

# Rendu 3 du Projet en Base de Données

BIN5\_WAVE



Iman Mellouk  
Herve Nguyen  
Groupe de TP5

# Table des matières

Table des matières	2
Présentation	3
Manuel Utilisateur	3
Remarques techniques	4
Description technique du site	7
Pistes d'améliorations	8
Organisation du travail	8

# Présentation

Notre projet en base de données consiste à développer une base de donnée qui répond aux besoins d'un site de streaming audio.

Le projet s'est déroulé en trois étapes, la conception de la base de données, la création de cette base de donnée et enfin l'application de cette BDD (Base de données) à travers un site web écrit en PHP.

Cette BDD doit gérer les informations à propos des morceaux, albums, utilisateur, groupes, playlists de la plateforme.

## Manuel Utilisateur

le site est accessible depuis ce lien : <https://etudiant.u-pem.fr/~iman.mellouk/index.php>

il est possible de se connecter avec les identifiants suivant : pseudo = 'Nousagi1', mdp =AZERTYUIOP

Le site PHP que nous avons développé a d'abord une page où l'utilisateur a le choix soit de s'inscrire ou de se connecter, pour s'inscrire il suffit d'entrer les informations requises pour l'inscription (et doivent être cohérentes). Il est tout aussi simple de se connecter, le pseudo et le mot de passe sont demandé.

En général, le site n'apprécie pas les retours en arrière via le navigateur en raison de l'utilisation quasi systématique des formulaires POST dans les pages.

L'utilisateur peut se déplacer à travers le site via les menus situé vers le haut qui permet de se rendre aux pages de recherches, de profil personnel, de suggestions et d'actualités.

Au sein d'une page, lorsque des informations sur des objets comme des albums / utilisateurs / groupes sont affichés, un bouton (en général, nommé « détails »)

correspondant à ces objets sont présent ce qui permet de se diriger vers la page spécifique concernant ces objets. Par exemple, pour un album le bouton va envoyer l'utilisateur vers la page qui contient ses informations tel que le groupe et les morceaux qui composent l'album etc.

Pour écouter un morceau, il suffit alors de cliquer sur le bouton détail correspondant à un morceau à travers d'une recherche, dans le profil de l'utilisateur ou dans les suggestions, puis de cliquer sur le bouton écouter.

Pour créer une playlist, il faut aller dans le profil de l'utilisateur puis de cliquer sur le bouton créer une playlist. Pour ajouter un morceau, il faut être dans la page spécifique au morceau et cliquer sur ajouter à \*nom de la playlist\*. Pour supprimer un morceau, il faut aller dans les détails de la playlist de l'utilisateur et appuyer sur le bouton « enlever de la playlist ».

Pour suivre, se désister envers un groupe ou un utilisateur, on peut simplement aller sur la page spécifique d'un groupe ou d'un utilisateur pour suivre et arrêter de suivre, car il y a également un bouton qui correspond à ces actions.

## Remarques techniques

La base de données en elle-même n'a pas du tout changé depuis le rendu 2. Voici le schéma relationnel de la BDD.

Voici le schéma E/A modifié suite aux remarques du 1 er rendu.

