



République Tunisienne  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université de Tunis El Manar  
École Nationale d'Ingénieurs de Tunis



## DEPARTEMENT GENIE ELECTRIQUE

# Mini Projet Programmation Orientée-Objets C++

## Application de gestion de réservation du vol

Réalisé par :

Kaddoudi Louy

Sabri Ben Naceur

Classe : 2AGE1

Encadré par :

Prénom NOM

Année universitaire 2020/2021

# Table des matières

<b>Table des figures</b>	<b>3</b>
<b>Liste des tableaux</b>	<b>4</b>
<b>1 Étude préliminaire du projet</b>	<b>6</b>
1.1 Introduction . . . . .	7
1.2 Présentation logicielle . . . . .	7
1.2.1 Qt Creator . . . . .	7
1.2.1.1 SQLite . . . . .	7
1.2.1.2 DB Browser for SQLite . . . . .	8
1.3 Étude préliminaire du projet . . . . .	8
1.3.1 Énoncé du projet . . . . .	8
1.3.2 Diagramme des cas d'utilisations . . . . .	9
1.3.2.1 Étude de base de données . . . . .	9
1.4 Conclusion . . . . .	11
<b>2 Réalisation de l'interface graphique du Notre Projet</b>	<b>12</b>
2.1 Introduction . . . . .	13
2.2 Description Générale de l'interface graphique . . . . .	13
2.3 Fenêtre d'accueil . . . . .	13
2.4 Fenêtre "Admin" . . . . .	14
2.5 Fenêtre "User" . . . . .	14
2.6 Conclusion . . . . .	14

# Table des figures

1.1	Logo du logiciel Qt Creator . . . . .	7
1.2	Logo du logiciel SQLite . . . . .	8
1.3	Logo du logiciel DB Browser (SQLite) . . . . .	8
1.4	Diagramme des cas d'utilisation) . . . . .	9
1.5	Schéma relationnel de base de données . . . . .	9
1.6	Tables de base de données . . . . .	10
1.7	Tables de base de données (PowerShell) . . . . .	11
2.1	Fonction principale . . . . .	13

## Liste des tableaux

# Introduction générale

Le texte du rapport peut être fourni en Français ou en Anglais. Les rapports seront imprimés sur format A4 avec des marges de 2,5 cm en haut, en bas, à gauche et à droite. Le titre du rapport, le(s) auteur(s), leur(s) encadrant(s) couvrent toute la largeur de la page de garde. Un résumé avec mots-clés est demandé. La table des matières, la liste des figures et des tableaux doivent être présentées. Une liste des abréviations peut être ajoutée selon le besoin après la liste des tableaux.

# **Chapitre 1**

## **Étude préliminaire du projet**

## 1.1 Introduction

Étant donné qu'il s'agit d'un nouvel outil de programmation C++, il est primordial de commencer par une étude générale portant sur les logiciels à utiliser pendant la réalisation de notre mini projet ainsi qu'une étude préliminaire de ce dernier. Le projet vise à réaliser d'une application graphique permettant de gérer les réservations des vols au niveau d'une agence de voyage.

## 1.2 Présentation logicielle

### 1.2.1 Qt Creator

Qt Creator est un environnement de développement intégré multiplateforme faisant partie du framework Qt. Il est donc orienté pour la programmation en C++. Il intègre directement dans l'interface un débogueur, un outil de création d'interfaces graphiques, des outils pour la publication de code sur Git et Mercurial ainsi que la documentation Qt. L'éditeur de texte intégré permet l'autocomplétion ainsi que la coloration syntaxique. Qt Creator utilise sous Linux le compilateur gcc. Il peut utiliser MinGW ou le compilateur de Visual Studio sous Windows. Qt Creator a été traduit en français par l'équipe Qt de Developpez.com.



FIGURE 1.1 – Logo du logiciel Qt Creator

#### 1.2.1.1 SQLite

SQLite est un système de base de données ou une bibliothèque proposant un moteur de base de données relationnelles. Il repose sur une écriture en C, un langage de programmation impératif, et sur une accessibilité via le langage SQL (Structured Query Language). SQLite présente la particularité d'être directement intégré aux programmes et dans l'application utilisant sa bibliothèque logicielle. Avec SQLite, la base de données est intégralement stockée dans un fichier indépendant du logiciel. Créé au début des années 2000 par D. Richard Hipp, SQLite propose un accès plus rapide aux données, mais aussi plus structuré et avec davantage de sécurité.



FIGURE 1.2 – Logo du logiciel SQLite

### 1.2.1.2 DB Browser for SQLite

DB Browser for SQLite est un outil open source graphique de haute qualité pour créer, concevoir et modifier des fichiers de base de données compatibles avec SQLite. L'interface utilisée est de type tableur familière.

- Les tables créées à l'aide de DB Browser à l'aide du système de base de données

SQLite, en plus de l'interface graphique offerte par Qt Creator, nous permettront de créer une application accessible à toute catégorie d'utilisateurs vu sa facilité d'utilisation. En plus, cette combinaison garantit une meilleure optimisation pour l'administrateur qui sera capable de contrôler et de manipuler facilement les places dans un vol.



