

RAPPORT BDD

Présenter la base de données.....	2
Ancienne version VS nouvelle version.....	5
MLD	7
Script SQL.....	8
• Constructeurs	8
• Requêtes utiles.....	9
Insérer les informations d'un client	9



Présenter la base de données

Avant de réaliser notre base de données, nous nous sommes demandé quelles sont les données qui vont nous être utiles.

Ensuite nous avons commencé avec une première version très peu optimisée, puis nous avons réalisé une version plus compacte, par le fait que nous avons commencé à développer le site.

Nous nous sommes donc aperçus qu'on pouvait retirer des choses inutiles.

Nous avons donc 9 tables :

- 1. Client**
- 2. Ajouter_note**
- 3. Encherir**
- 4. Publier**
- 5. Interesser**
- 6. Commenter**
- 7. Visionnage**
- 8. Annonce**
- 9. Associer_categorie**

1. Client

Dans cette table nous avons toutes les informations du client, c'est-à-dire son nom, prénom, date de naissance, adresse, ville, code postal, email, mdp = mot de passe (qui sera sécurisé), newsletter qui est un booléen indiquant si cet utilisateur souhaite recevoir de la pub dans sa boîte mail, et admin, un booléen également permettant ainsi de définir directement si l'utilisateur joue un rôle d'admin ou non.

De plus pour que chaque client soit reconnaissable facilement, nous avons un id_client qui permet à chaque client d'avoir une identification unique.

2. Ajouter_note

Dans cette table nous avons deux identifiants client, le premier concernera l'acheteur, l'autre le vendeur et nous aurons aussi la notation qui correspondra à la note qu'un client acheteur pourra adresser au client vendeur. Si le vendeur vend des produits intéressants, le client acheteur pourra lui donner une bonne note et inversement.

3. Encherir

Dans cette table nous avons, le prix_actuel de l'annonce, le nouveau prix que le client va insérer et la date. Nous avons choisi cette disposition de sorte que lorsqu'un client va enchérir, il va mettre le prix actuel à jour, avec le nouveau prix, et nous mettrons la

date permettant ainsi de pouvoir faire des statistiques sur l'évolution des enchères sur une annonce.

4. Publier

Cette table sert simplement à sauvegarder dans une table le client avec son annonce. Il pourra nous servir pour ajouter des statistiques comme sur l'évolution du nombre de vente d'un vendeur ou encore tout simplement le nombre d'annonce que ce client a publié depuis qu'il a créé son compte.

5. Interesser

Cette table est similaire à celui de publier, seulement que celui là va nous permettre de savoir quelles sont les annonces que le client a décidé de mettre en favoris, permettant ainsi de sauvegarder les annonces qui intéressent l'acheteur.

6. Commenter

Dans cette table, nous aurons l'identifiant du client et de l'annonce dans lequel nous aurons le commentaires que le client a saisi (acheteur ou client), permettant ainsi les échanges accompagné de la date pour pouvoir dater les commentaires pour savoir quel est le dernier commentaire publié ainsi que de visualiser si le vendeur est réactif. De plus cela peut être nous permettre de mettre en avant une annonce plutôt qu'une autre si elle a plus de commentaires car cela veut dire qu'il y a plus d'informations qu'une annonce sans commentaires.

7. Visionnage

Dans cette table, nous aurons comme précédemment, l'identifiant du vendeur et de l'annonce dans lequel nous allons incrémenter à chaque fois qu'une personne visionnera l'annonce avec la date. Cela pourra aider le vendeur à savoir si il a fait plus de vues un jour plutôt qu'un autre.

