

- Rapport de projet -
**Réalisation d'une base de données pour une
société de location de véhicules**

Université Cergy Pontoise

Auteurs : Babaedjou Hanyatou
Phetramphand Antoine

Sommaire

I. Présentation du projet	4
1. Contexte	4
2. Objet	4
3. Organisation	4
4. Environnement	4
5. Planification	4
II. Objectif	5
1. Description brève du logiciel	5
2. Le but de cette base de données	5
3. Liste des fonctionnalités	5
4. Contraintes	6
III. Livraisons attendues	6
IV. Conception du projet	7
1. Schéma MCD	7
2. Schéma MLD	7
V. Description du schéma conceptuel	9
1. Description des tables (Entités)	9
a) table AGENCE	9
b) table CLIENT	9
c) table EMPLOYE	9
d) table VEHICULE	9
e) table UTILITAIRE	10
f) table LOUER	10
2. Description des relations entre les tables	10
VI. Scripts SQL	12
1. Création des tables	12
a) table AGENCE	12
b) table EMPLOYE	12
c) table UTILITAIRE	12

d) table VEHICULE	12
e) table CLIENT	13
f) table LOUER	13
2. Attribution des clés étrangères	13
3. Les requêtes	14
a) requête n°1.....	14
b) requête n°2	14
c) requête n°3.....	15
d) requête n°4	15
e) requête n°5.....	16
f) requête n°6	16
g) requête n°7.....	16
h) requête n°8	17
VII. Scripts PHP	18
1. Quelques illustrations.....	18
2. Codes	20
VIII. Remarques sur le déroulement du projet	22
1. Méthode de travail au sein du groupe	22
2. Difficultés rencontrées	22
IX. Conclusion	23

I. Présentation du projet

1. Contexte

Dans le cadre de notre formation à l'Université de Cergy Pontoise, les étudiants en Licence *Informatique* troisième année réalisent des projets personnels afin de valider leur année scolaire. Ce type de réalisation exige donc un certain temps, un investissement personnel et l'usage de toutes les ressources disponibles. Ici, nous allons réaliser une base de données pour une société de location de véhicules.

2. Objet

Le but est la réalisation d'une application qui consiste à créer une base de données, puis manipuler les données à l'aide du langage *SQL*.

Ce document est destiné à identifier et décrire les besoins de la direction de la société de location de véhicules.

3. Organisation

Notre équipe est composée de deux membres à la charge du projet : Babaedjou Hanyatou et Phetramphand Antoine.

4. Environnement

Les ressources que nous disposons pour le développement de notre projet sont :

- **JMerise** : logiciel permettant de réaliser notre schéma *MCD*.
- **SQL Developer** : logiciel permettant de programmer nos tables, ainsi que nos requêtes *SQL* sous Oracle.
- **Sublime Text** : logiciel permettant d'éditer nos pages *PHP*.
- **Wampserver** : plateforme de développement *Web* de type *WAMP*, permettant de faire fonctionner localement des scripts *PHP*.

Les langages utilisés sont : *PHP* et *SQL*.

Le SGBDR que nous utilisons est *Oracle*.

5. Planification

Délai : le projet doit se finir avant le 04/12/2015.

II. Objectif

1. Description brève du logiciel

L'objectif du projet consiste à créer une base de données ou une application basée qui répond aux exigences du travail demandé.

Cette étude doit permettre à l'entreprise de connaître avec précision les apports et conséquences de la migration de sa base de données dans des conditions réelle afin de pouvoir, à terme, valider ou non la mise en place d'une telle migration de ses serveurs.

2. Le but de cette base de données

Notre programme consiste à gérer les employés et les clients selon les critères de cette société.

3. Liste des fonctionnalités

Cette application permettra de réaliser les fonctionnalités suivantes :

