

**UNIVERSITÉ DES MASCAREIGNES
FACULTÉ DES TECHNOLOGIES
INFORMATIQUES ET COMMUNICATION
DÉPARTEMENT D'INFORMATIQUE APPLIQUÉE**

RAPPORT PROJET GERER TAXIS

PAR : Sydney Mungly

PREMIÈRE ANNÉE

TRAVAIL REMIT A : **Mons. Toofanee**

AVRIL 2018

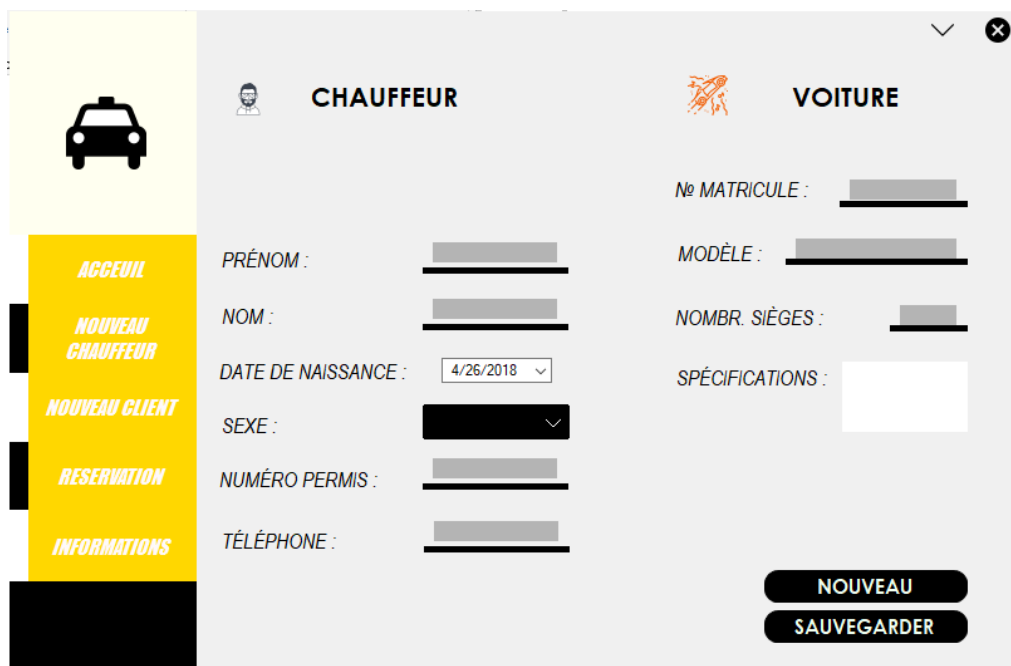
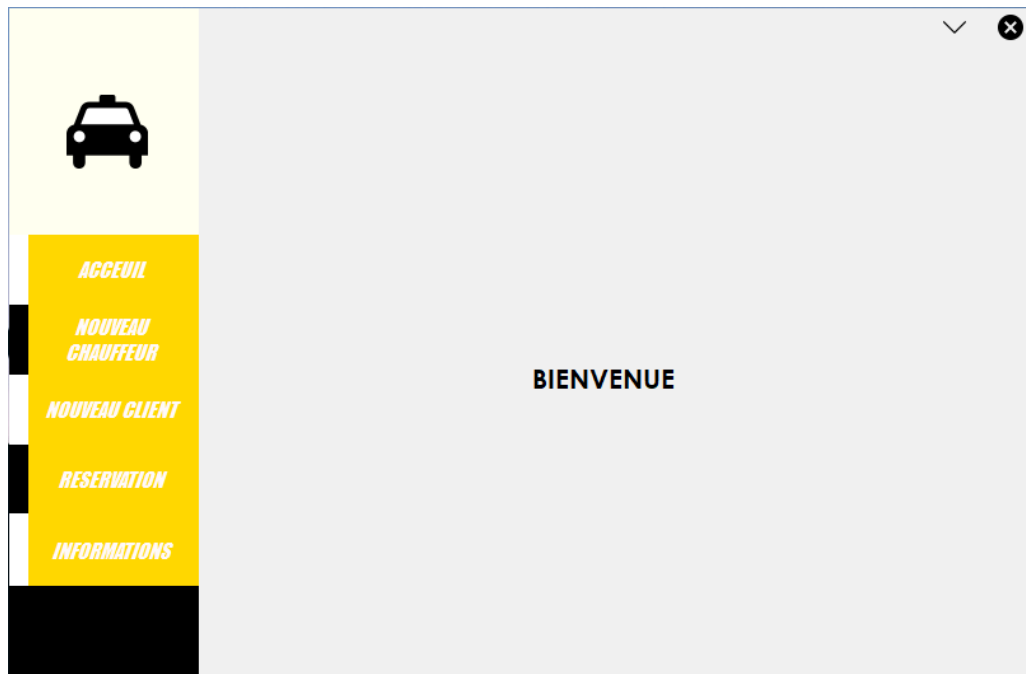
TABLE DES MATIERES


Préface	1
Table Des Matières	2
Introduction	3
Version 1.0 (Aperçu)	4-5
Version 1.0 (Problématiques).....	6
Version 2.0: Améliorations et Procédées	7-9
Source des photos et icones.....	10-11
Ajout des données (Oledb).....	12
La Connexion entre la BDD et le logiciel	13
Ajout des données (Client, Voiture, Chauffeur, Réservation)	14-17
Conception du design.....	18
Conception (Codage)	19
Problématiques de la Version 2.0.....	20
Version 3.0 (Aperçu)	21-23
Les Améliorations.....	24
Les Validations	25-26
Comment on a fait l'interface.....	27-29
Problématiques de la Version 3.0.....	30
Version 4.0.....	31-33

Introduction


Ce rapport parle du projet logiciel pour gérer les taxis créer en C# sur Visual Studio 2015.

