

MSG91 Campaign WooCommerce Plugin

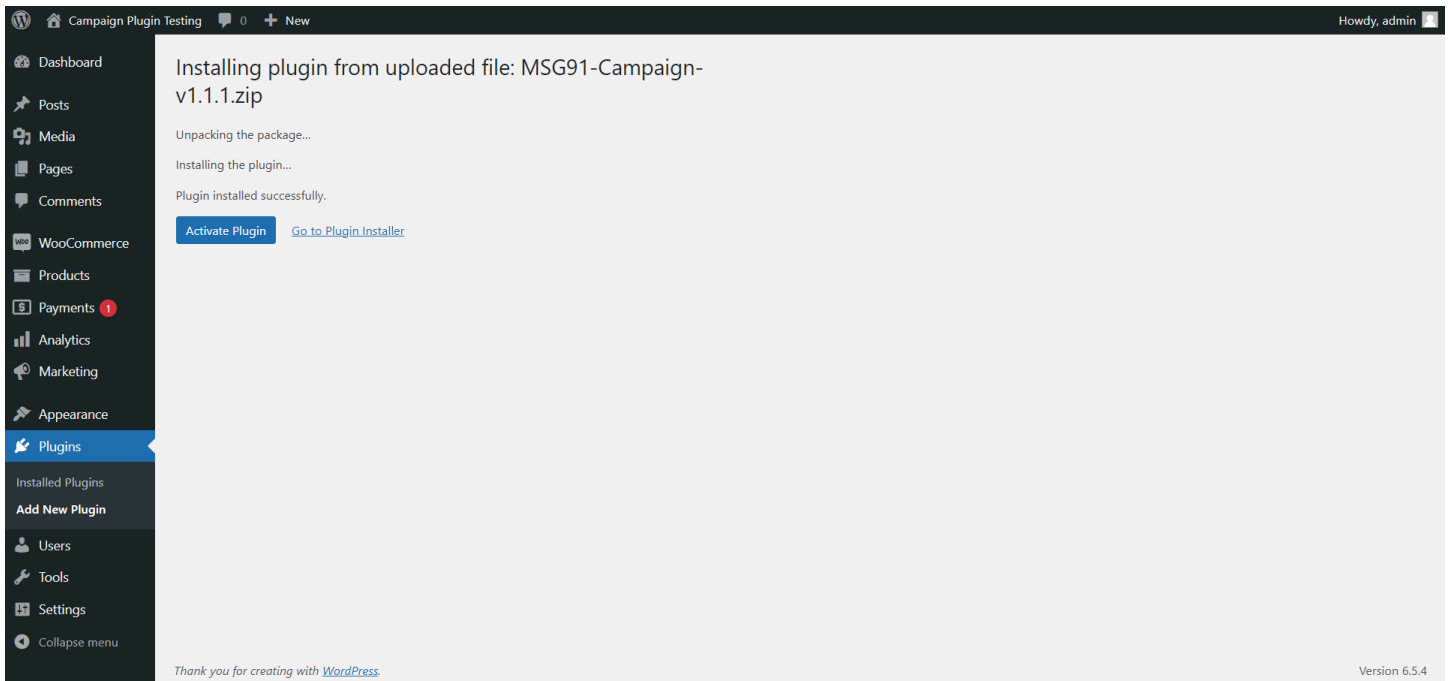
Plugin Link : <https://theweeky.com/msg91-campaign-woocommerce-plugin.php>

Support Email : contact@theweeky.com

- **How is it useful?** - You can send Email, SMS, WhatsApp, Telegram, Voice, etc directly from your WooCommerce Store. These notifications are triggered on default WooCommerce Order statuses like Order Creation, Order On Hold, Order Processing, Order Completed, Order Canceled based on the Campaigns created in your MSG91 Account.
- **How to get the Plugin?** - You would need to purchase the Plugin from this Page : <https://theweeky.com/msg91-campaign-woocommerce-plugin.php> Plugin is free for life time. No subscription or renewable fees. It's just that only 1 copy of the Plugin is valid for a single website, in case you have 2 websites you would need to purchase it again.
- **How to install the Plugin?** - Once you have purchased the Plugin, you will receive an email with the plugin download link. This email with download link is sent to the same email with which the payment is made and is an automated process. Sometimes, the email may land in your Spam folder.
 - After downloading the Plugin from the link, you would need to go to the Admin Panel of the website and then to 'Add New Plugin' under Plugins.