- Lister la liste des noms et adresses de tous les clients qui ont effectué au moins une location d'une voiture et d'un véhicule utilitaire.
- Lister la liste des modèles de véhicules n'ayant fait l'objet d'aucune location de la part des entreprises, vous indiquerez également la marque de chaque véhicule.
- Afficher les numéros et les noms des clients ayant effectué au moins une location pour laquelle l'agence de restitution est différente de l'agence où le véhicule a été loué.
- Afficher les agences dans lesquelles au moins un véhicule utilitaire de chacune des marques existantes dans l'entreprise est disponible à la location au moment où la requête est exécutée.
- Afficher les noms des responsables des agences dans lesquelles il est impossible de louer un véhicule de catégorie voiture. Aucune voiture n'est disponible au moment où la requête est exécutée.
- Afficher, magasin par magasin, le nom du magasin, le nom de son responsable, ainsi que le nombre de locations de plus de trois jours effectuées en 2013.
- Trouver le plus grand nombre de locations effectuées par un client et afficher les noms et adresses des clients qui ont effectué ce grand nombre de locations.
- Afficher, agence par agence, le nom de l'agence, le nom de son responsable, ainsi que le nombre de locations de plus de trois jours effectuées en 2015.
- Pour chaque véhicule de moins de 20 000km (au moment où la requête est effectuée), donner la somme totale des montants de toutes les locations effectuées par des entreprises dont il a fait l'objet au cours du mois de juillet 2015.

- Enregistrement d'une location et la restitution d'un véhicule.

4. Contraintes

- La base de données doit être conforme aux normes *SQL*.
- Le code d'interrogation de la base de données doit respecter les règles définies dans les normes *SQL*.

III. Livraisons attendues

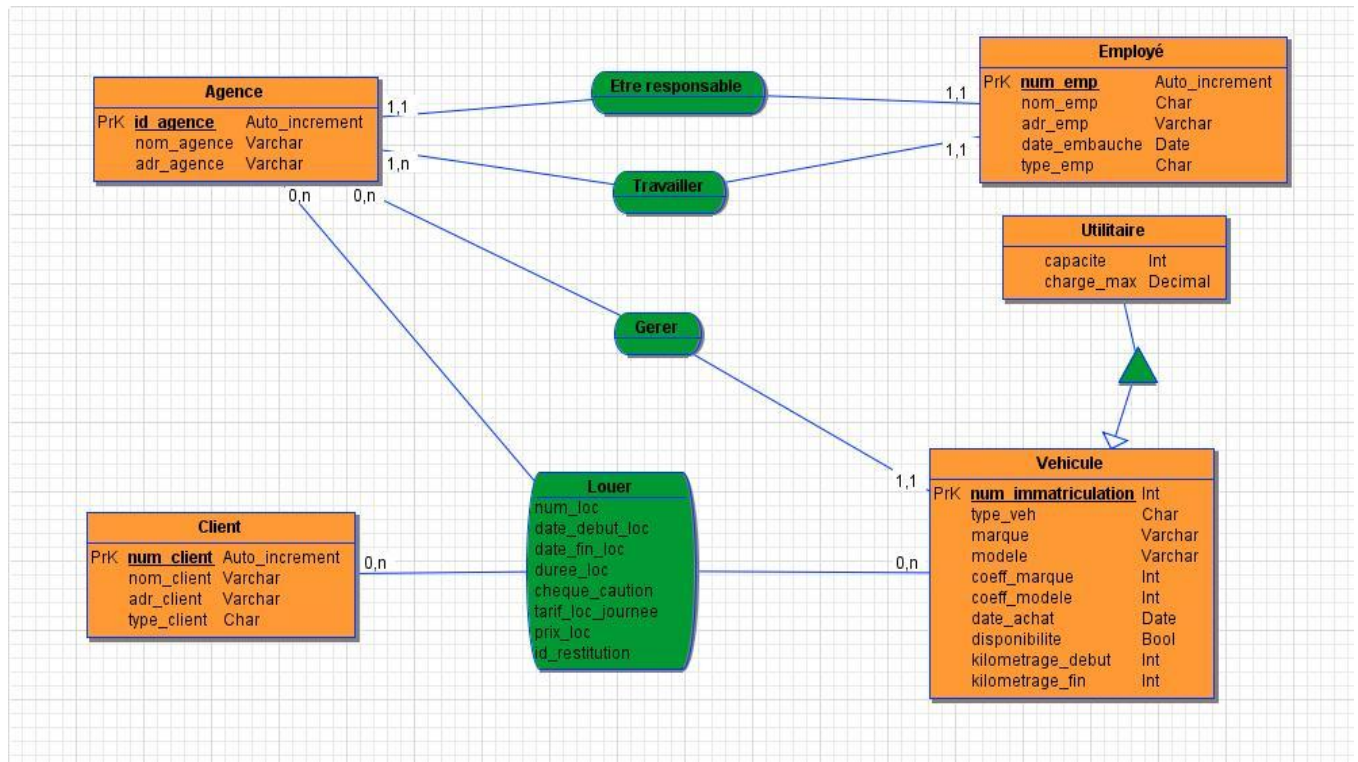
À la fin de notre projet, nous pensons rendre les documents suivants :

- des programmes en langage *SQL* comportant les tables et les requêtes demandées
- des pages *PHP*
- un rapport
- le schéma *MCD*

IV. Conception du projet

1. Schéma MCD

Voici le schéma MCD :



Légende :

- Les entités sont représentées en orange.
- Les relations sont représentées en vert.