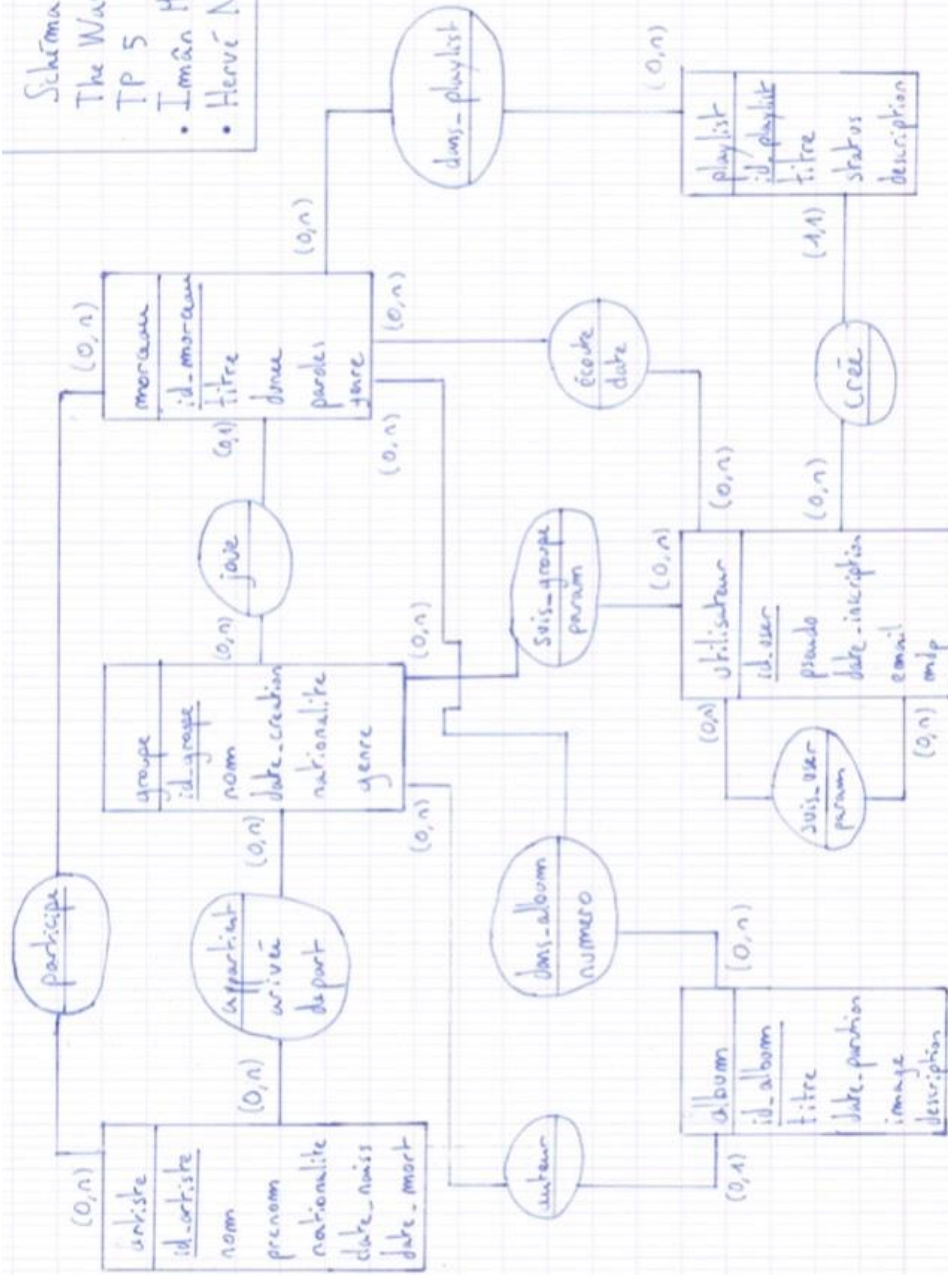
# Schema EA V2

The Wave

TP 5

• Imâm HELLOUK

• Hervé NGUYEN



## Schéma Relationnel

artiste ( id\_artiste, nom, prenom, nationalite, date\_naiss, date\_mort )

morceau ( id\_morceau, titre, donnee, paroles, genre, id\_groupe )

groupe ( id\_groupe, nom, date\_creation, nationalite, genre )

utilisateur ( id\_user, pseudo, date\_inscription, email, mdp )

playlist ( id\_playlist, titre, visible, description, id\_user )

album ( id\_album, titre, date\_parution, image, descript, auteur )

participe ( id\_artiste, id\_morceau )

appartient ( id\_artiste, id\_groupe, arrivée, depart )

suis\_groupe ( id\_groupe, id\_user, parametre)

ecoute ( id\_morceau, id\_user, date\_ecoute )

dans\_playlist ( id\_morceau, id\_playlist, numero)

suis\_user ( id\_user, id\_user2 )

dans\_album ( id\_morceau, id\_album, numero)

FK:

id\_artiste dans (appartient,participe) fait ref à la clé primaire id\_artiste de artiste  
id\_groupe / id\_groupe dans (appartient, suis\_groupe, morceau) fait ref à la clé primaire id\_groupe de groupe

id\_user / id\_user dans (suis\_groupe, suis\_user, playlist, ecoute) fait ref à la clé primaire id\_user de utilisateur

id\_user2 dans (suis\_user) fait ref à la clé primaire id\_user de utilisateur

id\_album dans (dans\_album) fait ref à la clé primaire id\_album de album

id\_morceau dans (participe, ecoute, dans\_album, dans\_playlist) fait ref à la clé primaire id\_morceau de morceau

auteur dans (album) fait ref à la clé primaire id\_groupe dans groupe.