Aperçu du logiciel (Version 1.0)





NOUVEAU CLIENT



PRENOM :


NOM :

N° TELEPHONE :


EMAIL :

ADRESSE L1:


ADRESSE L2:





NOUVEAU **SAVEGARDER**



FAIRE UNE RÉSERVATION







ID RÉSERVATION :

ID CHAUFFEUR :

ID CLIENT :

DE :

↓

: À

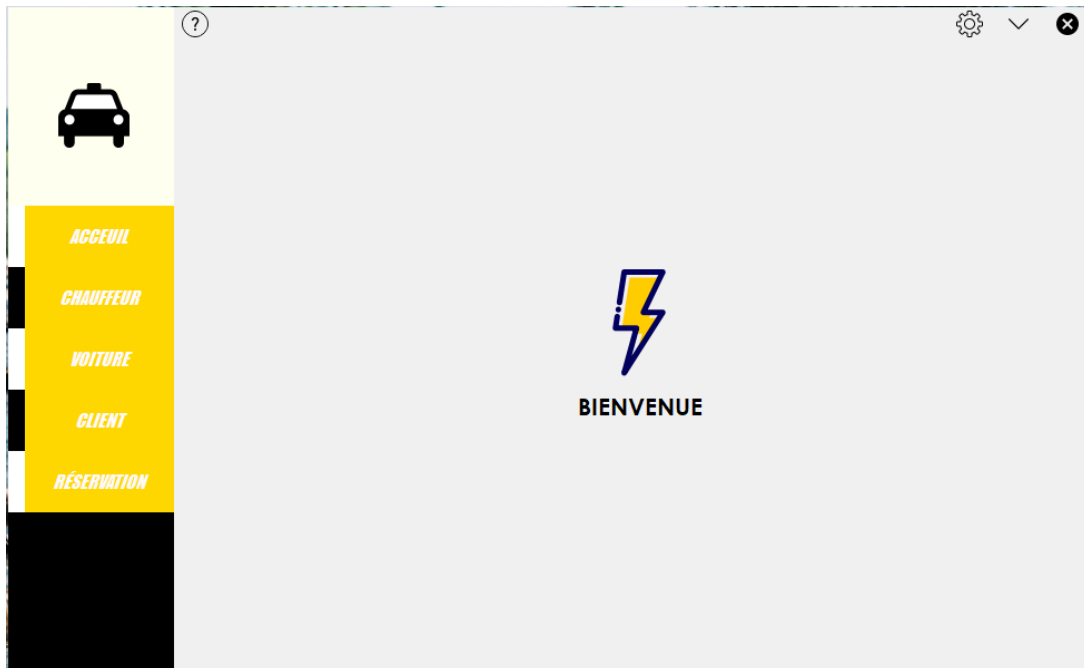
HEURE :

NOUVEAU **SAUVEGARDER**

Problématique de la version 1.0

- Du a la petite taille de l'application / logiciel, l'espace était assez limitée pour l'ajout de nouvelle chose.
- Les données ne se reliées pas à la base de données.

Aperçu du logiciel (Version 2.0)



This screenshot displays the 'NOUVEAU CHAUFFEUR' (New Driver) form. The sidebar is identical to the previous screen, with **CHAUFFEUR** highlighted. The main area has a light gray background and a title bar with a user icon, the text **NOUVEAU CHAUFFEUR**, and window control icons. The form contains the following fields and controls:

- PRÉNOM(S) :** A text input field.
- NOM :** A text input field with the placeholder text 'Nom De Famille'.
- N° TÉLÉPHONE :** A text input field with a '+230' prefix.
- SEXE :** A dropdown menu with a downward arrow.
- E-MAIL :** A text input field.
- NUMÉRO PERMIS :** A text input field.
- DATE DE NAISSANCE :** A text input field.

At the bottom of the form are two buttons: **NOUVEAU** and **SAVEGARDER**.

ACCEUIL

CHAUFFEUR

VOITURE

CLIENT

RÉSERVATION

?

VOITURE

N° MATRICULE :

DATE EXP. ASSURANCE :

MODELE :

NOMBRE SIEGES :

SPECIFICATIONS :

EN USAGE :

PHOTO :

NOUVEAU

SAVEGARDER

ACCEUIL

CHAUFFEUR

VOITURE

CLIENT

RÉSERVATION

NOUVEAU CLIENT

PRENOM(S) :

NOM :

N° TELEPHONE :

SEXE :


EMAIL :

ADRESSE L1:


ADRESSE L2:




NOUVEAU


SAVEGARDER





?

 FAIRE UNE RÉSERVATION







MATRICULE:

DE :

Aucun

ID CHAUFFEUR :

↓

ID CLIENT :

Aucun

 : À


HEURE :

12:00:00 AM

SAUVEGARDER

RESET

Pour ajouter les données, l'utilisateur va appuyer sur le bouton Modifier ce qui rend fait *apparaître* les Panel.



NOUVEAU CHAUFFEUR

PRÉNOM(S) :

NOM :

N° TÉLÉPHONE :

SEXE :

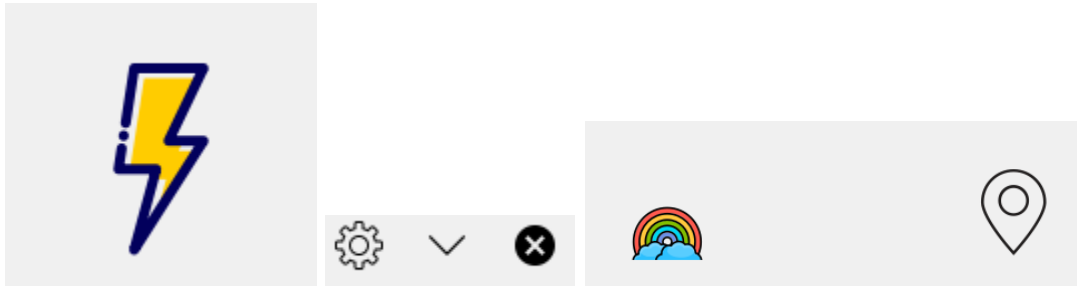
E-MAIL :

NUMÉRO PERMIS:

DATE DE NAISSANCE :

Source des photos et icones

Tous les icones sont du site <https://www.iconfinder.com>



Framework Personnalisée

Pour tous les textbox, boutons personnalisés on a utilisé le Framework Bunifu (trouvée sur <https://bunifuframework.com>)

