GABRIELE ROMANATO

Menu

Come importare gli allegati di un sito in WordPress usando le REST API

L'importazione di allegati da un'installazione di WordPress a un'altra può sembrare un compito arduo, ma grazie alle REST API di WordPress, questo processo può essere automatizzato e reso più efficiente. In questo articolo, vedremo passo dopo passo su come eseguire questa operazione.

Prima di tutto, è necessario ottenere una chiave API per l'installazione di WordPress sorgente per autenticarsi durante le chiamate API.

1. Accedi alla tua dashboard di WordPress.
2. Vai su **Utenti > Il tuo profilo**.
3. Scorri verso il basso fino a trovare la sezione delle **Applicazioni**.
4. Crea una nuova applicazione e annota la chiave API generata.

Possiamo quindi scrivere:

```
$per_page = 100; // Numero di elementi per pagina, regolabile
in base alle esigenze
$page = 1;
$all_attachments = [];
$source_url = 'https://example.com/wp-json/wp/v2/media';
$api_key = 'YOUR_API_KEY';
$total_pages = 1; // Inizialmente impostato a 1 per entrare
nel ciclo

do {
    $response = wp_remote_get($source_url . '?per_page=' .
$per_page . '&page=' . $page, [
        'headers' => [
            'Authorization' => 'Bearer ' . $api_key,
        ],
    ]);
```

```

    if (is_wp_error($response)) {
        break; // Interrompe il ciclo se si verifica un
errore
    }

    $headers = wp_remote_retrieve_headers($response);
    $body = wp_remote_retrieve_body($response);
    $attachments = json_decode($body, true);

    if (empty($attachments)) {
        break; // Interrompe il ciclo se non ci sono più
allegati
    }

    $all_attachments = array_merge($all_attachments,
$attachments);

    // Aggiorna il numero totale di pagine

    if (isset($headers['x-wp-totalpages'])) {
        $total_pages = intval($headers['x-wp-totalpages']);
    }

    $page++;
} while ($page <= $total_pages); // Continua fino a quando
non abbiamo recuperato tutte le pagine

// Importa gli allegati nel sito di destinazione

foreach ($all_attachments as $attachment) {
    $attachment_url = $attachment['source_url'];
    if (!function_exists( 'download_url')) {
        require_once ABSPATH . 'wp-admin/includes/file.php';
    }
    $file = download_url($attachment_url);
    $filename = basename($attachment_url);

    $upload = wp_upload_bits($filename, null, $file);

    if (!$upload['error']) {
        $wp_filetype = wp_check_filetype($filename, null);
        $attachment_post = [

```

```
        'post_mime_type' => $wp_filetype['type'],
        'post_title' => sanitize_file_name($filename),
        'post_content' => '',
        'post_status' => 'inherit',
    ];
    $attach_id = wp_insert_attachment($attachment_post,
$upload['file']);
    require_once(ABSPATH . 'wp-
admin/includes/image.php');
    $attach_data =
wp_generate_attachment_metadata($attach_id, $upload['file']);
    wp_update_attachment_metadata($attach_id,
$attach_data);
    }
    wp_delete_file($file);
}
```

Spiegazione del codice:

1. Configurazione Iniziale:

- `per_page` è impostato a 100 per recuperare 100 allegati per ogni chiamata API.
- `page` è inizializzato a 1 per iniziare dalla prima pagina.
- `all_attachments` è un array vuoto che memorizzerà tutti gli allegati recuperati.
- `total_pages` è inizialmente impostato a 1 per entrare nel ciclo.

2. Richiesta Paginata:

- Viene eseguita una richiesta API utilizzando `wp_remote_get` con l'header di autenticazione `Bearer`.
- Le intestazioni della risposta sono recuperate utilizzando `wp_remote_retrieve_headers`.
- Il corpo della risposta viene decodificato da JSON a un array associativo PHP.

3. Controllo e Memorizzazione degli Allegati:

- Se la risposta non contiene allegati (`empty($attachments)`), il ciclo si interrompe.
- Gli allegati ottenuti dalla risposta corrente vengono aggiunti a `$all_attachments` utilizzando `array_merge`.

4. Aggiornamento delle Pagine Totali:

- L'header `X-WP-TotalPages` viene letto dalle intestazioni della risposta e convertito in un intero.
- Il ciclo continua fino a quando tutte le pagine non sono state recuperate (`$page <= $total_pages`).

5. Importazione degli Allegati:

- Iteriamo su tutti gli allegati ottenuti e scarichiamo il file usando `download_url`.
- Carichiamo il file nel sistema di WordPress del sito di destinazione usando `wp_upload_bits`.
- Inseriamo l'allegato nel database di WordPress e generiamo i metadati necessari.

Conclusione

Importare gli allegati da un'installazione di WordPress a un'altra può essere semplificato utilizzando le REST API di WordPress. Questo metodo non solo risparmia tempo ma garantisce anche che tutti i dati vengano trasferiti in modo sicuro e accurato. Con un po' di codice e alcuni strumenti, puoi automatizzare questo processo e concentrarti su altre attività importanti.

Ricorda di eseguire sempre un backup completo dei tuoi siti prima di eseguire qualsiasi operazione di importazione o esportazione per evitare la perdita di dati.