

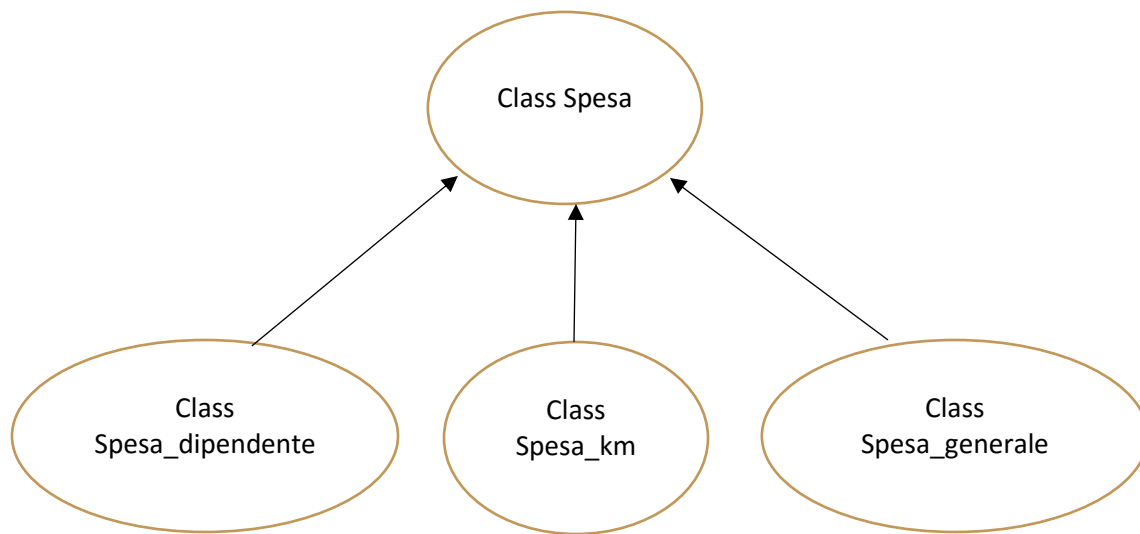
EasyJob

Preliminare:

EasyJob® è un gestionale diverso dal solito, EasyJob® tenta di facilitare la vita al lavoratore. Questa applicazione è un raccoglitore di dati giornalieri della vostra azienda personale, Spese e Entrate. EasyJob® è volta, soprattutto, ai lavoratori con partita Iva remunerati “ad ore”, tornati a casa, si concedono due minuti per memorizzare le spese e le entrate della loro giornata lavorativa così, una volta arrivato il momento di calcolare i soldi a debito e a credito, non devono andare a sfogliare decine di carte.

Descrizione:

EasyJob® utilizza due gerarchie di classe, una per le spese ed l’ altra per le entrate,,entrambe molto estendibili nel futuro.



L’ organigramma soprastante è quello per le spese, la classe “Spesa” è una classe base astratta con 3 metodi virtuali puri; il motivo per cui l’ ho resa virtuale è perché una Spesa, in termini generali, in se non esiste ma ogni spesa accomuna molti fattori, infatti ogni “Spesa” ha come campi dati privati la data di riferimento della data spesa e l’ essere o non essere stata pagata.

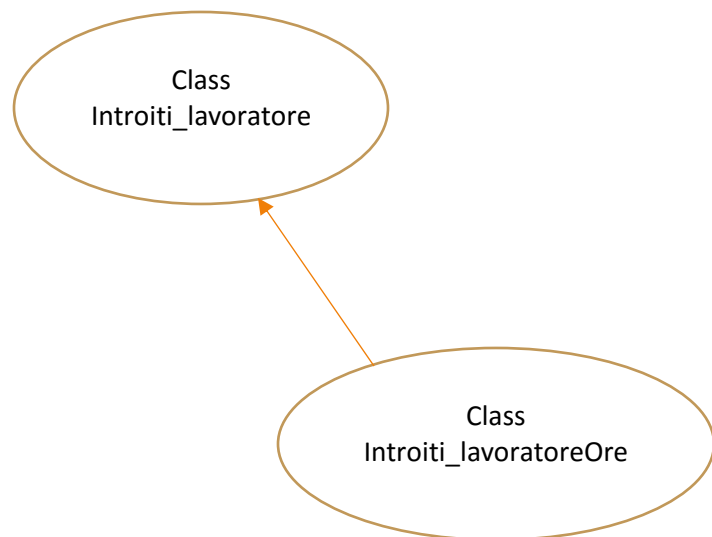
Le classi “Spesa_dipendente”, “Spesa_km”, “Spesa_generale” ereditano pubblicamente da “Spesa” e implementano, ovviamente, tutti i metodi virtuali puri della base.