Ajout des données

Pour l'ajout des données niveau code, on a utilisé la librairie OleDb.

```
3/*
    Nom : Sydney Mungly
    Version: 2.0
    Date: 17 AVRIL 2018
    Description: Logiciel qui gere des taxis en utilisant une base de donnee
    Page: CLIENT
*/
3using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Drawing;
using System.Data;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.OleDb; ←
//Creation de la connection
OleDbConnection vcon = new OleDbConnection(@"Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\Users\User\Documents\MODULES\VB.NET\Projet\ProjetGererTaxi\Projet Gerer Taxi\TAXIG.accdb");
DataTable dt = new DataTable();
```


Ainsi que le Sequel Language (SQL) pour communiquer avec la base de donnée.


```
//INSERER
OleDbCommand cmd = new OleDbCommand();
string sqlinserer = "INSERT INTO Chauffeur ( nomChauffeur, prenomChauffeur, mailChauffeur, sexeChauffeur, numPermisChauffeur, numTelephone, dateNaissance ) VALUES (
try
{
    cmd.Connection = vcon;
    cmd.CommandText = sqlinserer;
    cmd.ExecuteNonQuery();
    MessageBox.Show("Nouveau chauffeur ajouté !", "Message", MessageBoxButtons.OK, MessageBoxIcon.Information);
    resettb();
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message, "Message", MessageBoxButtons.OK, MessageBoxIcon.Error);
}
}
```

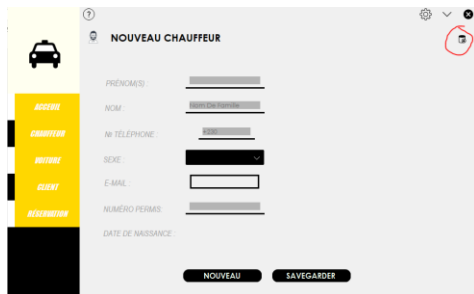
La connexion

Pour savoir si la connexion a été établie avec la base de donnée, on a 2 icones ainsi qu'un message au cas ou il y a eu un problème avec la connexion.

Les icones s'affichent à l'extrême droite en haut de chaque page.

Pour dire s'il y a eu un problème : 

Pour dire que la connexion a été établie : 



Ajout des données(Client)

On a une cliente Patty Murphy une femme de 24 ans qui réside au 21, Inkerman Street Rose-Hill ayant pour adresse mail Patty@mail.com avec comme numéro de téléphone +2305123456.

PRENOM(S) :
 NOM :
 N° TELEPHONE :
 SEXE :
 EMAIL :
 ADRESSE L1:
 ADRESSE L2:

Patty fut ajoutée à la base de donnée.

	CLIENT		
	prenomClient	nomClient	sexeClient
▶	Annie	Leblanc	Femme
	Sia	Ester	Femme
	Leslie	Jones	Femme
	Tyler	Oakly	Homme
	Shaun	Mendes	Homme
	Wendy	Williams	Femme
	The	Weekend	Homme
	Patty	Murphy	Femme

	CLIENT		
	sexeClient	telClient	mailClient
▶	Femme	+23058944114	annie@orange.fr...
	Femme	+23058944114	Sia@orange.fr#m...
	Femme	+23058944114	Leslie@gmail.co...
	Homme	+23058944114	Tyler@hotmail.co...
	Homme	+23058944114	Shaun@gmail.co...
	Femme	+23058944114	wendy@wendy.c...
	Homme	+23058944114	weekdays@we.o...
	Femme	+2305123456	Patty@mail.com

Ajout des données (Chauffeur)

On a un nouveau Chauffeur John Doe, homme, âgé de 24 ans (né le 12/12/1994) ayant comme numéro de permis 28374MU et email John@mail.com et une voiture (MERCEDES G WAGON 4 places) matriculée 7812 Zn 2019 assurée jusqu'au 13/02/2019 avec WIFI inclus (sans numéro de téléphone précisé).

PRÉNOM(S) :	<input type="text" value="John"/>	N° MATRICULE :	<input type="text" value="7812 Zn 2019"/>
NOM :	<input type="text" value="Doe"/>	DATE EXP. ASSURANCE :	<input type="text" value="2/13/2019"/> ▼
N° TÉLÉPHONE :	<input type="text" value="+230"/>	MODELE :	<input type="text" value="ERCEDES G-WAGON"/>
SEXE :	<input type="text" value="Homme"/> ▼	NOMBRE SIEGES :	<input type="text" value="4"/>
E-MAIL :	<input type="text" value="John@mail.com"/>	SPECIFICATIONS :	<input type="text" value="WIFI"/>
NUMÉRO PERMIS :	<input type="text" value="28374MU"/>	EN USAGE :	<input type="text" value="OUI"/> ▼
DATE DE NAISSANCE :	<input type="text" value="12/12/1994"/> ▼	PHOTO :	

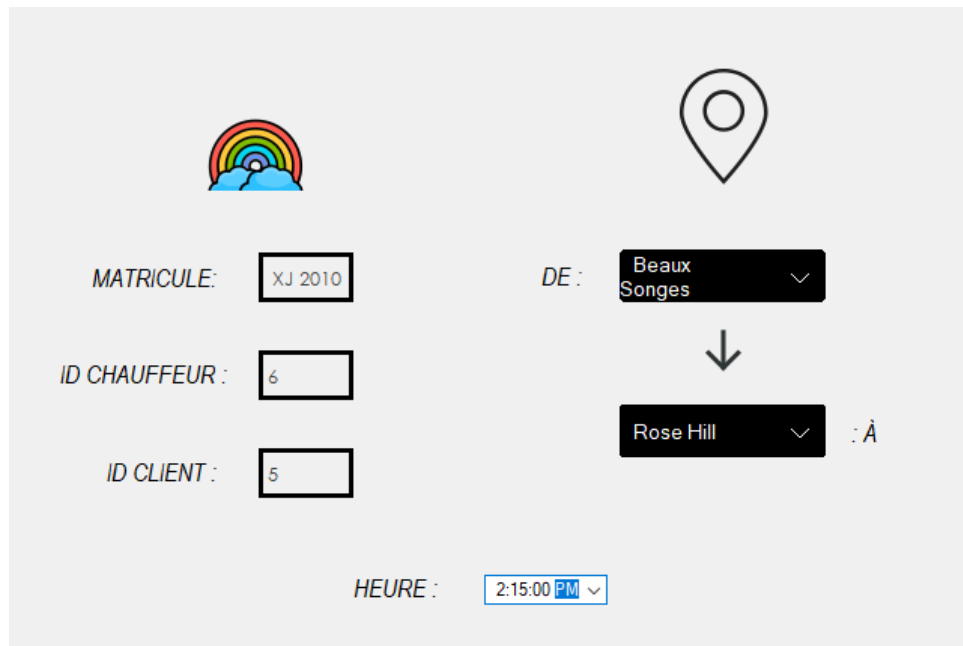
CHAUFFEUR			
IDChauffeur	nomChauffeur	prenomChauffeur	
6	Shane	Dawson	
7	Tom	Holmes	
8	Silver	Adams	
9	Lily	James	
10	Steven	Cav	
20	Doe	John	
* <input type="text"/>			

