

## TD N°8 : **uberisation**

### Thème : modélisation d'une application de transport de personnes

#### Contexte

La société de transport de personnes TOPCHRONO souhaite mettre en place une plateforme extranet à destination des particuliers. L'application proposera des services similaires à ceux proposés par la plateforme UBER :

- La plateforme répertorie les chauffeurs et les passagers. Diverses informations sont conservées à leur sujet : adresse email, prénom, nom, « hash » du mot de passe, etc. Rien n'empêche a priori qu'un chauffeur en repos puisse être passager !
- TOPCHRONO effectue une qualification de tous les chauffeurs, à savoir qu'un chauffeur effectue une inscription préalable. Son inscription n'est « convertie » en un compte chauffeur qu'une fois l'inscription validée. Afin de pouvoir afficher le périmètre d'intervention d'un chauffeur, on souhaite pouvoir conserver son adresse principale sous la forme d'un couple longitude et latitude. Il appartient au chauffeur de choisir son périmètre d'intervention. On suppose que le périmètre d'intervention du chauffeur est un cercle.
- Un passager propose une course en partance d'un lieu et à destination d'un autre lieu. Pour des raisons d'affichage encore, les lieux sont stockés sous la forme de couples longitude/latitude. La course intervient à une date et une heure donnée. Le logiciel se charge de fournir une estimation de la date et de l'heure d'arrivée.
- Les chauffeurs peuvent consulter les courses qui sont dans leur périmètre d'intervention. Si la course est dans ce périmètre, ils peuvent proposer leurs services à titre de candidats. Ils peuvent alors consulter les coordonnées du passager et le contacter. Il appartient au passager de choisir le chauffeur qui réalisera sa course. Pour l'accompagner dans sa décision, grâce à la plateforme, le passager peut consulter le profil des chauffeurs et son tarif. La plateforme fournit, pour chaque chauffeur une estimation du tarif pratiqué ;
- Dans la première version, le tarif du chauffeur est tarif forfaitaire (montant fixe) assorti d'un tarif kilométrique ;
- Pour chaque course, la longueur du trajet est stockée. Elle est arrondie au kilomètre le plus proche. Elle est obtenue grâce à l'API Google Maps. Cette longueur sert de base de calcul à l'estimation du tarif pratiqué par le chauffeur.
- Par ailleurs, TOPCHRONO, société soucieuse de respecter ses partenaires, chauffeurs et passages, prélève une commission d'apporteur d'affaire sur chacune des candidatures des chauffeurs. La commission est un montant fixe fonction de la longueur du trajet. Le montant est fixé au moyen d'un barème progressif (exemple : 5€ si le trajet a une longueur comprise entre 0km et 50km exclus, 9€ s'il est compris entre 50km et 100km exclus, etc.).

## Travail à faire

### 1. Modélisation

1.1.	Modéliser la base de données du logiciel sous forme de MCD (ou de diagramme de classes) au regard des informations fournies.
1.2.	Rédiger le schéma relationnel correspondant à votre MCD.
1.3.	Afin de mieux répondre aux attentes des chauffeurs, proposer une évolution leur permettant de fixer un tarif kilométrique dégressif, à savoir diminuant (ou augmentant) en fonction de la longueur du trajet.

### 2. Fonctions stockées

2.1.	Implémenter la fonction stockée <code>tarif_commission(distance INT) AS DECIMAL</code> qui retourne la commission pratiquée en fonction de la longueur d'un trajet.
2.2.	Implémenter la fonction stockée <code>tarif_chauffeur(numChauffeur INT, distance INT) AS DECIMAL</code> qui retourne le tarif pratiqué par un fournisseur en fonction de la longueur d'un trajet.
2.3.	Implémenter la fonction stockée <code>tarif_chauffeur(numChauffeur INT, distance INT) AS DECIMAL</code> en tenant compte de l'évolution 1.3. <i>N.B. : on pourra se servir d'un bloc <code>WHILE ... DO ... END WHILE ;</code></i>
2.4.	Quelle condition permet de vérifier si une course est dans le périmètre d'un chauffeur ? Exprimer cette condition de manière formelle.
2.5.	Rédiger la fonction stockée <code>est_dans_perimetre(...)</code> <code>AS BOOLEAN</code> qui permet de tester si une course est dans le périmètre d'un chauffeur. <i>N.B. : on pourra utiliser la fonction racine carrée <code>SQRT(Nombre DECIMAL)</code>.</i>

Exemple d'utilisation de `WHILE` pour parcourir un jeu d'enregistrements :

```

DECLARE curseur CURSOR FOR ( SELECT champ1, champ2 FROM table1 ... ) ; -- lignes à parcourir
DECLARE i AS INT ; -- compteur pour parcourir les lignes
SELECT count(*) INTO @Nb FROM table1 ... ; -- comptage du nombre de lignes à parcourir
i = 0
WHILE i < Nb DO
    FETCH curseur INTO var1, var2 ; -- récupère le contenu de la ligne courante et passé à la suivante
    ....
END WHILE ;
    
```

### 3. Procédures stockées

3.1.	Implémenter la procédure stockée <code>valider_inscription(numChauffeur INT)</code> permettant de valider l'inscription d'un chauffeur.
3.2.	Implémenter la procédure stockée <code>chauffeurs_proximite(...)</code> permettant de lister les chauffeurs disponibles en fonction d'une course, c'est-à-dire tous les chauffeurs dans le périmètre de la course.
3.3.	Implémenter la procédure stockée <code>tarif_moyen(distance INT)</code> qui affiche le tarif moyen (et l'écart type) pratiqué sur une distance donnée. <i>N.B. : on pourra se servir de l'agrégat <code>STD(...)</code> pour le calcul de l'écart-type, auquel cas on précisera comment l'on pourrait faire sans. STD pour Standard Deviation.</i>