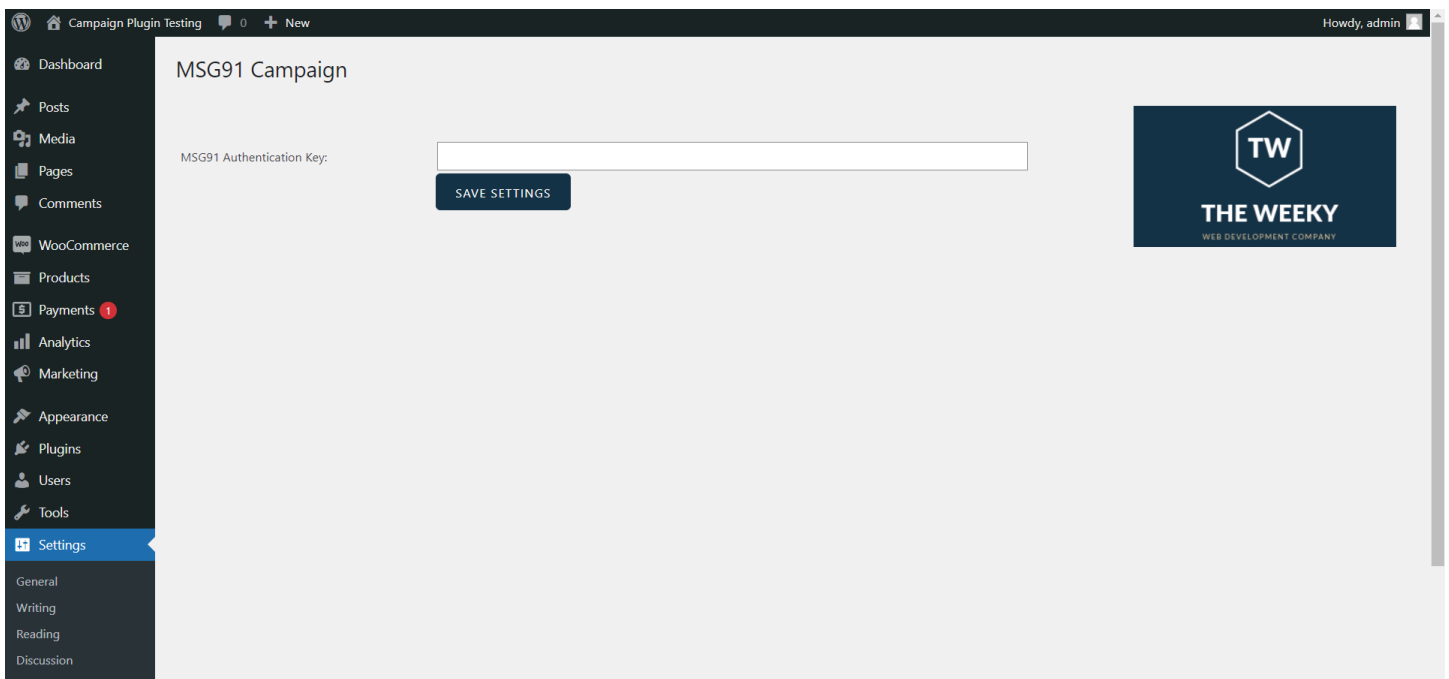
The screenshot shows the WordPress 'Add Plugins' page. The interface includes a sidebar with navigation options like Dashboard, Posts, Media, Pages, Comments, WooCommerce, Products, Payments, Analytics, Marketing, Appearance, and Plugins. The main content area is titled 'Add Plugins' and features a search bar and a grid of four featured plugins. Each plugin card displays its icon, name, a brief description, an 'Install Now' button, a star rating, the number of active installations, and the last update date. The plugins shown are: Classic Editor (1,172 stars, 10+ million installations), Akismet Anti-spam: Spam Protection (1,018 stars, 7+ million installations), Jetpack - WP Security, Backup, Speed, & Growth (2,071 stars, 4+ million installations), and Classic Widgets (248 stars, 2+ million installations).

