

Diario di lavoro

Luogo:	Canobbio
Data:	01.10.2019

Lavori svolti

Oggi ho iniziato a sviluppare le classi modello, quindi le classi model. Per le quali ho avuto un problema e non so come risolvere il problema in questione.

Il problema riguarda il sistema di classi astratte, che intendo utilizzare per lo sviluppo del software. L'idea è quella di creare un metodo statico nella Super classe che mi ritorni il `QueryBuilder` (che sarebbe una classe che mi possa creare tutte le query relative ai modelli). Questo metodo deve prendere dei valori dalla sottoclasse, che servono per creare la `QueryBuilder`.

Una possibile soluzione di questo problema sarebbe non creare questi metodi e questi attributi statici, ma questo mi obbligherebbe ad istanziare ogni volta un oggetto, per richiedere la classe `QueryBuilder`, che non sarebbe ottimale perché nel caso in cui in un futuro, durante lo sviluppo del progetto l'oggetto utilizzato per richiedere questo elemento venga riutilizzato potrebbe generare delle vulnerabilità nel codice. Essendo in difficoltà con questo problema ho chiesto aiuto al mio compagno Filippo Finke, con il quale abbiamo fatto delle prove, che hanno portato ad una soluzione parziale del problema.

Questa soluzione utilizza il sistema di chiamata delle classi in maniera dinamica.

```
public abstract class A {
    public static void m() {
        StackTraceElement[] elements = Thread.currentThread().getStackTrace();
        StackTraceElement element = elements[elements.length - 1];
        try {
            Class classe = Class.forName(element.getClassName());
            System.out.println(classe.getField("ATTR").get(classe));
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

public class B extends A {
    public static final String ATTR = "content";
    public static void main(String[] args) {
        B.m();
    }
}

public class C {
    public static void main(String[] args) {
        B.m();
    }
}
```

Questo è il codice di test che abbiamo scritto.

Nel metodo `m()` della classe `A`, viene preso l'elenco delle classi per cui passa la chiamata del metodo. Dal quale viene preso l'ultimo elemento, per il quale viene richiesto l'attributo `ATTR`.

Nella classe `B` viene eseguita la classe `A`, ed implementato l'attributo `ATTR`. Se il metodo `B.m()` viene eseguito nella classe `B`, viene ritornato `ATTR` mentre se viene eseguito dalle altre classi (come `C`) viene ritornato errore.

Per risolvere questo problema ho anche chiesto al Docente Luca Muggiasca, che sul momento non sapeva rispondermi, quindi domani chiederò aiuto al docente Andrea Albertini, in maniera di cercare di risolvere il problema.

Problemi riscontrati e soluzioni adottate

Ho avuto problemi con l'implementazione di alcune classi Java e non ho ancora trovato una soluzione.

Punto della situazione rispetto alla pianificazione

Sono ancora molto avanti rispetto alla pianificazione.

Programma di massima per la prossima giornata di lavoro

Per la prossima giornata dovrò risolvere il problema della classe.