



| | |
|---|----|
| Pushing events to Google Analytics from giosg | 1 |
| Enable Google Analytics events | 1 |
| Determine Google tracker service object | 2 |
| Events pushed to Google Analytics | 3 |
| Using with Google Tag Manager | 5 |
| *a) Google Analytics – Universal Analytics | 9 |
| *b) Google Analytics: GA4 Configuration | 10 |

Pushing events to Google Analytics from giosg

It is possible to integrate giosg to your existing Google Analytics or Google Tag manager account. You can find a list of events pushed to analytics under “Events pushed to Google Analytics”.

To start pushing events to your analytics account please follow the steps below.

Enable Google Analytics events

1. First you need to enable analytics events in giosg settings. Go to your Room settings, select the desired domain and under the “Tracking settings” header check “Enable tracking analytics events” checkbox. Then select the desired tracking object from the “Tracking service” drop-down menu below. To determine which Google tracker object is in use please see “Determine Google tracker object type” below. If your website is configured as a Single-Page application, check the single page app (AJAX app) tracking checkbox.

2. Click Save and you're done!

Determine Google tracker service object

Google Analytics can be used with many versions of tracker scripts and objects. To check which one you should use please consult your IT department or use your browser developer tools to check which tracker object is in use.

You can check the tracker object type by going to your company's website and opening your browser's developer tools console.

If you type "dataLayer" to the developer console, and it returns other than undefined, select "Google Tag Manager Data Layer (dataLayer.push)". For further configuration needed when using dataLayer, please see "Using with Google Tag Manager" below.

If you type "ga", and it returns other than undefined, select "Google Universal Analytics (analytics.js)".

DEPRECATED: If you type "_gaq" into the browser console, and it returns other than undefined, select "Google Analytics (ga.js)".

Note: ga.js is a legacy library. Google recommends migrating existing implementations to use the newest analytics.js library.

Custom analytics tracker object

If you have customized your Google tracker object name or are using some other analytics script it is possible to define a custom tracker object name. If you select “Custom tracker” as the tracking service type this object is checked from the window object and used if found.

If the type of that object is “function” then it is called with three parameters, for example: myTrackerObj(eventSender, eventName, eventData);

If the type of that object is “object” then we assume that the object is an array and the push function is used. For example:

```
myTrackerObj.push([eventSender, eventName, eventData]);
```

Below is an example custom tracker object that could be defined. In this case you should set the Analytics “Tracking service” to “Custom tracker” and the object name setting to “myTrackObj”:

```
window.myTrackObj = function (ev, name, data) { console.log(ev, name, data); };
```

Events pushed to Google Analytics

Here is the list of tracking events that will be pushed to Google Analytics when certain events occur.

Chat button clicked

```
eventCategory: "giosglive"  
eventAction: "chatbutton"  
eventLabel: "[Current room id]:[Current page URL]"
```

Visitor received an autosuggest message

```
eventCategory: "giosglive"  
eventAction: "autosuggested"  
eventLabel: "[Current room id]:[Current chat id]:[Current page
```

URL]" Visitor has chatted a real conversation

```
eventCategory: "giosglive"  
eventAction: "hasChatted"
```

```
eventLabel: "[Current room id]:[Current chat id]:[Current page URL]"
```

Visitor, who has received an autosuggest message, chatted a real conversation

```
eventCategory: "giosglive"  
eventAction: "autosuggestedHasChatted"  
eventLabel: "[Current room id]: [Current chat id]:[Current page
```

```
URL]" Visitor reached goal
```

```
eventCategory: "giosglive"  
eventAction: "visitorReachedGoal"  
eventLabel: "[Goal name]:[Goal id]:[Current page URL]" eventValue: [Goal value]
```

Lead form was shown to visitor - Deprecated

```
eventCategory: "giosglive"  
eventAction: "leadFormWasShown"  
eventLabel: "[Lead Form name]: [Lead Form id]: [Current page
```

```
URL]" Lead form was completed - Deprecated
```

```
eventCategory: "giosglive"  
eventAction: "leadFormWasCompleted"  
eventLabel: "[Lead Form name]:[Lead Form id]:[Current page URL]"
```

Why is eventLabel data separated by a colon? Why can't I have data XYZ?