2. Schéma MLD

Voici le schéma MLD :

AGENCE (id_agence, nom_agence, adr_agence, #num_emp)

CLIENT (num_cli, nom_cli, adresse_cli, type_cli)

EMPLOYE (num_emp, nom_emp, adresse_emp, date_embauche, type_emp, #id_agence)

VEHICULE (num_immatriculation, type_veh, marque, modele, coeff_marque, coeff_modele, date_achat, disponibilite, kilometrage_debut, kilometrage_fin, #id_agence)

UTILITAIRE (capacite, charge_max, #num_immatriculation)

Louer (num_cli#, num_immatriculation#, id_agence#, num_loc, date_debut_loc,
date_fin_loc, durée_loc, cheque_caution, tarif_loc_journée, prix_loc, id_restitution)

V. Description du schéma conceptuel

1. Description des tables (Entités)

a) table AGENCE

- *Id_agence* : numéro de l'agence
- *nom_agence* : nom de l'agence
- *adr_agence* : adresse de l'agence

b) table CLIENT

- *num_cli* : numéro du client
- *nom_cli* : nom du client
- *adresse_cli* : adresse du client
- *type_cli* : type du client

c) table EMPLOYE

- *num_emp* : numéro de l'employé
- *nom_emp* : nom de l'employé
- *adresse_emp* : adresse de l'employé
- *date_embauche* : date d'embauche de l'employé
- *type_emp* : type de l'employé

d) table VEHICULE

- *num_immatriculation* : numéro d'immatriculation du véhicule
- *type_veh* : type du véhicule
- *marque* : la marque du véhicule
- *modele* : le modèle du véhicule
- *coeff_marque* : coefficient de la marque du véhicule
- *coeff_modele* : coefficient du modèle du véhicule
- *date_achat* : date d'achat du véhicule
- *disponibilite* : état du véhicule actuel (loué, en panne ou en réparation)

- *kilometrage_debut* : début du kilométrage du véhicule
- *kilometrage_fin* : fin du kilométrage du véhicule

e) table UTILITAIRE

- *capacite* : la capacité du véhicule
- *charge_max* : la charge maximale du véhicule

f) table LOUER

- *num_loc* : numéro de location du véhicule
- *date_debut_loc* : la date de début de location du véhicule
- *date_fin_loc* : la date de fin de location du véhicule
- *durée_loc* : la durée de location exprimée en jours
- *cheque_caution* : chèque de caution
- *tarif_loc_journée* : tarif à la journée de location
- *prix_loc* : prix de location du véhicule
- *id_restitution* : agence de restitution

2. Description des relations entre les tables

Etre responsable

Chaque employé de type responsable n'appartient qu'à une seule et une seule agence.

Gérer

Un véhicule est au moins restitué par une seule et une seule agence.

Une agence peut gérer soit aucune ou plusieurs véhicules.

Louer

Un client peut n'avoir loué aucun véhicule, mais peut en louer plusieurs véhicules dans une agence.

Travailler

Un employé travaille dans une et une seule agence.

Une agence regroupe un ou plusieurs employés.

Véhicule-Utilitaire

Un véhicule peut être au moins un utilitaire.

VI. Scripts SQL

1. Création des tables

a) table AGENCE

```
CREATE TABLE Agence(  
    id_agence  NUMBER NOT NULL ,  
    nom_agence VARCHAR2 (25) ,  
    adr_agence VARCHAR2 (40) ,  
    num_emp    NUMBER(10,0) ,  
    CONSTRAINT Agence_Pk PRIMARY KEY (id_agence)  
);
```

b) table EMPLOYE

```
CREATE TABLE Employe(  
    num_emp    NUMBER NOT NULL ,  
    nom_emp    CHAR (30) ,  
    adr_emp    VARCHAR2 (40) ,  
    date_embauche DATE ,  
    type_emp   CHAR (25) ,  
    id_agence  NUMBER(10,0) ,  
    CONSTRAINT Employe_Pk PRIMARY KEY (num_emp)  
);
```

