

Rapport de Projet

Nom : AIT BIHI

Prénom : Boubaker

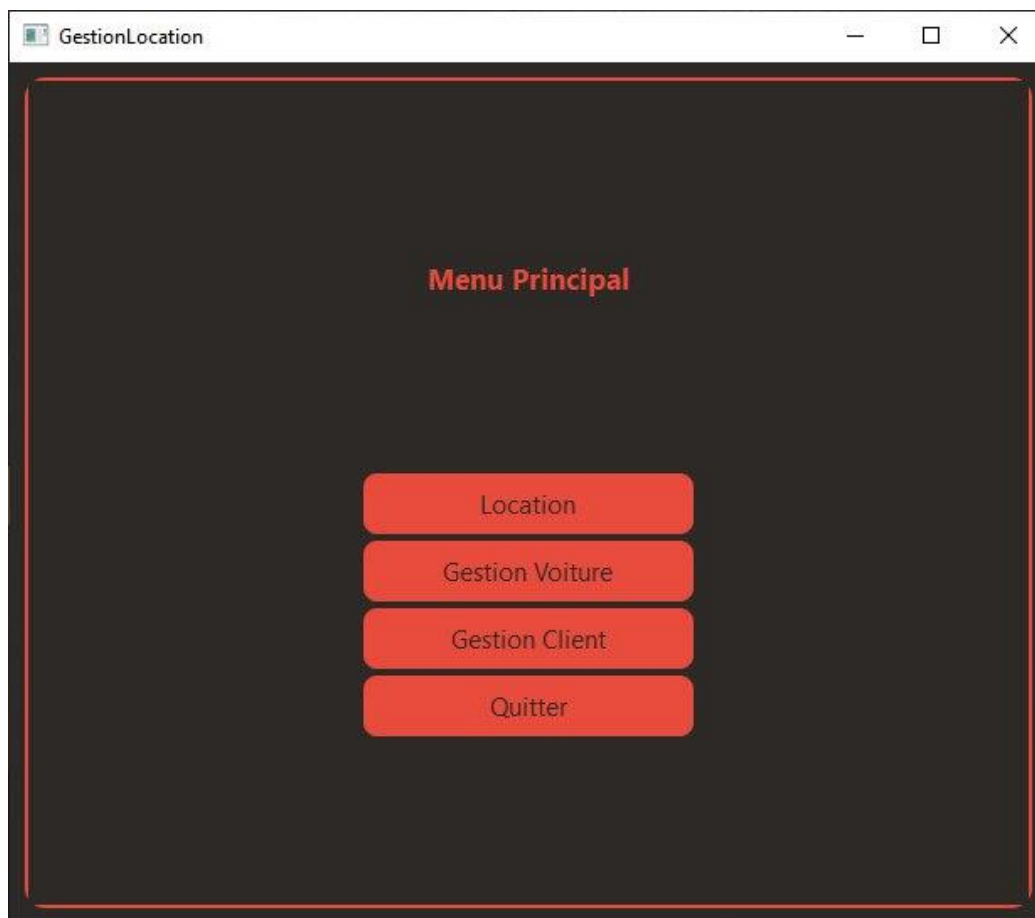
Filière : SMI4

I. Définition de projet

L'objectif de ce projet est de développer une application pour la gestion d'une agence de location de voitures. Cette dernière permettra de réaliser toutes les opérations nécessaires pour la gestion des clients (locataires), la gestion des voitures et la gestion des locations. et elle permet de stocker les données dans des fichiers et de les organiser, d'une manière facile d'accès et lire.

