Projet final ISN - Logiciel de gestion

**Problématique :** Comment apporter une solution informatique pour permettre la gestion de réservation du parc automobile d’une entreprise ?

**Plan :**

I - Cahier des charges et solution adoptée

II - Programmation de la base de données

III - Mise en place de la partie graphique

IV - Toutes les fonctionnalités du logiciel

**Présentation :**

L’objectif de notre groupe était de vouloir confectionner un logiciel qui serait utilisé pour pouvoir aider une personne ou un groupe de personnes. Nous avons donc fait des recherches et l’économat du lycée nous a proposé de réaliser un logiciel pour l’ensemble du lycée. L’idée était de créer un logiciel pour permettre la gestion du parc automobile de l’établissement Sainte Croix Saint Euverte. Nous avons donc accepté après avoir pris connaissance du cahier des charges imposé par le demandeur.

**I - Cahier des Charges :**

Le logiciel devra comporter deux parties : une pour l’administration et une pour l’utilisateur. Nous rappellerons les différences entre ces deux parties plus loin dans le CDG.

L’établissement possède 4 voitures de fonctions de différentes formes :

- 1 boxer (camion)

- 1 plateau

- 1 modus

- 1 transit

Ces 4 voitures peuvent être réservée pendant une demi-journée, une journée ou deux jours.

Il faut savoir que seul les personnes ayant accès à l’intranet peuvent réserver les voitures de fonctions.

Il faut que les deux économats du collège et du LEGT puisse également accéder au logiciel. Pour se connecter sur l’intranet, il n’y a pas de mot de passe demandé, aucune différenciation n’est faite avant l’accès au logiciel de ce fait.

Pour la partie graphique, il faudrait un calendrier mensuel (un affichage sur 4 à 5 semaines) avec la partie visible de réservations. L’administrateur doit pouvoir imprimer ce calendrier. Pour toute la partie graphique nous sommes libres à partir du moment où le logiciel reste intuitif et remplissent les conditions précisées précédemment.

**I - Solution adoptée :**

Réalisation d’un logiciel destiné à s’exécuter sur des systèmes exploitations Windows. Après un certain travail exhaustif, le langage le plus adapté sera donc le C# (avec support en .NET). Pour cela, nous utiliserons l’IDE mis à disposition par Microsoft nommé Visual Studio. Pour gérer la communication et la sauvegarde des données entre la partie administrateur et la partie utilisateur nous allons implanter mettre en place un serveur avec une base de données permettant ainsi aux deux parties d’interagir entres elles ainsi qu’un accès permanant aux données.

**II - Programmation de la base de données :**

**1 - Choix du type de serveurs**

Dès lors de la conception du logiciel, un problème c’est directement posé à nous. Comment faire communiquer la partie administrateur du logiciel et la partie utilisateur et permettre la sauvegarde des données tout en étant accessible tout le temps. Une solution parmi les plus courantes s’est imposée à nous : Utiliser un serveur. Seulement il existe plusieurs types de serveurs (Logiciel serveur indépendant, MySQL, Oracle, Apache …) utilisant tous des langages de programmation différents et remplissant des fonctions variées. Nous avons donc recherché quel type de serveur serait le plus adapté à notre situation permettant une communication simple avec notre logiciel en C#. Nous avons donc décidé d’utiliser un serveur MySQL (gestion de bases de données avec accès distant) pour notre logiciel, cependant, comment le mettre en place et le configurer ?



