

Come sviluppare un widget complesso (1/3)

In questa serie di tre articoli, vedremo come sviluppare un widget WordPress di una certa complessità con caratteristiche piuttosto avanzate. Il widget che svilupperemo è **un pretesto per vedere assieme due argomenti**: L'accesso al database di WordPress e la localizzazione (ovvero la traduzione in più lingue).

Nel primo tutorial imposteremo il widget, nel secondo lavoreremo sui database ed infine nel terzo vedremo come localizzarlo.

Darò per scontato che tu abbia letto ed acquisito i tutorial sui [plugin](#) e quello sui [widget](#), quindi in questa prima parte andrò piuttosto spedito.

Scenario

Svilupperemo un widget in grado di mostrarci gli ultimi post ed i post più commentati. Nel front-end queste due viste saranno mostrate grazie a delle tabs (come nella sidebar di YIW per i più commentati / in evidenza) realizzate con le UI di jQuery (fornite nativamente da WordPress). Il risultato finale sarà simile a questo

POSTPLUS



Per seguire più agevolmente il tutorial, puoi scaricare il documento che svilupperemo.

Warm Up

Nella cartella *plugins* di WordPress crea la cartella *postPlus* ed al suo interno il file *post.plus.widget.php*.

Crea l'intestazione del widget in questo modo:

Estendi la classe *WP_Widget* e dichiara il metodo costruttore come abbiamo già visto.

Il metodo form

Daremo la possibilità all'utente di configurare:

- il titolo del widget
- i titoli delle due tabs
- il numero di ultimi post da mostrare
- il numero di post più commentati da mostrare

Iniziamo dichiarando i valori di default.

E quindi dichiariamo il form così come abbiamo già visto in precedenza.

Il metodo update

Anche qui dichiariamo il metodo update così come abbiamo già visto.

Il metodo widget

Come ho anticipato all'inizio, visualizzeremo il nostro widget con delle tabs implementate tramite jQuery. Quindi vediamo brevemente la procedura.

Markup

Il markup per creare delle tabs con le UI di jQuery deve avere questa struttura:

- Un div contenitore all'interno del quale inserire
 - Una lista nella quale ogni elemento contiene un link. Questi link rappresenteranno la navigazione e fanno riferimento all'id delle rispettive sezioni di contenuto.
 - Le sezioni di contenuto saranno rappresentate da dei div con l'id corrispondente al link della navigazione.

Sembra molto complicato, ma basta guardare la [pagina di demo](#) (e visualizzare il codice) per capire immediatamente.

Javascript

Dovremo poi indicare che il nostro contenitore deve essere implementato come tabs. Lo faremo in modo semplicissimo con il metodo `tabs()`, in questo modo:

Naturalmente andrà inclusa la libreria jQuery, UI-tabs e tutte le sue dipendenze (ma come abbiamo visto, con WordPress è molto semplice).

Inoltre ci servirà il foglio di stile delle UI, senza il quale faremo ben poco. Iniziamo a procurarcelo. Ti insegno un trucchetto: Visto che ci serve unicamente lo stile per le tabs, è inutile utilizzare il foglio di stile completo. Dunque:

- Vai sulla [pagina di download delle UI](#)
- Deseleziona tutto
- Seleziona unicamente tabs (si selezioneranno automaticamente tutte le dipendenze)
- Scegli un tema
- Scarica il pacchetto, al suo interno troverai (cartella css) il foglio di stile contenente solo quello che serve per far funzionare le tabs.

Possiamo inoltre eliminare da questo documento tutta la parte relativa al framework delle icone, come pure la cartella immagini non ci serve.

Crea la cartella css all'interno della cartella del nostro plugin e copiaci il foglio di stile delle UI.

Siamo dunque pronti per scrivere il metodo widget.

Come puoi vedere ho dato al markup la giusta struttura per supportare le tabs di jQuery ed ho inserito dinamicamente i link della navigazione.

Per quello che concerne il contenuto, come vedi verrà inserito dai metodi `_getLastPost()` e `_getMostComment()` che fanno riferimento ai due metodi privati che svilupperemo nel prossimo

articolo. Per il momento li lasceremo commentati.

Dare vita alle tabs

Dovremo ora creare il file nel quale passare il metodo *tabs()* al contenitore che nel nostro caso ha id *pp-tabs-container*.

Crea il file *pp.jquery.tabs.js* nella cartella del plugin.

Ora dovremo includere gli script ed il foglio di stile. Iniziamo con lo stabilire il percorso della cartella del plugin, in questo modo:

Ed ora includiamo scripts e foglio di stile. Come abbiamo visto nei tutorial sui plugin, per gli script basterà richiedere l'inclusione di *pp.jquery.tabs.js* passando come dipendenza unicamente *jquery-UI-tabs*. Questo basterà per richiedere l'inclusione e gestione automatica di jQuery e delle dipendenze necessarie.

Scriviamo tutto in una funzione che passeremo al gancio degli scripts in questo modo:

Ed ora non resta che registrare il widget.

Ed il nostro plugin ora sarà visibile così:



Naturalmente potrai dargli l'aspetto che desideri lavorando sul foglio di stile.

Conclusione

In questo articolo abbiamo iniziato a predisporre il nostro widget. Nel prossimo capitolo vedremo

come si comunica correttamente con il database di WordPress in modo da poter recuperare i dati necessari per visualizzare le informazioni che ci interessano.

A presto dunque e... fino a qui è tutto chiaro?