

Personalizzare il Template dei Commenti di WordPress (Parte 1)

Con questa lezione giungiamo alla fine del corso dedicato alla realizzazione di un tema per WordPress che ci ha accompagnato in questi mesi. Per completare l'opera che c'eravamo preposti all'inizio del corso, oggi vedremo come personalizzare il template dei commenti del nostro tema. L'argomento di oggi è abbastanza ostico ma, se hai un minimo di conoscenze di programmazione php e hai voglia di imparare, può diventare davvero semplice e divertente affrontarlo insieme.

Ogni blog che si rispetti permette di commentare gli articoli pubblicati in modo da creare interazione e dialogo con i propri lettori. Che ne dici di creare un template per visualizzare i commenti sul nostro blog?

1. Creare il template per la visualizzazione dei commenti

Il primo passo da fare per visualizzare i commenti alla fine di ogni articolo consiste nel creare un file che sarà utilizzato come template per mostrare i commenti. Il file utilizzato da WordPress per svolgere questa funzione è denominato *comments.php*, creiamo quindi un nuovo file vuoto e rinominiamolo in questo modo.

1.1 Controllare se l'articolo è protetto da password

All'interno di questo nuovo file la prima cosa che facciamo è controllare se l'articolo è protetto da password, in caso affermativo visualizziamo un messaggio che invita il lettore a inserire la password per leggere e commentare l'articolo.

Per eseguire questo controllo utilizziamo la funzione di WordPress denominata *post_password_required()* la quale ci indica se l'articolo in questione è protetto o meno da password.

Nel caso in cui l'articolo è protetto da password l'utente verrà invitato a inserire quest'ultima per poterne leggere i commenti.

1.2 Controllare se l'articolo contiene dei commenti

Subito dopo questo controllo eseguiamo un altro controllo verificando se per l'articolo richiesto sono presenti dei commenti, per fare questo utilizziamo la funzione di WordPress denominata *have_comments()*:

Adesso andiamo ad aggiungere le istruzioni che vogliamo eseguire nel caso in cui il nostro articolo contenga dei commenti.

1.3 Mostrare un titolo e il numero di commenti totali presenti nell'articolo

Iniziamo con il mostrare un titolo che indichi il numero totale dei commenti presenti prima della lista vera e propria di quest'ultimi e facciamo lo mediante l'utilizzo della funzione [comments_number\(\)](#) la quale accetta tre parametri: il primo indica il testo da visualizzare nel caso in cui l'articolo non contenga ancora nessun commento, il secondo il testo da visualizzare nel caso sia presente un solo commento e l'ultimo il testo da visualizzare nel caso siano presenti più commenti.

Aggiungiamo quindi quanto detto all'interno del nostro file *comments.php*:

1.4 Mostrare l'elenco dei commenti presenti utilizzando una funzione personalizzata

Fatto questo andiamo a processare l'elenco di tutti i commenti presenti utilizzando la funzione [wp_list_comments\(\)](#):

Come puoi vedere dalla documentazione ufficiale la funzione [wp_list_comments\(\)](#) accetta diversi parametri. Noi ne abbiamo passato solo due tra i tanti supportati dalla funzione.

A cosa servono i due parametri che abbiamo utilizzato?

Molto semplice, i due parametri indicano nell'ordine:

- *'type'* => *'comment'*: il tipo di commenti che vogliamo mostrare, noi abbiamo scelto di mostrare solo la tipologia *"comment"*, ma si può scegliere di mostrare anche *"pingback"* o *"trackback"* o tutti e tre le tipologie insieme;
- *'callback'* => *'yiw_comment'*: il nome della funzione personalizzata (che dobbiamo ancora creare e sarà l'argomento della prossima lezione) che desideriamo utilizzare per mostrare l'elenco dei commenti, il parametro *callback* infatti serve proprio a definire il nome della funzione da richiamare per visualizzare l'elenco dei commenti, bypassando in questo modo la funzione predefinita di WordPress atta a svolgere questo compito.

Adesso all'interno del file *function.php* basterà creare una nuova funzione denominata *yiw_comment()* la quale conterrà il template (il codice XHTML per intenderci) da utilizzare per la visualizzazione di ogni singolo commento.

1.5 Mostrare, se presente, la paginazione dei commenti e la possibilità di

navigarla

Adesso controlliamo se è presente una paginazione nei commenti che abbiamo mostrato e in caso affermativo visualizziamola subito sotto l'elenco dei commenti:

Per navigare tra i commenti precedenti e successivi utilizziamo le due funzioni messe a disposizione da WordPress [previous_comments_link\(\)](#) e [next_comments_link\(\)](#).

1.6 Mostrare un messaggio nel caso in cui i commenti all'articolo non siano permessi

Se il nostro articolo non ha commenti, potrebbe trattarsi del caso in cui i commenti per l'articolo siano stati disabilitati dall'autore.

Se vogliamo mostrare un messaggio che avvisi l'utente che i commenti per quest'articolo sono stati disabilitati, dobbiamo eseguire un controllo per verificare se ci troviamo in questo caso ed eventualmente mostrare il testo desiderato.

Questa parte di codice va inserita subito dopo l'*else*, ovvero nella parte di codice da eseguire nel caso in cui non siano presenti commenti:

Per verificare se i commenti all'articolo sono permessi o meno, utilizziamo la funzione [comments_open\(\)](#) la quale restituisce un valore booleano (*true* o *false*) a secondo del caso in cui ci troviamo.

