

# Mini projet de Bases de Données Avancées

Université Jean Monnet  
Licence Informatique 2<sup>eme</sup> année

## Profils utilisateurs pour l'auto-complétion

Le site <https://qanswer-frontend.univ-st-etienne.fr/> permet la recherche directe d'information dans des bases de connaissances publiques. Actuellement, ce site ne gère pas les utilisateurs et leur profil, rendant impossible l'auto-complétion (compléter automatiquement la question en cours d'écriture). L'idée de ce projet est de constituer la base de données et les routines de base qui permettront une implémentation simple des profils et donc de l'auto-complétion.

Soit la base de donnée suivante proposée (et qui peut être améliorée) :

User(IP, name, firstname)  
Query(ID\_Q, #IP, text\_Q)  
Key\_word(ID\_KW, text)  
User\_KW(#IP, #ID\_KW, weight, lastUpDate)  
  
Connection(ID\_C, #IP, date, time, duration, #browserID)  
Browser(ID\_B, name\_B)

L'utilisateur est identifié par son adresse IP, et il a éventuellement donné son nom, prénom au système. Une question est un texte écrit par l'utilisateur dans l'interface. Les mots-clés sont des termes issus d'un lexique ou extraits des questions des utilisateurs. La table User\_KW constitue le profil de l'utilisateur : association pondérée (poids = weight) d'un utilisateur (IP) avec un mot-clé, et indication de la date de la dernière mise à jour de ce poids.

Créer la base/les tables<sup>1</sup> et les peupler.

Ecrire les fonctions ci-après, chaque fonction retournant un entier indiquant 1 (OK) ou 0 (erreur).

- Ajout/modification d'un utilisateur.
- Ajout/modification d'un mot-clé.
- Ajout dans User\_KW : soit on ajoute un tuple, soit on ajoute +1 au poids du tuple existant ; et dans tous les cas on indique la date du jour.
- Maintenance de la table User\_KW. Fonction avec un entier X en paramètre. Pour chaque tuple de la table, si la date est antérieure à X années, alors on remet le poids à 0.

Ajouter une table LesSimilaires(U, U1, U2, U3, U4, U5). Cette table portera les identifiants des 5 utilisateurs les plus proches de l'utilisateur U.

Ecrire une procédure Similarité (sans paramètre) qui calcule pour chaque utilisateur les 5 autres utilisateurs les plus proches et renseigne la table Similaires. La fonction de distance pourra être établie en utilisant le nombre de mots-clés en commun.

Ecrire une fonction Proche\_de(Nom\_Utilisateur, i) qui retourne, pour un nom d'utilisateur passé en paramètre, l'identifiant de l'utilisateur le plus proche de rang i. La fonction retourne NULL en cas d'erreur.

Ecrire un script qui exploite toutes ces routines.

Le compte-rendu de ce projet se limitera aux scripts commentés et exécutés<sup>2</sup>.

---

<sup>1</sup> La table Query est optionnelle. Elle ne sera pas utilisée dans le projet.

<sup>2</sup> TEE nomFichier permet la création d'un fichier de sortie, résultat de l'exécution de SOURCE. NOTEE désactive TEE.