



Projet PEVEN

GESTION D'UNE AGENCE DE LOCATION DE VOITURES

Bliidi Abderrahmane
Chouria Mohamed amine
Medyouni Saber

ISG TUNIS | 1LNIG8

Sommaire

I.	Étude de l'existence	3
II.	SOLUTION	4
III.	Réalisation.....	5
1.	La bienvenue.....	5
2.	L'inscription ou l'authentification	6
3.	Le choix.....	9
4.	La consultation et réservation d'une véhicule	10
5.	L'annulation d'une réservation du véhicule	14

Tableau des figures

Figure 1:l'interface de la bienvenue	5
Figure 2 : Les contrôles de saisie de l'inscription et la connexion	7
Figure 3 : fonction isValid	8
Figure 4 : fonction Checkcin.....	8
Figure 5 : existInCinFile	8
Figure 6: L'interface de l'inscription ou l'authentification	9
Figure 7 : L'interface de choix	10
Figure 8 : Les maison des voitures.....	10
Figure 9 : marque et prix du jour.....	11
Figure 10 : les conditions	11
Figure 11: la fonction addMarque.....	12
Figure 12 : code de contrôle de saisie des dates	12
Figure 13 : L'interface de la consultation et réservation d'une véhicule	13
Figure 14 : L'interface l'annulation d'une réservation d'un véhicule.....	15

I. Étude de l'existence

Ce projet Gestion d'une agence de location de voitures, a pour objectif l'implémentation d'une application simple basée sur l'interfaçage graphique, et permettant d'aider à la gestion de la location de voitures de l'agence « IKRI CAR ».

L'application, qui fera suite à cette implémentation, doit contenir une première interface d'accueil permettant l'accès à l'espace virtuel de l'agence. Ensuite, l'application devrait orienter l'utilisateur vers un ensemble d'interfaces qui lui sont appropriés pour faire les opérations suivantes :

- S'inscrire s'il s'agit d'un nouveau client ou bien de s'authentifier s'il s'agit d'un client déjà inscrit
- Consulter les voitures disponibles à l'agence
- Réserver une voiture
- Annuler une réservation

L'interface doit à chaque fois indiquer si l'opération a été effectuée avec succès ou pas.

II. SOLUTION

Il s'agit de développer une application Windows composé par plusieurs Forms. Ils représentent :

- La bienvenue
- L'inscription ou l'authentification
- Le choix de consulter et réserver ou annuler une réservation
- La consultation et réservation d'une véhicule
- L'annulation d'une réservation d'un véhicule

Pour rendre ces interfaces utilisables et pour implémenter cette application il s'agit de développer le code en vb.net qui permet d'assurer des contrôles de saisie et de rendre l'application dynamique.

III. Réalisation

L'application est formée par 5 Forms qu'on va les expliquer un par un :

1. La bienvenue

La page de la bienvenue est très simple. Elle composé par un fond qu'on a choisi avec le logo de notre agence « IKRI CAR » qu'on a conçu. Et par deux boutons :

Accédez à l'espace virtuel qu'il va nous emmènera vers l'interface de l'inscription et l'authentification par la commande « Form.Show() » et cacher la Form actuelle par la commande « Me. Hide () »

Et Quitter qui va fermer l'application en utilisant la commande Application. Exit ()



Figure 1: l'interface de la bienvenue

2. L'inscription ou l'authentification

Cette page permet l'utilisateur de se connecter s'il a un compte existant ou de s'inscrire sinon.

L'interface est un mélange entre des boutons, des zones de textes et des labelles qui rendent ces zones de textes compréhensible.

Deux boutons annuler permet de supprimer le contenu des ensembles des text boxes connexion et authentification en utilisant l'événement « Button. Click ». L'ors de l'erreur un message box s'affiche.

Deux autres boutons existent :

- Inscription et Connexion : en cliquant à ce bouton un contrôle de saisie vérifie des contraintes sur les zones de textes (voir figure 2) :
 - Tous les champs ne doivent pas être vide
 - Le nom et le prénom ne doivent pas être composé seulement le caractère espace et doivent être composé par des lettres majuscules et minuscules et des espaces. La condition c'est si la variable égale vide ou « Not(isValid(variable)) ». On a déclaré la fonction isValid avec une valeur de retour booléenne vrai si les contraintes qu'on a mentionnées sont respectées et false sinon (voir figure 3).
 - Le numéro de CIN doit être composé par 8 lettres et la première lettre doit être 0 ou 1. On a utilisé une fonction Checkcin (voir figure 4).

