

# Tag Template

Google Tag Manager



This guide explains how to use TradeTracker.com's Google Tag Manager Template that has been published in Google's [Community Template Gallery](#).

This tag template makes it easy for advertisers to add TradeTracker.com's conversion pixel to Google Tag Manager container. If you have any questions or need any assistance please contact your account manager, or email at [support@tradetracker.com](mailto:support@tradetracker.com).



# Contents:

Before setting up your tag	4
Sales tag variables & triggers	4
Leads tag variables & triggers	6
Installing & configuring your tag	7
Testing your tag	9
Other tag types	11



# Before setting up your tag

Make sure you have a **Campaign ID** and **Product ID** that have been provided by your account manager at TradeTracker.com. You will need to enter these during the tag configuration.

You will also need to ensure that you have GTM (Google Tag Manager) Variables configured that capture the data that you want to track, and GTM Triggers to fire the tag on these events.

## Sales tag variables & triggers

If you want to track retail sales from your website you will need to capture the transaction (order) details, and fire this tag on the purchase event. For this we highly recommend using [Google's Enhanced Ecommerce model](#) for pushing purchase events to the dataLayer. We require these two specific variables:

### Transaction ID:

This variable should capture the unique transaction or order id for the sale. This should be the same id that you use to track your sales internally, and it's important that it's unique for every sale.

In the Enhanced Ecommerce purchase model this variable can be found in the dataLayer at `ecommerce.purchase.actionField.id`.

If you need to create new GTM variable for this value, create a new dataLayer type variable, and enter this address in the "Data Layer Variable Name" field.



### Transaction amount:

This variable should capture the total monetary amount for the sale **excluding** any applicable tax such as VAT. It should be formatted as a “string”, and should be a dot-decimal number to two decimal places.

#### *Examples:*

*Sale amount: \$9.81 => “9.81”*

*Sale amount: € 2.145,99 => “2145.99”*

The transaction amount cannot always be taken directly from the standard Enhanced Ecommerce purchase object, as the “revenue” variable typically contains the total amount **including** tax.

If you don't already have a variable for the order amount excluding tax, the recommended way to capture this amount is with a “Custom JavaScript Variable” in GTM. You will first need to create dataLayer variables for the transaction revenue `ecommerce.purchase.actionField.revenue` and transaction tax `ecommerce.purchase.actionField.tax` amounts.

Once this is done you can then create a new Custom JavaScript Variable that should look something like this:

```
function () {  
  return ({{your_transaction_revenue_variable}} - {{your_transaction_tax_variable}}).toFixed(2)  
}
```

### Purchase trigger:

This GTM Trigger should be set to fire either when a “purchase” event is pushed to the dataLayer, or to match the url of your post-purchase “thank-you” page and fire the tag there.



Firing on the purchase event is a more reliable way to fire the tag, and if you use the Enhanced Ecommerce purchase model you can create a “Custom Event” trigger, and enter `purchase` as the “Event name” to create this trigger.

Alternatively if you need to create a trigger based on the thank-you page url or another event, test it thoroughly to make sure that it fires reliably and that the transaction detail variables will be available to the tag when the trigger fires it.

## Leads tag variables & triggers

If you want to track custom lead events such as email signups or demo requests, you will need to create some custom variables to provide to the TradeTracker.com tag. Due to the highly customizable nature of these events, we cannot offer as much advice as we can for the sales tag, but we do have a few guidelines.

### **Transaction ID:**

The Leads tag still requires a transaction id. In this case the transaction id tracks a unique user event on your website. If this will track something like email signups where an id is returned from the submitted form, you could provide this id so that these leads events can be matched from TradeTracker.com’s tracking to your own analytics.