c) table UTILITAIRE

```
CREATE TABLE Utilitaire(  
    capacite   NUMBER(10,0) ,  
    charge_max FLOAT ,  
    num_immatriculation NUMBER(10,0) NOT NULL ,  
    CONSTRAINT Utilitaire_Pk PRIMARY KEY (num_immatriculation)  
);
```

d) table VEHICULE

```
CREATE TABLE Vehicule(  
    num_immatriculation NUMBER(10,0) NOT NULL ,  
    type_veh           CHAR (25) ,  
    marque            VARCHAR2 (25) ,
```

```

    modele          VARCHAR2 (25) ,
    coeff_marque     NUMBER(10,0) ,
    coeff_modele     NUMBER(10,0) ,
    date_achat       DATE ,
    disponibilite     NUMBER (1) ,
    kilometrage_debut NUMBER(10,0) ,
    kilometrage_fin   NUMBER(10,0) ,
    id_agence        NUMBER(10,0) ,
    CONSTRAINT Vehicule_Pk PRIMARY KEY (num_immatriculation) ,
    CONSTRAINT CHK_BOOLEAN_disponibilite CHECK (disponibilite IN (0,1))
);
```

e) table CLIENT

```

CREATE TABLE Client(
    num_client  NUMBER NOT NULL ,
    nom_client  VARCHAR2 (25) ,
    adr_client  VARCHAR2 (40) ,
    type_client CHAR (40) ,
    CONSTRAINT Client_Pk PRIMARY KEY (num_client)
);
```

f) table LOUER

```

CREATE TABLE Louer(
    num_loc      NUMBER NOT NULL ,
    date_debut_loc  DATE ,
    date_fin_loc   DATE ,
    duree_loc     NUMBER(10,0) ,
    cheque_caution  FLOAT ,
    tarif_loc_journee  NUMBER(10,0) ,
    prix_loc      FLOAT ,
    id_restitution  NUMBER(10,0) ,
    num_client      NUMBER(10,0) NOT NULL ,
    num_immatriculation  NUMBER(10,0) NOT NULL ,
    id_agence       NUMBER(10,0) NOT NULL ,
    CONSTRAINT Louer_Pk PRIMARY KEY (num_client,num_immatriculation,id_agence)
);
```

2. Attribution des clés étrangères

```

ALTER TABLE Agence ADD FOREIGN KEY (num_emp) REFERENCES Employe(num_emp);
```

```
ALTER TABLE Employe ADD FOREIGN KEY (id_agence) REFERENCES Agence(id_agence);
ALTER TABLE Utilitaire ADD FOREIGN KEY (num_immatriculation) REFERENCES
Vehicule(num_immatriculation);
ALTER TABLE Vehicule ADD FOREIGN KEY (id_agence) REFERENCES Agence(id_agence);
ALTER TABLE Louer ADD FOREIGN KEY (num_client) REFERENCES Client(num_client);
ALTER TABLE Louer ADD FOREIGN KEY (num_immatriculation) REFERENCES
Vehicule(num_immatriculation);
ALTER TABLE Louer ADD FOREIGN KEY (id_agence) REFERENCES Agence(id_agence);
```

3. Les requêtes

a) requête n°1

Afficher la liste des noms et adresses de tous les clients qui ont effectué au moins une location d'une voiture et d'un véhicule utilitaire

```
Select distinct c.nom_client, c.adr_client
from CLIENT c, LOUER l, LOUER m, VEHICULE v, VEHICULE u
WHERE c.num_client=l.num_client
AND l.num_client=m.num_client
AND l.num_immatriculation=v.num_immatriculation
AND m.num_immatriculation=u.num_immatriculation
AND v.type_veh = 'voiture'
AND u.type_veh = 'utilitaire';
```

NOM_CLIENT	ADR_CLIENT
EZZARI	4 impasse des électriciens
CHEN	5 rue de la Pyramide

b) requête n°2

La liste des modèles de véhicules n'ayant fait l'objet d'aucune location de la part des entreprises, vous indiquerez également la marque de chaque véhicule.

```
Select distinct v.marque,v.modele
from CLIENT c, LOUER l, VEHICULE v
WHERE c.num_client=l.num_client
AND l.num_immatriculation=v.num_immatriculation
AND c.type_client <> 'entreprise'
```

