**University Timetable Scheduler**

**Introduction**

**1.1 Overview of the project**

University Timetable Scheduler is a user-friendly Qt based C++ GUI Application which helps the timetable coordinators or staff members to schedule and manage various class or laboratory daily schedules. The main agenda of this application is to generate teachers or class/laboratory schedules for an educational institution.

The application uses Qt as a front end for interacting with the user , Object Oriented C++ routines for managing user interfaces and for scheduling, a database engine popularly know as SQLite for storing and retrieving the user data or the generated schedules and some UNIX Shell commands to pre-process the given user input. It was developed by using the Git Version Control techniques to integrate code written by multiple developers.

**C++ Concepts**

**1. Inheritence**

In object-oriented programming, **inheritance** is when an object or class is based on another object or class, using the same implementation (inheriting from an object or class) or specifying a new implementation to maintain the same behavior (realizing an interface). Such an inherited class is called a **subclass** of its parent class or super class.

**2. Encapsulation**

In object-oriented programming,encapsulation is a mechanism of binding the data, and the functions together in a class and use them by creating an object of that class.

**3. Data Abstraction**

Data abstraction refers to, providing only essential information to the outside world and hiding their background details, i.e., to represent the needed information in program without presenting the implementation details. Data abstraction is a programming (and design) technique that relies on the separation of interface and implementation.

**Qt Application**

**Qt** is a cross-platform application framework that is widely used for developing application software that can be run on various software and hardware platforms with little or no change in the underlying codebase, while still being a native application with native capabilities and speed. Qt is currently being developed both by The Qt Company, a company listed on the Nasdaq Helsinki Stock Exchange and the Qt Project under open-source governance, involving individual developers and firms working to advance Qt. Qt is available with both commercial and open source GPL 2.0, GPL 3.0, and LGPL 3.0 licenses

Qt is used mainly for developing application software with graphical user interfaces (GUIs).Qt uses standard C++ with extensions including signals and slots that simplify handling of events, and this helps in development of both GUI and server applications which receive their own set of event information and should process them accordingly. Qt supports many compilers, including the GCC C++ compiler and the Visual Studio suite.

**SQLite Database Engine**

SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purpose, commercial or private. SQLite is the most widely deployed database in the world with more applications than we can count, including several high-profile projects.

SQLite is an embedded SQL database engine. Unlike most other SQL databases, SQLite does not have a separate server process. SQLite reads and writes directly to ordinary disk files. A complete SQL database with multiple tables, indices, triggers, and views, is contained in a single disk file. The database file format is cross-platform - you can freely copy a database between 32-bit and 64-bit systems or between big-endian and little-endian architectures. These features make SQLite a popular choice as an Application File Format.

**Git Version Control**

**System Requirements specification**

**Hardware Requirements:**

**Software Requirements:**

Operating System : Linux

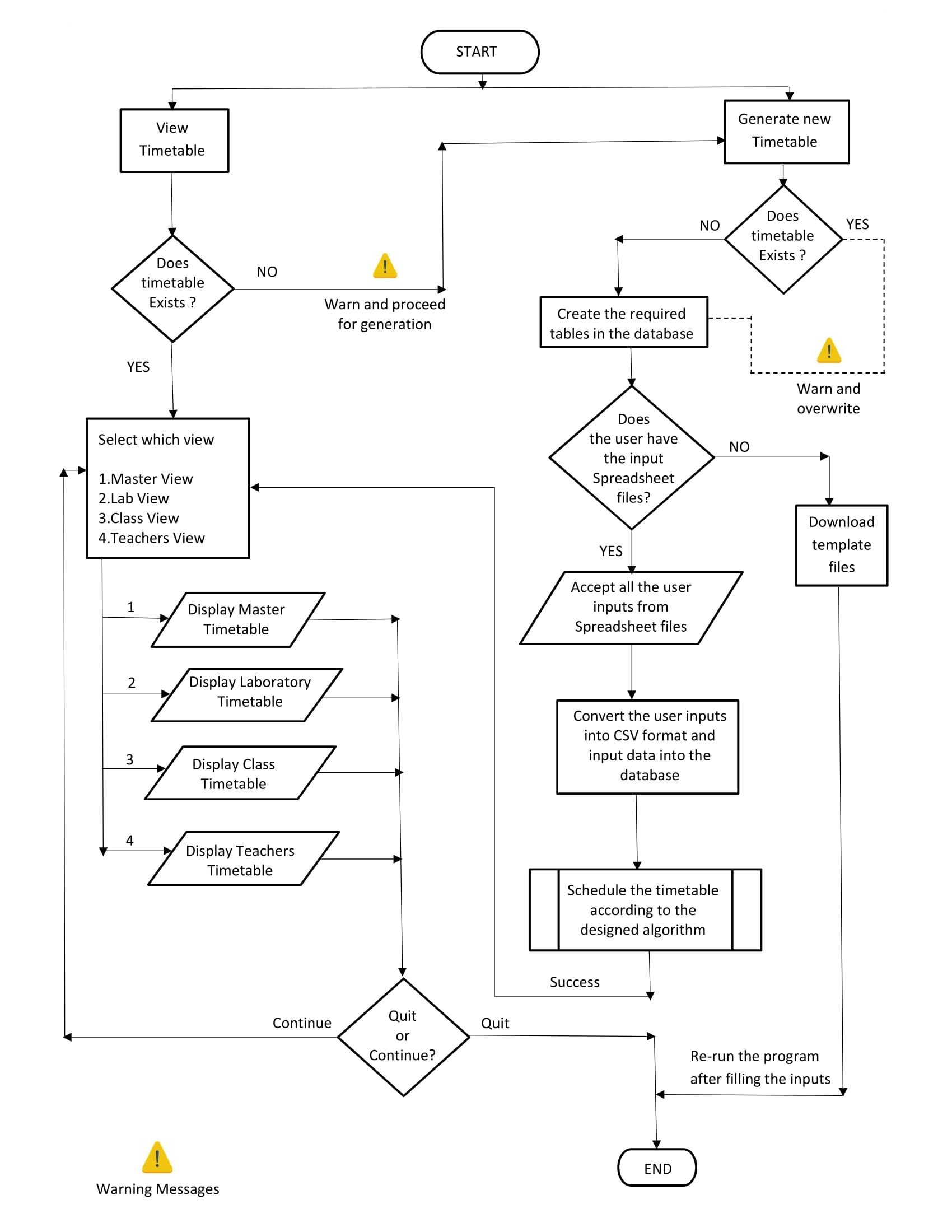
Qt Libraries : Qt5Core , Qt5gui , Qt5Sql , Qt5widgets

Compiler : GNU C++ compiler

Debugger : GNU Debugger (optional)

Other Applications : LibreOffice , Java Runtime (optional)

**Architectural Design**



**References**

* Programming Knowledge Video series

<https://www.youtube.com/user/ProgrammingKnowledge>

* Qt online Documentation

<https://doc.qt.io/qt-5>

* SQLite Documentation

<https://www.sqlite.org/docs.html>

* Tutorials point : SQLite

<http://www.tutorialspoint.com/sqlite>

* “The Book of Qt 4 The Art of Building Qt applications” - Daniel Molkentin
* “Advanced Qt Programming” - Mark Summerfield
* StackOverflow

[http://stackoverflow.com](http://stackoverflow.com/)

* C++ Documentation

<http://www.cplusplus.com/doc>

* Qt forums

[https://forum.qt.io](https://forum.qt.io/)

* SQLite

[https://www.sqlite.org](https://www.sqlite.org/)

* GIT (Version Control)

[https://git-scm.com](https://git-scm.com/)