Some Google Analytics tracker solutions are pretty limited considering the data format. Therefore, eventLabel is used to carry the extra data. However, if Google Tag Manager Data Layer is used, the extra data fields are also appended to the sent JSON. For example:

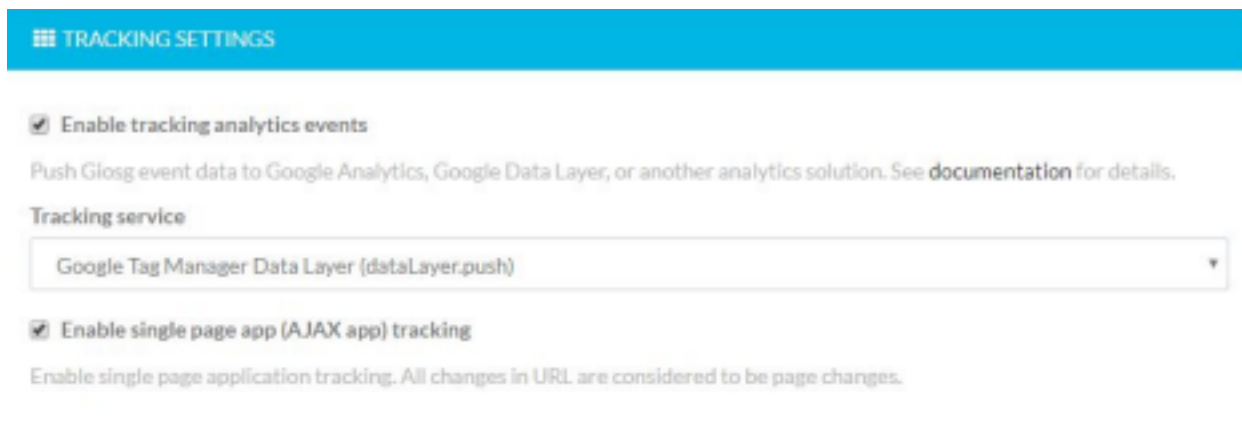
```
{  
  eventCategory: "giosglive"  
  eventAction: "hasChatted"  
  eventLabel: "[Current chat id]:[Current chat id]:[Current page  
URL]", chatId: "[Current chat id]",  
  roomId: "[Current room id]",  
  pageUrl: "[Current page URL]"
```

Using with Google Tag Manager

To use giosg Live analytics integration with Google Tag Manager you first need to select “Google Tag Manager DataLayer” as the Tracking Service in giosg settings, and then configure Google Tag Manager to capture events triggered by giosg.

Configuring giosg

Go to your Room settings, select your domain and under the “Tracking settings” header check “Enable tracking analytics events” checkbox. Then select ‘Google Tag Manager Data Layer (dataLayer.push)’ from the “Tracking service” drop-down menu below. If your website is configured as a Single-Page application, check the single page app (AJAX app) tracking checkbox.