8. Annonce

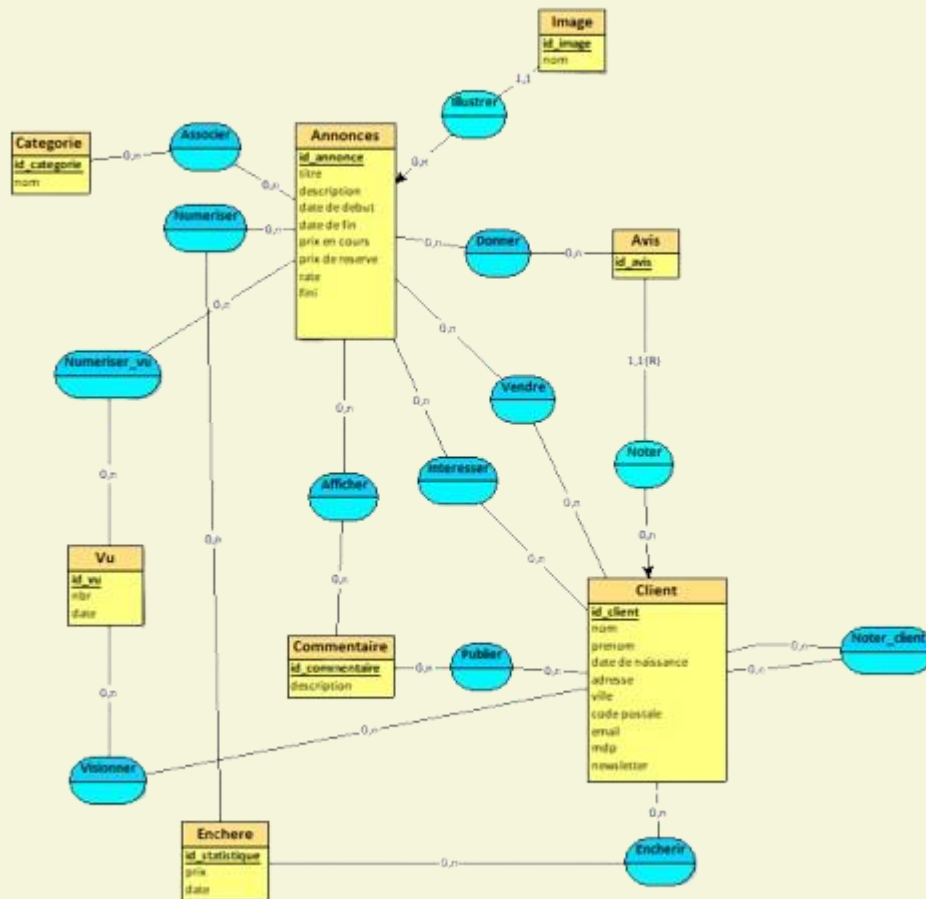
Comme pour la table client, nous allons récupérer plusieurs informations pour l'annonce. Nous aurons donc l'identifiant de l'annonce qui sera une clé primaire et qui sera une manière de référencer une annonce de façon unique. De plus nous aurons le titre, la description, la date de début de l'annonce, la date de fin pour pouvoir afficher un compteur du temps restant, le prix actuel, le prix de réserve et le prix de départ pour nous permettre de stocker toutes les enchères que l'annonce va avoir ainsi que son status (terminé, à valider, supprimer, en cours) et le nombre de vue, en corrélation avec la table visionnage.

