

Conception base de données

Petit topo sur la conception de base de données, plus vous passerez du temps sur la conception, moins vous aurez à revenir sur la structure de votre base, et éviter de modifier la base avec du contenu déjà inséré.

Il existe plusieurs méthodes, essentiellement la méthode Merise et Uml, je vais détailler la méthode Merise.

Qu'est-ce que la méthode Merise ? Merise est une méthode qui permet de traduire un modèle en une base de données. Cela permet de mieux structurer notre base de données avant de la créer.

Ci-dessous les différentes étapes :

Lister les fonctionnalités

à partir des éléments fournis (cahier des charges, mockup, créa...), nous allons d'abord lister les fonctionnalités, après une analyse.

Je prendrai l'exemple du blog :

- une page d'accueil, avec une liste d'articles
- un article est composé par :
 - un titre
 - une date
 - une catégorie
 - un auteur (nom, prénom)
 - le contenu de l'article
- une page propose tout le contenu de l'article, avec la possibilité d'ajouter des commentaires (pseudo, commentaire, date).
- d'après les mockups du panneau d'administration (admin), nous aurons la gestion des articles, des auteurs et des catégories

Remarques que l'on peut dire en conclusion de l'analyse :

- un Id est défini pour chaque article

- un article est relié à un seul auteur, et une seule - catégorie un auteur peut avoir plusieurs articles
- une catégorie peut contenir plusieurs articles
- un article peut contenir plusieurs commentaires
- un commentaire est relatif à un seul article

Définir les groupes de données

A partir de cette analyse nous allons définir les groupes de données (entités) de manière à répéter le moins possible le contenu de nos données, ce travail va permettre de définir nos futures tables.

dans l'exemple du blog :

- des articles (Post)
- des commentaires (Comment)
- des auteurs (Author)
- des catégories (Category)

Définir les champs visibles

Post :

- Title (titre)
- Contents (contenu)
- CreationTimestamp (date de création)

Comment :

- NickName (pseudo)
- Contents (contenu du commentaire)
- CreationTimestamp (date de création)

Author :

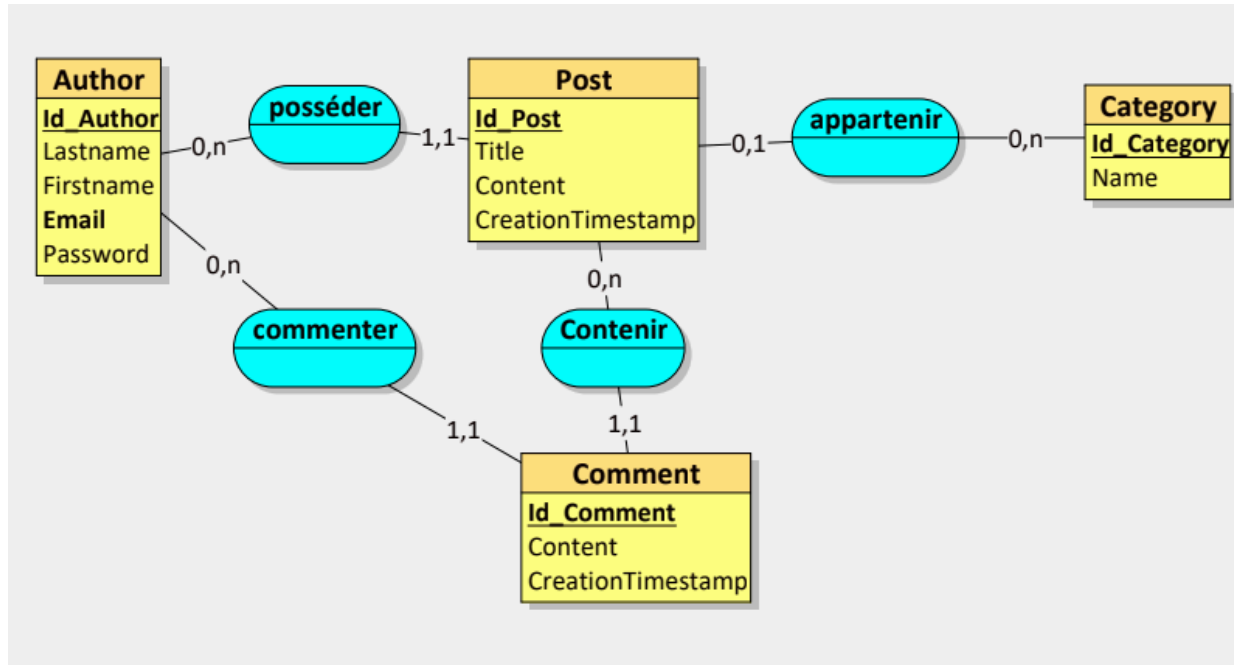
- LastName (nom)
- FirstName (prénom)

Category :

- Name (nom)

Modèle Conceptuel de données

Suite à cette définition, nous allons mettre en place les relations entre les entités, en créant un MCD. Un MCD (Modèle Conceptuel de Données) est composé de plusieurs entités et associations.



Entités

Une entité va permettre de modéliser votre première table de votre base de données. Elle va être caractérisée par un nom et des propriétés.

