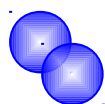


*Département Informatique*  
*initiation aux système informatique*

## **Mini Projet**



# Gestion de location de voiture

Préparé par :

- **Oularbi soufiane**

Encadré par :

- **El merabety yassine**

**Prof.EL BRAK**

1 ère année cycle ingénieur  
Génie électrique EEA

# **SOMMAIRE**

## **1) PRESENTATION :**

## **2) ANALYSE DU PROBLEME :**

## **3) PARTIE CONCEPTION:**

3-1) introduction

3-2) DICTIONNAIRE DE DONNEE :

3-3) REGLES DE GESTION :

3-4) dépendance fonctionnelle

3-5) Le Modèle conceptuel des données (MCD)

3-6) Le Modèle logique des données (MLD)

## **4) PARTIE REALISATION**

4-1) réalisation de la base de donnée sous accès:

4-2) formulaires

4-3) requêtes

4-4) macros

## **5) CONCLUSION :**

## **6) ANNEXES:**

Interface sous visuel basic

## **7) BIBLIOGRAPHIE**

## **1)Presentation**

La location des voitures est devenue un secteur en pleine expansion, dont la compétitivité augmente jours après jours, donc pour améliorer ces services une agence de location de voiture décide d'informatiser son système de gestion a travers la réalisation d'une base de données regroupant toutes les informations concernant ses clients, ses contrats, ces voitures, afin de faciliter la tâche aussi bien pour les clients que pour son personnel.

## **2) ANALYSE DU PROBLEME :**

La gestion traditionnelle d'une agence de location de voiture nécessite un grand effort de la part du personnel, il faut utiliser des formulaires en papier pour les clients, pour les contrats, des archives pour stocker toutes les anciennes opérations ou transactions, ce qui rend le travail désagréable, la nouvelle technologie informatique présente une solution à ses problèmes avec l'utilisation de la base de données relationnelle, cette solution consiste à automatiser et informatiser la gestion à travers l'utilisation de différents logiciels, le logiciel que l'on va utiliser dans cette application est Microsoft Access, ce logiciel permet de créer des tables, pour stocker les informations souhaitées, des formulaires pour saisir les informations facilement, des requêtes pour chercher, modifier des données et des macros pour automatiser certaines tâches.

### **3) PARTIE CONCEPTION:**

#### **3-1) Introduction:**

La conception de cette base de donnée s'appuiera sur la méthode merise :  
D'abord on va définir notre dictionnaire de donnée (DD), ensuite on va énoncé les règle de gestion (RG), après on élaborera les dépendance fonctionnelles (DF) qui nous aidera a élaborer notre modèle conceptuelle de données (MCD) et enfin sortir par notre modèle logique de donnée (MLD) qui sera implémenter dans la base de donnée.

#### **3-2) dictionnaire de donnée :**

**Client** : N°client ; Nom; Prénom ; code postal ; rue ; ville; Tel; numéro de permis; date de permis.

**Contrat**: N° contrat; type; date de départ ; date de retour.

**Facture**: N° facture ; date de facture; compteur KM; montant ; montant a payé (après une réduction)

**Parking**: N° parking ; capacitée ; rue ; arrondissement.

**Voiture** : Immatriculation ; marque ; carburant ; capacité habitacle ;

### **3-3) Règle de gestion :**

- 1- un client peut faire plusieurs contrats.
- 2- Un contrat ne concerne qu'un seul client
- 3- Un client peut avoir plusieurs factures.
- 4- Une facture ne concerne qu'un seul client.
- 5- Un client peut louer plusieurs voitures.
- 6- Une voiture ne peut être loué que par un seul client.
- 7- Une voiture peut être dans un parking
- 8- Un parking peut contenir plusieurs véhicules.

### **3-4) les dépendances fonctionnelles :**

<u>N°client</u> →	nom _____
→	Prénom
→	Code postal
→	Rue
→	Ville
→	Tel
→	Numéro de permis
→	Date de permis.

<u>N° contrat</u>	Type _____
→	Date de départ
→	Date de retour
→	N° client
→	Immatriculation

