EEP702-Software Lab
Assignment 8 : Basics of Designing Applicaton in Qt

Harshit Kumar Gupta
2013EET2369

Computer Technology
Department Of Electrical Engineering
IIT DELHI

April 15, 2014

Contents

1 PROBLEM STATEMENT 2

2 ABSTRACT 3

3 INTRODUCTION 4

4 SPECIFICATIONS AND ASSUMPTIONS 6

5 LOGIC USED/METHODOLOGY 7

6 FLOWCHART 8

7 RESULTS AND CONCLUSIONS 9

8 OUTPUTS OF PROGRAM 10

Chapter 1

PROBLEM STATEMENT

Develop a Library Management System:

1. Design a user interface(GUI) in Qt.The user will be asked to enter a number in numeric (say, 55) and will be provided a button named "Convert to text".On pressing this button, the entered number should be displayed in words (fifty five).

2. Add one more feature to the above GUI, which will enable user to enter a number in text (fifty five). Add one button named "Convert to Number", on pressing which the entered number in words will be shown in numeric digits (55).

3. Add one more feature to the above GUI, which will enable user to enter a number in text (fifty five) and display a comma separated value as it is typed in (note that the internal representation will be without commas of course).

Use Qt to design the Application.

Chapter 2

ABSTRACT

The intention is to learn designing in Qt.

1. Qt is a cross-platform application framework that is widely used for developing application software with a graphical user interface (GUI) (in which cases Qt is classified as a widget toolkit), and also used for developing non-GUI programs such as command-line tools and consoles for servers.

2. Qt uses standard C++ but makes extensive use of a special code generator (called the Meta Object Compiler, or moc) together with several macros to enrich the language. Qt can also be used in several other programming languages via language bindings. It runs on the major desktop platforms and some of the mobile platforms. It has extensive internationalization support. Non-GUI features include SQL database access, XML parsing, thread management, network support, and a unified cross-platform application programming interface (API) for file handling.

Chapter 3

INTRODUCTION

Qt, when it was first released, relied on a few key concepts:

1. Complete abstraction of the GUI – When first released, Qt used its own paint engine and controls, emulating the look of the different platforms it runs on when it drew its widgets. This made the porting work easier because very few classes in Qt depended really on the target platform; however, this occasionally led to slight discrepancies where that emulation was imperfect.

   Recent versions of Qt use the native style APIs of the different platforms, on platforms that have a native widget set, to query metrics and draw most controls, and do not suffer from such issues as much.[53]

   On some platforms (such as MeeGo and KDE) Qt is the native API.

   Some other portable graphical toolkits have made different design decisions; for example, wxWidgets uses the toolkits of the target platform for its implementations.

2. Signals and slots - a language construct introduced in Qt for communication between objects[54] which makes it easy to implement the Observer pattern while avoiding boilerplate code. The concept is that GUI widgets can send signals containing event information which can be received by other controls using special functions known as slots.

3. Metaobject compiler - The metaobject compiler, termed moc, is a tool that is run on the sources of a Qt program. It interprets certain macros from the C++ code as annotations, and uses them to generate added C++ code with Meta Information about the classes used in the program. This meta information is used by Qt to provide programming features not available natively in C++: signals and slots, introspection and asynchronous function calls.

Chapter 4

SPECIFICATIONS AND ASSUMPTIONS

Specifications

1. Design a user interface(GUI) in Qt.
2. User should be asked to enter either number or text to convert it to text and numbers correspondingly.
3. Converted number should be formated in indian number format Ex: 11,12,256.

Assumptions

1. If user enters wrong number output displays as zero.
2. If user enters wrong text output displays as 0.
3. If user entered number is negative or out of range, output displays as unsupported.

