Spécification Technique - Modèle de Données

# 1. Présentation Générale

Le modèle de données de l'application a été conçu pour supporter un réseau social lucratif reposant sur un système freemium. Il permet la gestion des utilisateurs, des publications publiques et premium, des abonnements, des paiements, ainsi que de la messagerie privée.

# 2. Description des Entités Principales

## 2.1 Utilisateur (`users`)

Cette table stocke les informations de base sur chaque utilisateur, qu'il soit créateur de contenu, abonné ou administrateur. Chaque utilisateur a un rôle défini et peut interagir différemment avec la plateforme selon ce rôle.

## 2.2 Publications (`posts`)

Les publications sont créées par les utilisateurs de type créateur. Elles peuvent être publiques ou réservées aux abonnés (premium). Chaque publication peut contenir du contenu texte et être associée à un ou plusieurs médias.

## 2.3 Médias (`media`)

Les médias sont les fichiers (images ou vidéos) associés aux publications. Chaque média est lié à un post spécifique.

## 2.4 Abonnements (`subscriptions`)

Les abonnements permettent aux utilisateurs d'accéder à tous les contenus premium d’un créateur pendant une durée déterminée. Un abonnement est actif entre une date de début et une date de fin, et peut être de type mensuel ou paiement unique.

## 2.5 Accès individuel à un post (`post\_access`)

Cette table gère les achats de posts à l’unité. Contrairement à un abonnement, cet accès est permanent. Elle est utilisée pour les utilisateurs qui veulent acheter un post premium sans s’abonner à un créateur.

## 2.6 Paiements (`payments`)

Les paiements permettent de tracer chaque transaction effectuée par les utilisateurs. Un paiement peut être lié à un abonnement (via `subscription\_id`) ou à l’achat d’un post unique (via `post\_id`).

## 2.7 Commentaires et Likes

Les utilisateurs peuvent commenter et liker les publications. Ces interactions sont enregistrées dans les tables `comments` et `likes`, chacune liée à un post et à un utilisateur.

## 2.8 Messagerie Privée (`messages`)

La messagerie privée est gérée via une seule table `messages`. Chaque ligne représente un message échangé entre deux utilisateurs, identifiés par `sender\_id` et `receiver\_id`. Ce design est optimisé pour des conversations simples en tête-à-tête.

# 3. Logique d'accès au contenu premium

Pour accéder à un contenu premium, l'application vérifie deux conditions :  
1. Si l'utilisateur a un abonnement actif au créateur du post (via `subscriptions`).  
2. Si l'utilisateur a acheté ce post en accès individuel (via `post\_access`).  
L'accès est accordé si l'une des deux conditions est remplie.

# 4. Cohérence et évolutivité

Le modèle est conçu pour être extensible. Il permet de suivre précisément les paiements, les accès, et les interactions. Il facilite aussi la mise en place de statistiques, de reporting, ou d’évolutions comme la gestion de badges ou de niveaux de créateurs.