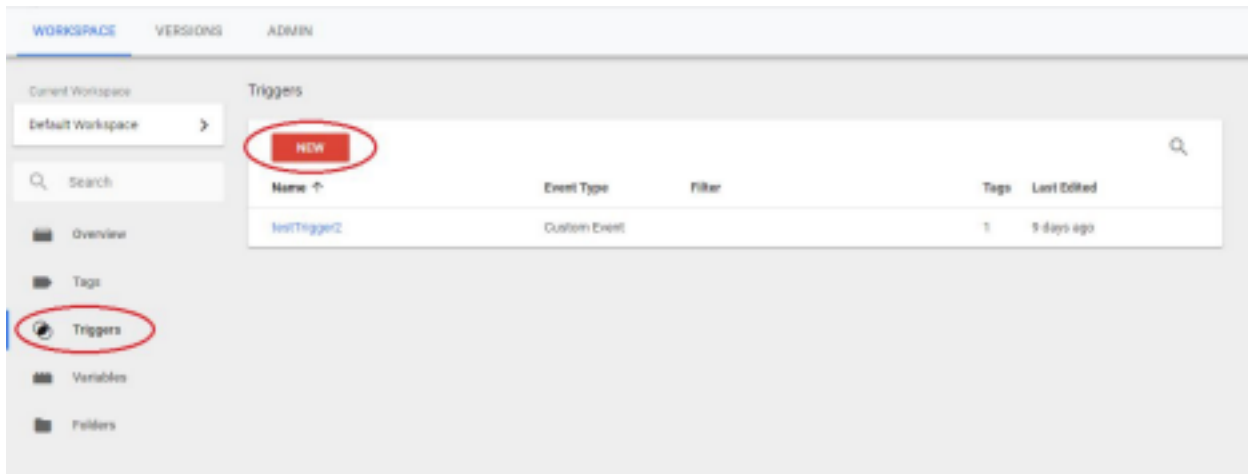


The screenshot shows the 'TRACKING SETTINGS' section of a web application. It features a blue header with a grid icon and the text 'TRACKING SETTINGS'. Below the header, there are two main settings sections. The first section, 'Enable tracking analytics events', has a checked checkbox and a description: 'Push Giosg event data to Google Analytics, Google Data Layer, or another analytics solution. See [documentation](#) for details.' Below this is a 'Tracking service' dropdown menu with 'Google Tag Manager Data Layer (dataLayer.push)' selected. The second section, 'Enable single page app (AJAX app) tracking', also has a checked checkbox and a description: 'Enable single page application tracking. All changes in URL are considered to be page changes.'

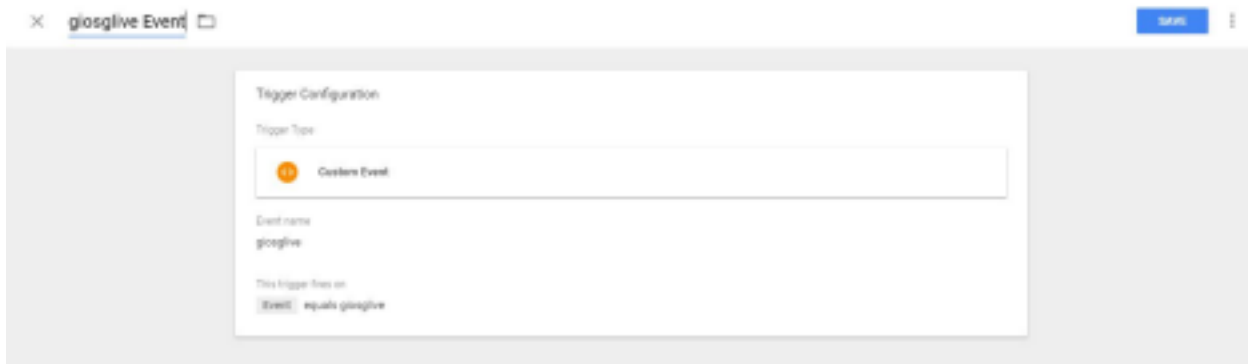
Configuring Google Tag Manager

To set up GTM to allow receiving giosg events you need to add a trigger, a tag and a few variables to GTM. Follow steps below to make these:

1. First login to your tag manager account at <https://www.google.com/tagmanager/>
2. Start by creating a trigger for the tag that we will create later. Under the “Triggers” menu click “New”.



3. Select “Custom Event” as the trigger type and use “giosglive Event” as the Trigger and Event name.
4. Under “This trigger fires on” select “Some Custom Events”, then select “Event” to the first dropdown and “Equals” to the second dropdown. In the value text box type “giosglive”. When done, click save.