- Le numéro de CIN ne doit pas être déjà existant dans le fichier des CIN. On a utilisé la fonction existInCinFile (voir figure 5).
- La longueur du mot de passe ne doit pas être inférieure à 8
- Le numéro de permis doit être composé par 8 lettres et la première lettre doit être 0 ou 1. On a utilisé la même fonction Checkcin de contrôle de CIN (voir figure 4).

Si toutes les contraintes sont respectées on va remplir chaque donné dans le fichier qui lui correspondant et le programme va nous emmenera vers l'interface choix.

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    Dim nom = TextBox3.Text
    Dim pre = TextBox4.Text
    Dim cin = TextBox5.Text
    Dim mdp = TextBox6.Text
    Dim num = TextBox7.Text
    Dim per = TextBox8.Text
    If (nom = "" Or Not (IsValid(nom))) Then
        MsgBox("Verifier le nom")
    ElseIf (pre = "" Or Not (IsValid(pre))) Then
        MsgBox("Verifier le prenom")
    ElseIf (Not (Checkcin(cin))) Then
        MsgBox("Verifier le CIN")
    ElseIf (existInCinFile(cin)) Then
        MsgBox("CIN deja inscrit !")
    ElseIf (mdp.Length < 8) Then
        MsgBox("Le mot de passe doit contenir au moins 8 caractères")
    ElseIf (Not (Checknum(num))) Then
        MsgBox("Verifier le numero")
    ElseIf (Not (Checkcin(per))) Then
        MsgBox("Verifier le numéro de permis")
    End If
End Sub
```

Figure 2 : Les contrôles de saisie de l'inscription et la connexion


```

Function isValid(ByVal str As String) As Boolean
    Dim i As Integer
    Dim c As Char
    For i = 0 To (str.Length) - 1
        c = str(i)
        If (Not (c >= "A" And c <= "Z") And Not (c >= "a" And c < "z") And c <> " ") Then
            Return False
        End If
    Next
    Return True
End Function

```

Figure 3 : fonction isValid

```

Function Checkcin(ByVal ch As String) As Boolean
    If (ch.Length = 8 And IsNumeric(ch) And ch.IndexOf(".") = -1 And (ch(0) = "0" Or ch(0) = "1")) Then
        Return True
    End If
    Return False
End Function

```

Figure 4 : fonction Checkcin

```

Function existInCinFile(ByVal ch As String) As Boolean
    Dim fs As New FileStream("cin.txt", FileMode.Open, FileAccess.Read)
    Dim sr As New StreamReader(fs)
    Dim exist As Boolean = False
    Dim chtest As String
    While (sr.Peek() > -1)
        chtest = sr.ReadLine()
        If (chtest = ch) Then
            exist = True
        End If
    End While
    sr.Close()
    fs.Close()
    Return exist

```

Figure 5 : existInCinFile

- Connexion : en cliquant à ce bouton si le CIN existe dans le fichier des CIN et le mot de passe saisi est égale au mot de passe qui correspond au CIN, le programme va nous emmenera vers l'interface choix. Il existe une case à cocher qui permet d'afficher le mot de passe caché.

Inscription ou authentification

IKRI CAR

S'inscrire ou s'authentifier

CIN

Mot de passe ☐ Afficher

Connexion **Annuler**

Nom

Prénom

CIN

Mot de passe *8 Caractères

Numéro de téléphone

Numéro de permis

Inscription et Connexion **Annuler**

Retour **Quitter**

Figure 6: L'interface de l'inscription ou l'authentification

3. Le choix

Cette page donne à l'utilisateur le choix de consulter et réserver des voitures ou d'annuler une réservation. On trouve un labelle contenant le CIN de l'utilisateur et deux boutons :

- Consulter et Réserver : nous emmène vers l'interface de La consultation et réservation d'une véhicule.
- Annuler une réservation : nous emmène vers l'interface de L'annulation d'une réservation.



Figure 7 : L'interface de choix

4. La consultation et réservation d'une véhicule

L'interface de la consultation et réservation d'une véhicule est composé par deux « combo boxes » :

- Combo box des maisons des voitures formé par 6 maisons (Citroën, Kia, Peugeot, Renault, Toyota, Volkswagen) comme il représenter dans la figure 8.

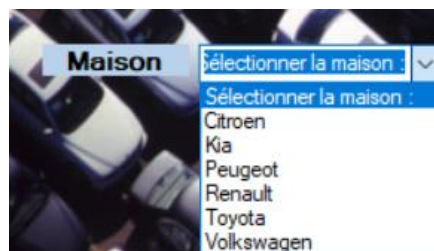


Figure 8 : Les maison des voitures

- Combo box des marques se remplit selon la marque à partir de 6 fichiers remplis à l'avance contenant les marques d'une maison et son prix de jour (voir figure 9).

Quand l'utilisateur choisit la voiture à louer le prix du jour s'affiche dans un labelle (voir figure 9).



Figure 9 : marque et prix du jour

Ces instructions se font avec un code (voir la figure 10). Selon l'index sélectionné le programme va ouvrir le fichier correspondant à la maison choisie et fait appel à la fonction addMarque. Cette fonction parcourt le fichier jusqu'à la fin. Chaque ligne est de la forme « marque/prix » le combo box va prendre seulement la marque (voir la figure 11).