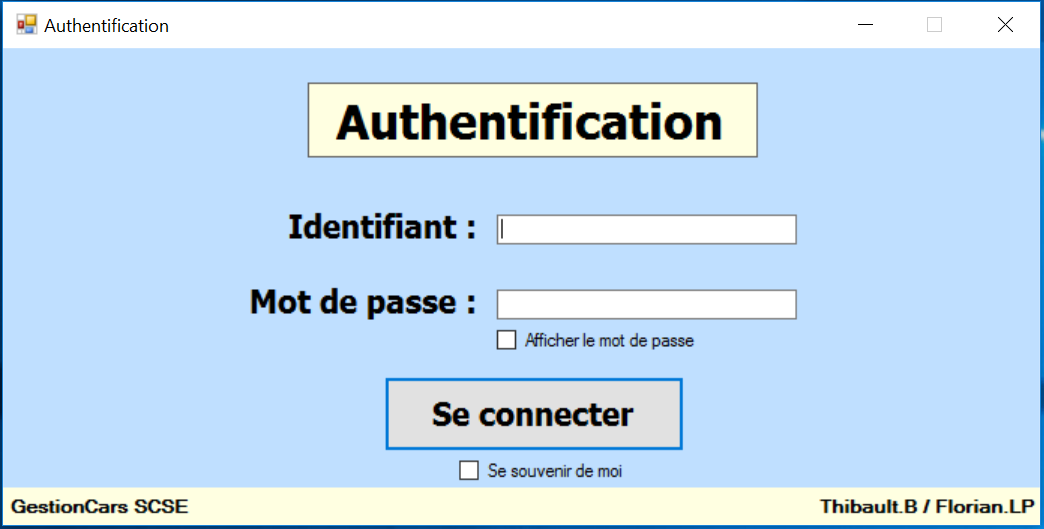
**2 - Mise en place et configuration du serveur MySQL**

L’avantage d’utiliser un serveur MySQL c’est que de nombreux hébergeurs utilisent le Langage MySQL. Nous allons donc utiliser un l’hébergeur WAMP Serveur qui inclut une version préconfigurée d’un serveur MySQL avec de nombreux outils, nous permettant de modifier assez facilement notre base de données et de remplir les fonctions dont nous avions besoin. Une fois après avoir créé notre serveur Wamp et utilisé une configuration locale (IP : 127.0.0.1), nous pouvions accéder à notre base de données à l’aide de phpMyAdmin (outil graphique d’accès et de gestion des bases de données MySQL). Une base de données est composée de tables et d’informations composants ces tables. Nous avons dû créer 4 tables pour notre projet. La première pour les « nom d’utilisateur » et « mot de passe », une autre stockant les demandes de réservations, une pour stocker et gérer les journées du calendrier ainsi qu’une dernière pour conserver un historique des réservation acceptés par un administrateur.

