

Relazione Progetto qCharts

Corso di laurea triennale in informatica – Università di Padova

Progetto del corso di programmazione ad oggetti A.s. 2012/2013



Introduzione:

qCharts è un programma sviluppato in C++/Qt che permette all'utente di creare, archiviare, modificare e visualizzare charts, ovvero diagrammi di dati.

Progettazione e sviluppo:

La scelta progettuale è stata basata principalmente sull'idea di permettere all'utente di inserire qualsiasi tipo di dato, numerico e non in modo facile e chiaro. Sarà poi compito di qCharts interpretare i dati e visualizzarli nel miglior modo possibile.

La base dati su cui si basa qCharts è la classe **DataTable**, derivata da *QWidget* contiene al suo interno un puntatore ad un oggetto *QTableWidget* che altro non è che un widget tabella. *DataTable* contiene inoltre diverse altre informazioni riguardanti il chart in oggetto quali il titolo del grafico, le etichette degli assi, un puntatore al vettore di colori *QColor* per distinguere le varie serie di dati, oltre che a diversi metodi per gestire l'inserimento/rimozione delle righe e delle colonne della tabella.

Grazie a questa scelta è stato facile implementare anche la fase di salvataggio/recupero dei dati. qCharts infatti permette di salvare il proprio lavoro salvando tutti i dati del grafico e tutte le altre informazioni quali titolo del grafico, titolo delle serie, colore delle serie, impostazioni varie in un file *.qch che utilizza lo standard XML per formattare il testo. Grazie alle librerie Qt *QxmlStreamReader* e *QxmlStreamWriter* è stata realizzata la classe **FileManager** contenente solamente metodi statici per salvare ed aprire la configurazione di qCharts.

Per quanto riguarda le operazioni di visualizzazione dei charts è stata creata la classe **DrawChart** che è un *QWidget* utilizzato come canvas, una lavagna dove disegnare sfruttando il relativo *paint event*.

Ogni classe che descrive una certa tipologia di grafico in qCharts deriva dalla classe **AbstractChart**.

