# Home Work Weekend Silvio Barbato

Il progetto è stato così gestito:

come prima cosa ho creato un branch features dal branch develop e nella mia repository locale ho creato una nuova directory nominata “HomeWorkWeekend”. Da qui ho aperto il prompt dei comandi e utilizzando il template archetype per progetti java, ho creato un nuovo progetto Maven passandogli nome del progetto e del gruppo (rispettivamente “CurriculaSilvio” e “it.silvioCurricula”; quest’ultimo sarà il nome del package che conterrà il main del porgetto java) . Successivamente, dalla directory del progetto maven aperto, lancio da riga di comando il comando <mvn package> per far scaricare tutte le librerie necessarie alla compilazione del progetto, e lo integro con un ambiente di sviluppo, Spring in questo caso, tramite il comando <mvn eclipse:eclipse>. Da Spring quindi importo il progetto maven così creato. Dato che il progetto in esame richiede anche la connessione a un database, nel file di configurazione del progetto maven (nel pom.xml) aggiungo la dipendenza di connessione al database: dato che questa dipendenza è stata già utilizzata in passato essa è già presente nella repository locale (sul disco locale del pc), altrimenti essa veniva scaricata dalla repository remota di maven.

Per quanto riguarda la parte implementativa del progetto java esso è diviso in diversi package in cascata: oltre a quello che contiene il main è stato previsto un package che contiene tutte le interfacce del progetto, uno che contiene le classi che estendono le interfacce (e i corrispettivi metodi) e che contiene anche la classe per la connessione al database (il database è di tipo relazionale quindi si è sfruttato un RDBMS, in particolare MySQL) e uno che contiene tutte le classi concrete, tante quante sono le tabelle nel database, e conterranno tutti gli attributi di una tupla (una riga di una tabella del database identifica un oggetto della corrispettiva classe). Per quanto riguarda i metodi implementati, stando alle richieste del problema, essi sono di inserimento, di selezione e di aggiornamento, ognuno dei quali “interroga” il database, rispettivamente, di inserire una nuova tupla/oggetto in una tabella specifica, selezionare solo parte dei dati del database e aggiornare i dati di una tupla. Tutti e tre i metodi prevedono anche la gestione degli errori lanciati nel caso di non connessione al database (bisogna aver fatto l’accesso a MySQL per non incorrere in errore) o nel caso non trova la definizione di una classe.

1. Per l’inserimento di un nuovo candidato e del suo numero, passo al metodo un oggetto di candidato e una stringa che rappresenta il suo numero. Quindi una volta stabilita la connessione al database utilizzo due query: la prima a cui passo tutte gli attributi del candidato e la seconda a cui passo la stringa identificativa del numero di cellulare e la chiave primaria, immessa nella prima query, identificativa di quel candidato.
2. Per la selezione dei titoli di studio di un candidato, dato che un candidato potrebbe avere più titoli di studio, mi faccio restituire un Vettore di oggetti “TitoloStudio”; al metodo passo un intero che rappresenta la chiave primaria (di tipo intero quindi) del candidato che vogliamo cercare. Al solito quindi una volta stabilita la connessione utilizzo una query di selezione tra “Candidato”, “Istruzione” e “titoloStudio” e come condizione di ricerca(clausa) passo la chiave primaria del candidato.
3. Per l’aggiornamento dei dati di un candidato, passo al metodo un intero che rappresenta la chiave primaria identificativa del candidato che vogliamo modificare e tre stringhe rappresentative dei campi della residenza (via, cap e città): utilizzo quindi una query di update a cui passo quindi i tre campi della residenza e la chiave primaria del candidato che utilizzerò per la condizione di ricerca(clausa) della stesso.

Alla fine quindi ho fatto la pull request al reviewer del progetto (in questo caso lei quindi) per mergiare il nuovo branch al branch develop.

P.S. Ho notato che c’era un file (provastash.txt), nel branch develop, che creava conflitto nella richiesta di merge. Quindi quello che ho fatto è stato allineare il mio branch al develop (con il comando “git merge origin/develop”), dopo di che ho cancellato il file come suggerito da git e ho pushato di nuovo le modifiche. Il conflitto ora è risolto e non dovrà più essere lei a risolverlo.