```
Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ComboBox1.SelectedIndexChanged
    Dim pos = ComboBox1.SelectedIndex
    Select Case pos
        Case 1
            Dim fs As New FileStream("citroen.txt", FileMode.Open, FileAccess.Read)
            Dim sr As New StreamReader(fs)
            addMarque(fs, sr)
            sr.Close()
            fs.Close()
        Case 2
            Dim fs As New FileStream("kia.txt", FileMode.Open, FileAccess.Read)
            Dim sr As New StreamReader(fs)
            addMarque(fs, sr)
            sr.Close()
            fs.Close()
        Case 3
            Dim fs As New FileStream("peugeot.txt", FileMode.Open, FileAccess.Read)
            Dim sr As New StreamReader(fs)
            addMarque(fs, sr)
            sr.Close()
            fs.Close()
        Case 4
            Dim fs As New FileStream("renault.txt", FileMode.Open, FileAccess.Read)
            Dim sr As New StreamReader(fs)
            addMarque(fs, sr)
        Case 5
            Dim fs As New FileStream("toyota.txt", FileMode.Open, FileAccess.Read)
            Dim sr As New StreamReader(fs)
            addMarque(fs, sr)
            sr.Close()
            fs.Close()
        Case 6
            Dim fs As New FileStream("volkswagen.txt", FileMode.Open, FileAccess.Read)
            Dim sr As New StreamReader(fs)
            addMarque(fs, sr)
            sr.Close()
            fs.Close()
    End Select
End Sub
```

Figure 10 : les conditions

```

Private Sub addMarque(ByVal fs As FileStream, ByVal sr As StreamReader)
    Dim ch As String
    Dim marque As String
    Dim i As Integer
    ComboBox2.Items.Clear()
    While (sr.Peek() > -1)
        ch = sr.ReadLine
        i = 0
        marque = ""
        While (ch(i) <> "/")
            marque = marque + ch(i)
            i = i + 1
        End While
        ComboBox2.Items.Add(marque)
        ComboBox2.SelectedIndex = 0
    End While
End Sub

```

Figure 11: la fonction addMarque

Dans l'interface aussi on trouve deux calendriers ils prennent la date de sortie et la date de retour sous un contrôle de saisie qui affiche un message d'erreur lorsque l'utilisateur choisit une date de sortie au passé ou une date de retour inférieur ou égale à la date de sortie (voir le code de contrôle de saisie dans la figure 12).