AND l.num_loc is NOT NULL;

MARQUE	MODELE
Citroen	Jumpy
Citroen	Nemo
Nissan	Qashqai

c) requête n°3

Afficher les numéros et les noms des clients ayant effectué au moins une location pour laquelle l'agence de restitution est différente de l'agence où le véhicule a été loué.

```
SELECT c.num_client, c.nom_client
FROM CLIENT c, LOUER l
WHERE c.num_client = l.num_client
AND l.ID_AGENCE != l.ID_RESTITUTION
AND l.ID_RESTITUTION IS NOT NULL;
```

NUM_CLIENT	NOM_CLIENT
5	EZZARI
7	GAMA
8	GASMI

d) requête n°4

Afficher les agences dans lesquelles au moins un véhicule utilitaire de chacune des marques existantes dans l'entreprise est disponible à la location au moment où la requête est exécutée.

```
SELECT DISTINCT a.nom_agence
FROM AGENCE a, VEHICULE v
WHERE a.id_agence=v.id_agence
AND v.TYPE_VEH='utilitaire'
AND v.disponibilite = '1' ;
```

NOM_AGENCE
Cooldrive
Car Go
SAAI

e) requête n°5

Afficher les noms des responsables des agences dans lesquelles il est impossible de louer un véhicule de catégorie voiture.

```
SELECT distinct e.NOM_EMP
FROM EMPLOYE e, AGENCE a, VEHICULE v
WHERE e.NUM_EMP = a.NUM_EMP
AND a.ID_AGENCE = v.ID_AGENCE
AND v.TYPE_VEH = 'voiture'
AND v.DISPONIBILITE='0';
```

NOM_EMP

NGUYEN

f) requête n°6

Trouver le plus grand nombre de locations effectuées par un client et afficher les noms et adresses des clients qui ont effectué ce grand nombre de locations.

```
SELECT c.nom_client , c.adr_client, COUNT(*) as nombre_locations, max(c.NUM_CLIENT)
FROM CLIENT c, LOUER l
WHERE c.num_client = l.num_client
GROUP BY c.nom_client , c.adr_client
HAVING COUNT(*) >= 2;
```

NOM_CLIENT	ADR_CLIENT	NOMBRE_LOCATIONS	MAX(C.NUM_CLIENT)
-----	-----	-----	-----
EZZARI	4 impasse des électriciens	2	5
CHEN	5 rue de la Pyramide	2	1

g) requête n°7

Afficher agence par agence, le nom de l'agence, le nom de son responsable, ainsi que le nombre de locations de plus de trois jours effectuées en 2015.

```
SELECT a.NOM_AGENCE , e.NOM_EMP , COUNT(*) nombre_locations
FROM AGENCE a , EMPLOYE e, LOUER l
WHERE a.ID_AGENCE = e.ID_AGENCE
AND a.ID_AGENCE = l.ID_AGENCE
AND l.DATE_DEBUT_LOC BETWEEN ('01/01/2015') and ('31/12/2015')
```



```
AND I.DATE_FIN_LOC BETWEEN ('01/01/2015') and ('31/12/2015')
AND e.TYPE_EMP='responsable'
AND I.DUREE_LOC > 3
GROUP BY a.NOM_AGENCE, e.NOM_EMP;
```

NOM_AGENCE	NOM_EMP	NOMBRE_LOCATIONS
Livop	RALAIMANANTSOA	1

h) requête n°8

Pour chaque véhicule de moins de 20 000 km (au moment où la requête est effectuée), donner la somme totale des montants de toutes les locations effectuées par des entreprises dont il a fait l'objet au cours du mois de juillet 2015.

```
SELECT SUM(I.PRIX_LOC) montants_locations
FROM VEHICULE v, LOUER l, CLIENT c
WHERE I.NUM_IMMATRICULATION = v.NUM_IMMATRICULATION
AND c.NUM_CLIENT = I.NUM_CLIENT
AND c.TYPE_CLIENT='entreprise'
AND v.KILOMETRAGE_FIN < 20000
AND I.DATE_DEBUT_LOC BETWEEN ('01/07/2015') and ('31/07/2015')
AND I.DATE_FIN_LOC BETWEEN ('01/07/2015') and ('31/07/2015');
```

