



Getting started with Google Analytics 4

Installation, Events, Conversions

Created by



Julius Fedorovicius
Founder of [Analytics Mania](#)
julius@analyticsmania.com
[@fedorovicius](#)



I can help you learn GTM and GA4

Hi, my name is Julius. I founded Analytics Mania in late 2016. Since then, the blog has grown to over 100 000 monthly visits. Here I share my knowledge, experiments, learnings related to web analytics, mainly Google Tag Manager (GTM) & Google Analytics (GA).

I also run the Google Tag Manager community on Facebook, where awesome people from around the world help each other.

Last but not least, I actively help others to learn Google Tag Manager by running workshops and [GTM online courses](#) where you can:

- Learn GTM in the fastest way possible
- Save a lot of time and money
- Get practical knowledge/skills
- Become a tag management professional
- Get quick help from me when you're stuck

Table of Contents

Chapter I. Install GA 4 with GTM.....	4
#1.1. If you are currently using the older version of GA	5
#1.2 Create a Google Analytics 4 Property & Data Stream	6
#1.3. Install Google Analytics 4 with Google Tag Manager	9
#1.4. DebugView in Google Analytics 4	11
#1.5. Publish your changes in Google Tag Manager	16
#1.6. Google Analytics 4 Realtime Reports	16
#1.7. Frequently asked questions	17
Chapter II. Track events with GA4	19
#2.1. Bye-bye to Event Category, Action, and Label.....	20
#2.2. 4 categories of events in Google Analytics 4	21
#2.2.1. Category #1. Automatically captured events.....	23
#2.2.2. Category #2. Events tracked via Enhanced Measurement	23
#2.2.3. Category #3. Recommended events	28
#2.2.3.1. Example of a Recommended Event: login tracking	29
#2.2.4. Category #4. Custom events	36
#2.2.4.1. Example of a Custom Event in GA4: Menu Link Click	37
#2.2.4.2. IMPORTANT: Register custom definitions.....	45
#2.3. Useful Feature: Create new events (GA4 interface)	47
#2.4. Another useful feature: Modify Events (in GA4 interface)	52
#2.5. Test your events in GA4	54
#2.6. Where can I find events data in GA4 reports?	56
#2.7. How to plan your events in GA4.....	62
#2.8. Event limits in Google Analytics 4	65
Chapter III. Conversions in GA4.....	66
#3.1. What is a conversion in general?	67
#3.2. Goodbye, goals. Hello, conversions!	67
#3.3. Predefined Conversions	69
#3.4. Conversions that can be switched on	70
#3.5. Create events and mark them as conversions.....	71
#3.6. Or try planning your event naming convention better	75
#3.7. Check the data in Google Analytics 4 DebugView	76
#3.8. Where can I see the conversion data in GA 4 reports?	77

Chapter I

Install GA 4 with GTM

In late 2020, Google renamed its App + Web properties to Google Analytics 4 and officially graduated from beta (at least, that's what Google is saying). This means that from now on, the default property (when you create it) is GA4. It is a completely revamped platform (compared to the previous versions). New interface, new tag template in Google Tag Manager, new data model, etc.

In this chapter, we'll take a look at one of the basic (but fundamental) topics - how to install Google Analytics 4. And we'll do that with Google Tag Manager.

Here's the process of how to install Google Analytics 4 with Google Tag Manager:

- Create a new GA4 property
- Create a data stream (in this e-book, I focused on the Web stream)
- Copy the Measurement ID
- Create a Google Analytics 4 Configuration tag
- Paste the Measurement ID and set the tag to fire on All Pages
- Preview/test the new tag
- Publish the changes in Google Tag Manager

#1.1. If you are currently using the older version of GA

What if you are already using Universal Analytics (the older version of GA) on your website? Should you ditch it completely and replace it with Google Analytics 4? Should you use them both?

To answer that, [I have published a blog post \(and a video tutorial\)](#) on this topic.