If you are tracking an event that does not have an provide it’s own id, you could create a unique identifier for events by combining a random number and a timestamp in a GTM Custom JavaScript variable. For example:

```
function () {  
  return Date.now() + Math.random()  
}
```

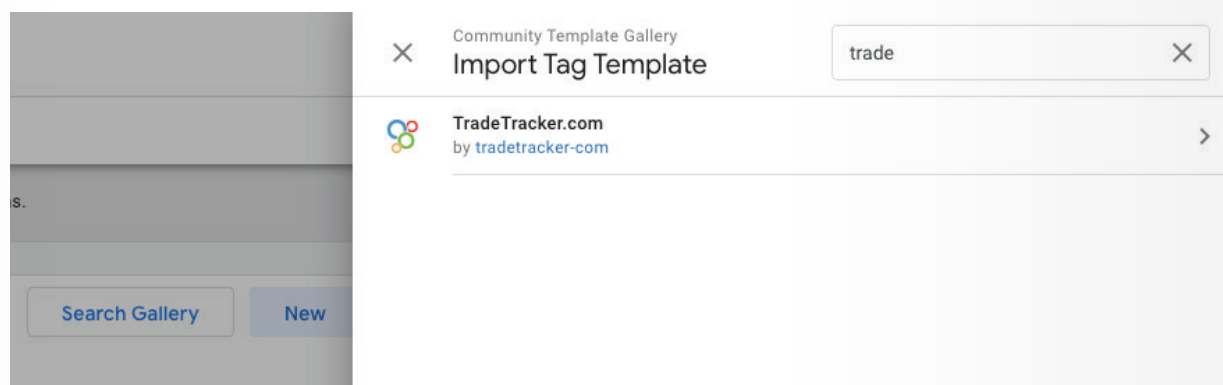


## Event trigger

The trigger for your leads tag will be defined by the type of user event that you wish to track. If the event includes a click on a particular button, or a post-signup page you can create GTM triggers for these events. Alternatively you may want to look at creating a custom event trigger. More information on how to set up these triggers can be found on [Google's Trigger types article](#).

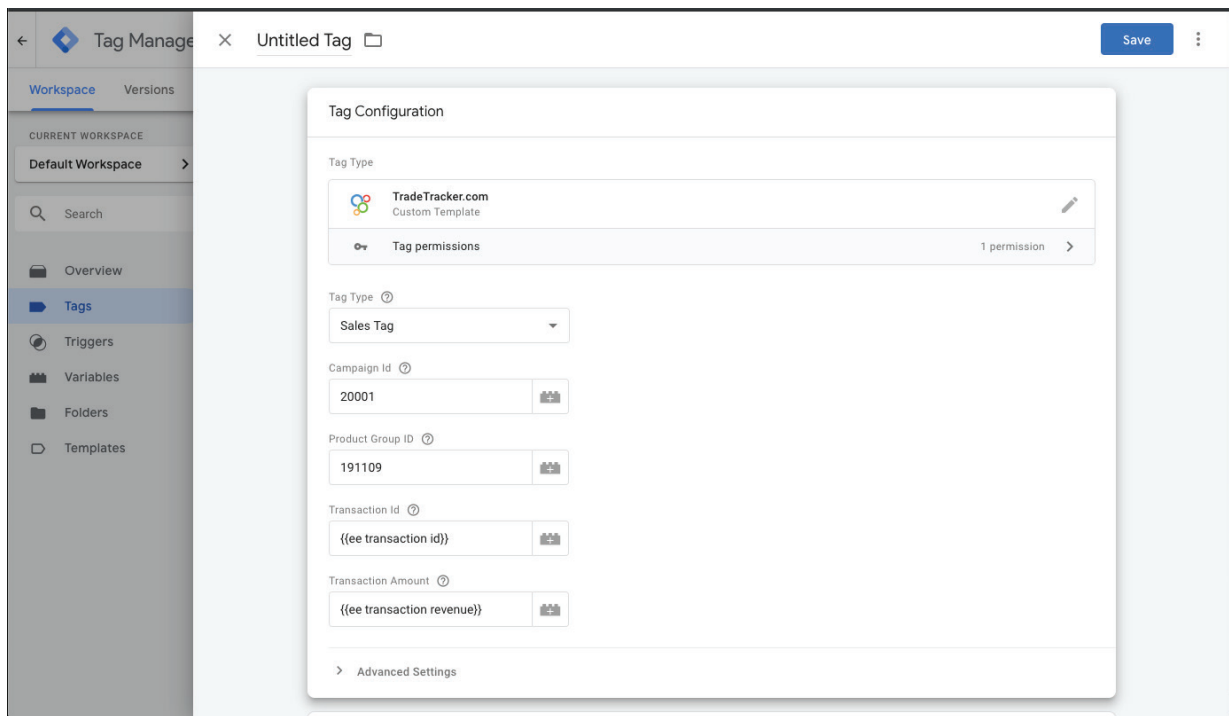
# Installing & configuring your tag

1. To install the TradeTracker.com tag template in your GTM workspace, navigate to the Templates tab, click Search Gallery in the Tag Templates section. Search for "TradeTracker":