MONTANTS_LOCATIONS
220

VII. Scripts PHP

1. Quelques illustrations

Requêtes

- 1. la liste des noms et adresses de tous les clients qui ont effectué au moins une location d'une voiture et d'un véhicule utilitaire.
- 2. la liste des modèles de véhicules n'ayant fait l'objet d'aucune location de la part des entreprises, vous indiquerez également la marque de chaque véhicule.
- 3. les numéros et les noms des clients ayant effectué au moins une location pour laquelle l'agence de restitution est différente de l'agence où le véhicule a été loué.
- 4. les agences dans lesquelles au moins un véhicule utilitaire de chacune des marques existantes dans l'entreprise est disponible à la location au moment où la requête est exécutée.
- 5. les noms des responsables des agences dans lesquelles il est impossible de louer un véhicule de catégorie voiture.
- 6. le plus grand nombre de locations effectuées par un client et afficher les noms et adresses des clients qui ont effectué ce grand nombre de locations.
- 7. agence par agence, le nom de l'agence, le nom de son responsable, ainsi que le nombre de locations de plus de trois jours effectuées en 2015.
- 8. Pour chaque véhicule de moins de 20 000 km (au moment où la requête est effectuée), donner la somme totale des montants de toutes les locations effectuées par des entreprises dont il a fait l'objet au cours du mois de juillet 2015.

OK

Connexion ok !

Nom client	Adresse client
EZZARI	4 impasse des électriciens
CHEN	5 rue de la Pyramide

Page qui permet de tester les requêtes demandées au projet.

Locations

Numéro location	Date début	Date fin	durée	Chèque caution	Tarif journée	Prix	Numéro client	Numéro immatriculation	ID agence	ID restitution
1	20/11/10	27/11/10	7	60	30	150	4	555901	4	
2	13/12/12	16/12/12	3	50	30	110	5	555907	8	
3	02/08/12	09/08/12	7	55	30	120	2	555905	3	
4	11/08/14	18/08/14	7	65	30	120	3	555906	4	
5	20/07/15	27/07/15	7	80	30	120	5	555906	5	7
6	14/07/15	15/07/15	1	50	30	100	7	555909	6	5
7	20/10/15	23/10/15	3	70	30	110	1	555903	7	
8	14/11/15	17/11/15	3	60	30	110	1	555910	8	
41	04/12/15	04/12/15	4	55	30	150	8	555912	10	2

Enregistrez une nouvelle location

Restituez une location

Page (**locations.php**) qui affiche toutes les locations existantes et offre deux options : soit d'enregistrer une nouvelle location ou de la restituer via des liens *HTML*.

Accueil
Tables
Requêtes
Locations

Locations

Agence Car Go
Client COLL
Véhicule Dacia - voiture - Sandero
Durée de location 2

Chèque caution à payer 50
Tarif journée 30
Prix location 150

Validez

1
2
3
4
5
6
7

Cette image montre comment on enregistre une nouvelle location d'un véhicule. Ici, il suffit simplement de saisir tous les champs avant de pouvoir valider sinon cette page affichera une erreur.

Locations

Numéro location	Date début	Date fin	durée	Chèque caution	Tarif journée	Prix	Numéro client	Numéro immatriculation	ID agence	ID restitution
1	20/11/10	27/11/10	7	60	30	150	4	555901	4	
2	13/12/12	16/12/12	3	50	30	110	5	555907	8	
3	02/08/12	09/08/12	7	55	30	120	2	555905	3	
4	11/08/14	18/08/14	7	65	30	120	3	555906	4	
5	20/07/15	27/07/15	7	80	30	120	5	555906	5	7
6	14/07/15	15/07/15	1	50	30	100	7	555909	6	5
7	20/10/15	23/10/15	3	70	30	110	1	555903	7	
8	14/11/15	17/11/15	3	60	30	110	1	555910	8	
41	04/12/15	04/12/15	4	55	30	150	8	555912	10	2

Numéro de location ? Agence de restitution

Quel véhicule ?

Cette image montre comment on restitue un véhicule à une agence. Ici, on saisit le numéro de location qui existe, ensuite, on choisit le nom de d'agence de retour, ainsi que le nom du véhicule concerné.