VOITURE				
MatriculeID	expAssurance	modele	non	
1234 ZN 2015	12/1/2020	HONTA CIVIC	4	
4567 AB 2018	2/19/2021	BMW	4	
8911 XY 2012	2/13/2021	JEEP	4	
8752 XJ 2010	5/20/2019	MERCEDES J W...	4	
1237 WQ 2918	1/2/2019	NISSAN	4	
7812 ZN 2019	2/13/2019 1:31 ...	MERCEDES G...	4	
* <input type="text"/>				

Ajout des données (Réservation)

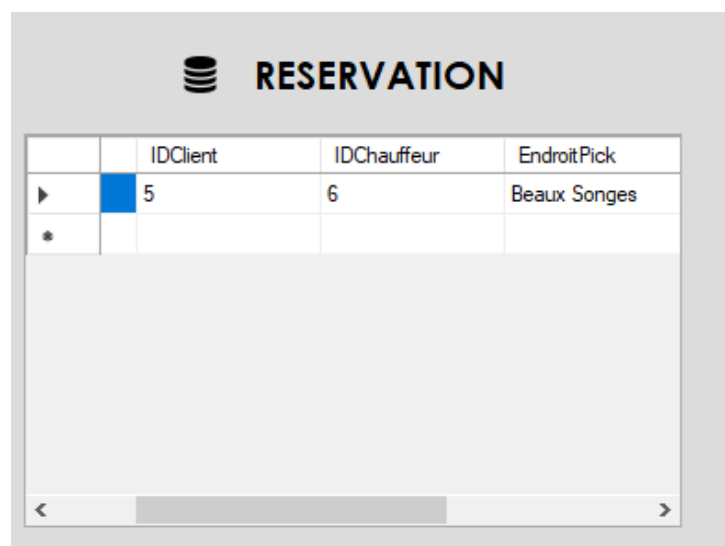
On a une réservation de la part du Client Shaun Mendes (identifiant 5) pour faire le trajet Beau-Songes (Départ) à Rose-Hill (Destination) a 02 :15 PM.

On lui assigne le Chauffeur Wendy Williams (identifiant 6) avec la voiture matriculée 8752 XJ 2010.



The form is titled 'Ajout des données (Réservation)'. It features a rainbow icon on the left and a location pin icon on the right. The form contains several input fields and dropdown menus:

- MATRICULE:** A text input field containing 'XJ 2010'.
- ID CHAUFFEUR:** A text input field containing '6'.
- ID CLIENT:** A text input field containing '5'.
- DE:** A dropdown menu with 'Beaux Songes' selected.
- À:** A dropdown menu with 'Rose Hill' selected.
- HEURE:** A dropdown menu with '2:15:00 PM' selected.

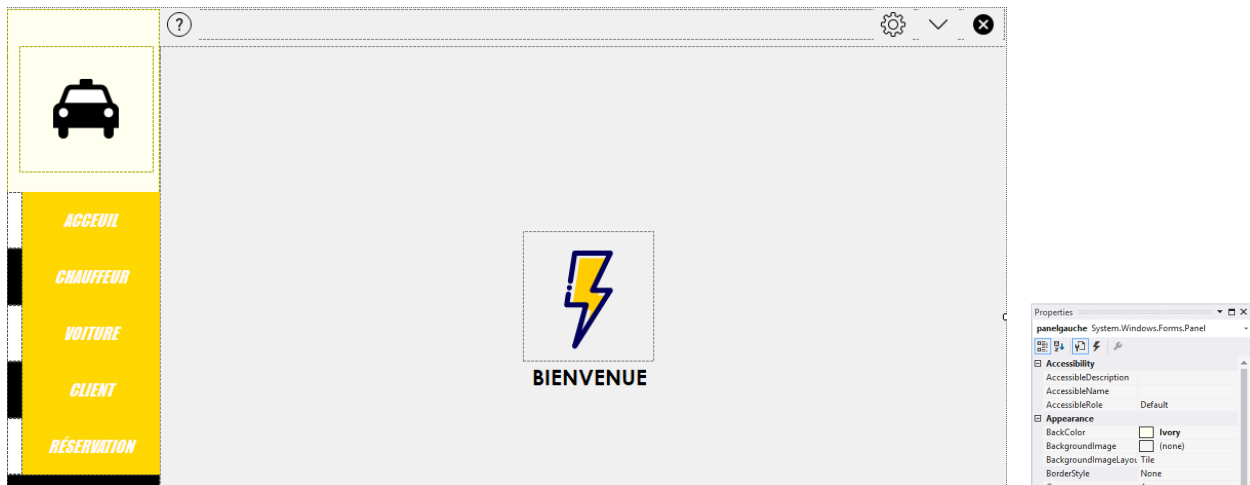


The table is titled 'RESERVATION' and has four columns: IDClient, IDChauffeur, EndroitPick, and a blank column. The first row shows a reservation for client 5, driver 6, starting at Beaux Songes. The second row is empty, and the third row is a placeholder for a new reservation.

	IDClient	IDChauffeur	EndroitPick
▶	5	6	Beaux Songes
*			

Conception du design du logiciel

Pour concevoir le logiciel, on a fait usage de UserControl et de Panels.