**III - Mise en place de la partie graphique :**

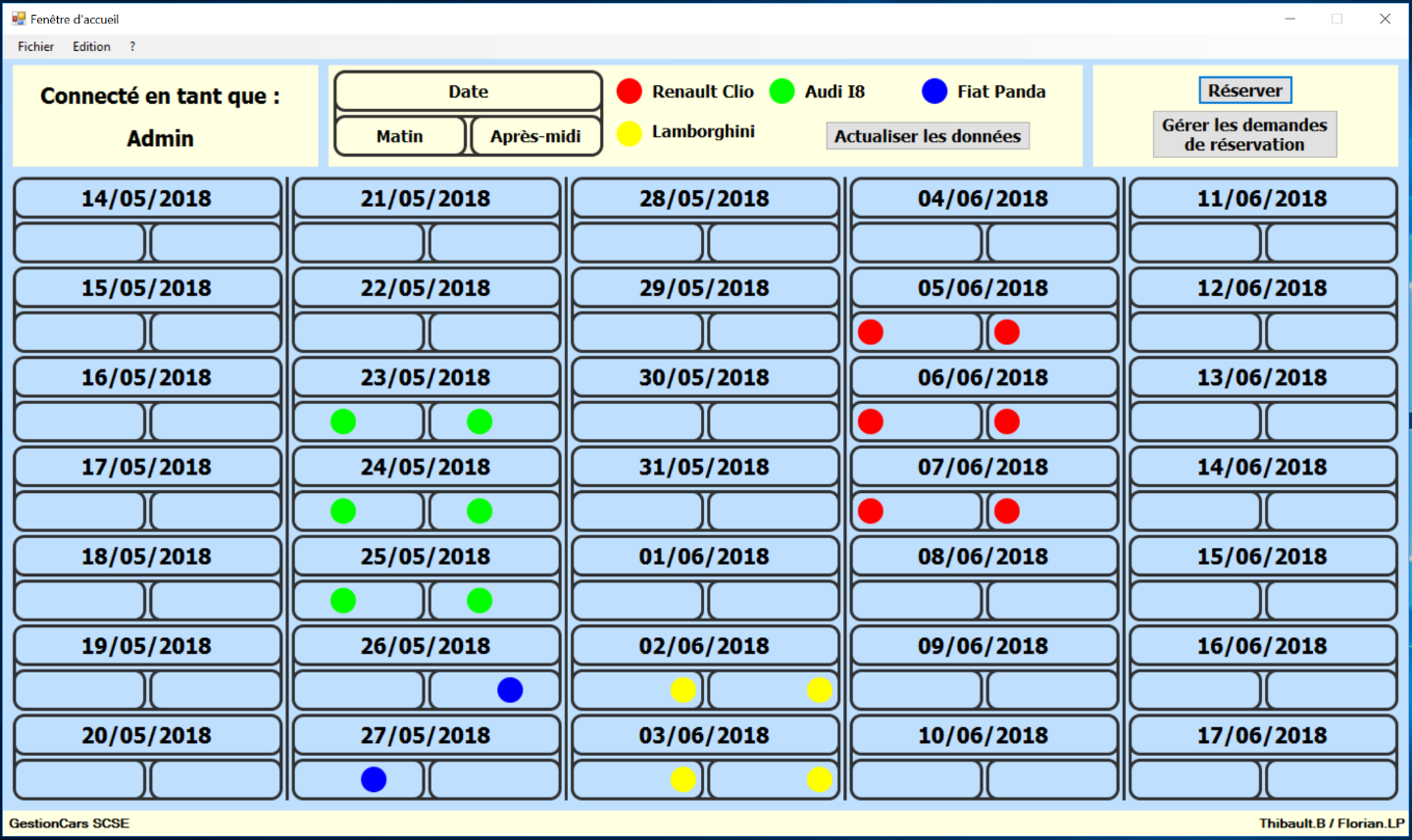
**1 - Fenêtre pour l’authentification**

En utilisant le concepteur graphique (WinForms) fournit par l’IDE Visual Studio, nous avons créé une première page : d’identification où l’on doit rentrer notre login et mot de passe pour se connecter. Si ces deux informations sont reconnues dans la base de données alors cette page se ferme pour laisser place à la fenêtre principale, sinon une erreur est affichée à l’aide d’un pop-up.