- Here click on 'Upload Plugin' and then on 'Choose File'.
- Finally select the downloaded file and click on 'Install Now'.

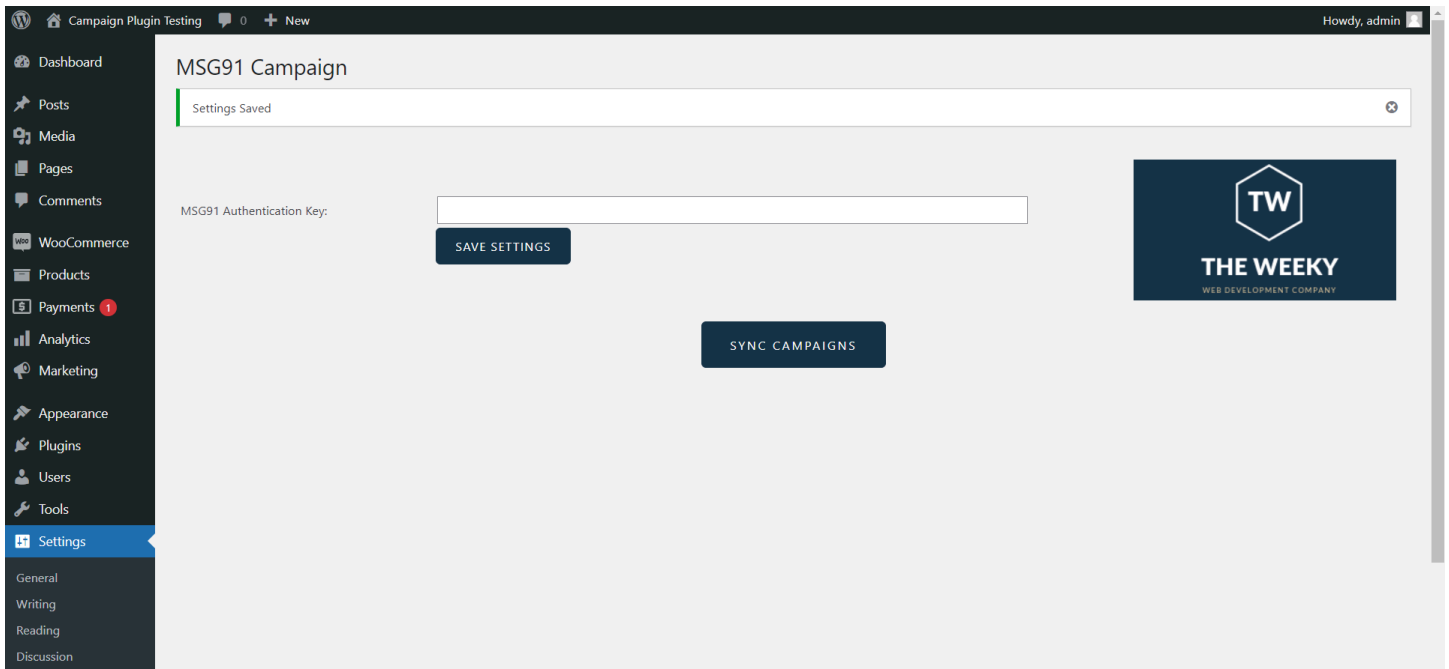


- After installation, click on 'Activate Plugin'