2. Codes

```

(isset($_GET['agence']) AND isset($_GET['client']) AND isset($_GET['vehicule']) AND isset($_GET['duree']) AND isset($_GET['tarif']))

include 'utils.inc.php';
$dbdd = new PDO(HOST, USER, MDP) or die();
$today=$_GET['date'];
$duree=$_GET['duree'];
$caution=$_GET['caution'];
$tarif=$_GET['tarif'];
$prix=$_GET['prix'];
$client=$_GET['client'];
$vehicule=$_GET['vehicule'];
$agence=$_GET['agence'];

/*Insère les nouvelles données pour la nouvelle location, soit une nouvelle ligne*/
$req = $dbdd->prepare("INSERT INTO LOUER (DATE_DEBUT_LOC,DUREE_LOC,CHEQUE_CAUTION,TARIF_LOC_JOURNEE,PRIX_LOC,NUM_CLIENT, NUM_IMM
(CURRENT_DATE,$duree,$caution,$tarif,$prix,$client,$vehicule,$agence)");
// $req->execute(array($_POST['id']));
/* $req->bindParam(1,$today);
$req->bindParam(2, $_GET['duree']);
$req->bindParam(3, $_GET['caution']);
$req->bindParam(4, $_GET['tarif']); */

$req->execute();
$req->setFetchMode(PDO::FETCH_ASSOC);

/*signale que le véhicule n'est plus disponible*/
$req = $dbdd->prepare("UPDATE VEHICULE SET DISPONIBILITE='0' WHERE NUM_IMMATRICULATION=$vehicule");
$req->execute();
$req->setFetchMode(PDO::FETCH_ASSOC);

```

Code qui permet de réaliser une nouvelle location d'un véhicule.

```
(isset($_GET['agence']) AND isset($_GET['vehicule']) AND isset($_GET['numero'])) {  
  
    /*$bdd = new PDO(HOST, USER, MDP) or die();*/  
    $today='CURRENT_DATE';  
    $num=$_GET['numero'];  
    $vehicule=$_GET['vehicule'];  
    $ag=$_GET['agence'];  
  
    /*Met à jour les données de l'allocation*/  
    $requete = $bdd->prepare("UPDATE LOUER SET DATE_FIN_LOC=CURRENT_DATE, ID_RESTITUTION=$ag WHERE NUM_IMMATRICULATION=$vehicule AND NUM_");  
    $requete->execute();  
    $requete->setFetchMode(PDO::FETCH_ASSOC);  
  
    /*signale que le véhicule est disponible*/  
    $requete = $bdd->prepare("UPDATE VEHICULE SET DISPONIBILITE='1' WHERE NUM_IMMATRICULATION=$vehicule");  
    $requete->execute();  
    $requete->setFetchMode(PDO::FETCH_ASSOC);  
  
    echo "<font color='red'>location restituée !</font>";  
}
```

Code qui permet de réaliser une restitution d'un véhicule à une agence.

VIII. Remarques sur le déroulement du projet

1. Méthode de travail au sein du groupe

Après avoir choisi et analysé le sujet, nous avons rédigé le cahier des charges.

Ensuite, nous nous réunissions pour mettre en commun nos idées, puis nous nous répartissons les tâches pour effectuer chacun une partie du travail.

Puis, nous nous réunissions afin d'effectuer des tests de notre projet et de corriger les erreurs.

Nos travaux étaient enregistrés sur nos ordinateurs personnels pour éviter toute perte de données.

2. Difficultés rencontrées

- l'exécution des requêtes.
- la connexion de la base de données en PDO.
- la programmation en PHP.

IX. Conclusion

Par conséquent, nous avons donc créé notre programme construit avec des fonctionnalités grâce au logiciel SQL Developer.

Pour réaliser notre projet, nous nous sommes fixés des objectifs de programmation. Notre produit doit être simple d'utilisation, les résultats du calcul devaient être clairs et compréhensibles, le programme doit utiliser des méthodes de SQL Developer et le code doit être très commenté pour permettre à l'utilisateur de comprendre le code malgré quelques difficultés rencontrées.

La réalisation de ce projet nous a appris ce qu'est le travail en groupe. Entre autre, nous avons appris à répartir les tâches entre deux personnes, sécuriser les données pour éviter toute perte et mettre en commun le code créé individuellement.

De plus, au niveau informatique nous avons découvert et maîtriser de nouvelles techniques de programmation et perfectionné notre programmation SQL.

Ceci nous aura été bénéfique pour des projets futurs que ce soit dans les études ou dans la vie professionnelle.

Si nous devons refaire ce projet, nous déciderons d'optimiser cette fois la simulation des comportements des souris avec plus de fonctionnalités selon nos propres idées.