentées

1. **Authentification (Login/Logout) :**
 - Les utilisateurs peuvent se connecter et se déconnecter de l'application.
 - La gestion des sessions assure que les informations de l'utilisateur sont persistantes dans l'application.
2. **Flux d'Actualité Personnalisé :**
 - Les utilisateurs ont accès à un flux d'actualité différent qui affiche les photos en fonction du type d'abonnement.
3. **Affichage de Profil :**
 - Les profils utilisateurs permettent à chacun de voir les informations et les photos associées à un utilisateur spécifique.
 - Cela inclut la visualisation des photos postées par l'utilisateur et les détails de son profil, comme les badges.
4. **Gestion des Photos CRUD (Ajouter, Modifier, Supprimer, Afficher) :**
 - Il est possible de cliquer sur une photo afin d'y voir les informations supplémentaires comme les tags et les commentaires.
 - Les utilisateurs peuvent ajouter de nouvelles photos.
 - La modification des photos permet aux utilisateurs de mettre à jour les détails de leurs photos après publication.
 - La suppression de photos permet la suppression d'une photo.
5. **Système de Badges (Trigger sur Post de Photo) :**
 - Une fonctionnalité d'attribution de badges a été implémenté pour récompenser les utilisateurs lorsqu'ils postent des photos.
 - Le premier badge est décerné à l'utilisateur lors de la publication de sa première photo, à l'aide d'un trigger

Fonctionnalités non implémentées ou partiellement

- Gestion des Tags
- Gestion des Dossiers
- Gestion des Commentaires
- Gestion des Réactions
- Gestion des Abonnements et des Suivis