FIGURE 1.3 – Logo du logiciel DB Browser (SQLite)

## 1.3 Étude préliminaire du projet

### 1.3.1 Énoncé du projet

L'objectif est de développer une application graphique permettant de gérer les réservations des vols au niveau d'une agence de voyage. L'utilisateur de cette application peut :

- Ajouter un vol.
- Rechercher et consulter des vols existant par date départ, destination etc.
- Ajouter un voyageur et lui réserver un vol
- Il peut aussi modifier ou annuler un vol.



### 1.3.2 Diagramme des cas d'utilisations

Le diagramme des cas d'utilisations nous permettons de mieux comprendre actions réalisées par le système en interaction avec les acteurs en vue d'une finalité. L'ensemble des cas d'utilisation permet ainsi de décrire les exigences fonctionnelles d'un système en adoptant le point de vue et le langage de l'utilisateur final.

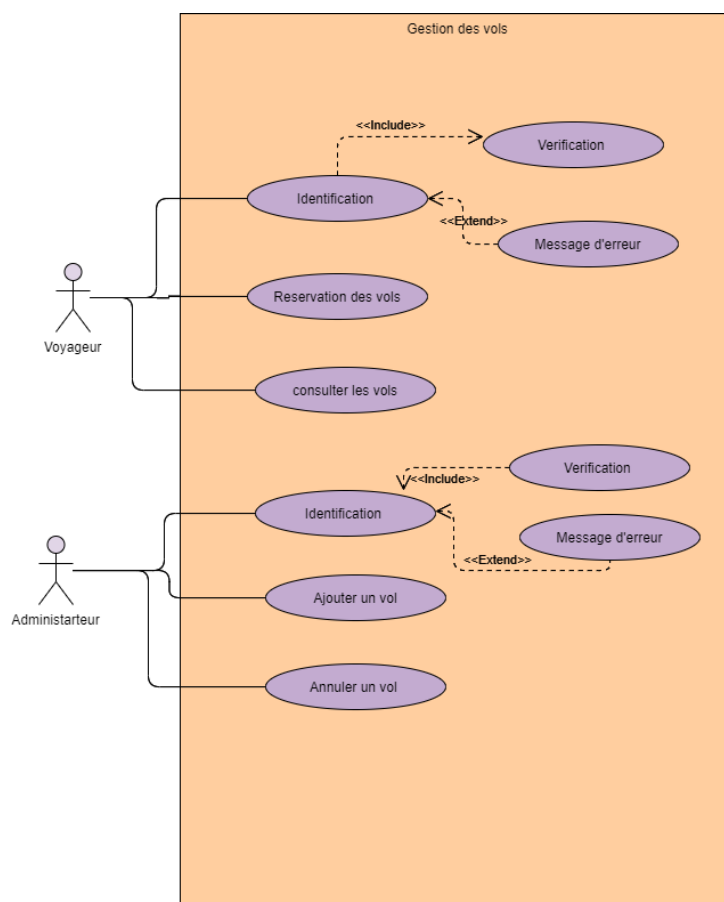


FIGURE 1.4 – Diagramme des cas d'utilisation)

#### 1.3.2.1 Étude de base de données

Pour faire une bonne conception de base de données nous commençons par la réalisation de schéma relationnel suivant :

FIGURE 1.5 – Schéma relationnel de base de données

Une fois le modèle choisi, il faut implémenter la base de données. Pour la création de notre base de données, nous avons utilisé le logiciel DB Browser (SQLite) et avons suivi les étapes suivantes :

1. création d'une nouvelle base de données Il est assez simple de créer une nouvelle base de données dans DB Browser (SQLite).

Notre base de données est créée sous le nom "GestionVol". Quand cette étape est terminée, la création des tables peut commencer.

2. création des tables Nous avons créé les tables suivantes :

