

# Protocollo di generazione “action” e scambio dati

## 1. Obiettivo del documento

Il capitolo successivo fornirà le specifiche con cui trattare le informazioni contenute in documenti xml con schema descritto nel documento “Definizione dell'xml schema per la gestione di “action””.

Più precisamente:

- Come il client dovrà usare un file xml dato dal server per generare una “action”, ovvero una wizard con cui l'utente potrà interagire
- Come il client dovrà eventualmente alterare il documento di partenza per tornare al server i dati forniti dall'utente
- Come il server dovrà interpretare i dati tornati dal client per valorizzare il database.

## 2. Generazione action

In questo capitolo verrà analizzato in dettaglio come il client dovrà interpretare il file di creazione action fornito dal server

### 2.1. <action>

Con <action> si deve generare una nuova wizard con titolo uguale all'attributo *title* dell'action.

Gli step, ovvero i figli, fanno aggiunti come sequenza di pannelli, ognuno con i contenuti specificati dello <step> e con la possibilità di navigare avanti e indietro fra l'uno e l'altro.

### 2.2. <step>

Con <step> si deve generare un pannello da inserire nella main window (ovvero l'action). Se l'attributo *type* è impostato a “in”, allora il passo conterrà dei campi di inserimento dati per l'utente. Di conseguenza, quando verrà premuto il tasto avanti o indietro per navigare verso gli altri step, bisognerà assicurarsi che le informazioni aggiuntive vengano opportunamente inserite nel file xml di partenza (e salvato). In questo modo, in caso di crash, non bisognerà ripetere l'analisi da zero.

### 2.3. <text>

Si dovrà generare un semplice textfield largo quanto specificato nell'attributo *width* (non più largo dello spazio a disposizione) e alto quanto specificato dal campo *row* (non più alto dello spazio a disposizione). L'altezza di una row dovrà essere proporzionata alla grandezza del carattere.

Il contenuto del textfield è il testo interno al tag.

NB: <text> è readonly.

### 2.4 <select> e <option>

I due campi sono strettamente correlati. La generazione degli option dipende dalle impostazioni del <select>:

- *mutex = true* : dovranno essere generate delle opzioni a selezione mutuamente esclusiva. Quindi, per ogni <option>, bisogna generare un campo con testo uguale al contenuto testuale dell'entità. Nel caso l'attributo *selected* sia uguale a true, il pulsante dovrà risultare selezionato. E' necessario che, con più option, se ne viene selezionata una, le altre rimangano deselectionate.
- *mutex = false* : le opzioni dovranno essere selezionabili in modo indipendente le une dalle altre. Il resto delle specifiche rimane uguale al caso “mutex = true”.

Il salvataggio delle modifiche in questo caso viene effettuato impostando l'attributo *selected* della/e option selezionata/e a true.

### 2.5. <inputField>

Bisogna generare due campi: uno descrittivo dell'input atteso (ovvero il label del textfield) e il secondo il textfield scrivibile. Il textfield ha dimensioni pari a quelle specificate dagli attributi *width* e *row* (come per <text>).

Il label deve essere posizionato dove specificato dall'attributo *layout*: con **up** sopra al text field, con **down** sotto, con **left** a sinistra e con **right** a destra. Il testo del label dovrà essere disposto su più linee nel caso lo spazio

rimanente a destra o sinistra del textfield sia troppo ristretto. Lo stesso vale se la lunghezza (del label) supera quella massima della finestra nel caso sia posizionato sopra o sotto il label.

Il salvataggio in questo caso viene impostato valorizzando il contenuto testuale del <inputField> col test presente nel textfield.

## 2.6. <image>

Nel caso l'immagine sia già stata fornita dal server, bisognerà decodificare la stringa in base64 nel tag e renderizzarla come immagine, della grandezza specificata dagli attributi width e height.

Altrimenti bisognerà comunque predisporre un'area della grandezza data con all'interno, invece dell'immagine, un pulsante "scatta", che cliccato avvia la fotocamera. Una volta scattata la fotografia, essa andrà salvata e visualizzata nello spazio a preimpostato.

La *caption* andrà posizionata sotto l'immagine (se presente).

Nel caso di salvataggio, la stringa base64 che rappresenta l'immagine andrà inserita come contenuto testuale del tag.

## 2.7. <recording>

Lo stesso discorso del <image> vale per il recording. Solo che invece di un'area con un'immagine, bisognerà visualizzare un riproduttore con pulsanti play/stop/record.

La *caption* andrà inserita sotto.

In caso di salvataggio, la rappresentazione testuale della registrazione andrà inserita nel contenuto testuale del tag.

## 3. Salvataggio nel database

## 4. Creazione action lato admin