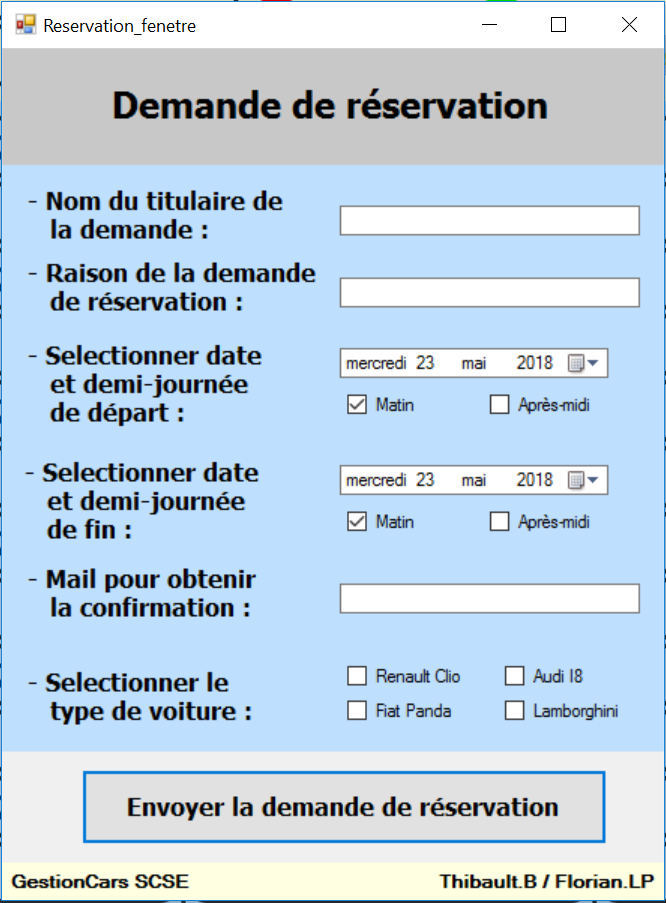
**2 - Fenêtre principale**

Nous devions faire figurer dans cette page les différentes fonctionnalités par rapport à l’authentification (admin/Utilisateur) ainsi que le calendrier présentant les réservations. Nous avons donc affiché dans le coin haut gauche le type de connexion (admin/utilisateur) et dans celui de droit le type d’accès (accepter les réservations pour l’admin ou faire une demande de réservation pour les utilisateurs). Au centre nous avons fait figurer la légende du calendrier et ce dernier occupe la majeure partie de la place de la fenêtre pour le mettre en valeur et bien visualiser son contenu. Pour l’administrateur, cliquer sur la pastille colorée d’une réservation en affiche les détails dans un petit pop-up.



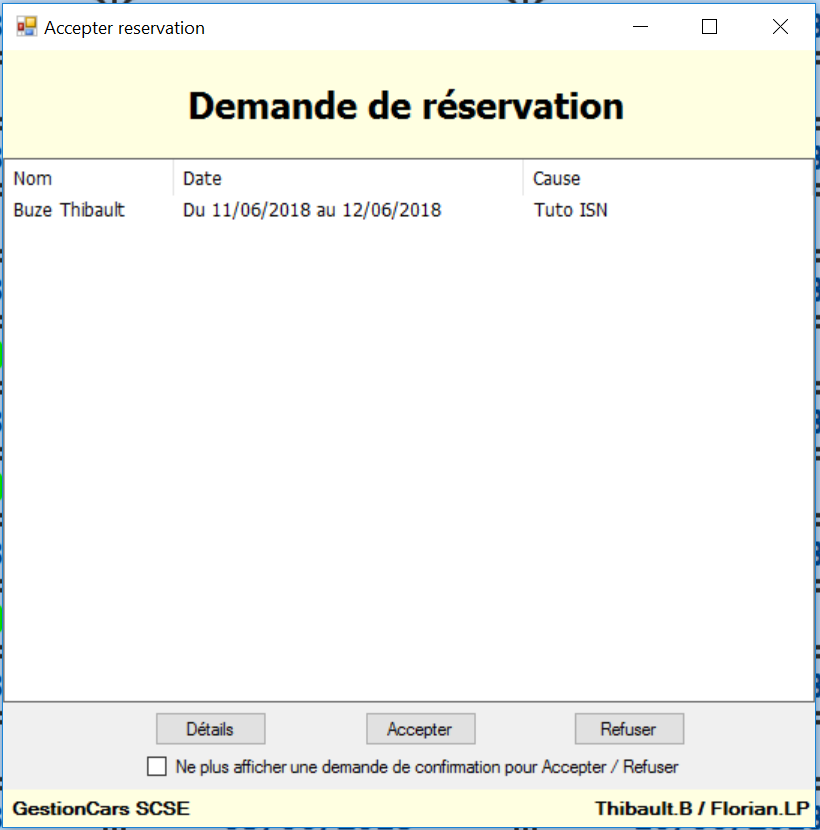
**3 - Fenêtre pour la demande de réservation**

Ici nous avons créé une fenêtre ou l’utilisateur peut faire une demande de réservation. Pour cela il va devoir remplir plusieurs champs obligatoires afin que sa demande de réservation soit validée (Nom / Raison de la demande / Date / Mail pour l’envoi de la réponse). Suite à cela, la demande de réservation est envoyée sur le serveur et sera accessible par l’administrateur lors de sa connexion. Dans le cas d’un administrateur, si la demande est valide, elle est directement acceptée.