The screenshot shows a form titled **NOUVEAU CLIENT**. The form has a light gray background. At the top left is a lightbulb icon, and at the top right is a small icon of a document with a checkmark. The form contains the following fields and controls:

- PRENOM(S) :** A text input field.
- NOM :** A text input field.
- N° TELEPHONE :** A text input field with a placeholder '+230'.
- SEXE :** A dropdown menu with a black background and a white checkmark.
- EMAIL :** A text input field.
- ADRESSE L1 :** A text input field.
- ADRESSE L2 :** A text input field.

At the bottom of the form are two buttons: **NOUVEAU** and **SAVEGARDER**.

Le UserControl(UC) est implémenté et connectée avec les lignes de codes suivantes(Par exemple le UC Client) :

```
// Pour que le main herite du UserControl|
public static Client _instance;
    5 references
    public static Client instance
    {
        get
        {
            if (_instance == null)
            {
                _instance = new Client();
            }
            return _instance;
        }
    }
    1 reference
```

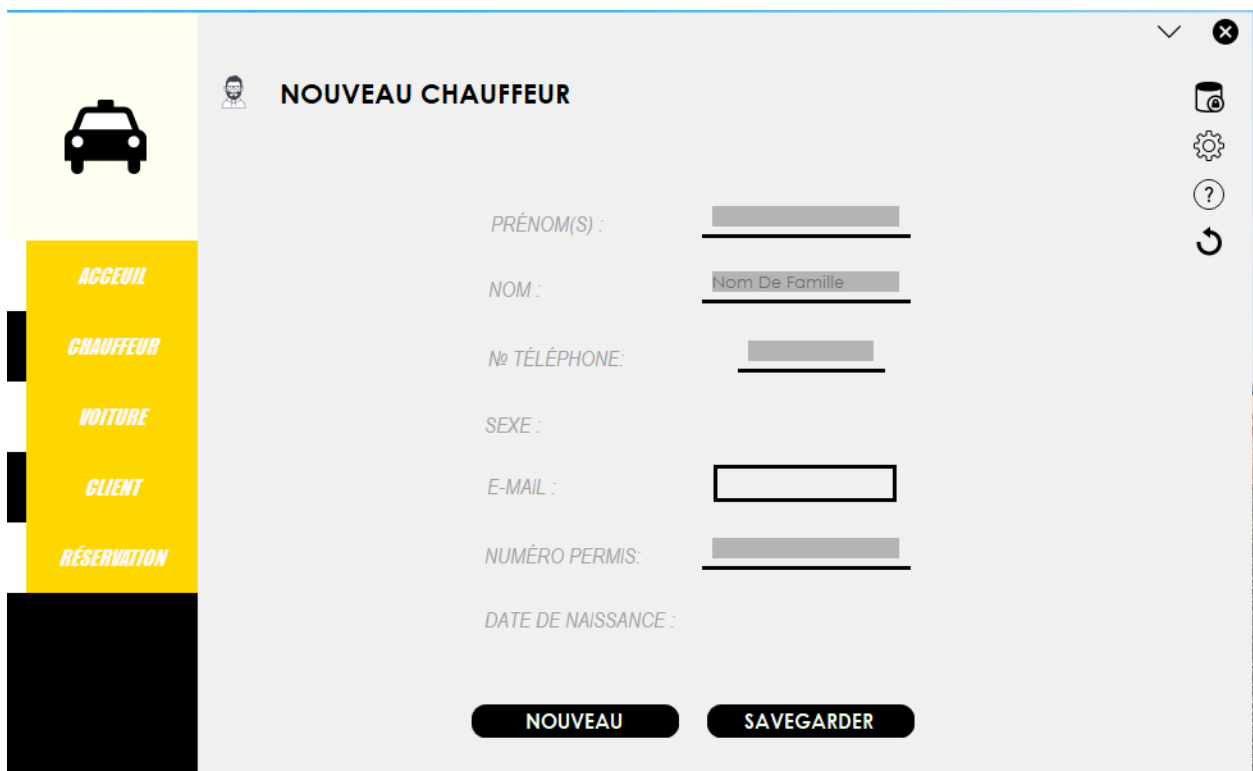
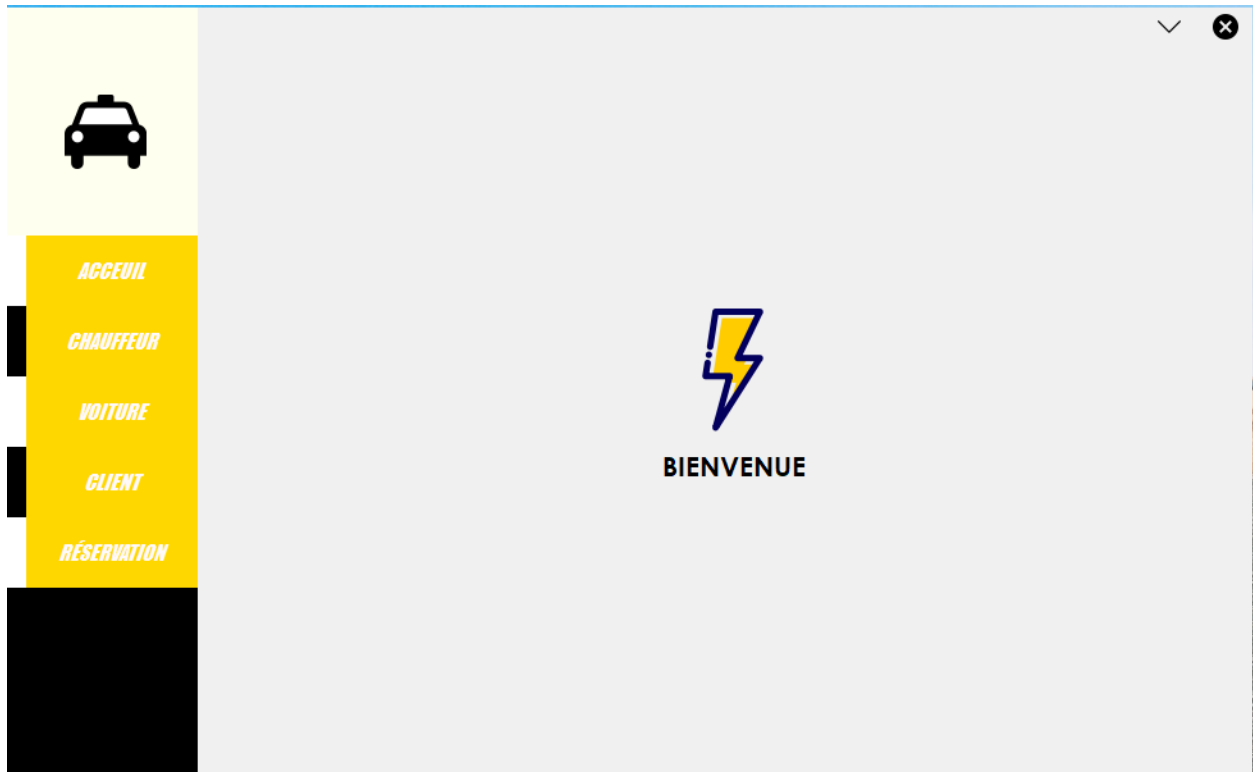
Puis dans le formulaire principal avec les lignes suivantes (Par exemple le UC Client) :


```
1 reference
private void btnreserv_Click(object sender, EventArgs e)
{
    if (!MainPan.Controls.Contains(Client.instance))
    {
        MainPan.Controls.Add(Client.instance);
        Client.instance.Dock = DockStyle.Fill;
        Client.instance.BringToFront();
    }
    else
        Client.instance.BringToFront();
}
```


