

RELAZIONE ELABORATO 20: RASH + EVOGRAPHS

1 INTRODUZIONE

L'elaborato è stato svolto seguendo le linee guida indicate nella descrizione. In particolare il lavoro è consistito in un'integrazione nel codice JavaScript della libreria Rash per permettere la conversione di dati csv in grafici, senza modifiche al codice HTML (escludendo l'inclusione delle risorse richieste dal plugin EvoGraphs).

2 FUNZIONAMENTO

Le modifiche apportate permettono ad un documento Rash contenente un elemento `` con valore class del tipo `CHARTTYPE_chart` (dove `CHARTTYPE` può essere `vertical_bar`, `horizontal_bar` o `pie` a seconda del tipo di grafico che si vuole mostrare) di sostituire l'`img_block` che lo contiene con un grafico, sfruttando le potenzialità del plugin EvoGraphs. Il grafico viene creato a partire da dati in formato csv che devono essere forniti in input attraverso l'attributo "src" dell'elemento ``, in due possibili modalità: passando all'interno dell'attributo il nome del file .csv in cui essi sono contenuti o indicando l'id dello script interno in cui essi sono stati inseriti. I dati vengono letti e processati in coppie chiave-valore, come richiesto da EvoGraphs, prendendo in considerazione le prime due righe non vuote; dalla prima vengono estratte le chiavi e dalla seconda i valori numerici.

NB: L'utilizzo di file .csv esterni per il passaggio di dati è possibile solo per documenti il cui accesso avviene in remoto, come spiegato più nel dettaglio successivamente.

3 LINGUAGGI E LIBRERIE UTILIZZATI

Come accennato nell'introduzione, le modifiche introdotte sono state principalmente implementate in linguaggio JavaScript, sfruttando, in particolare, la libreria jQuery. Viene inoltre utilizzata, attraverso il metodo `jQuery.get()`, la tecnologia AJAX per accedere ai file .csv esterni; per questo motivo un documento nel quale si voglia inserire un grafico a partire da un file .csv esterno deve essere acceduto in remoto, dato che tale tecnologia non permette l'accesso a file in locale (in fase di sviluppo questa funzionalità è stata testata simulando l'accesso al file da server attraverso localhost).

Non sono state utilizzate ulteriori librerie esterne, a parte quelle di EvoGraphs.

4 DETTAGLI DI IMPLEMENTAZIONE

Di seguito vengono sinteticamente descritti i dettagli di implementazione delle integrazioni fatte all'interno dei file del progetto.

4.1 MODIFICHE AL FILE RASH/DOCUMENTATION/EXAMPLE.HTML

Come da indicazioni, le uniche modifiche al file html di riferimento riguardano l'inclusione di risorse esterne necessarie per il funzionamento del plugin EvoGraphs. Nello specifico sono state incluse una risorsa css e due JavaScript attraverso le seguenti linee di codice:

```
19.      <link rel="stylesheet" type="text/css" href="http://code.evoxlabs.org/evographs.css" />

[...]
```

```
24.      <script src="http://code.evoxlabs.org/evographs.min.js"></script>

25.      <script src="http://code.jquery.com/jquery-migrate-1.2.1.min.js"></script>
```

4.2 MODIFICHE AL FILE RASH/JS/RASH.JS

L'integrazione al sistema si basa principalmente sull'aggiunta della funzione `convertToGraph()`, che viene applicata a ciascun elemento di classe `"img_block"` verificando se al suo interno ci sia un elemento `` da convertire in grafico, attraverso uno switch che ne individua anche il tipo specifico, e che, in caso positivo, estrae i dati dalla fonte indicata e sostituisce la struttura html dell'elemento secondo le specifiche richieste da EvoGraphs.