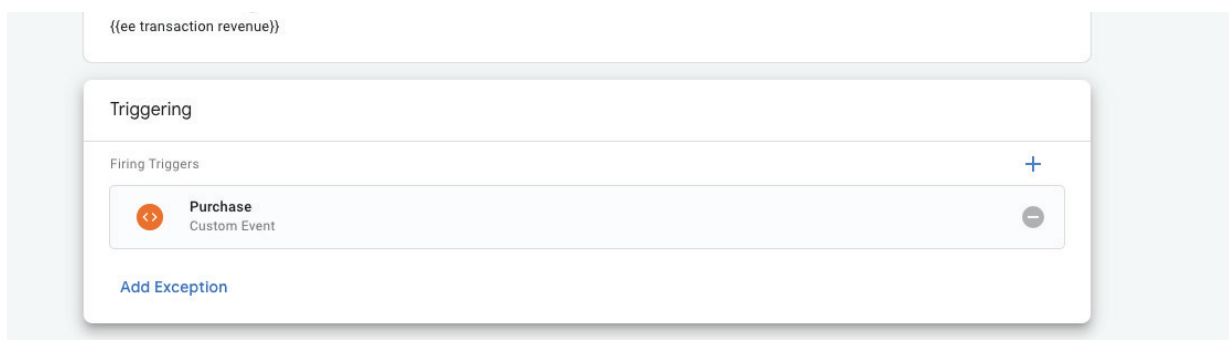


2. You will need to accept permissions for "Sends Pixel" so that the TradeTracker.com tag can communicate your campaign data back to our tracking servers.
3. Once the template has been added, navigate to the Tags tab and create a new tag for TradeTracker.com.
4. Select a Tag Type from the drop-down menu. Add your Campaign ID as text or enter the variable name.
5. Add your ProductGroup Id as text or enter the variable name.





6. Enter the variable name for your Transaction ID variable.
7. Enter the variable name for your Transaction amount variable ("Sales" tag type only).
8. Select the correct firing trigger for the tag.



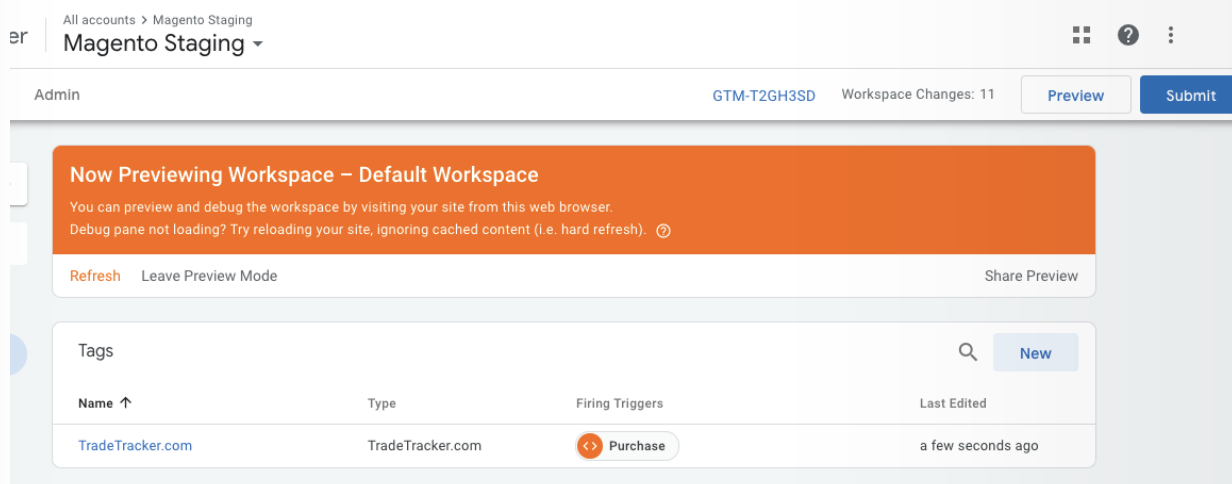
9. Save the tag. If you haven't already entered a tag name, You will be prompted to add a tag name before saving.