5. Next we need to define the variables. Select “Variables” from the left sidebar, and under “User-Defined Variables” click “New”.

| | | |
|------------------|-------------------|---------------------|
| Variables | Old History State | Data Layer Variable |
| Folders | Page Hostname | URL |
| | Page Path | URL |
| | Page URL | URL |
| | Referrer | HTTP Referrer |

| User-Defined Variables | | |
|---|---------------------------|-------------|
| NEW | | |
| Name ↑ | Type | Last Edited |
| giosglive event action | Data Layer Variable | 8 days ago |
| giosglive event category | Data Layer Variable | 8 days ago |
| giosglive event label | Data Layer Variable | 8 days ago |
| giosglive event value | Data Layer Variable | 8 days ago |
| Google Analytics Settings - service-staging.giosg.com | Google Analytics Settings | 3 days ago |
| room id | URL | 3 days ago |
| URL query | URL | 3 days ago |

- Type “giosglive event category” as the Variable Name. Select “Data Layer Variable” as the variable type.
- Type “eventCategory” as the Data Layer Variable Name, and save.

×
giosglive event category
Save

Variable Configuration

Variable Type

Data Layer Variable

Data Layer Variable Name

eventCategory

Data Layer Version

Version 2

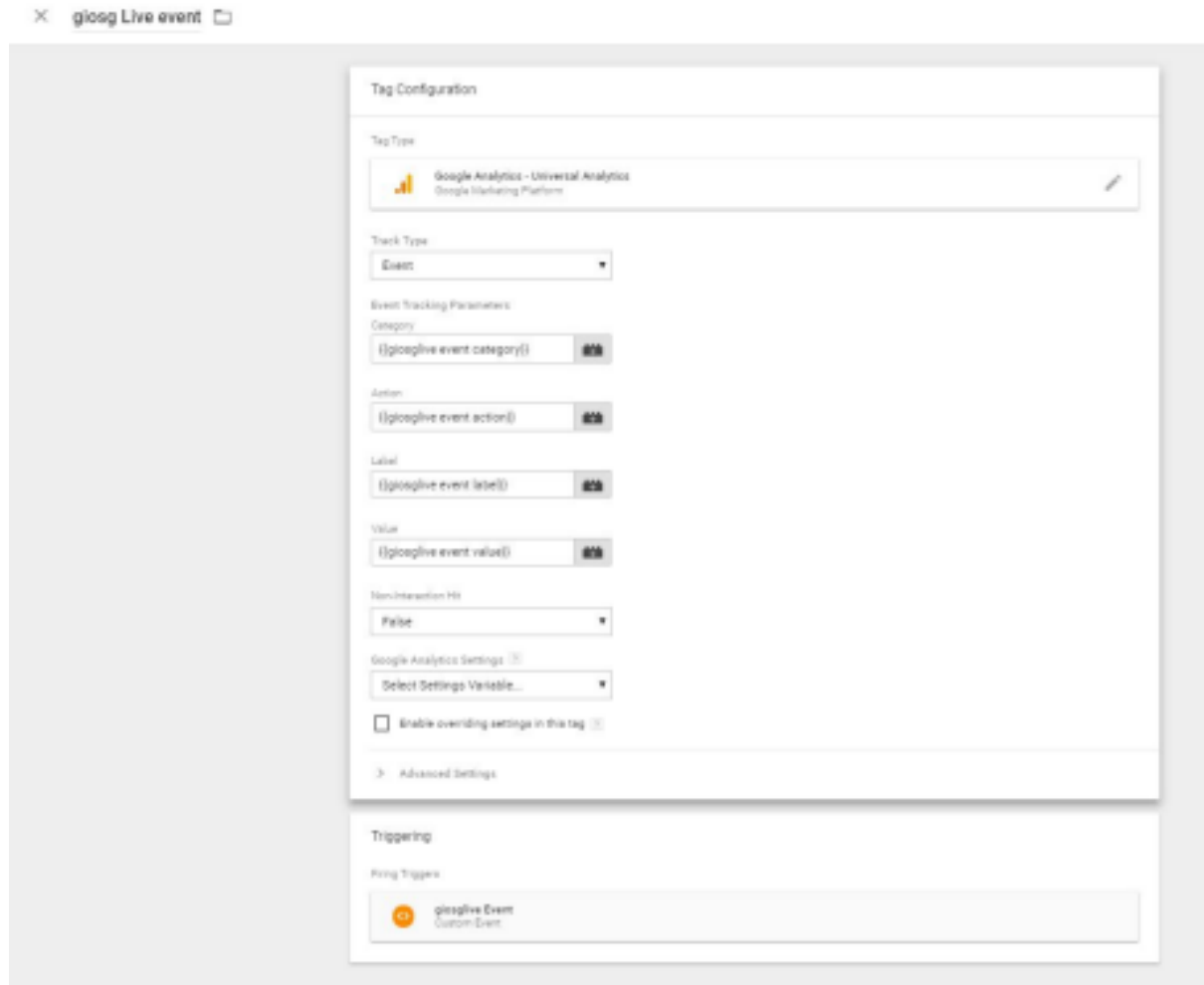
Repeat step 5 for following items:

| Variable Name | Variable Type | Data Layer Variable Name |
|------------------------|---------------------|--------------------------|
| giosglive event label | Data Layer Variable | eventLabel |
| giosglive event value | Data Layer Variable | eventValue |
| giosglive event action | Data Layer Variable | eventAction |
| | | |

6. After creating the Trigger and Variables we can create the actual Tag object. Select “Tags” from the left sidebar and click “New”.
7. On the tag creation page write “giosg live event” as the Tag Name.
8. Under “Tag Configuration” Select “Google Analytics – Universal Analytics(*a)” or if you are using GA4 choose “Google Analytics: GA4 Configuration(*b)” as the Tag Type(See next steps based on what analytics version you are using)

*a) Google Analytics – Universal Analytics

On the Track Type dropdown select “Event” and configure the event parameters as shown on the image below. If you haven’t defined your Google Analytics Settings, you can select “Enable overriding settings in this tag”



The screenshot shows the 'Tag Configuration' window for a 'giosg Live event' tag. The 'Tag Type' is set to 'Google Analytics - Universal Analytics'. The 'Track Type' is set to 'Event'. The 'Event Tracking Parameters' section includes fields for 'Category' ({{giosglive event category}}), 'Action' ({{giosglive event action}}), 'Label' ({{giosglive event label}}), and 'Value' ({{giosglive event value}}). The 'NonInteraction Hit' is set to 'False'. The 'Google Analytics Settings' section has a dropdown for 'Select Settings Variable...' and a checkbox for 'Enable overriding settings in this tag'. The 'Triggering' section shows 'Firing Triggers' with 'giosglive Event' (Custom Event) selected.

Now under “Triggering” select “giosglive Event” as the Firing Trigger and click Save.



Everything is now configured and you just need to publish your new configuration. You can publish your configuration with these instructions in the Overview tab :

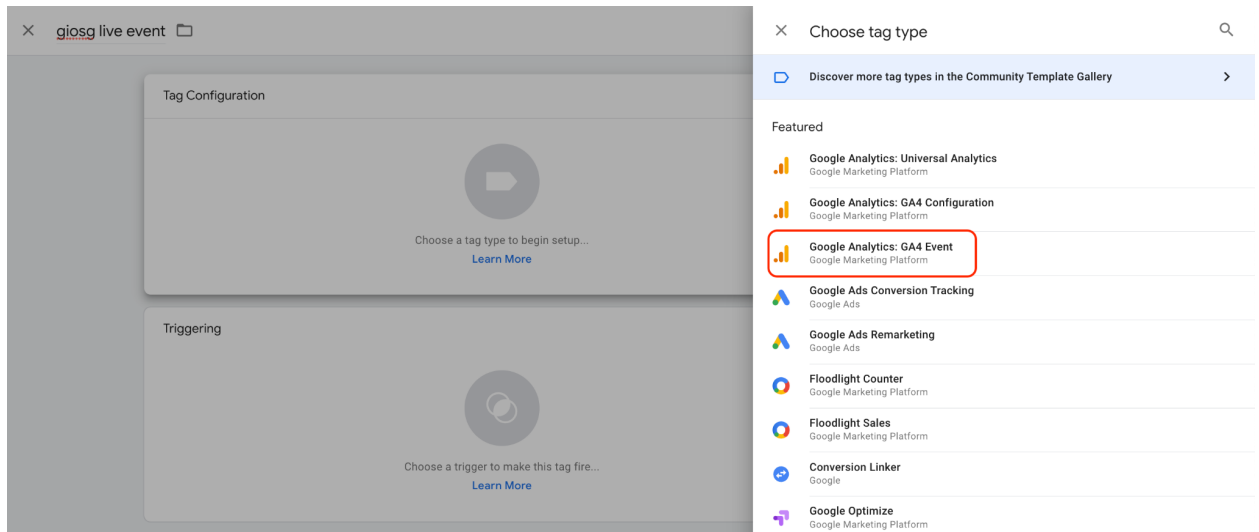
1. Click Submit at the top right hand side of the screen. The Submit Changes screen will appear, with options to publish the container and save a version of your container.
2. Select Publish and Create Version if it is not already selected.
3. Review the Workspace Changes section to see if your configuration appears as you expect.
4. Enter a Version Name and Version Description.
5. If you have Tag Manager configured to use multiple environments, use the Publish to Environment section to select which environment you'd like to publish to.
6. Click Publish. After this you can go to your website and login to your Analytics account. On your website click the giosg chat button and you should be able to see events appearing in your analytics real-time view.