9. Associer_categorie

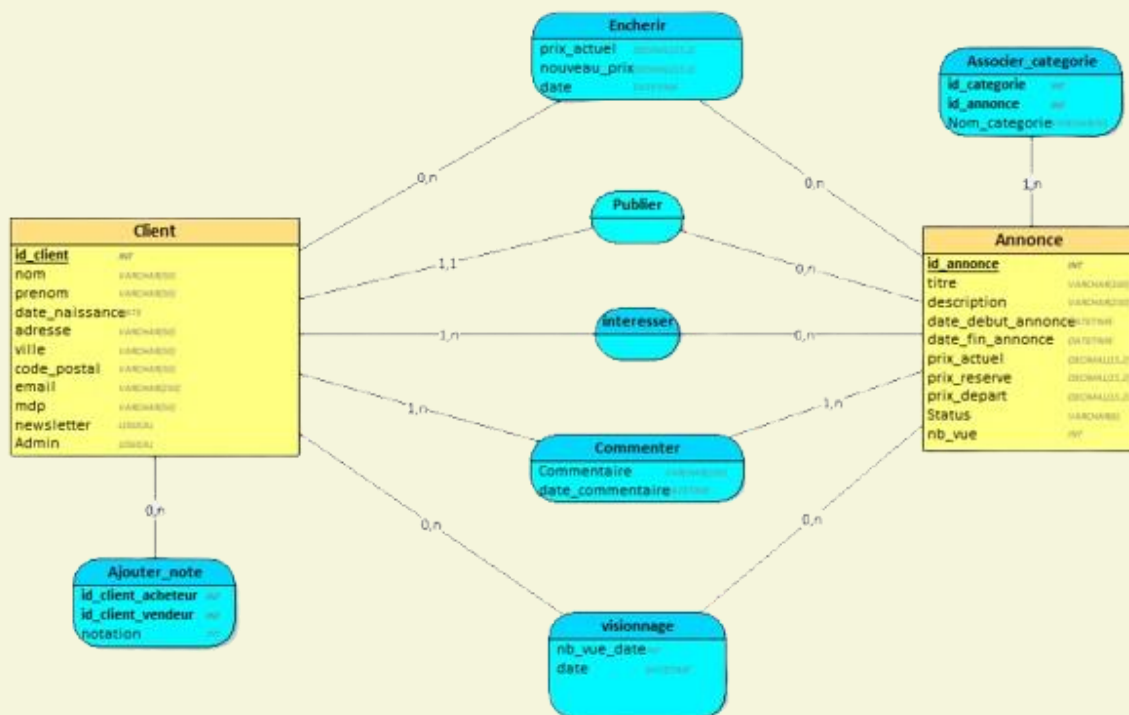
Cette table va nous permettre de catégoriser une annonce, si cela concerne de l'art, une voiture, ... nous pourrons y ajouter des mots clés pour pouvoir de trier les annonces lorsque le possible acheteur voudra chercher une annonce.

Ancienne version VS nouvelle version

Nous avons amélioré notre ancienne version en enlevant des tables qui nous étaient plus dérangeantes qu'utiles et nous avons enlevé la classe image car les images seront sauvegardées dans un dossier du projet via php. Cette nouvelle version est donc plus efficace, plus simple et plus facilement exploitable.



Ancienne version



Nouvelle version

MLD

Annonce = (id_annonce INT, titre VARCHAR(100), description VARCHAR(250), date_debut_annonce DATETIME, date_fin_annonce DATETIME, prix_actuel DECIMAL(15,2), prix_reserve DECIMAL(15,2), prix_depart DECIMAL(15,2), Status VARCHAR(6), nb_vue INT);

Client = (id_client INT, nom VARCHAR(50), prenom VARCHAR(50), date_naissance DATE, adresse VARCHAR(50), ville VARCHAR(50), code_postal VARCHAR(50), email VARCHAR(250), mdp VARCHAR(50), newsletter LOGICAL, Admin LOGICAL, #id_annonce);

Interesser = (#id_annonce, #id_client);

Publier= (#id_annonce, #id_client);

Visionnage = (#id_annonce, #id_client, nb_vue_date INT, date_ DATETIME);

Encherir = (#id_annonce, #id_client, prix_actuel DECIMAL(15,2), nouveau_prix DECIMAL(15,2), date_ DATETIME);

Commenter = (#id_annonce, #id_client, Commentaire VARCHAR(550), date_commentaire DATETIME);

Ajouter_note = (#id_client_acheteur INT, #id_client_vendeur INT, notation INT);

Associer_categorie = (id_categorie INT, #id_annonce INT, Nom_categorie VARCHAR(50));

