



## Pushing events to Google Analytics from giosg

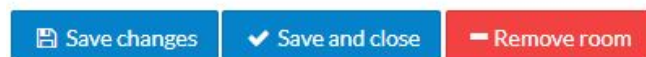
It is possible to integrate giosg to your existing Google Analytics or Google Tag Manager account. You can find list of events pushed to analytics under “Events pushed to Google Analytics”.

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 your 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" in to browser console, and it returns other than undefined, select "Google Analytics (ga.js)".

**Note:** ga.js is a legacy library. Google recommends to migrate 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 it is found.

If the type of that object is "function" then it is called with six parameters, for example: myTrackerObj(eventSender, eventName, eventDataAsString, eventDataAsPlainObject, eventValue, isUserInteraction);

If the type of that object is "object", we then assume that the object is an array containing a "push" method (i.e. in the same form as the normal dataLayer object). In this case the object's push function is used. For example:

```
myTrackerObj.push([eventSender, eventName, eventDataAsString,
eventDataAsPlainObject, eventValue, isUserInteraction]);
```

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

```
window.myTrackObj = function (event, eventName, dataString, dataPlainObject,
eventValue, userInteraction) { console.log(event, eventName, dataString,
dataPlainObject, eventValue, userInteraction); };
```

Also if your dataLayer object is renamed to something else (e.g. customLayer) and you want to send the data through your custom layer, you’ll have to use the Custom tracker object to correctly send the data through your customized dataLayer. You’ll first have to define the function to the window object, and inside the function call your customized dataLayer’s push -method. Here’s an example of how:

```
window.myTrackerObj = function (event, eventName, dataString, dataPlainObject,
eventValue, userInteraction) {
  const layerData = {
    'event': eventSender,
    'eventCategory': eventSender,
    'eventAction': eventName,
    'eventLabel': eventDataAsString,
    'nonInteraction': (userInteraction ? 'False' : 'True')
  };

  if (eventValue) {
    layerData.eventValue = eventValue;
  }

  layerData = Object.assign(layerData, dataPlainObject);
  window.customLayer.push(layerData);
}
```