Problématiques de la version 2.0







- Les données ne peuvent à présent qu'être uniquement ajoutées et non effacées ou mises-a-jour.
- Il n'y a pas de connexion entre les chauffeurs et les TAXIS, on peut dire que les TAXIS sont attribués au Chauffeur.
- Les champs de textes ne sont pas mis à zéro après avoir ajouter des données.
(Résolu)

Aperçu du logiciel (Version 3.0)





 **VOITURE**



N° MATRICULE :


DATE EXP. ASSURANCE :

MODELE :

NOMBRE SIEGES :

SPECIFICATIONS :

EN USAGE :

PHOTO : 



 **NOUVEAU CLIENT**



PRÉNOM(S) :

NOM :


N° TÉLÉPHONE :


SEXE :





EMAIL :


ADRESSE L1:


ADRESSE L2:



 **FAIRE UNE RÉSERVATION**







MATRICULE:

DE:

Aucun

ID CHAUFFEUR:

↓

ID CLIENT:

Aucun

 : À

HEURE:

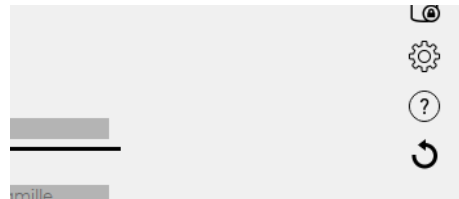
12:00:00 AM

NOUVEAU

Améliorations

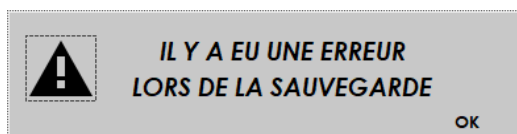
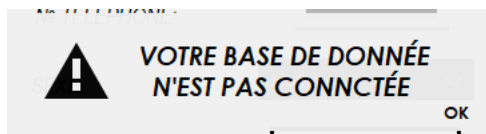
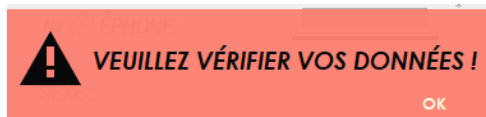
La version 3.0 de ce logiciel apporte plus de changements au niveau de l'interface. Pour chaque *formulaire* l'utilisateur peut voir toutes les données individuellement sur la page sur laquelle l'utilisateur est.

On apporte aussi un bouton reset qui va aider à mettre à zéro les TextBox ainsi que le bouton aide qui était déjà présent à la version 2.0.

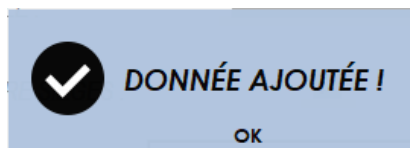


Cette nouvelle version est une version sans MessageBox mais avec des notifications personnalisées grâce à des User Control. On a :

- Les notifications pour les erreurs.



- Les notifications pour indiquer que les données ont été sauvegardées.



Les Validations

La validation des données se fait avec les codes pour vérifier si les TextBox ne sont pas vides, sont en bon format (pour les email en utilisant `using System.Text.RegularExpressions` et `Regex`) et ne contiennent que des chiffres pour le numéro de téléphone.

```
private void verification()
{
    // Validation des text-box

    flag = false;
    if (String.IsNullOrEmpty(TBPrenom.Text) || String.IsNullOrWhiteSpace(TBPrenom.Text))
    {
        error1.Visible = true;
        flag = true;
    }
    else
    {
        error1.Visible = false;
    }
    if (String.IsNullOrEmpty(TBNom.Text) || String.IsNullOrWhiteSpace(TBNom.Text))
    {
        error2.Visible = true;
        flag = true;
    }
    else
    {
        error2.Visible = false;
    }

    if (String.IsNullOrEmpty(TBMail.Text) || String.IsNullOrWhiteSpace(TBMail.Text))
    {
        error4.Visible = true;
        flag = true;
    }

    Regex reg = new Regex(@"\w+([-+.] \w+)*@\w+([-.] \w+)*\.\w+([-.] \w+)*");
    if (!reg.IsMatch(TBMail.Text))
    {
        error4.Visible = true;
        flag = true;
    }
    else
    {
        error4.Visible = false;
    }

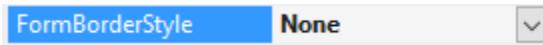
    int parsedValue;
    if (!int.TryParse(TBTel.Text, out parsedValue) || (TBTel.Text.Length >= 9))
    {
        error3.Visible = true;
        flag = true;
    }
    else
    {
        error3.Visible = false;
    }
}
```

On notifie l'utilisateur en affichant une étoile (*).