Chapter 5

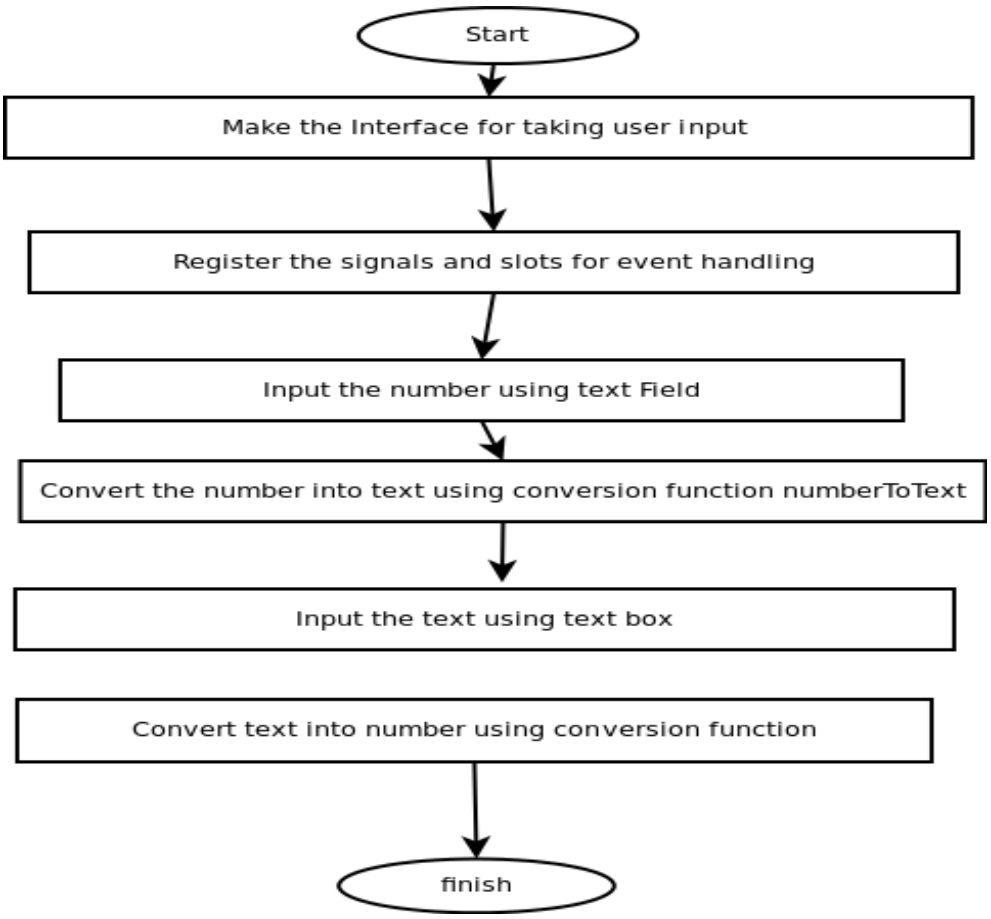
LOGIC USED/METHODOLOGY

The methodology that is used for developing the program is defined below:

1. Storing all strings for digits in arrays.
2. Deviding the number with corresponding weight and concating to global variable result the words according to digits and weight.
3. Implemented above functionality in a function and calling the function when user clicks the button.
4. Text to numbers implemented using python and calling the python script from Qt using QtProcess.
5. Formating the number into indian number format using string stream.

Chapter 6

FLOWCHART



Chapter 7

RESULTS AND CONCLUSIONS

1. Could convert number to text.
2. Could convert text to number.
3. Could format number to indian number format.
4. We can use espeak command and QtProcess type to produce voice for given text.

Chapter 8

OUTPUTS OF PROGRAM

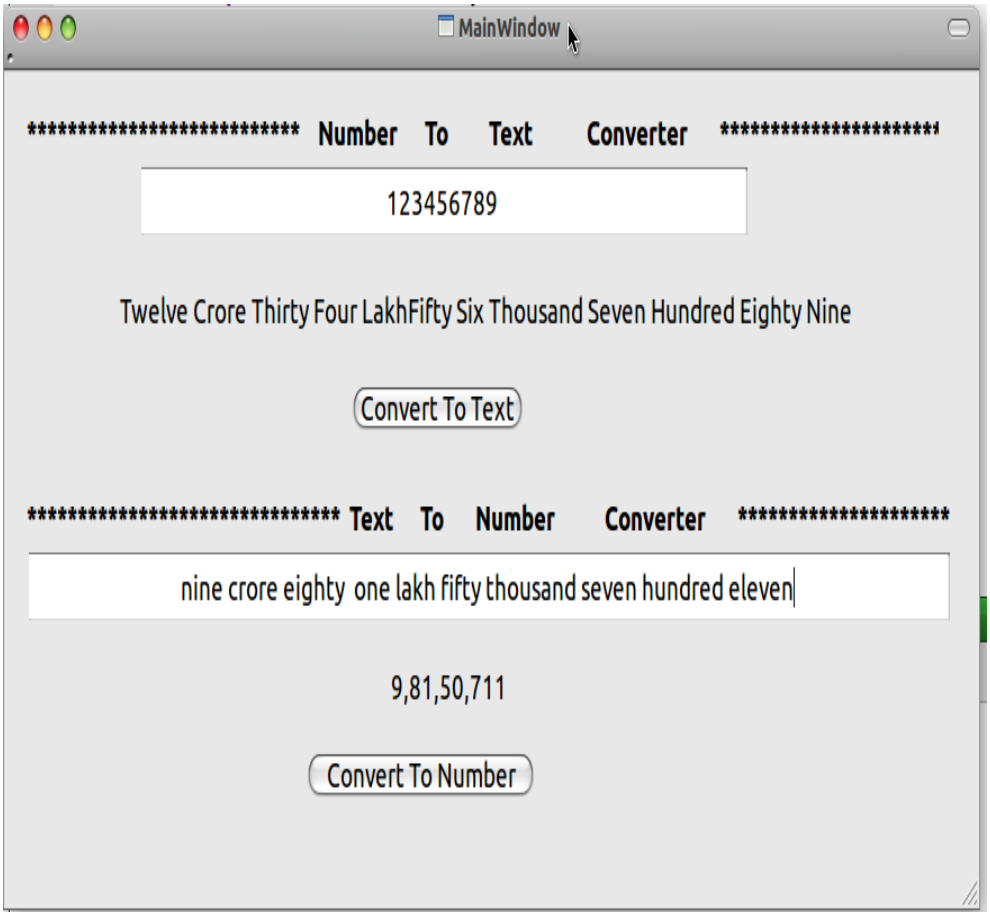


Figure 2: Application GUI Screen with Results .