## 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]"
```

## Lead form was shown to visitor

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

## Lead form was completed

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

## Visitor reached goal

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

## Why is the 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 the correct “Tracking service” type 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.

**TRACKING SETTINGS**

**Enable tracking analytics events**  
Push Giosg event data to Google Analytics, Google Data Layer, or another analytics solution. See [documentation](#) for details.

Tracking service

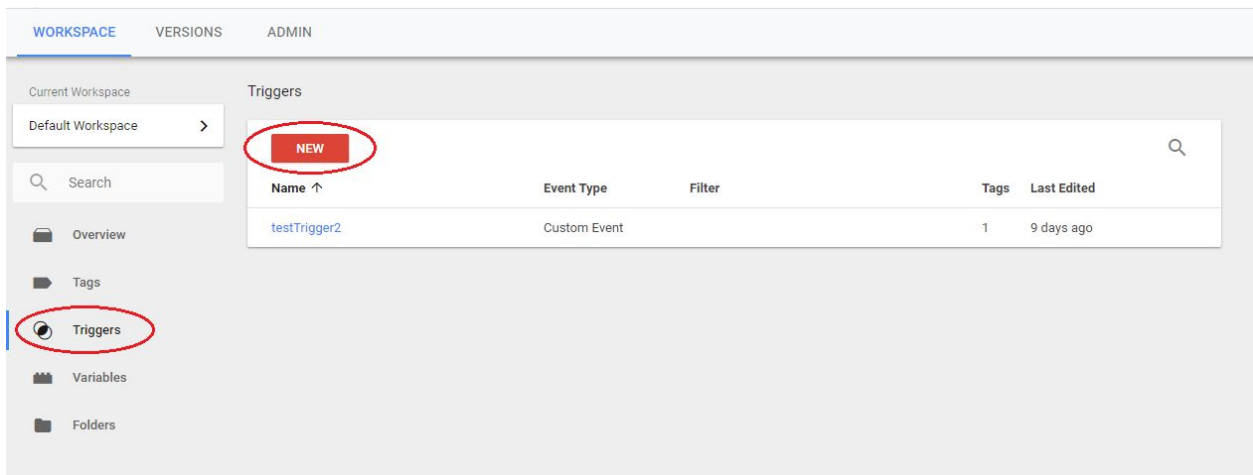
Google Tag Manager Data Layer (dataLayer.push) ▼

**Enable single page app (AJAX app) tracking**  
Enable single page application tracking. All changes in URL are considered to be page changes.

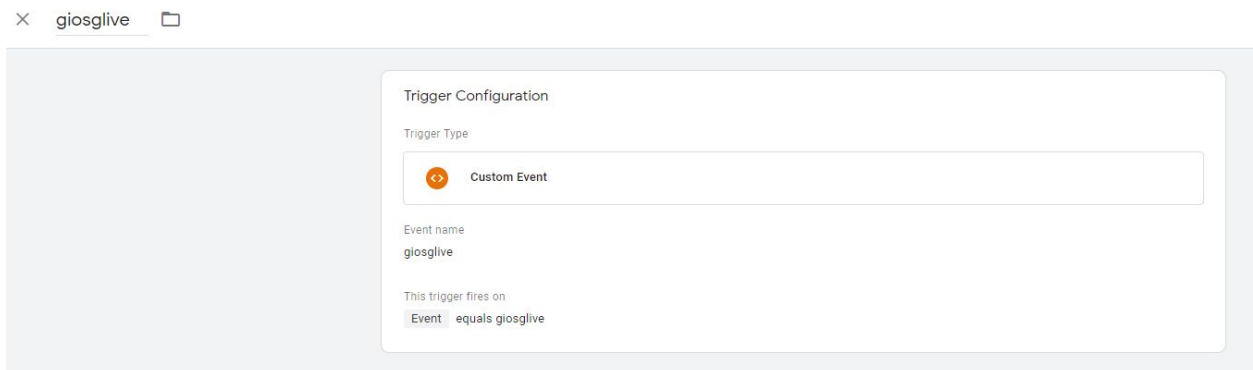
## Configuring Google Tag Manager

To setup 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:

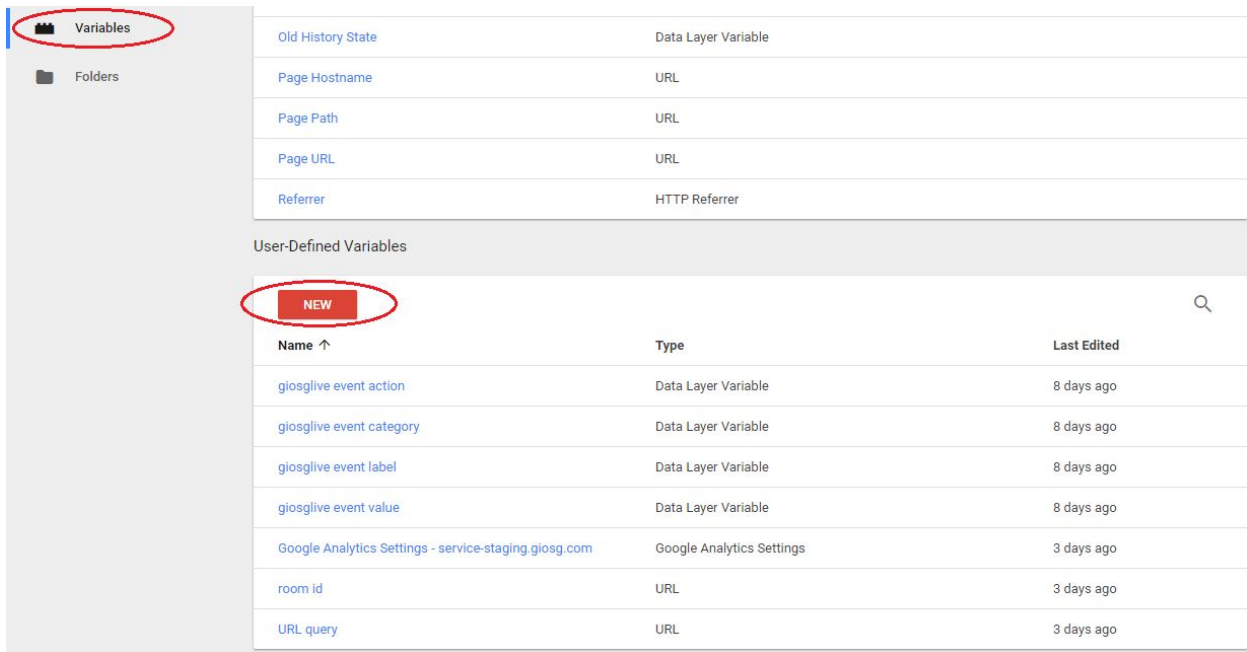
1. First login to your tag manager account at <https://www.google.com/tagmanager/>
2. First create a trigger for the the tag that we will create later. Under “Triggers” menu click “New”.



3. Select “Custom Event” as the trigger type and use “giosglive” as the Trigger name, and as Event name use “giosglive”. 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 box type “giosglive”. When done, click save.

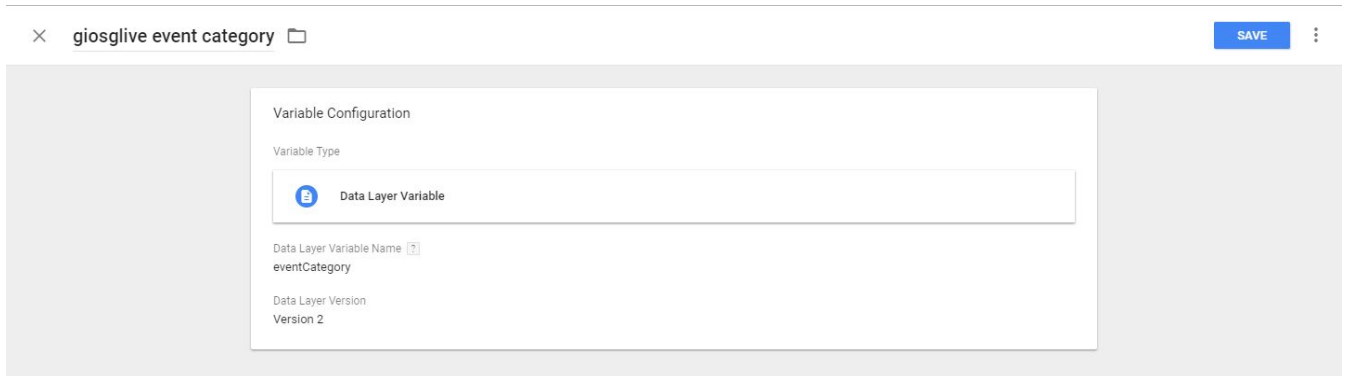


4. Next we need to define the variables. Select “Variables” from the left sidebar, and under “User-Defined Variables” click “New”.



Name ↑	Type	Last Edited
<a href="#">giosglive event action</a>	Data Layer Variable	8 days ago