1.7 Mostrare il modulo per inserire i commenti

Dopo aver visualizzato l'intera lista di commenti presenti, è il momento di aggiungere (nel caso in cui i commenti siano permessi) il modulo che permette ai lettori di lasciare il proprio commento.

La prima cosa da fare è quindi verificare che i commenti siano permessi attraverso l'utilizzo della funzione [comments_open\(\)](#) vista nel paragrafo precedente:

In caso affermativo mostriamo il modulo che permette ai lettori di inserire il proprio commento all'articolo, ovviamente se i commenti sono chiusi non si può inserire nessun commento e il modulo non è necessario.

1.7.1 Controllare se è necessario essere registrati per inviare un commento

Se i commenti sono permessi il prossimo controllo che dobbiamo eseguire è verificare se è necessario essere registrati per inviare un commento:

Dopo aver aggiunto un titolo, la prima istruzione condizionale controlla se è necessario essere registrati per inviare un commento e se l'utente che sta visualizzando la pagina non è loggato. Se l'istruzione condizionale è soddisfatta, è mostriamo un messaggio che invita l'utente a eseguire il login.

1.7.2 Visualizzare il modulo

Nel caso in cui la registrazione non è richiesta o l'utente ha già eseguito il login, lo script andrà avanti saltando questo messaggio e visualizzando il modulo necessario per lasciare il commento.

Iniziamo dunque con l'inserire il form dei commenti il quale dovrà puntare alla pagina *wp-comments-post.php* come mostrato sotto:

Subito dopo eseguiamo un nuovo controllo condizionale: se l'utente è loggato mostriamo un messaggio che indica il suo nome utente (con link che rimanda al suo profilo) e che gli permetta anche di eseguire il logout:

Adesso aggiungiamo la textarea, dove l'utente scriverà il proprio commento:

Se l'utente non è loggato mostriamo anche gli altri campi del form: nome, e-mail e sito web cosicché possa compilarli insieme al commento che desidera aggiungere, diversamente, se l'utente fosse già loggato sarebbe inutile fargli inserire nuovamente nome, e-mail e sito web.

Questo codice dovrebbe essere abbastanza auto-esplicativo. Adesso aggiungiamo il pulsante d'invio e completiamo così il nostro form per l'invio dei commenti:

In quest'ultimo passaggio abbiamo inserito il pulsante d'invio, un campo nascosto contenente il futuro ID da assegnare al commento e una riga di codice () necessaria per far funzionare il modulo dei commenti.

2. Aggiungere i commenti all'interno del template degli articoli

Adesso che abbiamo il nostro file dei commenti completamente pronto per l'uso ci basta

semplicemente richiamarlo all'interno del template che avevamo realizzato per visualizzare il contenuto degli articoli: `single.php` e il gioco è fatto.

In questo modo ogni articolo mostrerà in fondo alla pagina l'elenco dei commenti - quando presenti - e il modulo per inserirne uno nuovo.

Per fare questo basta aprire il file `single.php` e richiamare la funzione `comments_template()` nel punto in cui vogliamo mostrare l'elenco dei commenti:

Questa funzione messa a disposizione da WordPress serve a richiamare in modo dinamico il contenuto del file `comments.php` che abbiamo realizzato in questa lezione.

Se adesso sul nostro blog andiamo a visualizzare un articolo che contiene dei commenti, possiamo vedere il nostro file in azione (Figura 1 e Figura 2).

Nota bene: *Se per scopo didattico desideri visualizzare correttamente l'elenco dei commenti alla fine di questa lezione dobbiamo eliminare il secondo parametro (callback) dalla funzione utilizzata per richiamare l'elenco dei commenti, questo è necessario perché al momento non abbiamo ancora creato la funzione personalizzata `yiw_comment` necessaria per visualizzare l'elenco dei commenti. La prossima settimana questo sarà l'argomento della seconda parte di quest'ultima lezione.*

3. Download dei file

Per scopo didattico ho messo a disposizione il tema finora realizzato, se lo desideri, puoi scaricarlo cliccando sul pulsante sottostante.

4. Conclusione

Siamo partiti completamente da zero e nel corso di queste lezioni siamo riusciti a personalizzare in modo completo un tema per il nostro sito/blog sviluppato su piattaforma WordPress. Nella lezione di oggi abbiamo visto come realizzare il file dei commenti e come implementarlo all'interno del template degli articoli. Nella prossima lezione vedremo come personalizzare l'elenco di visualizzazione dei commenti mediante la creazione di una funzione personalizzata, non mancare.

5. Indice degli articoli trattati in questa guida

Per agevolarti la consultazione della guida alla fine di ogni articolo riporterò l'indice comprendente tutti gli argomenti trattati.

1. [Come Installare WordPress in Locale?](#)
2. [WordPress: Diamo Uno Sguardo al Pannello di Amministrazione](#)
3. [Come Creare un Template per WordPress Partendo da Zero?](#)
4. [WordPress: Come Realizzare l'Header del Template? \(Parte 1\) | \(Parte 2\)](#)
5. [Diamo Uno Sguardo al Loop di WordPress](#)
6. [Aggiungiamo la Sidebar al nostro tema](#)
7. [WordPress: come rendere dinamici Header e Sidebar del template?](#)
8. [Gerarchia dei template e loro utilizzo: come creare un template per gli articoli?](#)
9. [Come Creare Template Personalizzati per le Pagine di WordPress?](#)
10. [WordPress: predisporre la sidebar per l'uso di widget](#)
11. [Personalizzare il Template dei Commenti di WordPress \(Parte 1\) | \(Parte 2\)](#)