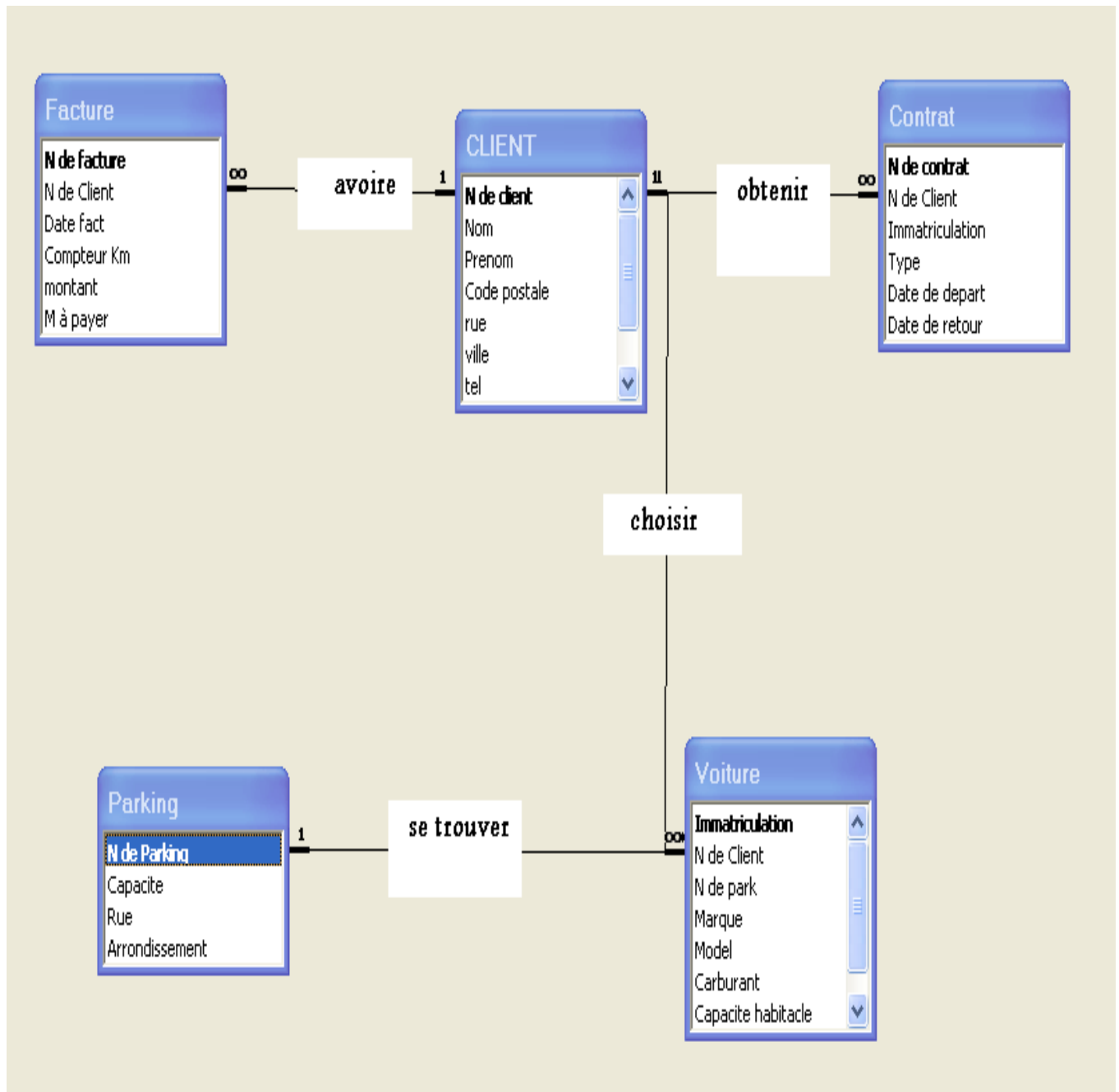
.

<u>N° facture</u>	Date de facture
→	Compteur KM
→	N° client
→	Montant
→	Montant à payer

<u>Immatriculation</u>	marque
→	Modèle
→	N° client
→	Capacité habitacle
→	Compteur KM
→	N° parking
→	Carburant

<u>N° parking</u>	capacité
→	Rue
→	Arrondissement

**3-5) le modèle conceptuel des données (MCD):**



### **3-6) modèle logique donnée:**

Le modèle logique de donnée:

**Client** (N°client ; Nom; Prénom ; code postal ; rue ; ville; Tel; numéro de permis; date de permis).

**Contrat**:( N° contrat; type; date de départ ; date de retour; #N°client; #immatriculation).

**Facture**: (N° facture ; date de facture; compteur KM; montant ; montant a payé (après une réduction);#N°client).

**Parking** :( N° parking; capacité ; rue ; arrondissement).

**Voiture** : (Immatriculation ; marque ;modele; carburant ; capacité habitacle ; #N°client; #N°parking ).

Le modele logique donnée est normaliser car;

-toutes les relations possèdent une clé primaire.

-les attribues de chaque relation dépendent de son clé.

-et rien que de cette clé.

## **4)Partie réalisation:**

### **4-1) réalisation de la base de donnée sous accès:**

On crée les relation obtenue à partir modèle logique de donnée sous forme de table sous Microsoft accès :

On a 5 tables:

Client ; contrat, facture, parking et voiture.