Script SQL

- **Constructeurs**

```
DROP DATABASE IF EXISTS site_enchere ;
```

```
CREATE DATABASE site_enchere;
```

```
USE site_enchere;
```

```
CREATE TABLE Annonce ( id_annonce INT, titre VARCHAR(100), description
VARCHAR(250), date_debut_annonce DATETIME, date_fin_annonce DATETIME,
prix_actuel DECIMAL(15,2), prix_reserve DECIMAL(15,2), prix_depart
DECIMAL(15,2), status VARCHAR(6), nb_vue INT, PRIMARY KEY (id_annonce) );
```

```
CREATE TABLE Client ( id_client INT, nom VARCHAR(50), prenom
VARCHAR(50), date_naissance DATE, adresse VARCHAR(50), ville VARCHAR(50),
code_postal VARCHAR(50), email VARCHAR(250), mdp VARCHAR(50), newsletter
BOOLEAN, admin BOOLEAN, PRIMARY KEY (id_client));
```

```
CREATE TABLE Interesser ( id_annonce INT, id_client INT, PRIMARY KEY
(id_annonce, id_client), FOREIGN KEY (id_annonce) REFERENCES
Annonce(id_annonce), FOREIGN KEY (id_client) REFERENCES Client(id_client) );
```

```
CREATE TABLE Visionnage ( id_annonce INT, id_client INT, nb_vue_date INT,
date_ DATETIME, PRIMARY KEY (id_annonce, id_client), FOREIGN KEY
(id_annonce) REFERENCES Annonce(id_annonce), FOREIGN KEY (id_client)
REFERENCES Client(id_client) );
```

```
CREATE TABLE Encherir ( id_annonce INT, id_client INT, prix_actuel
DECIMAL(15,2), nouveau_prix DECIMAL(15,2), date_ DATETIME, PRIMARY KEY
(id_annonce, id_client), FOREIGN KEY (id_annonce) REFERENCES
Annonce(id_annonce), FOREIGN KEY (id_client) REFERENCES Client(id_client) );
```

```
CREATE TABLE Commenter ( id_annonce INT, id_client INT, commentaire
VARCHAR(550), date_commentaire DATETIME, PRIMARY KEY (id_annonce,
id_client), FOREIGN KEY (id_annonce) REFERENCES Annonce(id_annonce),
FOREIGN KEY (id_client) REFERENCES Client(id_client) );
```

```
CREATE TABLE Ajouter_note ( id_client_acheteur INT, id_client_vendeur INT,
notation INT, PRIMARY KEY (id_client_acheteur, id_client_vendeur), FOREIGN
```



```
KEY (id_client_acheteur) REFERENCES Client(id_client), FOREIGN KEY
(id_client_vendeur) REFERENCES Client(id_client) );
```

```
CREATE TABLE Associer_categorie ( id_categorie INT, id_annonce INT,
nom_categorie VARCHAR(50));
```

```
CREATE TABLE Publier (id_client INT, id_annonce INT, PRIMARY KEY(id_client,
id_annonce), FOREIGN KEY (id_client) REFERENCES Client(id_client), FOREIGN
KEY (id_annonce) REFERENCES Annonce(id_annonce));
```

• Requetes utiles

Insérer les informations d'un client

```
INSERT INTO client (nom, prenom, date_de_naissance, adresse, ville, code_postale,
email, mdp, newsletter, admin) VALUES ("GARNIER", "Jimmy", "2006/10/11", "48
rue du perlinpinpin", "Nevers", "58000", "jimmy.garnier1110@gmail.com",
"csecretça", 0);
```

Rechercher un client via son courriel et son mot de passe

```
SELECT * FROM client WHERE email = email and mdp = password;
```

Trier les annonces du plus cher au moins cher

```
SELECT * FROM Annonce ORDER BY prix_actuel DESC;
```

Rechercher un mot dans un texte

```
SELECT * FROM Annonce WHERE titre LIKE '%voiture%';
```

Limiter le nombre de résultats

```
SELECT * FROM Annonce LIMIT 5;
```