

Projet de Base de données

L3 Informatique

1 Le sujet

Musique en ligne. Vous êtes développeur Web et souhaitez créer un site de musique en ligne. Avant de créer l'interface Web, il convient de créer proprement la base de données sous-jacente et de réfléchir aux fonctionnalités futures.

2 Les Entités-Associations

Dans ce qui suit, certaines entités ne sont pas listées explicitement mais apparaissent dans la description d'autres entités. A vous de les reconnaître. Quand on parle de date, on considère indifféremment la date et l'heure car *mysql* gère ces informations dans un même type.

- Chaque artiste possède un numéro qui permet de l'identifier. Dans ses attributs, on retrouve un nom, un descriptif et un lien (texte) vers une photo.
- Un artiste enregistre des albums. Chaque album est identifié par un numéro, possède un nom et un lien (texte) vers une image.
- Un album contient des morceaux. Chaque morceau possède un lien vers un fichier MP3 qui lui sert d'identifiant, un nom, une durée et peut même contenir un lien vers une vidéo. Un morceau peut être associé à un ou plusieurs styles de musique.
- Les usagers sont identifiés par leur pseudonyme. Il existe différents types d'usager : les temporaires, les inscrits et les abonnés. Les temporaires ont un pseudonyme engendré aléatoirement au moment de leur connexion (géré par le site Web, pas par la base de données). On souhaite néanmoins garder le même pseudonyme pour un utilisateur temporaire se connectant plusieurs jours d'affilée, afin de conserver les informations sur ses goûts musicaux. Pour cela, on conserve son adresse ip. Les inscrits et les abonnés ont renseigné d'autres types d'information : un nom, un prénom, une adresse mail, une adresse réelle, un descriptif, un mot de passe. Les abonnés payent un forfait. Il peut exister plusieurs types de forfaits et on ne souhaite pas les définir à l'avance. On sait juste que chaque forfait a un nom et un coût. Lorsqu'un usager paye un forfait, on veut connaître la date de début et de fin du forfait.
- Les usagers qui sont inscrits ou abonnés peuvent être amis.

- Un usager possède des playlists. Une playlist possède un nom et réunit une liste de morceaux. Lorsqu'un morceau apparaît sur une playlist, on souhaite connaître sa position dans la liste.
- On souhaite également stocker la liste des morceaux qu'un usager a écouté, ainsi que la date de début et de fin de l'écoute d'un morceau. Si un usager a écouté plusieurs fois le même morceau, on veut connaître le nombre de fois où il l'a fait.
- Une agence de publicité possède un nom, une adresse postale, une adresse mail et un numéro de téléphone. Une agence paye pour une ou plusieurs publicités. Une publicité possède un nom, un prix et un lien vers le fichier MP3 qui contient le message publicitaire. Un usager peut subir une publicité s'il n'est pas abonné. On souhaite connaître, pour chaque usager, la date à laquelle il a subi une publicité.

3 Action à effectuer sur la base

Écrire les requêtes *sql* permettant d'afficher les informations suivantes :

- la liste des usagers ayant écouté un morceau.
- la liste des usagers ayant écouté un groupe.
- le style musical d'un groupe est défini par la liste des styles des morceaux qu'il a composé. On souhaite obtenir le style musical d'un groupe (par exemple, selon un site de musique en ligne, Janis Joplin a le style musical suivant : "blues, psychédélique, 70's, rock, classic rock, blues rock, southern rock, 60's").
- pour un usager donné, la liste des playlists de ses amis. Attention, certaines playlists peuvent avoir le même nom.
- On souhaite connaître pour un artiste donné en entrée, la liste des artistes similaires, c'est à dire possédant le même style musical.
- On souhaite connaître, pour un usager donné, la durée cumulée d'écoute sur les dernières 24 heures. Cette requête doit permettre au site Web de pouvoir limiter, s'il le souhaite, le temps pour lequel l'utilisateur va pouvoir écouter de la musique dans une journée.
- On souhaite connaître le nombre d'utilisateurs qui se sont inscrits au cours du mois (en commençant la date au premier jour du mois)
- On souhaite connaître la somme d'argent obtenue grâce aux abonnements obtenus lors du mois en cours.
- On souhaite pouvoir envoyer des factures aux agences publicitaires. Pour cela, on veut pouvoir lister sur un mois donné, toutes les dates auxquelles un usager a subi une publicité d'une agence. On voudra voir apparaître la date, le pseudonyme de l'utilisateur, le nom et le prix de la publicité.
- Dans une deuxième requête, on utilisera la liste précédente pour calculer le coût total des frais de publicité. Pour cela, on fera la somme des prix obtenus dans la liste précédente.

4 Projet : dates importantes et organisation

- Pour le vendredi 25 octobre : rendre le schéma de la base de données dans le modèle entité-association.
- Pour le vendredi 22 novembre : rendre le schéma de la base de données dans le modèle relationnel. Vous donnerez également la liste des dépendances d'inclusions et des dépendances fonctionnelles qui vous semblent pertinentes.
- Pour le Mercredi 18 novembre : envoyer par mail la liste de vos fichiers *sql* permettant : la création de la base de données (*create.sql*), sa destruction (*clean.sql*), la liste de requêtes (*queries.sql*).
- Vendredi 20 décembre : soutenance de projet.

Organisation :

- à chaque étape, une correction vous sera donnée. Vous devrez vous baser sur cette correction pour préparer l'étape suivante.
- les notes intermédiaires ne vous seront pas communiquées.
- un retard à une étape équivaut à un 0 pour la note de cette étape. Rendre le devoir en retard signifie "après que j'ai distribué la correction".
- vous pouvez rendre une étape en avance, mais la correction sera donnée à tout le monde en même temps, le jour du rendu d'une étape.
- les corrections seront disponibles en séances de TD et sur mon site web : <http://lipn.fr/~david/>
- lors de la soutenance, les requêtes *sql* devront tenir compte du nom exact des relations et attributs donnés dans la correction du modèle relationnel. En effet, je préparerais un script *sql* pour remplir automatiquement les bases de données que vous aurez créées. Je testerai ensuite vos requêtes sur cette base de données.