



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 :

Louay Kaddoudi

Sabri Ben Naceur

Classe : 2AGE1

Encadré par :

Prénom NOM

Année universitaire 2020/2021

Table des matières

Table des figures	3
Liste des tableaux	4
1 Étude préliminaire du projet	6
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 Étude et planification du projet	9
1.3.2.1 Étude de base de données	9
1.4 Conclusion	11

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	Schéma relationnel de base de données	9
1.5	Tables de base de données	10
1.6	Tables de base de données (PowerShell)	10

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 son parking.



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 (numéro vol, date et heure départ, date et heure d'arrivée, Aéroport départ, aéroport d'arrivée, numéro avion).
- 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 Étude et planification du projet

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 :

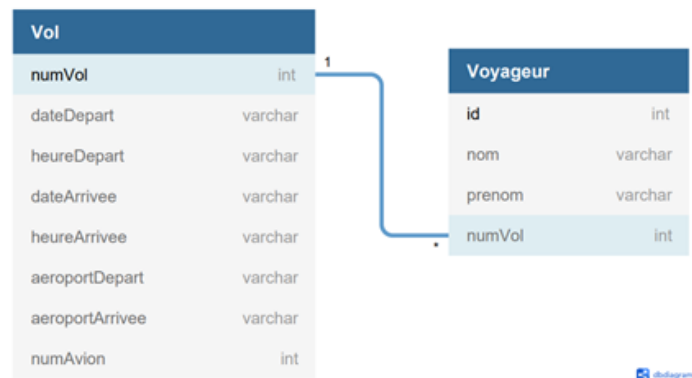


FIGURE 1.4 – Schéma relationnel de base de données

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 :

The figure consists of three screenshots from a database management tool. The top-left screenshot shows a tree view of the database schema with tables 'Vol' and 'Voyageur'. The top-right screenshot shows the 'Vol' table definition with columns: numVol (INTEGER, NOT NULL, UNIQUE, PRIMARY KEY), dateDepart (TEXT, NOT NULL), heureDepart (TEXT, NOT NULL), dateArrivee (TEXT, NOT NULL), heureArrivee (TEXT, NOT NULL), aeroportDepart (TEXT, NOT NULL), aeroportArrivee (TEXT, NOT NULL), and numAvion (INTEGER, NOT NULL, UNIQUE). The bottom screenshot shows the 'Voyageur' table definition with columns: id (INTEGER, NOT NULL, PRIMARY KEY, AUTOINCREMENT, UNIQUE), nom (TEXT, NOT NULL), prenom (TEXT, NOT NULL), and numVol (INTEGER, NOT NULL, FOREIGN KEY REFERENCES 'Vol'('numVol')).

Name	Type	NN	PK	AI	U
numVol	INTEGER	✓	✓		✓
dateDepart	TEXT	✓			
heureDepart	TEXT	✓			
dateArrivee	TEXT	✓			
heureArrivee	TEXT	✓			
aeroportDepart	TEXT	✓			
aeroportArrivee	TEXT	✓			
numAvion	INTEGER	✓			✓

Name	Type	NN	PK	AI	U	Default
id	INTEGER	✓	✓	✓	✓	
nom	TEXT	✓				
prenom	TEXT	✓				
numVol	INTEGER	✓				

FIGURE 1.5 – Tables de base de données

```

Select Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Here We Go Again> cd C:\MP_V4\DataBase
PS C:\MP_V4\DataBase> sqlite3 GestionVol.db
SQLite version 3.34.1 2021-01-20 14:10:07
Enter ".help" for usage hints.
sqlite> .schema
CREATE TABLE sqlite_sequence(name,seq);
CREATE TABLE IF NOT EXISTS "Vol" (
  "numVol"      INTEGER NOT NULL UNIQUE,
  "dateDepart"  TEXT NOT NULL,
  "heureDepart" TEXT NOT NULL,
  "dateArrivee" TEXT NOT NULL,
  "heureArrivee" TEXT NOT NULL,
  "aeroportDepart" TEXT NOT NULL,
  "aeroportArrivee" TEXT NOT NULL,
  "numAvion"    INTEGER NOT NULL UNIQUE,
  PRIMARY KEY("numVol")
);
CREATE TABLE IF NOT EXISTS "Voyageur" (
  "id"      INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,
  "nom"     TEXT NOT NULL,
  "prenom"  TEXT NOT NULL,
  "numVol"  INTEGER NOT NULL,
  FOREIGN KEY("numVol") REFERENCES "Vol"("numVol")
);
sqlite> .

```

FIGURE 1.6 – 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.