- La classe “Spesa_km” incapsula la spesa che si sostiene, in termini di carburante, per raggiungere un determinato posto ed ogni “Spesa_km” deve contenere la distanza percorsa, prezzo del carburante, km/l del mezzo ed il luogo raggiunto.

- La classe "Spesa_dipendente" incapsula la spesa che si sostiene quando una persona ti aiuta in quella giornata (cosa molto frequente quando si è pagati ad ora). Per incapsulare meglio i dati privati di questa classe ho implementato una classe estera a questa chiamata durata che incapsula la durata della giornata ed essa ha come campi privati 3 Qtime(inizio, fine e durata pranzo), questa mi ha aiutato molto alla realizzazione della classe. Ogni classe "Spesa_dipendente" deve contenere il nome del dipendente, il luogo del lavoro effettuato, durata della giornata di tipo e retribuzione oraria del dipendente.
- La classe "Spesa_generale" incapsula la spesa che si sostiene giornalmente per attrezzi, pagamento commercialista, etc. Ogni "Spesa_generale" deve contenere una breve descrizione della spesa e l'importo.

Ho scelto di realizzare in questo modo la mia gerarchia in modo che sia totalmente ampliabile in futuro, ad esempio se un domani voglio implementare una classe "Spesa_commercialista" posso benissimo farlo con il minimo sforzo, poi ho scelto di realizzarla così per il totale polimorfismo infatti grazie a questa gerarchia posso ordinare, in una lista ordinata per data, tutte le spese del mio gestionale in modo totalmente polimorfo.

Infine grazie a questo polimorfismo per calcolarmi, ad esempio, l'importo totale delle spese non devo preoccuparmi del tipo di spese e del modo in cui si calcolano.



L'organigramma sovrastante è invece quello delle entrate giornaliere del intestatario, la classe base "Introiti_lavoratore" è una classe astratta che implementa 3 metodi virtuali puri.

Come nella gerarchia per le spese la classe base incapsula tutti i fattori che sono condivisibili in tutti gli introiti. La classe base ha come campi dati privati la data del lavoro e l'essere o non essere stati pagati. La classe "Introiti_lavoratoreOre" eredita pubblicamente da "Introiti_Lavoratore" ed, ovviamente, implementa tutti i metodi virtuali puri della base.

- La classe "Introiti_LavoratoreOre" usa, similmente ad "Spesa_dipendente" la classe "Durata" per incapsulare in modo corretto la durata del lavoro.
"Introiti_LavoratoreOre" deve contenere la durata del lavoro, il nome del cliente, il luogo e la propria retribuzione oraria (non statica perchè può variare moltissimo in base al cliente).

Avrei potuto anche non realizzare organigramma in questo modo e quindi implementare solo una classe chiamata "Introiti_LavoratoreOre" ma avrei perso tutta l'estendibilità del codice che ho, per esempio, nelle classi delle spese.

Polimorfismo:

Nel mio progetto, EasyJob, il polimorfismo è molto importante perchè mi permette di dichiarare qualunque spesa e/o introito e di sapere all'istante la mia somma delle spese e/o introiti senza andare a verificare di che tipo di dati stiamo sommando. Ad esempio se io dichiaro 1 spesa di tipo generale, 1 spesa di tipo dipendente, 1 spesa di tipo spesa_km e vado ad invocare il calcolo dell'importo ivato mi basta, semplicemente, iterare ed invocare il metodo virtuale puro "calcolaImportoIva" su tutta la lista delle spese (nodi linkati con campo dati Spesa*) e del tutto dimenticarmi di che tipo di spese siano, ci perserà il polimorfismo ad invocare il metodo corretto rispetto alla classe corretta per ogni tipo di spesa. Stessa cosa avviene con la gerarchia degli introiti, ovviamente renderà più l'idea quando verranno implementati nuovi tipi di introiti, come per esempio una fantomatica classe "Introiti_lavoratoreMensile".