Associations

Une association est une relation entre deux entités.

Dans notre exemple, l'association permet de relier nos entités "Post" et "Category" de la manière suivante :

Un Post possède une Category et une Category est possédée par un Post.

Les associations vont nous permettre de lier les entités entre elles. Et par la suite de créer les clés étrangères de notre base de données.

Pour l'entité Author c'est la même chose.

Au sujet de l'association "Post" et "Comment", Un Post possède 0 ou plusieurs Comment, et un Comment possède un seul Post.

Cardinalités

Les cardinalités expriment le nombre minimum et maximum de fois où l'entité participe à la relation c'est-à-dire :

La cardinalité de notre entité "Post" (pour l'association avec Category) est de 0,1 donc au minimum, un Post a 0 Category et au maximum, il a une Category.

Nous dirons souvent que "Le Post possède une seule et unique Category".

La cardinalité de notre entité "Category" est de 0,N donc au minimum une Category contient 0 Post et au maximum, elle contient N Post avec N étant un nombre entier positif.

Cardinalités minimales

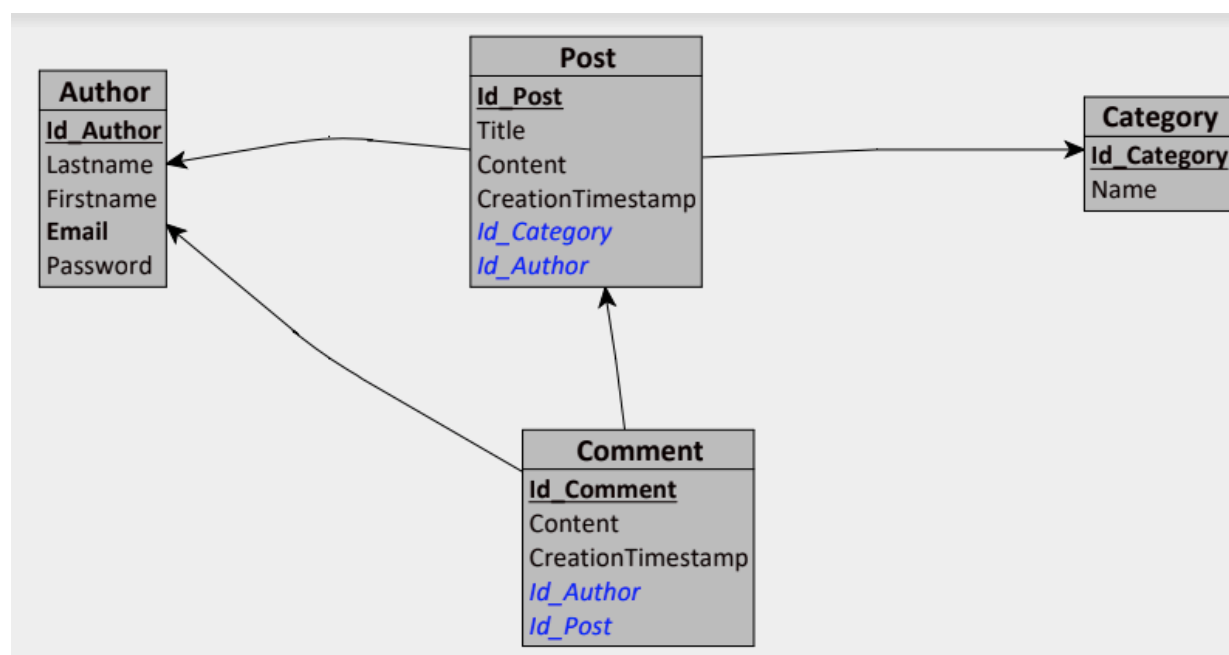
Une cardinalité minimale est de 0 ou 1 avec 0 signifiant que l'entité ne participe pas à l'association et 1 signifie que l'entité participe à l'association.

Cardinalités maximales

Une cardinalité maximale est de 1 ou N avec N un entier positif. 1 signifie que l'entité ne participe au maximum qu'une seule fois et N, qu'elle participe au maximum un certain nombre de fois.

Modèle logique de données

Le MLD (Modèle Logique de données) découle du MCD. Dans cette partie, nous allons transformer les associations.



Entités faibles/fortes

Une entité est faible si sa cardinalité maximale est de 1.

Une entité est forte si sa cardinalité maximale est de N.

Exemples :

L'entité "Post" est dite faible car sa cardinalité 0,1 possède un maximum de 1 Category.

L'entité "Category" est dite forte car sa cardinalité 0,N possède un maximum de N.

Lorsqu'une entité est faible, elle absorbe l'identifiant de l'entité forte et dans ce cas, une nouvelle propriété apparaît dans l'entité faible que nous appellerons clé étrangère.

Remarque :

Notre Post étant une entité faible, il va obtenir l'identifiant de l'entité "Category" et l'association disparaît.

Modèle physique de données

Le modèle physique de données (MPD) découle du modèle logique de données (MLD). Ce modèle permettra d'avoir le schéma de notre base de données.

Dans cette étape, les entités vont devenir des tables, toutes les propriétés vont obtenir un type de données (Varchar, integer, date, time, etc.) et les identifiants vont devenir des clé primaires.