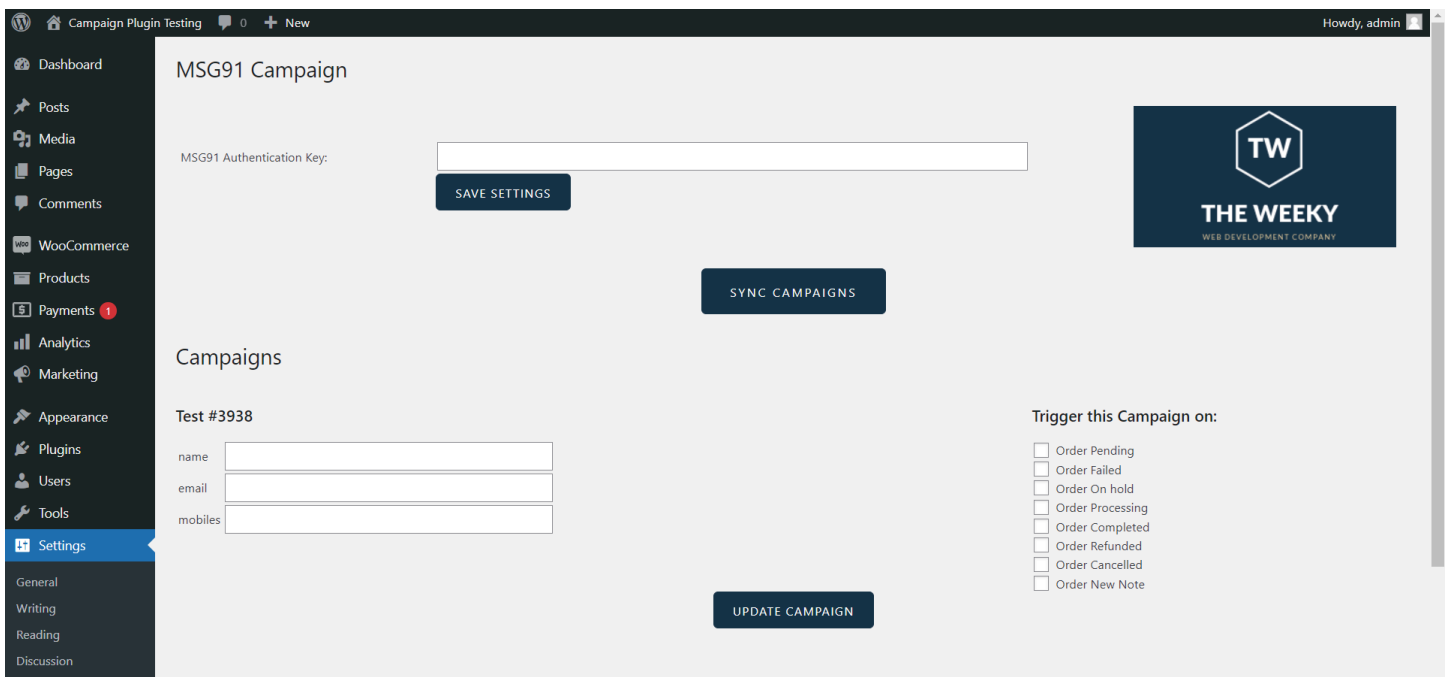
- **How to configure the Plugin?** - After activating the Plugin, go to 'Settings' and then to 'MSG91 Campaign'.



MSG91 Authentication Key - This can be obtained from your MSG91 account and is unique to every account. This is required so that Plugin can communicate with MSG91 using your account. Enter your Authentication Key and click Save Settings.



Click on Sync Campaigns to Sync your Campaigns from your MSG91 account to your WooCommerce Store.



It will list all your Campaigns. Under each Campaign, on the left side there are variables which you would need to map for the Orders Data so that Plugin can map that variable with the particular order value. Name can be mapped with “_billing_first_name” so Plugin can fetch the value of _billing_first_name for an order and can trigger MSG91 with the value.

On the right side we have the triggering part, here you can check the Order statuses on which you want this Campaign to be Active.

How to map Campaign Variables with Order data?

You can map Order data with the campaign variables in 3 ways:

1. Order Meta Data : You can map Order meta data which is stored in Order Meta Table using Meta Key. Simply enter the Meta key in the Input field and the value along that key will be passed to MSG91. For example : “_billing_first_name” (without quotes) for fetching the Customer’s first name, “_billing_phone” (without quotes) for fetching the Customer’s Phone.