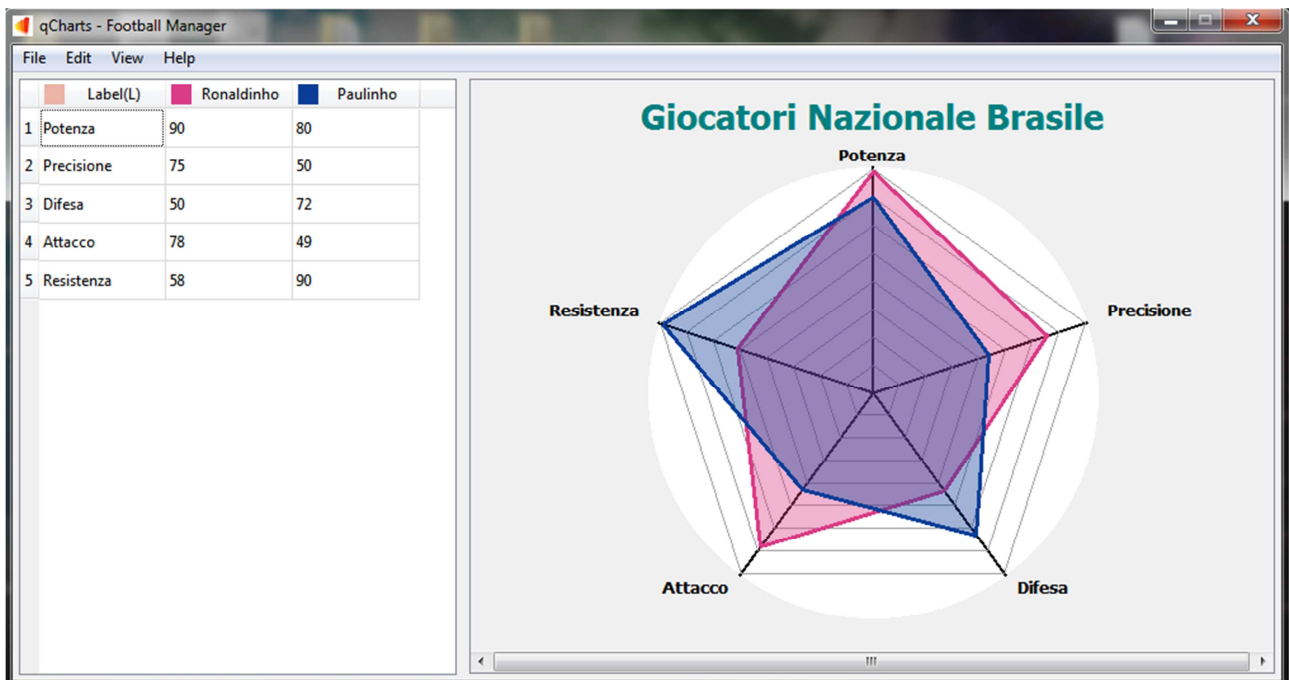
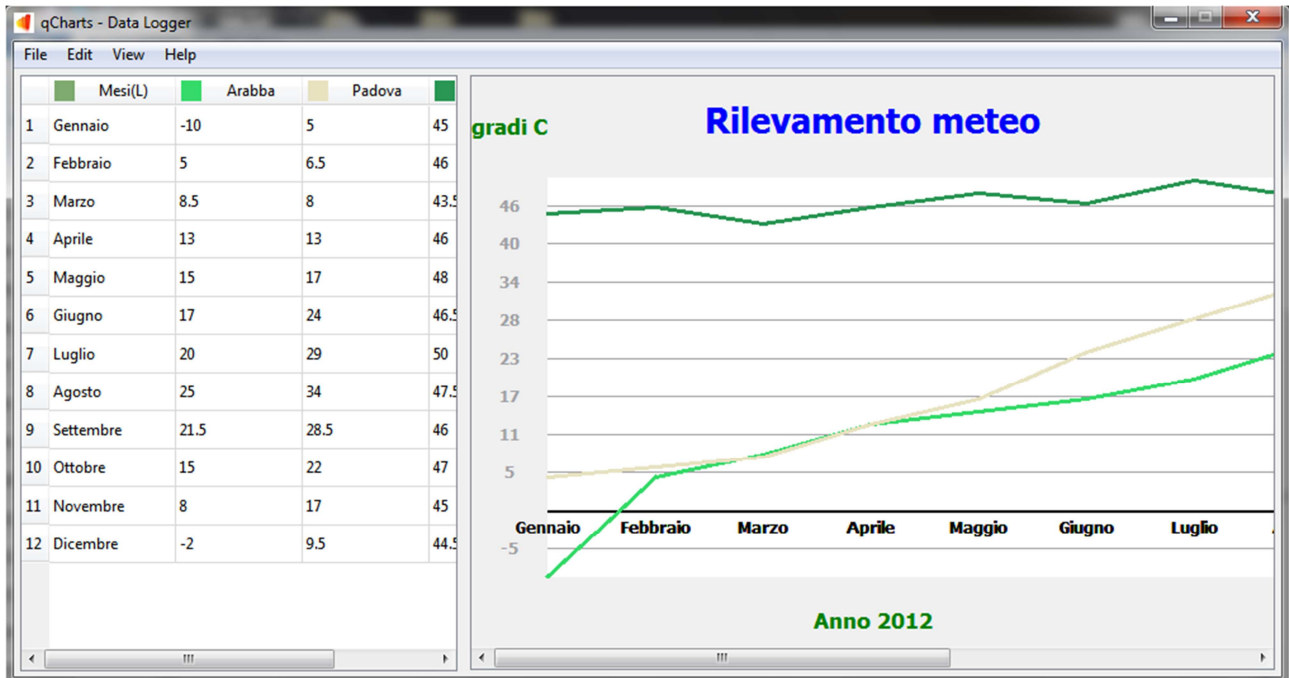
AbstractChart è una classe astratta contenente il metodo virtuale puro *draw()* che sarà implementato ad hoc in ogni classe che deriva da essa. In questo modo sarà possibile in futuro aggiungere altre classi di dati riguardanti nuovi tipi di charts in modo facile e veloce.