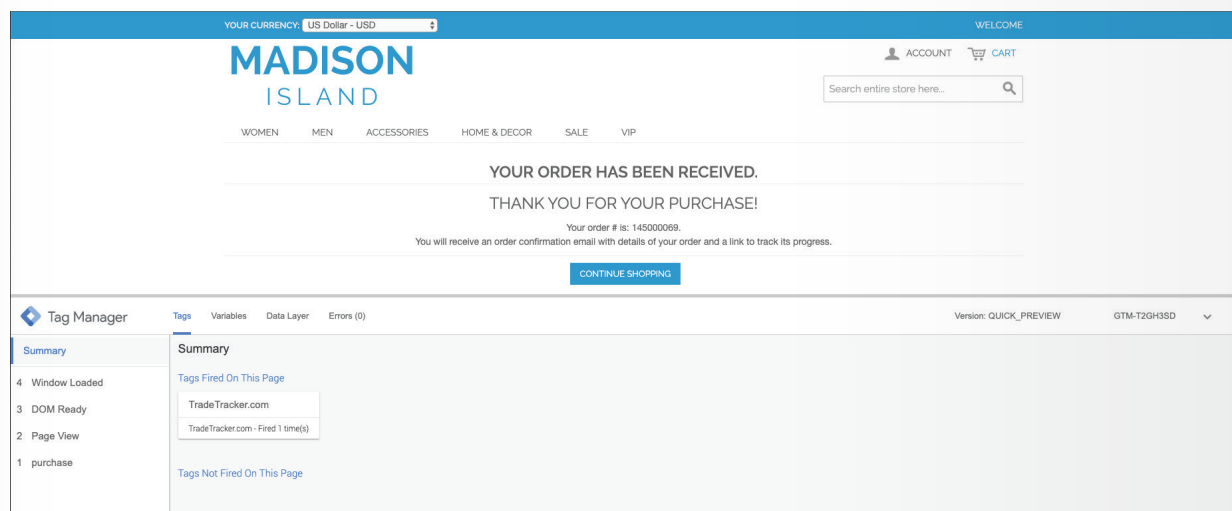


# Testing your tag

1. Once your tag as been configured, enter the workspace preview mode using the “Preview” button at the top right of the dashboard. Once the preview has been loaded, you should see an orange banner.



2. Place a test order on your shop, on the thank-you page you should see that the TradeTracker.com tag has fired in the Google Tag Manager debugging panel.



3. Clicking on the tag card will display the details of exactly how the tag fired.

The screenshot displays the Google Tag Manager interface. On the left, a sidebar shows a list of events: '4 Window Loaded', '3 DOM Ready', '2 Page View', and '1 purchase'. The 'purchase' event is selected. The main content area shows the 'Summary' tab for a tag named 'TradeTracker.com'. Below the tag name, the 'Properties' section contains a table with the following data:

Name	Value
Type	TradeTracker.com
Transaction Amount	65
Product Group ID	'1234567'
Transaction Id	'145000069'
Campaign Id	'12345'

Below the table is a 'Show More' link. Under the 'Messages Where This Tag Fired' section, a box indicates '1 purchase'. The 'Firing Triggers' section shows a 'Purchase' trigger with filters set to '\_event' equals 'purchase'.

4. At this point your tag should be fully configured and tested, and you can ask your account manager to check that the test transaction was correctly registered in our system.



# Other tag types

This Google Tag Manager template only supports the most common tracking scenarios for TradeTracker.com marketing campaigns. If you run a more complex performance marketing campaign that makes use of multiple campaigns for different brands, multiple product groups, exclusive voucher codes or needs to support multiple countries, please get in touch with your TradeTracker.com account manager for further support and guidance on how implement our tracking. If you're unsure of how to contact your account manager, please email us at [support@tradetracker.com](mailto:support@tradetracker.com).