```

Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click
    Dim ds = MonthCalendar1.SelectionRange.Start.ToString("yyyy/MM/dd")
    Dim dr = MonthCalendar2.SelectionRange.Start.ToString("yyyy/MM/dd")
    Dim dt = Today.ToString("yyyy/MM/dd")
    Dim maison = ComboBox1.SelectedItem
    Dim marque = ComboBox2.SelectedItem

    If (ComboBox1.SelectedIndex <= 0) Then
        MsgBox("Selectionner la maison")
    ElseIf (ComboBox2.SelectedIndex < 0) Then
        MsgBox("Selectionner la marque")
    ElseIf (ds < dt) Then
        MsgBox("Vous ne pouvez pas choisir une date de sortie au passé")
    ElseIf (dr <= ds) Then
        MsgBox("La durée de location doit etre 1 jour au moins")
    End If
End Sub

```

Figure 12 : code de contrôle de saisie des dates

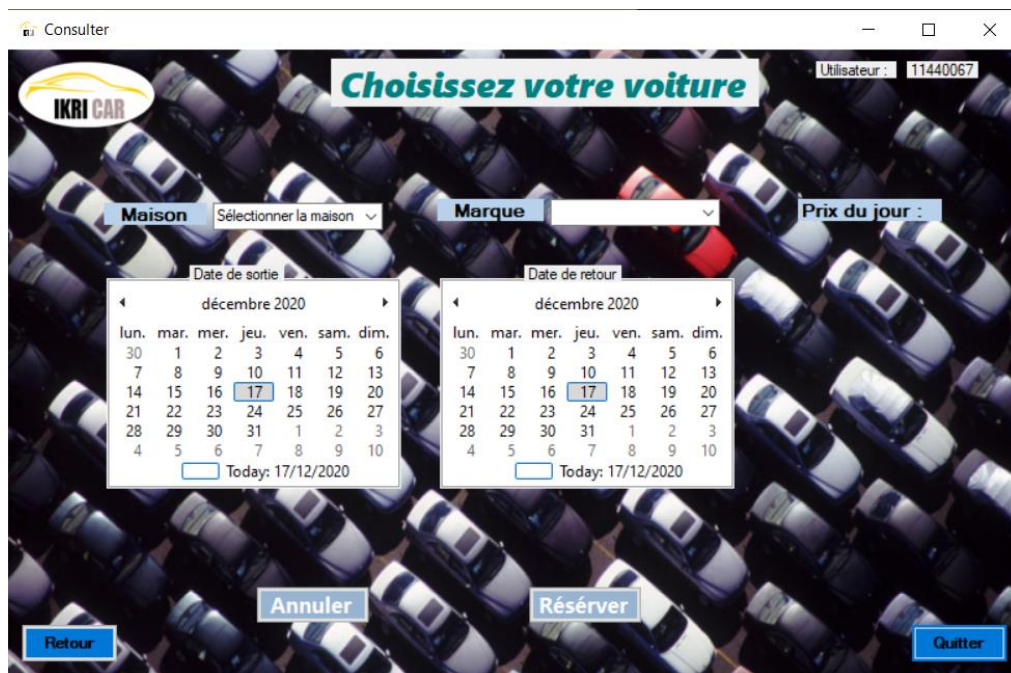


Figure 13 : L'interface de la consultation et réservation d'une véhicule

Il y a aussi 4 boutons :

- Annuler permet de vider tous les champs
- Réserver permet de remplir les informations de la réservation dans un fichier « reservations.txt », ajouter de maison marque et prix/jour dans le fichier "voitures reserves.txt" qui contient les voitures réservées, supprimer la marque du fichier de la maison correspondante et vider tous les champs en appelant la fonction bouton. Performclick () (le bouton est : annuler) si les informations de la réservation ne sont pas les même dans une autre réservation.
- Retour permet de retourner à l'interface de choix.
- Quitter permet de quitter le programme

5. L'annulation d'une réservation du véhicule

L'interface est formée par un List box qu'elle va se remplir à partir du labelle qui contient le numéro de CIN et le fichier « reservations.txt ». Les réservations sont écrites sous la forme « Voiture * Date de Sortie (AAAA/MM/JJ) * Date de Retour (AAAA/MM/JJ) * Prix de réservation ».

Si vous cliquer sur annuler sans la sélection d'une réservation un message d'erreur va s'afficher sinon le programme va supprimer la ligne de réservation de fichier « reservations.txt » et mettre à jour la List box puis il va déplacer la voiture correspondante du fichier « voitures reserves.txt » sans le nom de la maison vers le fichier de maison correspondante.

Il y a aussi deux autres boutons :

- Retour permet de retourner à l'interface de choix.
- Quitter permet de quitter le programme



Figure 1.4 : L'interface l'annulation d'une réservation d'un véhicule