***b) Google Analytics: GA4 Configuration**

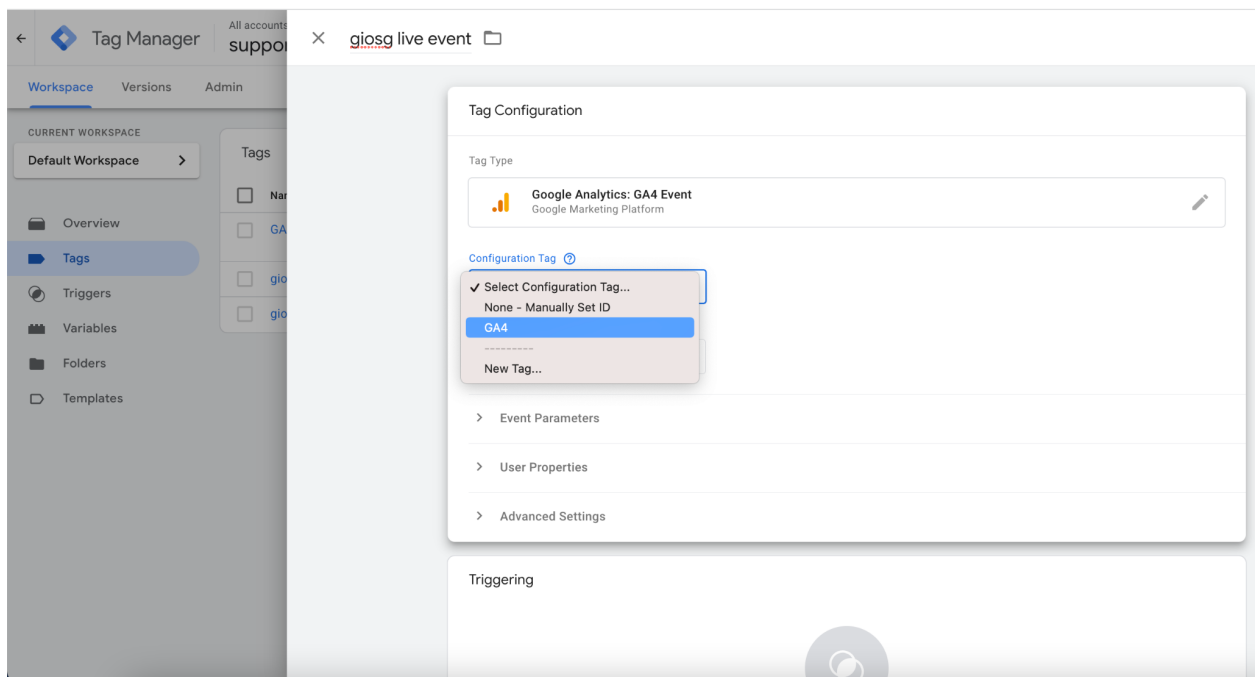
(This part of the guide assumes that your GA4 tracking service is already in use on your page)

After completing previous steps in this guide to connect the events from giosg through GTM to GA4, you need to create a new tag that uses the new GA4 Event type.

