Custom Field WordPress: Cosa Sono e Perché Sono Importanti

Introduzione

Quando scriviamo un articolo o un contenuto in generale, notiamo la presenza di alcuni dettagli di base che vengono visualizzati in basso, come ad esempio il nome degli autori, date etc. Questi dati vengono salvati come “meta dati” e prendono il nome di “custom field”; WordPress, di solito, li aggiunge automaticamente ma necessitano di essere abilitati.

Con i custom field (CF) è possibile modificare e aggiungere informazioni ulteriori all’interno della pagina con l’obiettivo di personalizzare ulteriormente il contenuto.

Per l’aggiunta di campi personalizzati è possibile procedere sia tramite plugin sia tramite l’aggiunta di codice direttamente nel tema. In questo articolo vedremo come aggiungere CF direttamente da codice.

Campi Personalizzati o Custom Field: a Cosa Servono?

L’utilità di un campo personalizzato è quella di poter mostrare, all’interno della pagina, informazioni aggiuntive oltre al contenuto principale e alle informazioni di base (come data di pubblicazione e autore).

Possiamo utilizzare i campi personalizzati in WordPress per aggiungere ad esempio un’informazione di approfondimento o una curiosità relativa al contenuto da mostrare in un determinato punto nella pagina.

Un custom field può anche essere utilizzato per prendere una

decisione su una parte di contenuto da mostrare o nascondere in pagina. Ad esempio potremmo aggiungere un custom field di tipo “text” per stabilire se mostrare la sidebar nel front-end del nostro sito web (vedremo tra poco come fare).

Oppure con un campo relazionale per mettere in relazione due tassonomie e restituire un valore in base al verificarsi di un dato evento.

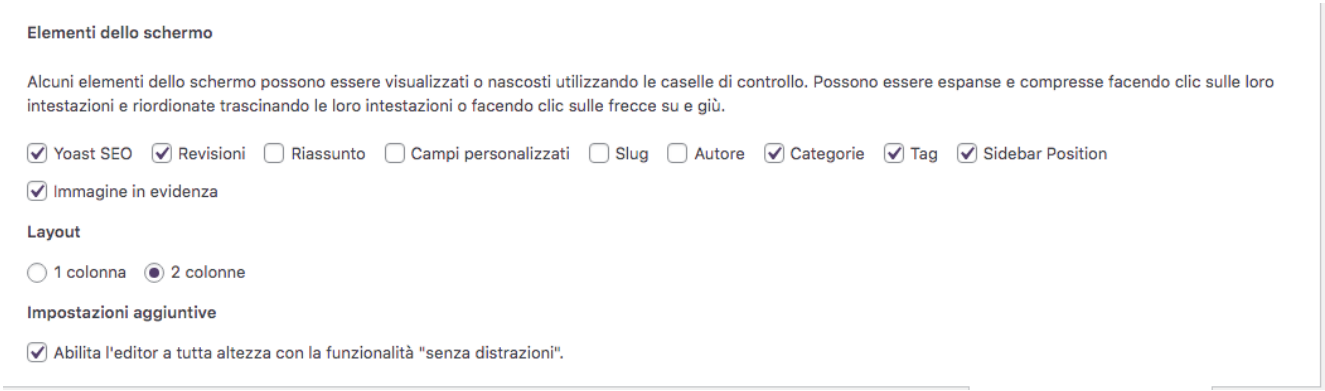
Supponiamo di dover creare un sistema di gestione dipendenti, avremo bisogno di alcune informazioni di base:

- nome e cognome (li abbiamo già in WordPress)
- data di nascita
- email (l’abbiamo già in WordPress)
- codice fiscale
- etc