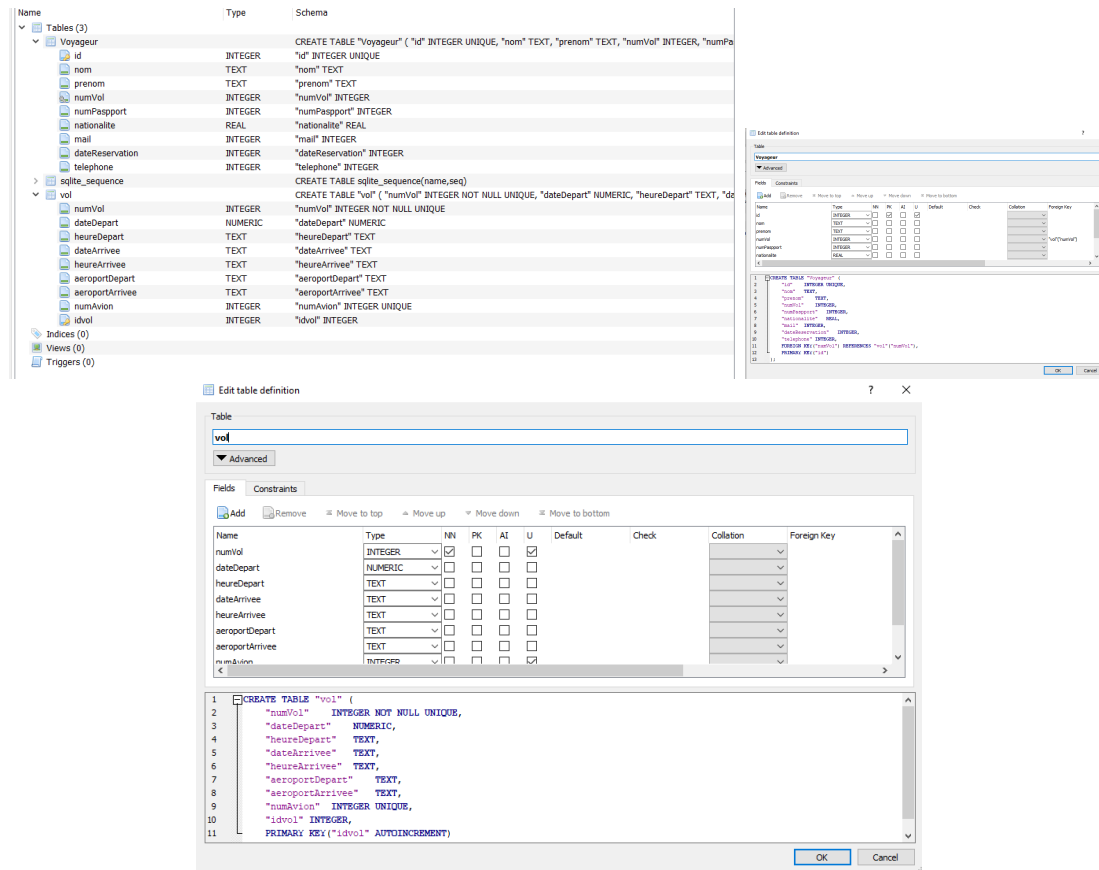


FIGURE 1.6 – Tables de base de données

```

sqlite> .schema
CREATE TABLE IF NOT EXISTS "Voyageur" (
  "id"      INTEGER UNIQUE,
  "nom"     TEXT,
  "prenom"  TEXT,
  "numVol"   INTEGER,
  "numPaspport" INTEGER,
  "nationalite" REAL,
  "mail"    INTEGER,
  "dateReservation" INTEGER,
  "telephone" INTEGER,
  PRIMARY KEY("id"),
  FOREIGN KEY("numVol") REFERENCES "vol"("numVol")
);
CREATE TABLE IF NOT EXISTS "vol" (
  "numVol"      INTEGER NOT NULL UNIQUE,
  "dateDepart"  NUMERIC,
  "heureDepart" TEXT,
  "dateArrivee" TEXT,
  "heureArrivee" TEXT,
  "aeroportDepart" TEXT,
  "aeroportArrivee" TEXT,
  "numAvion"    INTEGER UNIQUE,
  "idvol"       INTEGER,
  PRIMARY KEY("idvol" AUTOINCREMENT)
);

```

FIGURE 1.7 – Tables de base de données (PowerShell)

## 1.4 Conclusion

Ce chapitre nous a permis de se placer dans le cadre du projet, et ce, en connaissant les logiciels de travail ainsi que les tables et les interfaces graphiques à développer. Dans le chapitre suivant nous allons détailler la solution qu'on a proposé.

## **Chapitre 2**

# **Réalisation de l'interface graphique du Notre Projet**

## 2.1 Introduction

## 2.2 Description Générale de l'interface graphique

notre application peut être divisée en deux fenêtres :

- Fenêtre accueil.
- Fenêtre secondaire.

La première fenêtre se charge de l'étape de validation des coordonnées. La deuxième fenêtre est divisée en plusieurs onglets et porte des différences liées aux natures de l'utilisateur.

A noter que notre application distingue deux types d'utilisateur :

- Voyageurs.
- Administrateur.

## 2.3 Fenêtre d'accueil

Une fenêtre d'accueil dans le quel l'utilisateur est demandé de remplir deux champs :

- Le nom d'utilisateur.
- Le mot de passe.

L'utilisateur de l'application quelque soit un voyageur ou un administrateur doit remplir ses coordonnées afin de valider son accès. après l'étape de saisie l'utilisateur spécifie la nature de son compte : compte "user" ou compte "Admin" puis il valide ses coordonnées par appui sur un bouton de validation.

Pour notre fonction principale "main" elle est la suivante

```
#include "mainwindow.h"
#include <QApplication>

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    MainWindow w;
    w.show();

    return a.exec();
}
```

FIGURE 2.1 – Fonction principale

La fonction main construit un objet du type "MainWindow" qui correspond à notre fenêtre principale. À l'aide de la fonction "show()" cette dernière est affichée sur l'écran.

## **2.4 Fenêtre "Admin"**

## **2.5 Fenêtre "User"**

## **2.6 Conclusion**