Contenitore:

Ho deciso di sviluppare da zero due contenitori di tipo lista linkata per la natura di questo progetto, cioè ho deciso di organizzare la mia raccolta di oggetti in ordine di Data e quindi possono esserci svariate aggiunte ed eliminazioni per cui, per la natura della lista, la delete di un oggetto dentro ad essa è a tempo costante, contrariamente al Vector.

Progetto sviluppato con Qt 5. 4.2 in ambiente Windows 10 con compilatore MinGw 4.9 32bit .

Manuale Utente GUI:

Appena avviata EasyJob si presenta con una semplice schermata con la possibilità di inserimento delle principali dati della vostra ditta personale, Nome, Nazionalità della P.Iva, numero P.Iva, piccola descrizione della società (massimo 30 caratteri) premendo il bottone inserisci si cre la propria ditta.



Dopo l' inserimento si aprirà la vera e propria app, per spostarsi tra i menù si può usare la semplice navigationBar laterale. Cliccando in Informazioni si avranno a disposizione tutte le informazioni della propria ditta e di fare una delete Completa di tutte le spese e entrate compresa la ditta Stessa.

Cliccando su “Inserisci spesa” si avranno a disposizione tutti gli inserimenti dei tipi di spesa come, ad esempio, Spesa_dipendente. (presente nell’immagine)avendo la possibilità di compilare tutti i dati ed inserire la spesa nella propria lista in modo ordinato.

The screenshot shows a window titled "Easy Job" with a subtitle "Gestionale di Dario Luca". The interface has a sidebar on the left with four menu items: "Info Società", "Inserisci Spese" (highlighted in blue), "Inserisci Entrata", and "Funzionalità". The main area has three tabs: "Spesa Generale", "Spesa Viaggio", and "Spesa Dipendente" (selected). The form contains the following fields: "NOME DIPENDENTE:" with a text box containing "Fabio"; "DATA LAVORO:" with a date picker showing "11/02/2016"; "LUOGO:" with a text box containing "Bologna"; "RETRIBUZIONE ORARIA:" with a text box containing "10"; "Pagato?" with an unchecked checkbox; "Inizio Lavoro:" with a time picker showing "08:00"; "Fine Lavoro:" with a time picker showing "18:00"; and "Durata Pranzo:" with a time picker showing "01:00". An "Inserisci" button is located at the bottom right.

Cliccando su “Inserisci Entrata” si avrà a disposizione l’inserimento dell’ proprio Introito giornaliero con la compilazione ed Inserimento del tutto uguale alle Spese.

The screenshot shows a window titled "Easy Job" with a subtitle "Gestionale di Dario Luca". The interface has a sidebar on the left with four menu items: "Info Società", "Inserisci Spese", "Inserisci Entrata" (highlighted in blue), and "Funzionalità". The main area has a tab labeled "Entrate Intestatario". The form contains the following fields: "NOME CLIENTE:" with a text box containing "Giovanni"; "DATA LAVORO:" with a date picker showing "11/02/2016"; "LUOGO:" with a text box containing "Roma"; "RETRIBUZIONE ORARIA:" with a text box containing "32"; "Pagato?" with an unchecked checkbox; "Inizio Lavoro:" with a time picker showing "08:00"; "Fine Lavoro:" with a time picker showing "18:00"; and "Durata Pranzo:" with a time picker showing "01:00". An "Inserisci" button is located at the bottom right.

Cliccando su “Funzionalità” ci saranno due Major Feature.
 La prima è l’ estrazione delle spese e degli introiti organizzati per mese ed anno con la possibilità dell’ eliminazione, chiaramente sia profonda che dello widget, del settaggio del campo pagato, ovviamente anche questo sia in modo profondo che dl widget e del calcolo, relativo all’ anno e mese selezionato, dell importo, iva e importo non pagato.
 La seconda, Major Feature, è il calcolo su base Annua dell’ iva da dare allo stato, delle entrate, del netto, etc.

Dario Luca, Matricola: **1097935**