1. Create a new tag in GTM, Tags -> Create new Tag
2. Give the new tag a name: "giosg live event"
3. Under Tag type, choose Google Analytics: GA4 Event



4. Under Configuration Tag, either choose the tag you have created for your GA4 configuration or manually type in your GA4 tracking ID (this is so that the tag only triggers after the actual GA4 script has loaded and the events connect to the correct GA4 instance)



5. Under Event Name type in: giosglive

Tag Configuration

Tag Type

Google Analytics: GA4 Event
Google Marketing Platform

Configuration Tag ⓘ

GA4

Event Name ⓘ

giosglive

> Event Parameters

> User Properties

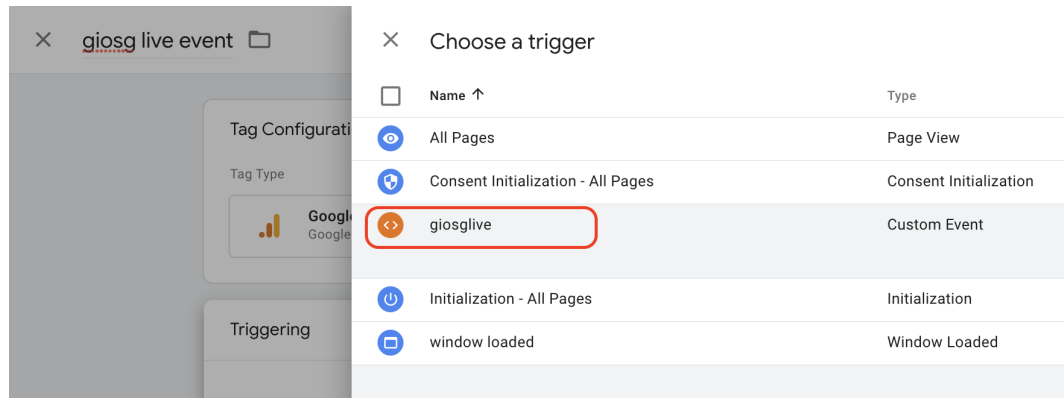
> Advanced Settings

Save

6. The following step is to add the event parameters to the tag so that you can filter the events in GA4. Click on “Event parameters” and add following events there and choose the respective trigger for the event (defined in step 5).

| Event Name | Event Value |
|-----------------------|----------------------------|
| giosg_event_label | {{giosglive event label}} |
| giosglive_event_value | {{giosglive event value}} |
| giosg_action_type | {{giosglive event action}} |

7. Leave all the other options as they are by default (User Properties and Advanced Settings)
8. Last thing before saving the new tag is to choose the trigger for this tag. You should use the trigger created earlier named: giosglive



9. When all this is set, the tag should look like this:

The image shows the 'Tag Configuration' window for a 'Google Analytics: GA4 Event' tag. The 'Tag Type' is 'Google Analytics: GA4 Event'. The 'Configuration Tag' is set to 'GA4'. The 'Event Name' is 'giosglive'. The 'Event Parameters' section shows three parameters: 'giosg_event_label', 'giosglive_event_value', and 'giosg_action_type'. The 'Triggering' section shows the 'giosglive' trigger.

Tag Configuration

Tag Type: Google Analytics: GA4 Event

Configuration Tag: GA4

Event Name: giosglive

Event Parameters

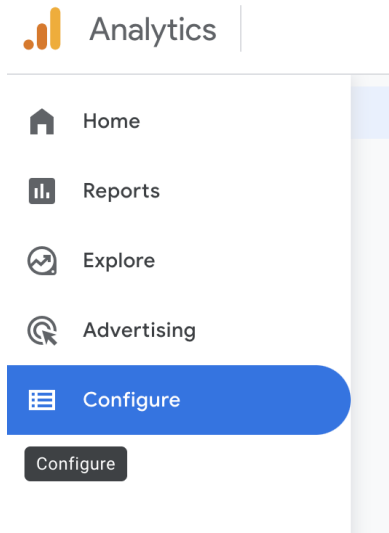
| Parameter Name | Value |
|-----------------------|----------------------------|
| giosg_event_label | {{giosglive event label}} |
| giosglive_event_value | {{giosglive event value}} |
| giosg_action_type | {{giosglive event action}} |