Après création des tables on établie les relation entre les différent tables suivant le modèle conceptuel de donnée et en appliquant l'intégrité référentielle qui nous permet d'établir les cardinalités convenable entre les différentes relations.



#### 4-2) les formulaires :

pour saisir les données facilement on va créer des formulaires pour chaque relation ces formulaires nous permettront de saisir toutes les données relatives soit au client, au contrat, au véhicule ou à la facture et aussi nous permettront de faire quelques calculs comme on peut voir dans cet exemple:

#### ***exemple de formulaire facture:***

The screenshot shows a web form titled 'Facture'. It contains the following fields and buttons:

- N de facture:
- N de Client:
- Date fact:
- Compteur Km:
- montant:
- M à payer:
- Buttons: 'montant à payer' and 'montant sans réduction'

At the bottom, there is a navigation bar with the text 'Enr : 1 sur 5' and navigation icons.

pour remplir le formulaire facture, on saisit les données N°de facture, N°client et date de facture et compteur KM ; pour avoir le montant il suffit de cliquer sur le bouton 'montant sans réduction' qui sera calculé automatiquement, on peut aussi faire une réduction d'un pourcentage défini par l'utilisateur en cliquant sur le bouton 'montant à payer' ce qui nous donnera le M à payer.

#### 4-3) les requêtes:

##### **\* Requête facture client:**

Afficher le numéro de client, son nom, son numéro et date de facture.

**\*Requête calcule montant :**

Calcule le montant à payer pour un client dont le numéro est saisi par l'utilisateur en faisant l'opération suivante:

Montant= 300\*([date de retour]-[date de départ]) , avec 300dH par jour est le coût de location par jour.

**\* Requête montant à payer:**

Calcule le montant a payer après un taux de réduction entré par l'utilisateur en effectuant l'opération suivante :

Montant a payer= [montant]-([montant]\*[donner la réduction:]/100)

4-4) les macros:

Les macros sont des petits programmes qui permettent de faciliter et d'automatiser quelque opération ; ici on a utilisé les macros suivantes:

Nouveau client

Nouvelle facturation

Nouvelle contrat

Nouvelle voiture

Ces macros permettent d'ouvrir les formulaires sur un nouvel enregistrement pour saisir des nouvelles données.

La macros calcule montant: permet de faire l'opération du calcule de montant dans le formulaire facture.

La macro montant à payer : permet de faire l'opération du calcule du montant réduit dans le formulaire facture.

**5)conclusion:**

Malgré que ça nécessite encore des amélioration ; Ce mini- projet nous a permet de savoir utiliser nos connaissances acquissent dans le module d'initiation aux système d'information dans des application réel est concrète et aussi solidifier et complété ses connaissances que nous auront besoin dans d'autre module et aussi sur terrain.

## 6) Annexe:

Utilisation d'une interface visual basic



Cette interface est muni de 4 boutons commande (**Ouvrir formulaire client, ajouter client, rechercher client et facturation**) qui gère quelque fonctionnalités de notre base de donn e.

On n'a pas eu la chance de faire fonctionner les 4 boutons, car on n'a pas pu se connecter   notre base de donn es pour modifier ou pour ajouter des donn es mais on a r eussi   imprimer une facture et ouvrir un formulaire client.

Form3

*Ajout d'un nouveau client*

Adodc1

*num clien*

*nom*

*prenom*

*code*

*rue*

*ville*

*TEI*

*num permis*

*date de permis*

ajouter

Dans cette interface client, on a voulu ajouter un client par saisir toutes les information qui lui concerne et puis appuyer sur le bouton ajouter pour enregistrer ces information dans la table client de notre base de données, ainsi on a utiliser le code suivant pour le bouton ajouter :

```
Private Sub Command1_Click ()  
Adodc1.Recordset.AddNew  
End Sub
```



formulaire client

**Fclient**

num client: 2

code: 90000

TEI: 06 13 78 18 99

num permis: 487667

date de permis: 13/06/2009

ville: Tanger

nom: Oularbi

prenom: Soufiane

rue: Geneve

suivant

On a conçu cette interface pour afficher les clients qui existent dans notre base de données, et cela en s'appuyant sur le contrôle MSAdodc qui est nécessaire à la liaison de la Form à notre base de données.

## **7) bibliographie :**

Cours d'initiation aux systèmes d'information par Pr.EL BRAK

TP d'Access et de visual basic par Pr. EL BRAK

-www.google.com

-[http://www.supinfo-projects.com/fr/2004/base\\_donnees\\_vb/3/](http://www.supinfo-projects.com/fr/2004/base_donnees_vb/3/)