Légende : Clé primaire, Clé étrangère

# Description technique du site

Le site utilise beaucoup de formulaires POST pour envoyer les informations vers une page cible. Les informations à propos de l'utilisateur connecté est gardé dans la variable \$\_SESSION, on y retrouve l'id\_user de l'utilisateur et son pseudo.

Dans quasiment toutes les pages, le site vérifie si l'utilisateur est connecté, sinon en l'envoie vers l'index.

```
$search = "%{$champ}%";
$q = $dbh->prepare("SELECT id_playlist, titre, pseudo
FROM playlist NATURAL JOIN utilisateur
WHERE (lower(titre) LIKE :champ) OR (pseudo LIKE :champ) OR (lower(descript) LIKE :champ)
AND (visible = 't' OR pseudo = :pseudo)");
$q->execute(['champ' => $search, 'pseudo' => $pseudo]);
$result = $q->fetchAll();
```

Les requêtes du site sont assez simple pour la plupart, par exemple :

Ici, la requête répond à une recherche de playlist (publique ou personnelle) dans la page de recherche. On y retrouve des conditions sur le pseudo, des conditions sur l'attribut visible etc.

```
$q = $dbh->prepare("SELECT count(+), id_morceau, titre, nom, id_groupe
FROM ecoute NATURAL JOIN (SELECT id_morceau, titre, nom, morceau.id_groupe FROM morceau INNER JOIN groupe ON morceau.id_groupe = groupe.id_groupe) AS max2021
WHERE id_user = :usr
GROUP BY id_morceau, titre, nom, id_groupe
ORDER BY count DESC
LIMIT 1");
$q->execute(['usr' => $id_user]);
$gl = $q->fetch();
$idgl = $gl['nom'];
echo "<h3> Les utilisateurs qui écoutent $idgl écoutent aussi: </h3>";
$q = $dbh->prepare("SELECT +
FROM (SELECT DISTINCT B.id_morceau, B.titre, B.nom
FROM (ecoute NATURAL JOIN (SELECT id_morceau, titre, nom, morceau.id_groupe FROM morceau INNER JOIN groupe ON morceau.id_groupe = groupe.id_groupe) AS max2021) AS A,
(ecoute NATURAL JOIN (SELECT id_morceau, titre, nom, morceau.id_groupe FROM morceau INNER JOIN groupe ON morceau.id_groupe = groupe.id_groupe) AS max2021) AS B
WHERE A.id_user = B.id_user
AND A.id_morceau <= B.id_morceau
AND (B.id_groupe = :id OR A.id_groupe = :id)
LIMIT 5) AS peko
ORDER BY random()
");
```

Par contre les requêtes sur les suggestions sont un peu plus complexe :

Comme celle-ci où la première requête consiste à obtenir le groupe le plus écouté de l'utilisateur. Puis la deuxième requête cherche aléatoirement des morceaux écoutés par les auditeurs du groupe en question.

On peut remarquer que les requêtes n'utilisent pas « morceau NATURAL JOIN groupe », cela est en raison d'une colonne commune « genre » entre ces deux tables. Ce problème a été énoncé dans le rapport du rendu 2 sur les vues.

Nous avons choisi de ne pas corriger ce problème mineur car nous considérons notre base de donnée au rendu 2 définitif.

## Pistes d'améliorations

Des améliorations possible peuvent être visible au niveau de l'interface utilisateur et de l'ergonomie du déplacement dans le site. L'affichage des informations sur les morceaux et albums peuvent être complétés d'images et d'animations.

Une possible amélioration peut être aussi la correction du « problème » de la BDD énoncé précédemment.

## Organisation du travail

Dans le rendu 1, la conception de la base de donnée a été faite systématiquement à deux pour avoir une compréhension commune de la structure de la base de donnée.

Dans le rendu 2, l'objectif était de traduire simplement le schéma entité-association du rendu 1 en schéma relationnel puis de le porter au SQL. Iman est la principale contributrice du schéma relationnel obtenu, et le portage au SQL a été assez simple donc un membre du binôme avait certaines tables à écrire et l'autre membre du binôme le reste des tables. Les choix sur les contraintes ont été décidés sur discussion au sein du binôme.

Dans le rendu 3, le rôle d'Iman était autour des pages d'inscriptions, les playlists, le fonctionnement des suivis, la connection. Tandis que Hervé était porté sur les pages de recherches, d'actualités et de suggestions et les écoutes.