Triggering

Firing Triggers: giosglive (Custom Event)

10. Save the new tag and don't forget to submit the new changes to your page

Please note that it might take up to 24hrs before the events show up in Google Analytics. To track specific custom events in GA 4, as per Google's documentation, you need to add a custom event metric in GA4 to track giosg events: *"Custom events don't show up in most standard reports so you need to set up custom reports or explorations for meaningful analysis"* (https://support.google.com/analytics/answer/9322688?hl=en&ref_topic=9756175) To create a custom event metric in GA4, navigate to the "Configure" page from the menu on the left side of the page. Here you can create an event for each giosg event.



Perform the following steps for each giosg event (listed below):**

- chatbutton
- autosuggested
- hasChatted
- autosuggestedHasChatted
- visitorReachedGoal

1. Choose "Events" -> Create Event
2. In the Event configuration give this event a new name.
3. Add event parameter called: giosg_action_type
4. Choose operator: Equals
5. Connect matching value to event e.g autosuggestedHasChatted

If you'd like to track giosg goals individually, create separate events for each goal and add a second condition to each goal event, as in the picture below. For the value of parameter giosg_event_label, add the name of the goal you'd like to track.

Configuration

Custom event name ⓘ

goal_tracking

Matching conditions

Create a custom event when another event matches ALL of the following conditions

| Parameter | Operator | Value | |
|-------------------|----------|-----------------------|---|
| giosg_action_type | equals | visitorReachedGoal | ⊗ |
| giosg_event_label | contains | Name of the goal here | ⊗ |

Add condition

Parameter configuration

☒ Copy parameters from the source event

Modify parameters ⓘ

Add modification

** below you can find Images of the setup page for each event type

Chat button clicked event

Configuration

Custom event name ⓘ

Chat_button_clicked

Matching conditions

Create a custom event when another event matches ALL of the following conditions

| Parameter | Operator | Value |
|-------------------|----------|------------|
| giosg_action_type | equals | chatbutton |

Add condition

Parameter configuration

☒ Copy parameters from the source event

Modify parameters ⓘ

Add modification

Autosuggest event

Configuration

Custom event name ⓘ

Autosuggested

Matching conditions

Create a custom event when another event matches ALL of the following conditions

| Parameter | Operator | Value |
|-------------------|----------|---------------|
| giosg_action_type | equals | autosuggested |

Add condition

Parameter configuration

☒ Copy parameters from the source event

Modify parameters ⓘ

Add modification

Visitor Has chatted event

Configuration

Custom event name ⓘ

RealChat

Matching conditions

Create a custom event when another event matches ALL of the following conditions

| Parameter | Operator | Value |
|------------------------------|-------------------|-----------------------|
| <div>giosg_action_type</div> | <div>equals</div> | <div>hasChatted</div> |

Add condition

Parameter configuration

☒ Copy parameters from the source event

Modify parameters ⓘ

Add modification

Visitor Has chatted with autosuggest event

Configuration

Custom event name ⓘ

autosuggestedwithRealChat

Matching conditions

Create a custom event when another event matches ALL of the following conditions

| Parameter | Operator | Value |
|------------------------------|-------------------|------------------------------------|
| <div>giosg_action_type</div> | <div>equals</div> | <div>autosuggestedHasChatted</div> |

Add condition

Parameter configuration

☒ Copy parameters from the source event

Modify parameters ⓘ

Add modification

Visitor reached goal event

Configuration

Custom event name ⓘ

goals

Matching conditions

Create a custom event when another event matches ALL of the following conditions

| Parameter | Operator | Value |
|------------------------------|-------------------|-------------------------------|
| <div>giosg_action_type</div> | <div>equals</div> | <div>visitorReachedGoal</div> |

Add condition

Parameter configuration

☒ Copy parameters from the source event

Modify parameters ⓘ

Add modification