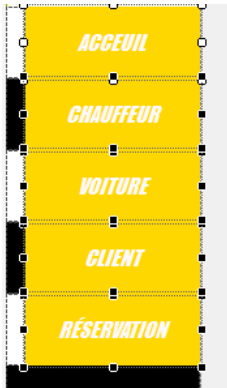
PRÉNOM(S) :	<input type="text"/>	*
NOM :	<input type="text" value="Nom De Famille"/>	*
N° TÉLÉPHONE:	<input type="text"/>	*
SEXE :	<input type="text" value="Homme"/>	
E-MAIL :	<input type="text"/>	*
NUMÉRO PERMIS:	<input type="text"/>	*
DATE DE NAISSANCE :	<input type="text" value="5/2/1997"/>	

Comment on a fait l'interface

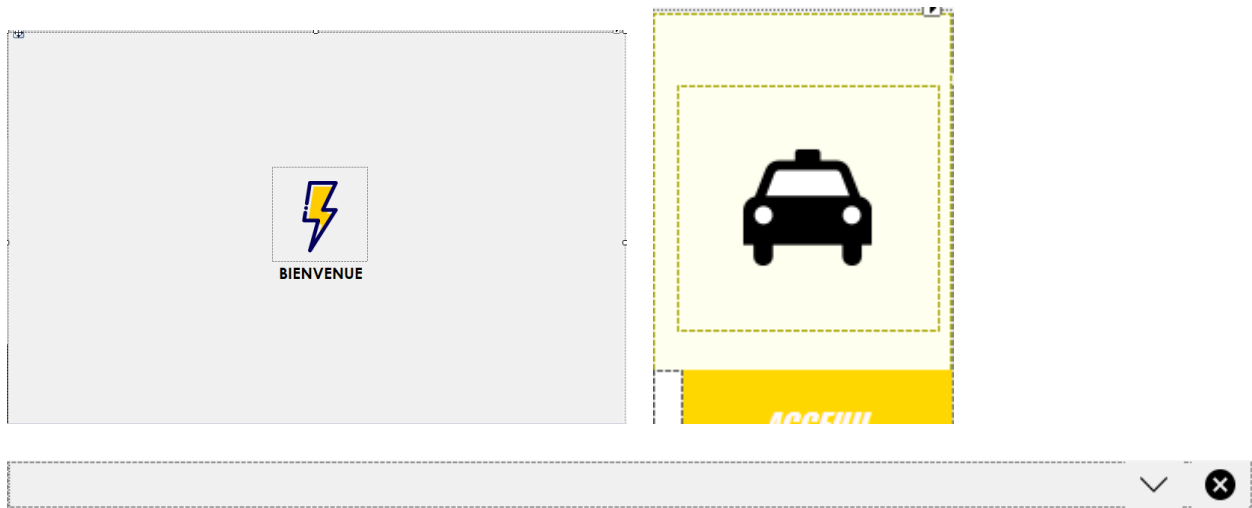
1. On a pris un formulaire et désactiver les *border*.



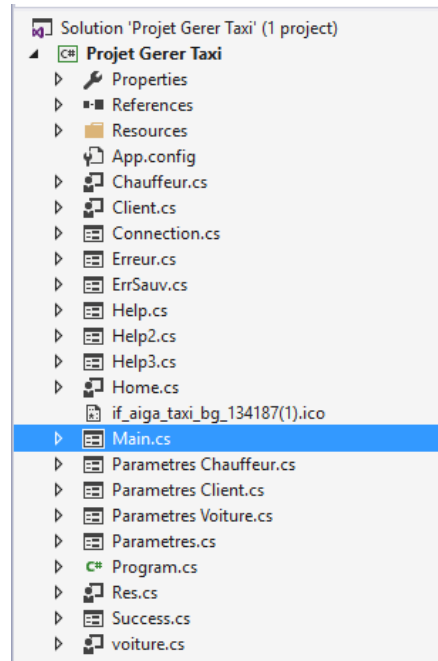
2. Puis on a ajouté des boutons sur le coté



3. Pour qu'on puisse intégrer les fonctionnalités sur chaque côté et que le logiciel soit pas instable on a fait l'utilisation de panel.



4. Pour qu'à chaque fois qu'on appuie sur un bouton, une page apparaisse, on a fait utilisation d'un panel, qu'on appelle le *MainPanel* pour que les User Control se placent sur ce *MainPanel*. En gros, chaque page (Réservation, Accueil, Voiture, Chauffeur) n'est en fait qu'un user control qui se place sur *MainPanel*.



- i. Grace à ce bout de code sur le formulaire principal pour chaque User Control.

```
1 reference
private void accueil_Click(object sender, EventArgs e)
{
    if (!MainPan.Controls.Contains(Home.instance))
    {
        MainPan.Controls.Add(Home.instance);
        Home.instance.Dock = DockStyle.Fill;
        Home.instance.BringToFront();
    }
    else
        Home.instance.BringToFront();
}
```

- ii. Les User Control doivent hériter du Main donc sur chaque User Control.

```
// Pour que le main herite du UserControl
public static Client _instance;
5 references
public static Client instance
{
    get
    {
        if (_instance == null)
        {
            _instance = new Client();
        }
        return _instance;
    }
}
```

Problématiques de la Version 3.0

Pour le reste, le logiciel a les mêmes fonctionnalités que la version précédente. Mais au niveau des problématiques :

- Le logiciel permet de voir et ajouter des données seulement et non les mettre à jour ou les effacer de la base de données.
- Si le logiciel doit être implémenté sur un autre ordinateur, il va falloir changer la source pour chaque formulaire.