II. Description de projet

- **Menu principal**



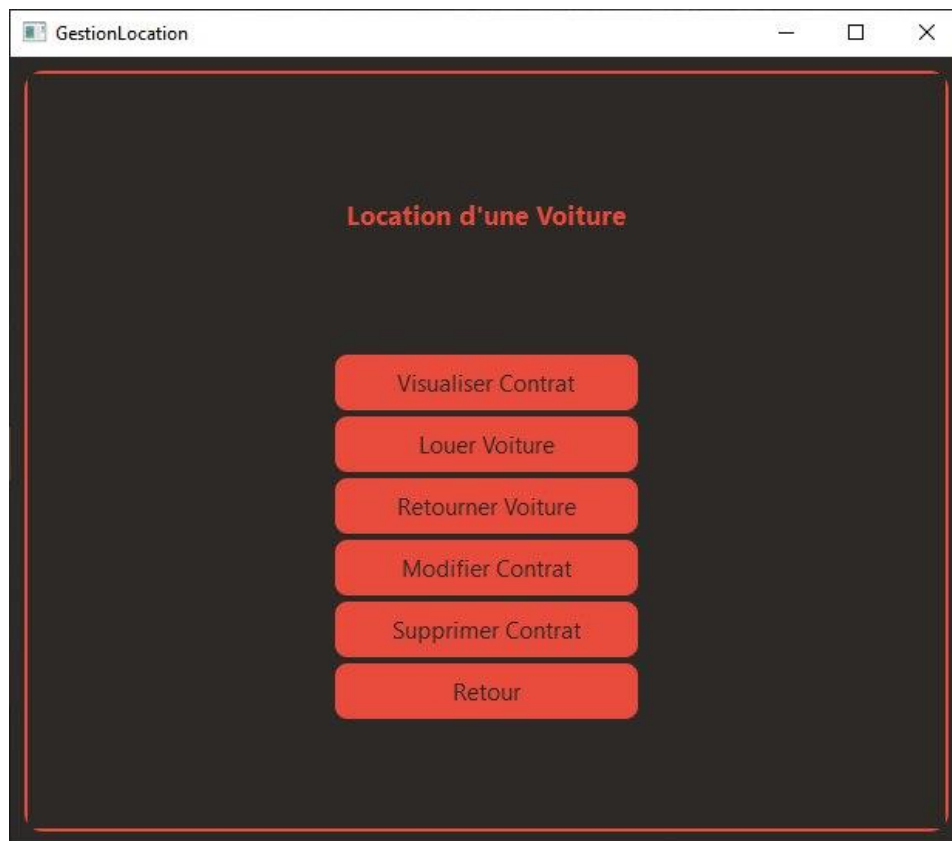
chaque bouton lorsqu'il est cliqué, exécute une fonction qui charge la page correspondante

```
void GestionLocation::on_rental_clicked()
```

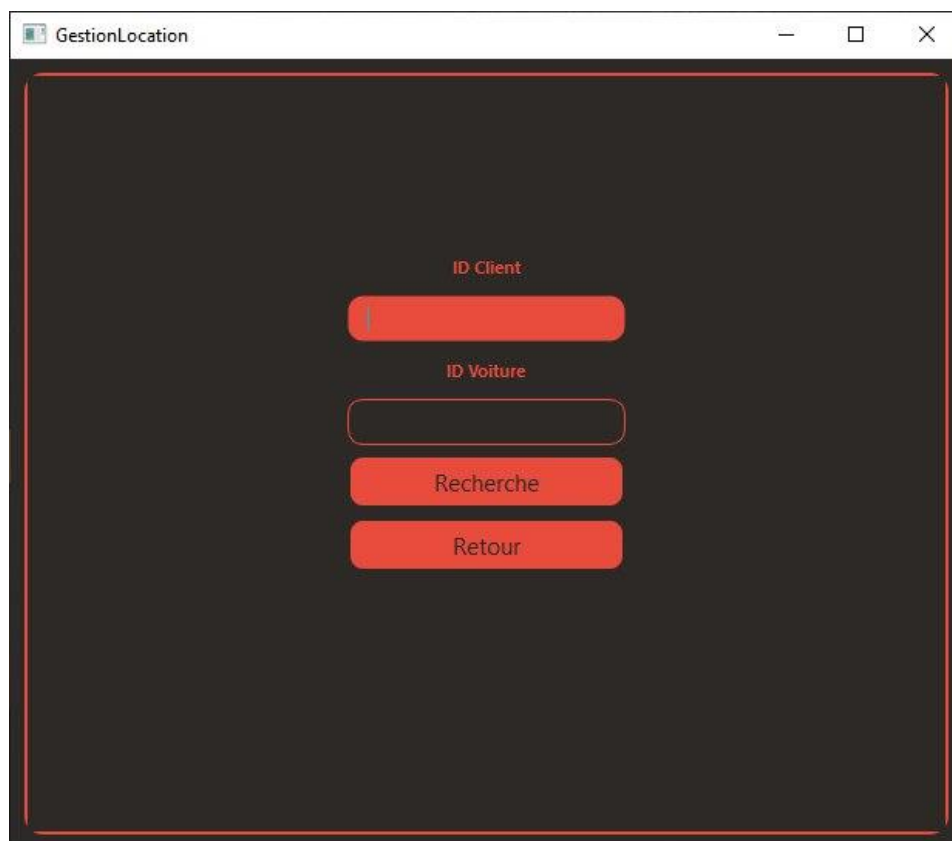
```
void GestionLocation::on_managecar_clicked()
```

```
void GestionLocation::on_manageclient_clicked()
```