<a href="#">giosglive event category</a>	Data Layer Variable	8 days ago
<a href="#">giosglive event label</a>	Data Layer Variable	8 days ago
<a href="#">giosglive event value</a>	Data Layer Variable	8 days ago
<a href="#">Google Analytics Settings - service-staging.giosg.com</a>	Google Analytics Settings	3 days ago
<a href="#">room id</a>	URL	3 days ago
<a href="#">URL query</a>	URL	3 days ago

5. Type “giosglive event category” as the Variable Name. Select “Data Layer Variable” as the variable type. As the Data Layer Variable Name write “eventCategory” and save.



Variable Configuration

Variable Type

**Data Layer Variable**

Data Layer Variable Name

Data Layer Version  
Version 2

**SAVE**

Repeat step 5 for following items:

Variable Name	Variable Type	Data Layer Variable Name
giosglive event category	Data Layer Variable	eventCategory
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”. On the tag creation page write “giosglive event” as the Tag Name. Under “Tag Configuration” Select “Google Analytics – Universal Analytics” as the Tag Type. 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” and set your Tracking ID and other settings here.

Tag Configuration

Tag Type

Google Analytics - Universal Analytics  
Google Marketing Platform

Track Type

Event

Event Tracking Parameters

Category

{{giosglive event category}}

Action

{{giosglive event action}}

Label

{{giosglive event label}}

Value

{{giosglive event value}}

Non-Interaction Hit

False

Google Analytics Settings

Select Settings Variable...

Enable overriding settings in this tag

> Advanced Settings



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



8. 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.