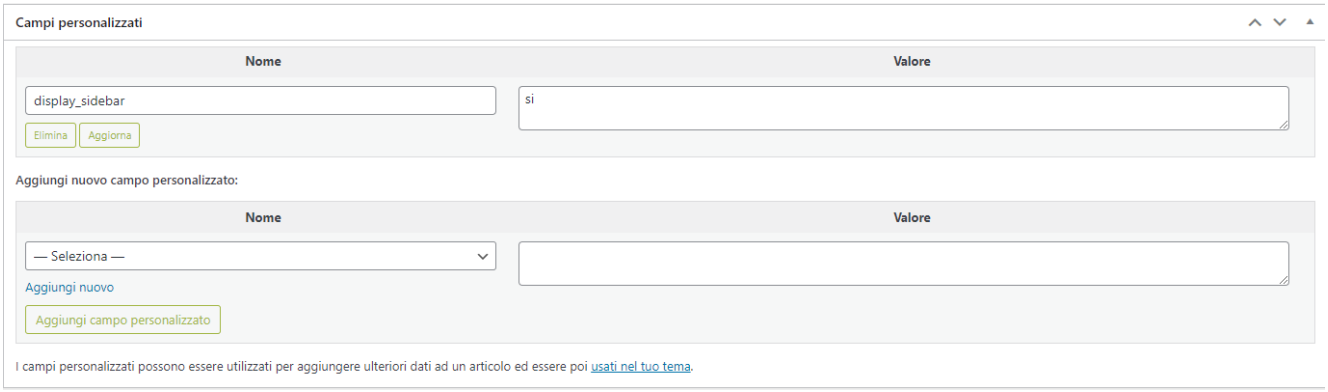
queste informazioni non sono tutte disponibili all’interno del sistema di gestione utenti di WordPress, quindi dovremmo aggiungere dei custom field.

Custom Field WordPress: Come Implementarli?

I campi personalizzati sono, per impostazione predefinita di WordPress, disabilitati, quindi dobbiamo abilitarli. Dalle **impostazioni schermata** della nostra pagina di modifica articolo, aggiungiamo un flag nella casella “Campi Personalizzati”.



Dopo aver abilitato la casella, spostarsi in basso e compilare il campo personalizzato con un nome descrittivo. Vogliamo decidere, per ogni articolo, se mostrare la sidebar oppure no.



Assegniamo come nome “display_sidebar” e come valore “si”, quindi stiamo passando come stringa il valore “si” che WordPress memorizzerà in database.

A questo punto dobbiamo aggiungere del codice all’interno del file single.php che si occupa di mostrare i dettagli del singolo articolo. All’interno del loop che mostra i dati del singolo articolo, aggiungiamo il seguente codice PHP:

```
$display_sidebar_si_no = get_post_meta($post->ID, 'display_sidebar', true);
```

Abbiamo creato una variabile chiamata **$display_sidebar_si_no** e al suo interno aggiungiamo il valore del nostro campo personalizzato. A questo punto intercettiamo la classe CSS della nostra sidebar e aggiungiamo una condizione IF che la mostrerà se il valore del CF è “si”. Nel nostro caso la classe CSS è “sidebar_developress_article”.

```
<?php
if ($display_sidebar_si_no == si) {
// Non facciamo nulla quando la variabile ha valore SI
}
else {
// Aggiungiamo uno style per nascondere la classe CSS
?>
   <style>
.sidebar_developress_article {
    display:none;
}
</style>
<?php
}
?>
```

Se abbiamo la necessità di mostrare agli utenti il valore di un campo personalizzato è sufficiente aggiungere, all'interno del loop (nella posizione che vogliamo), il seguente codice:

```
echo get_post_meta($post->ID, 'display_sidebar', true);
```

In alternativa è possibile anche lavorare con gli hook di WordPress per decidere la posizione esatta in cui mostrare il Custom Field.

Se vuoi approfondire il tema degli hook di WordPress puoi leggere questo articolo:

Hook di Azione: Cosa Sono e Come si Creano

Campi Personalizzati: Crearli con Advanced Custom Field

Uno dei migliori plugin per la creazione di campi personalizzati è Advanced Custom Field (ACF), una soluzione professionale e molto potente anche nella versione gratuita.

ACF permette di creare e lavorare con CF avanzati, del tipo: **campi relazionali tra custom post type** e tipologie di campi HTML avanzati come ad esempio il **caricamento di un file, color/data picker** etc.

ACF non verrà affrontato in questo articolo ma vorrei dedicare una guida dettagliata con esempi pratici di lavori che ho effettuato, in uno dei prossimi articoli. Intanto potete vedere la documentazione ufficiale, molto ben fatta e ricca di esempi:

Documentazione Advanced Custom Field

Se hai bisogno di supporto, puoi contattarmi o tramite email (la trovi nel footer) oppure tramite i profili social (li trovi nella mio biografica qua in basso).