2. Order Object Functions : You can input the Order Object Function name through which Plugin will fetch the value.

Some of the examples are:

```
$order->get_id();
$order->get_order_key();
$order->get_formatted_order_total();
$order->get_cart_tax();
$order->get_currency();
$order->get_discount_tax();
$order->get_discount_to_display();
$order->get_discount_total();
$order->get_total_fees();
$order->get_formatted_line_subtotal();
$order->get_shipping_tax();
$order->get_shipping_total();
$order->get_subtotal();
$order->get_subtotal_to_display();
$order->get_tax_location();
$order->get_tax_totals();
$order->get_taxes();
$order->get_total();
$order->get_total_discount();
$order->get_total_tax();
$order->get_total_refunded();
$order->get_total_tax_refunded();
$order->get_total_shipping_refunded();
$order->get_item_count_refunded();
$order->get_total_qty_refunded();
$order->get_qty_refunded_for_item();
$order->get_total_refunded_for_item();
$order->get_tax_refunded_for_item();
$order->get_total_tax_refunded_by_rate_id();
$order->get_remaining_refund_amount();
$order->get_items_key();
$order->get_items_tax_classes();
$order->get_item_count();
$order->get_item_total();
$order->get_downloadable_items();
$order->get_coupon_codes();
$order->get_line_subtotal();
$order->get_line_tax();
$order->get_line_total();
$order->get_shipping_method();
$order->get_shipping_methods();
```

```
$order->get_shipping_to_display();
$order->get_customer_id();
$order->get_user_id();
$order->get_user();
$order->get_customer_ip_address();
$order->get_customer_user_agent();
$order->get_created_via();
$order->get_customer_note();
$order->get_address_prop();
$order->get_billing_first_name();
$order->get_billing_last_name();
$order->get_billing_company();
$order->get_billing_address_1();
$order->get_billing_address_2();
$order->get_billing_city();
$order->get_billing_state();
$order->get_billing_postcode();
$order->get_billing_country();
$order->get_billing_email();
$order->get_billing_phone();
$order->get_shipping_first_name();
$order->get_shipping_last_name();
$order->get_shipping_company();
$order->get_shipping_address_1();
$order->get_shipping_address_2();
$order->get_shipping_city();
$order->get_shipping_state();
$order->get_shipping_postcode();
$order->get_shipping_country();
$order->get_address();
$order->get_shipping_address_map_url();
$order->get_formatted_billing_full_name();
$order->get_formatted_shipping_full_name();
$order->get_formatted_billing_address();
$order->get_formatted_shipping_address();
$order->get_payment_method();
$order->get_payment_method_title();
$order->get_transaction_id();
$order->get_checkout_payment_url();
$order->get_checkout_order_received_url();
$order->get_cancel_order_url();
$order->get_cancel_order_url_raw();
$order->get_cancel_endpoint();
$order->get_view_order_url();
$order->get_edit_order_url();
$order->get_status();
$order->get_checkout_order_received_url();
```

If you want to fetch the value of Shipping Country then enter this “get_shipping_country()” (without quotes) in the input field. Plugin already has a \$order object with it while “()” at the end tells Plugin that this is the function which needs to be called. In the case of meta keys we don't use () because they are not functions and they are values which are fetched directly from the database.

3. Using Plugin Filter : This is mostly required for Complex or Filtered Texts when you don't just need raw values.

Plugin uses filter `twy_campaign_order_msg` to filter the values before sending them to MSG91 and has 3 parameters.

1. `$value` is the value of the text that you enter in the input field.
2. `$order` object itself (WC_Order). You can use this to fetch the values.
3. `$campaign` contains the Campaign details.

Example:

```
add_filter('twy_campaign_order_msg',function($value,$order,$campaign){
    $result = "";
    if($value == 'get_date_created'){
        $result = wc_format_datetime($order->get_date_created());
    }
    return $result;
},10,3);
```

You would need to add this code in theme's `functions.php` or where you are maintaining your custom PHP code.

Enter `"get_date_created"` (without quotes) in the input field so that `{}` can tell the plugin to fire the filter and then filter custom code where we are checking this filter is getting executed for which input value using if condition.

If you want to use filters for multiple values simply add more ifs and then alter your `$result`. Example:

```
add_filter('twy_campaign_order_msg',function($value,$order,$campaign){
    $result = "";
    if($value == 'get_date_created'){
        $result = wc_format_datetime($order->get_date_created());
    }
    if($value == 'twy_id_formatted'){
        $result = "#TWY".$order->get_id();
    }
    return $result;
},10,3);
```

How to map Campaign Variables with Static data?

If you have static value for any Campaign Variable that will remain the same for all the Orders. You can simply prepend your value with `*` to make it static and Plugin will simply send this value to MSG91 without looking for it in the Orders.

For example: To send your Brand Name for each Order, you can use `"*The Weekly"` (without quotes) and it will send The Weekly to MSG91.

Don't forget to hit 'Update Campaign' after making your changes.