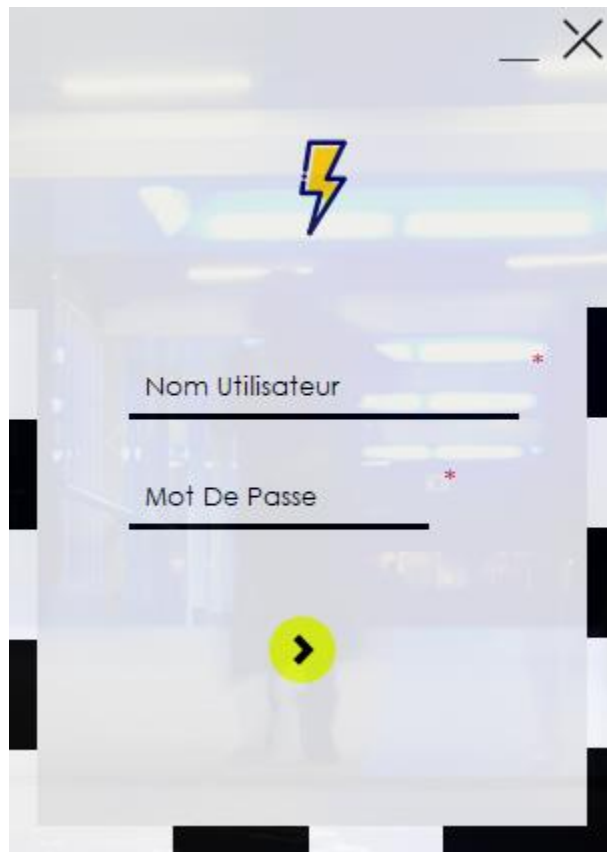
```
//Création de la connexion  
OleDbConnection vcon = new OleDbConnection(@"Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\Users\User\Documents\MODULES\VB.NET\Projet\ProjetGenerTaxi\Projet Gener Taxi\TAXIG.accdb"  
DataTable dt = new DataTable();
```

- Il n'y a pas de validations pour les dates mais les données ne seront pas sauvegardées car les validations ont été fait sur Access aussi.
- La version .exe ne peut être transférée sur un autre ordinateur sans créer un setup.

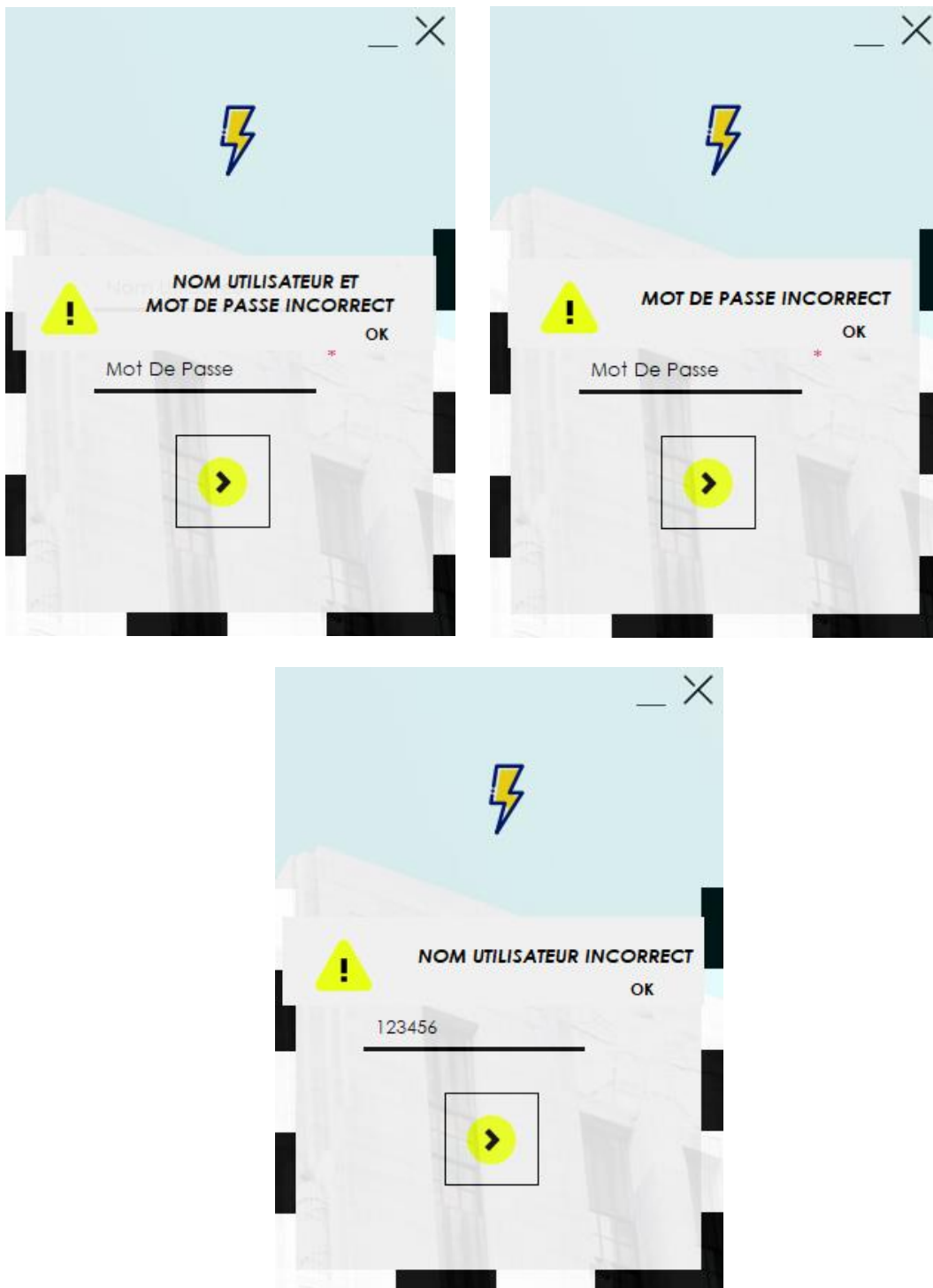
Version 4.0 (Version finale)

La dernière version du logiciel n'apporte qu'un seul changement : un login de démarrage pour apporter une certaine sécurité au logiciel et au données. Le reste du logiciel reste le même en plus d'un bouton pour retourner au mode Login quand voulu.

Aperçu :



On apporte aussi des messages d'erreurs personnalisées en cas de mot de passe/ nom utilisateur erronées ou champs vides, introduites dans les version 3.0.



Dans le programme principal on introduit le bouton



qui va permettre a l'utilisateur de retourner au mode login en temps voulu.