- **Menu location d'une voiture**

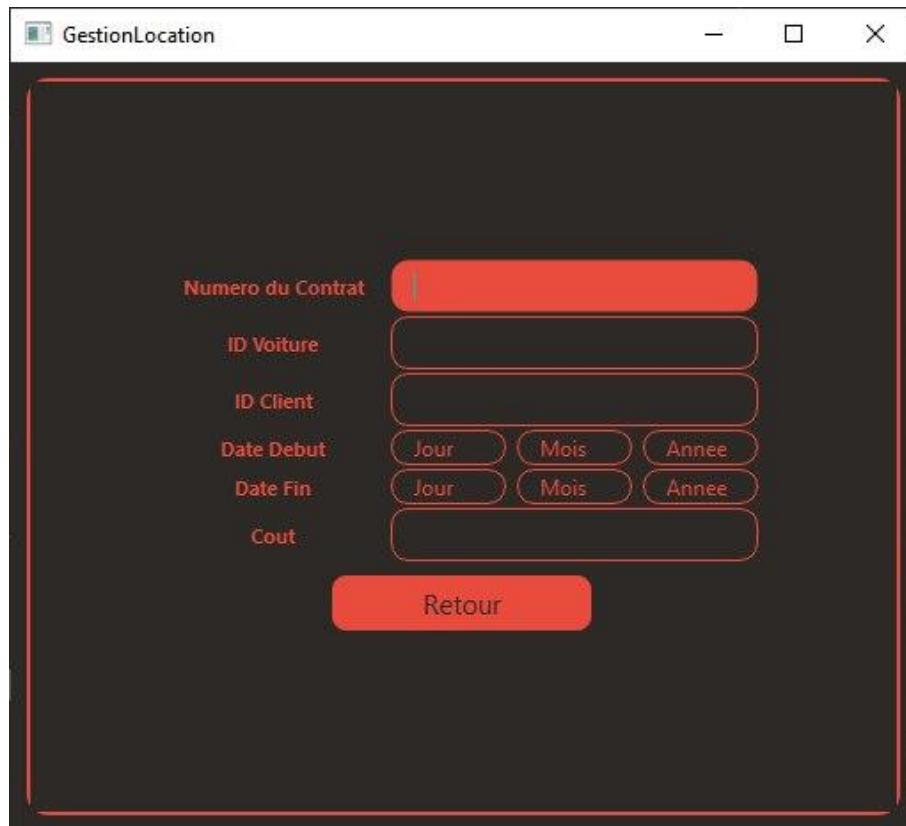


- **Visualiser contrat**



Cette page demande à l'utilisateur de donner l'ID de voiture et client, il cherche s'il existe et si elle trouve le contrat correspondant elle le transporte à la page suivante.

- **Affichage de contrat**



The screenshot shows a window titled "GestionLocation" with standard Windows window controls (minimize, maximize, close). The main content area has a dark background with a light gray border. It contains a form with the following fields and labels:

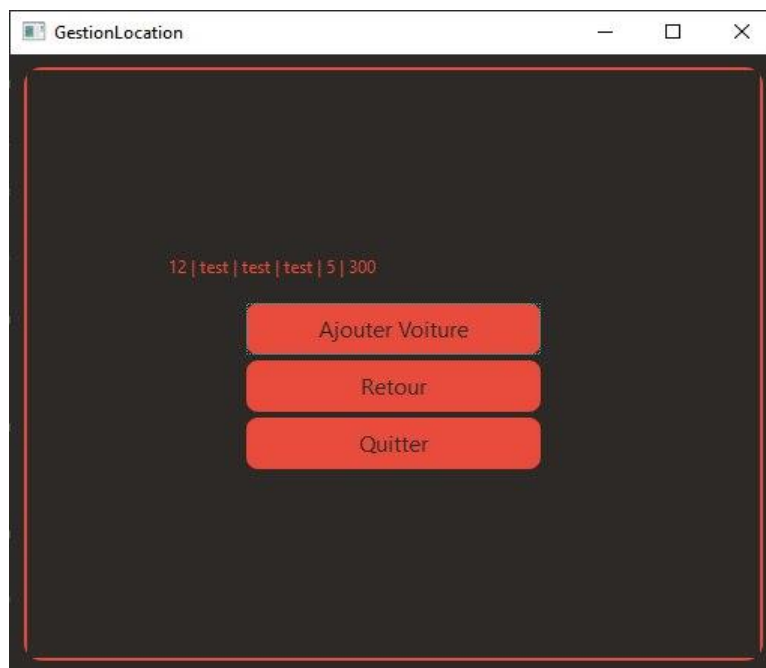
- Numero du Contrat**: A single-line text input field.
- ID Voiture**: A single-line text input field.
- ID Client**: A single-line text input field.
- Date Debut**: A date selection interface with three buttons labeled "Jour", "Mois", and "Annee".
- Date Fin**: A date selection interface with three buttons labeled "Jour", "Mois", and "Annee".
- Cout**: A single-line text input field.
- Retour**: A large, rounded rectangular button centered below the input fields.

Cette page affiche les informations de contrat de client qui est saisi dans la page précédente.

- **Menu Gestion des voitures**



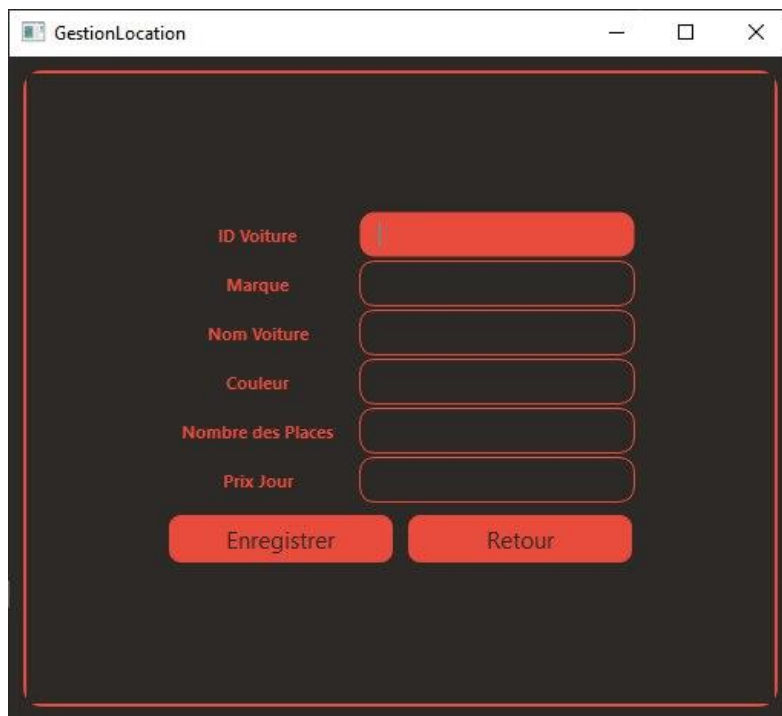
- **Liste des Voitures**



lorsque on clique sur le bouton « liste des voitures » dans la page précédente, il affiche une liste des voitures enregistrées à l'aide de la fonction :

Void GestionLocation ::on_listcar_clicked()

- **Ajouter voiture**



The screenshot shows a web application window titled 'GestionLocation'. Inside, there is a form with a dark background and red text and borders. The form contains the following fields and buttons:

- ID Voiture**: A red input field with a vertical cursor.
- Marque**: A red input field.
- Nom Voiture**: A red input field.
- Couleur**: A red input field.
- Nombre des Places**: A red input field.
- Prix Jour**: A red input field.
- Enregistrer**: A red button.
- Retour**: A red button.

Cette page permet de saisir les informations nécessaires pour ajouter une voiture à l'agence de location, et l'enregistrer dans le fichier « Voitures ».