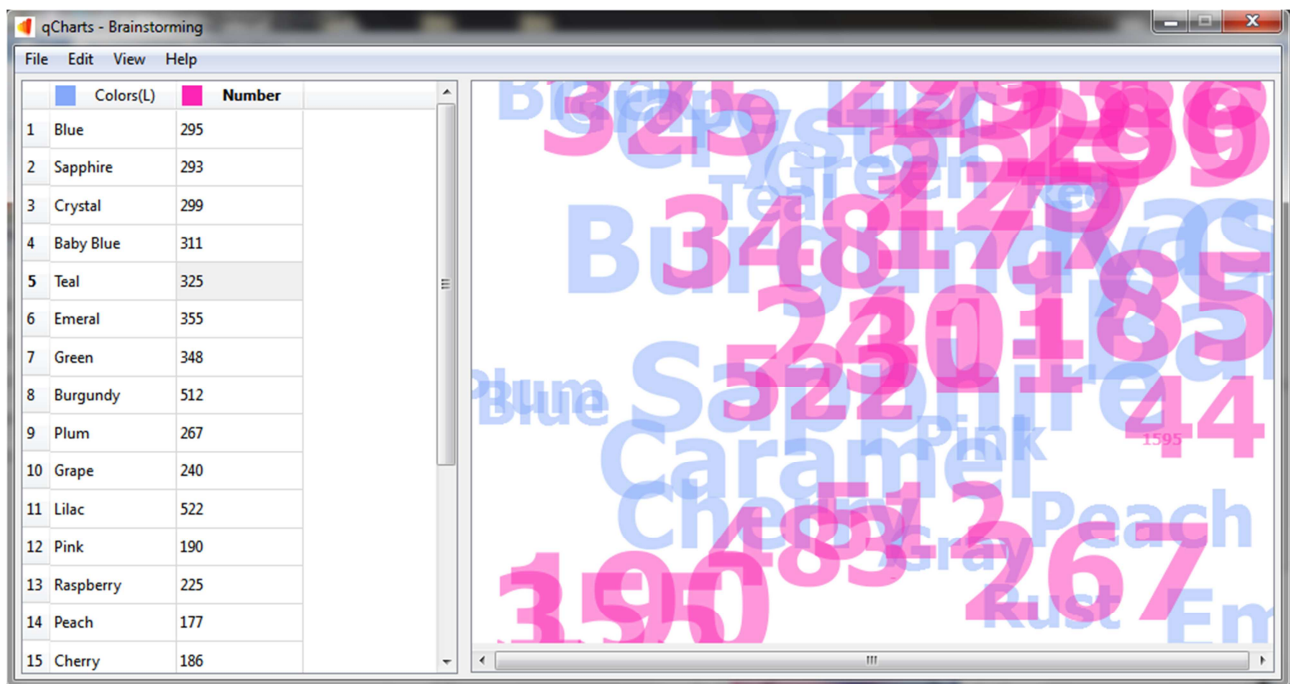
In questo progetto sono state implementate le classi **LineChart**, **BarChart**, **CaosChart** e **RadarChart** che concretizzano *AbstractChart* tramite l'implementazione del metodo virtuale puro *draw()*. Avendo accesso alla base dati tramite il puntatore alla tabella, ogni classe può interpretare i dati a suo piacimento e visualizzarli nel modo più opportuno.

Infine la classe che gestisce tutta l'interfaccia grafica è la classe **Layout** che si preoccupa di fornire all'utente tutte le funzionalità di qCharts tramite il menù ed i shortcuts.

Esempi di utilizzo:

I seguenti esempi sono disponibili anche dal menù di qCharts **File->New**:





Informazioni:

Il progetto è stato sviluppato in ambiente Windows (Windows7 PRO) utilizzando le librerie QT4.8.4, il compilatore GCC4.4.0 e l'ambiente di sviluppo QtCreator 2.6.2.

E' possibile compilare il progetto semplicemente generando un Makefile appropriato con il comando *qmake -project*, *qmake -makefile* e successivamente lanciare il comando *make*.

Files consegnati:

RELAZIONE: relazione.pdf

HEADERS: AbstractChart.h BarChart.h CaosChart.h DataTable.h DrawChart.h FileManager.h Layout.h LineChart.h RadarChart.h

SOURCES: AbstractChart.cpp BarChart.cpp CaosChart.cpp DataTable.cpp DrawChart.cpp FileManager.cpp Layout.cpp LineChart.cpp main.cpp RadarChart.cpp

RESOURCES: qCharts.qrc

IMAGES(cartella "immagini"): menuArrow.png qChartsIcona.png qChartsLogo.png

ESEMPIO FILE XML(cartella "esempio file xml"): Batman.qch