

Relazione sul progetto di
Programmazione ad oggetti
a.a. 2015/2016

Calendar for your buisness

Zanon Edoardo
Matricola : 1097467

Ambiente di sviluppo

- Compilatore: GCC 4.9.1, 64 bit
- Versione Qt Creator: 3.1.1
- Versione Qt: 5.3.0
- Sistema Operativo: Windows 10 / Ubuntu 14.10

Materiale consegnato

La cartella consegnata nominata: "ZanonEdoardoProgetto" contiene i seguenti elementi:

- Source : cartella nella quale sono presenti tutti i file sorgenti, .h e cpp per la compilazione del progetto;
- Relazione.pdf : file in pdf contenente la suddetta relazione;

Compilazione ed esecuzione

Opzione 1:

Aprire da qt Creator il progetto, selezionando il compilatore preferito, in seguito eseguire il compile, e l'esecuzione.

Per la corretta compilazione, è necessario usare il file progetto.pro, che oltre a fare gli include necessari, tiene conto anche del posizionamento delle varie cartelle che ho utilizzato per "ordinare i vari file".

1.Introduzione

Il progetto è stato pensato come evoluzione di un calendario, rivolto principalmente a liberi professionisti e lavoratori autonomi.

Esso si suddivide in due componenti:

- La gestione degli impegni, mediante il salvataggio degli impegni giornalieri
- La gestione della clientela, associando l'impegno con un determinato cliente, esso verrà considerato come una prestazione a favore di quel cliente, sarà quindi possibile calcolare il conto, da una certa data ad un'altra di quel cliente.

Questo progetto si propone come una soluzione basilare alla calendarizzazione e conteggio delle prestazioni per una moltitudine di clienti.

2.Parte logica

La gerarchia di classi pensata per questo progetto è abbastanza basilare, e si suddivide principalmente in due componenti:

- Persone: è una piccola gerarchia di due classi:
 - Persona: rappresenta una persona con nome e dati personali
 - Cliente: è un sott'oggetto di persona e rappresenta un cliente, ha quindi
- Impegni: suddivisa in 3 classi
 - Impegno: classe astratta che funge da interfaccia per tutti gli impegni
 - Personale: è un tipo di impegno che non ha il campo cliente, e che quindi riguarda un impegno al di fuori dall'ambito lavorativo
 - impegnoLavorativo: è un'impegno che possiede il campo cliente, ma è anchessa astratta e comprende sue sottoclassi:
 - Tempo dedicato: è un'impegno che prevede il calcolo come prezzo base, e corrisponde ad una prestazione "Standard per quel cliente"
 - Appuntamento: è un'impegno che prevede un surplus durante il calcolo dell'importo, che rappresenta il tempo dedicato ad un cliente "face-to-face"

Abbiamo anche due contenitori:

- Lista: consiste in una lista ad interimento in testa, di Impegno*. Lo scopo di questo contenitore è di "tenere in ordine" gli impegni di un singolo giorno.
- VectorDay: è un vector di Lista che si comporta in maniera singolare, ovvero ogni cella rappresenta un giorno, il vettore istanzia inizialmente 366 celle, una per ogni giorno dell'anno, a partire dal giorno di installazione. Con un calcolo approssimativo, se il software viene utilizzato, mediamente ci dovrebbe essere almeno un impegno al giorno e quindi lo spazio sprecato è influente rispetto il vantaggio di poter avere accesso casuale basandosi sulla posizione e correlandola al giorno. E' possibile quindi ottenere la lista di impegno di un giorno con accesso casuale, facendo una semplice calcolo sulle date.

3. Controller

Il controller è formato da due classi:

- ImpegniController: è il controller incaricato della gestione del vectorDay e delle liste, esso interagisce tra la mainWindow e la struttura dati, offre le funzionalità di inserimento e visualizzazione.
- ClientiController: gestisce una struttura dati QVector<Clienti> e la seconda finestra (GestioneClienti), offre anche funzionalità per ImpegniController.

I controller gestiscono anche le funzioni di salvataggio su file dei dati.

4. Parte Grafica

La parte grafica si suddivide in due finestre che gestiscono separatamente i due nuclei del programma:

- mainWindow: gestisce la parte della calendarizzazione (inserimento-cancellazione e visualizzazione degli impegni)
- GestioneCliente: gestisce inserimento e cancellazione dei clienti, oltre che il calcolo tra due date del conto per quel cliente.

Informazioni per l'utilizzo.

Mentre per l'inserimento ci sono degli appositi campi, per la cancellazione è sufficiente cliccare nella listView l'impegno o il cliente nel quale si desidera agire, e poi preme il tasto di cancellazione.

Attenzione: Per l'inserimento di un Impegno, se diverso da personale è obbligatorio selezionare un cliente già inserito.