- **Modifier Voiture**

Modifier Voiture

ID de Voiture pour modifier

Rechercher

ID Voiture	<input type="text" value="12"/>
Marque	<input type="text" value="fvve"/>
Nom Voiture	<input type="text" value="ewfew"/>
Couleur	<input type="text" value="dffg"/>
Nombre des Places	<input type="text" value="334"/>
Prix Jour	<input type="text" value="4444"/>

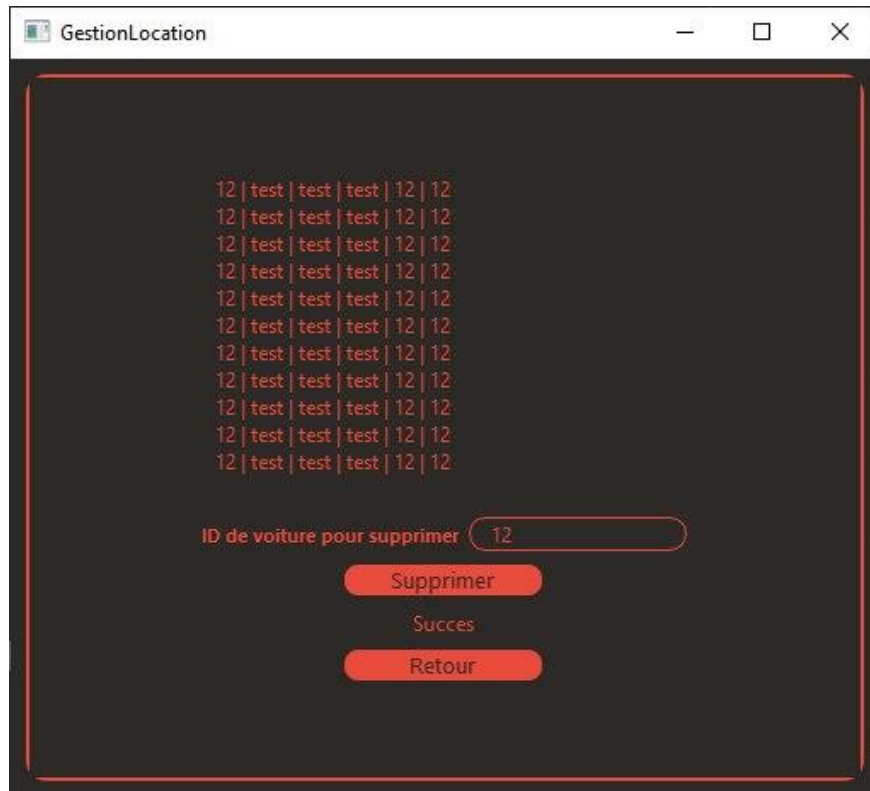
Enregistrer **Retour**

cette page vous demande de donner l'identifiant de la voiture que vous souhaitez modifier et lorsque vous appuyez sur rechercher, elle recherche le fichier pour voiture avec le même identifiant et si elle la trouve elle vous permet de modifier son contenu et si elle ne trouve pas cela, vous ne pouvez rien modifier. Et elle fait ça avec les fonctions suivantes

Void GestionLocation::on_savereturn_2_clicked()

Void GestionLocation ::on_recherche_clicked()

- **Supprimer voiture**



Cette page affiche tous les contenus de fichier Voitures et demande à l'utilisateur à saisir l'ID de votre pour le supprimer, et elle le fait en copiant le contenu du fichier dans un autre fichier sauf la ligne qui contient l'ID que l'utilisateur a saisi, et à la fin elle supprimer le premier fichier et renomme le deuxième Voitures, elle fait ça a l'aide des fonctions suivantes :

```
Void GestionLocation ::on_deletocar_2_clicked()
```

```
Void GestionLocation ::on_delcar_clicked()
```

III. Conclusion et Difficulté

1. Conclusion

Ce projet m'a montré l'importance d'organiser mon code et de nommer les variables sous différents noms sans rapport car il devient plus difficile de se souvenir de ce que fait chaque fonction si elles ont des noms similaires, et le rôle des commentaires pour expliquer ce que fait chaque fonction pour gagner du temps au lieu de perdre le temps de comprendre ce que j'ai fait précédemment, et comment structurer mon code, et aussi comment créer un projet ou un programme composé de plusieurs fichiers et comment lier chacun d'eux.

2. Difficulté

Lors de la création de ce projet, j'ai eu quelques difficultés à comprendre C++ et comment les classes, les objets et les méthodes se traduisent en c, et comment créer des fenêtres, des menus et des boutons évolutifs, aussi plus important encore comment programmer dans QT et créer des transitions entre les pages, et j'ai eu quelques problèmes pour optimiser mon code et mes fonctions.