**4 - Fenêtre pour accepter ou non les demandes de réservations**

Une dernière fenêtre ajoutée au projet est cette fois ci seulement accessible aux administrateurs uniquement. En effet, cette fenêtre permet d’afficher les demandes de réservations et de les accepter ou bien de les refuser. En effet dès l’ouverture de la page, le logiciel charge l’ensemble des informations de la table où sont enregistrés les demandes de réservations et les affiches à la suite dans un tableau. Trois choix s’offrent alors aux administrateurs : demander plus d’informations sur la demande de réservation, l’accepter ou la refuser. Suite aux choix de l’administrateur et à la fermeture de la fenêtre, le calendrier sera actualisé avec les réservations validées et les rajoutera sur l’affichage du calendrier. Les réservations refusées sont supprimées tandis que les réservations ni acceptées ni refusées sont conservées.



**IV - Toutes les fonctionnalités du logiciel**

Toutes les parties présentées précédemment permettent de remplir en partie le cahier des charges mais pas de proposer un « logiciel » propre et intéressant pour une entreprise. Nous avons donc rajouté certaines fonctionnalités permettant de compléter le programme pour le rendre utilisable et mieux adapté dans le cadre des objectifs qui nous avaient été imposés.

**1 - Les messages Pop-Up**

Afin de rendre l’utilisation du logiciel plus simple aux utilisateurs, nous avons ajoutés l’apparition de fenêtre Pop-Up (petite fenêtre à titre informative ouverte automatiquement) à certains moments clés (lors de l’authentification, pour confirmer une demande de réservation …). Elles sont aussi là pour avertir les utilisateurs comme par exemple si une nouvelle demande de réservation a été enregistrée, elle permet de prévenir l’administrateur. Elles apportent toutes les informations complémentaires qui ne figurent pas directement sur les 4 pages principales.

**2 - « Se souvenir de moi »**

Le fait de devoir s’authentifier à chaque début d’utilisation du logiciel est assez embêtant pour un administrateur ouvrant ce logiciel plusieurs fois par jours. Nous avons donc rajouté sous le bouton « Connexion » de la page d’accueil une petite option « se souvenir de moi » si l’utilisateur ou l’administrateur décide de cocher cette option, alors dès sa prochaine ouverture, le logiciel aura déjà rempli les cases d’identifications avec les informations rentrées lors de la précédente connexion.

**3 - Barre de menu sur la page principale**

Afin de rendre le logiciel plus professionnel, nous avons ajouté une barre de menu sur le haut de la page principale. Cette barre possède 3 options : Fichier, Edition et Aide. Dans l’option fichier, vous pouvez réduire, fermer la fenêtre ainsi qu’imprimer le calendrier pour les administrateurs. Dans l’option Edition, vous pouvez ouvrir la fenêtre de réservation ou de gestion de réservation en fonction de vos permissions. Tandis que la fonction Aide, regroupe plusieurs questions que l’utilisateurs puisse se poser et en clinquant dessus, une aide lui est proposé. Une petite option « A propos » décris pourquoi ce logiciel a été réalisé, par qui et pour qui.

**4 - Impression du calendrier**

Une des caractéristiques du cahier des charges était de pouvoir imprimer le calendrier. Ayant considéré qu’imprimer seulement le calendrier rendrait sa compréhension difficile car nous n’aurions aucune information sur les différentes réservations. Nous avons donc créé un PDF avec un capture d’écran du calendrier et par la suite un détail sous forme de liste de chaque réservation pour que l’administrateur ai tous les informations nécessaires. Le fichier PDF est créé sur le bureau lorsque l’administrateur clique sur « Créer un PDF » et il ne lui reste plus qu’à accéder au PDF créé et s’il le souhaite, l’imprimer.

**5 – Les micro fonctionnalités du programme (arrière-plan)**

Pour fonctionner et gérer les données que le programme récupère et transmet que ce soit avec l’utilisateur ou la basse de données, nous avons dû créer de nombreuses fonctionnalités internes au programmes et invisibles pour l’utilisateur. Ces fonctionnalités permettent par exemple de gérer la communication avec le serveur SQL : charger les données, mettre à jour les données serveur/programme et envoyer les données lors de la fermeture. Elles permettent aussi la gestion d’arrière-plan du calendrier en structurant les jours dans l’ordre et en y associant les données qui y sont liées. Enfin ces fonctionnalités permettent de gérer les conversions depuis les réservations brutes vers les données exploitables pour le calendrier.