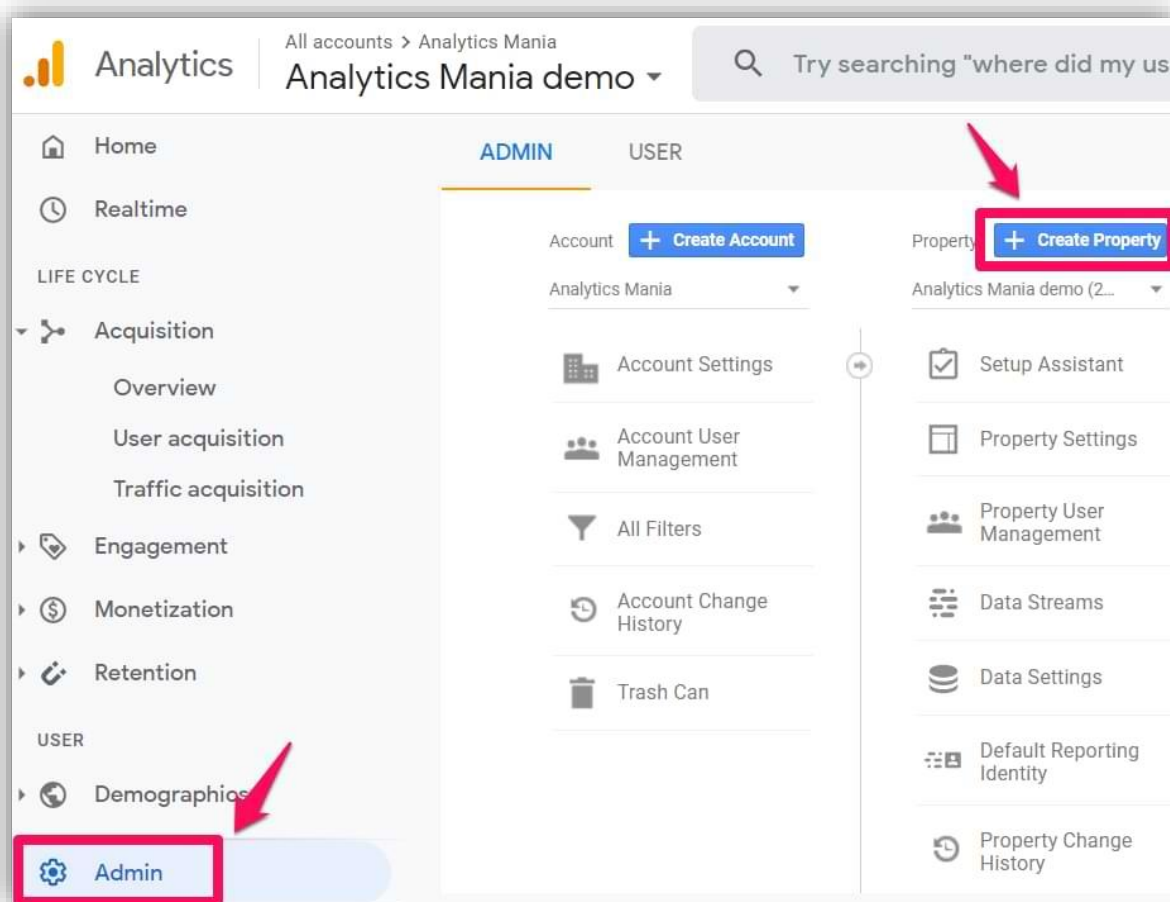
In a nutshell, if you are reading this e-book in late 2020 or 2021, chances are that Google Analytics 4 is still not mature enough to be used as a standalone web tracking platform. Hence you should use both tools (UA and GA4) at the same time.

But if you are reading this e-book in late 2021 or even more in the future, it will probably be enough to completely switch to Google Analytics 4.

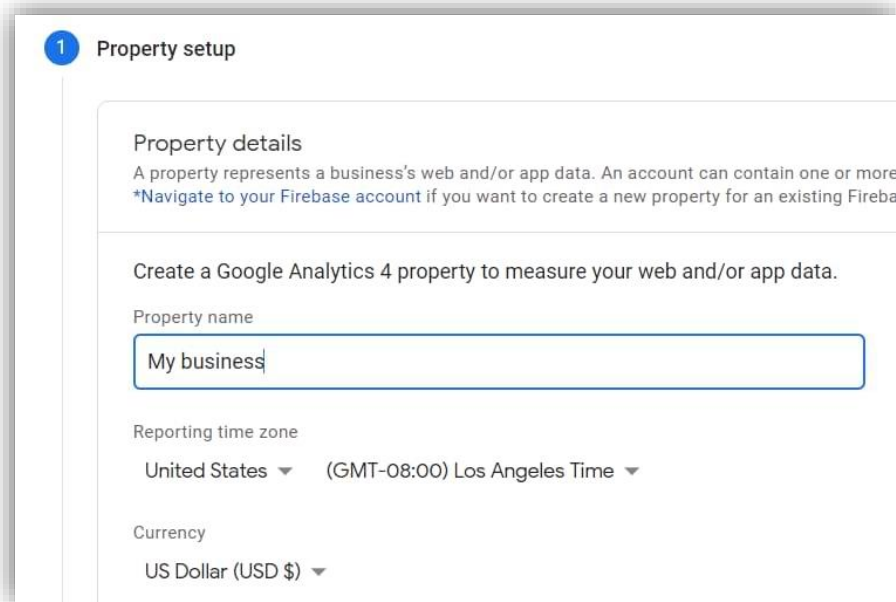
And now, let's learn how to install Google Analytics 4 with Google Tag Manager.

#1.2 Create a Google Analytics 4 Property & Data Stream

If you want to create a new Google Analytics 4 property (and you are not using the [Upgrade to GA4](#), then go to the *Admin* section of your Google Analytics interface (by clicking the *Admin* at the bottom-left corner) and then (in the *Property* section) click **Create Property**.

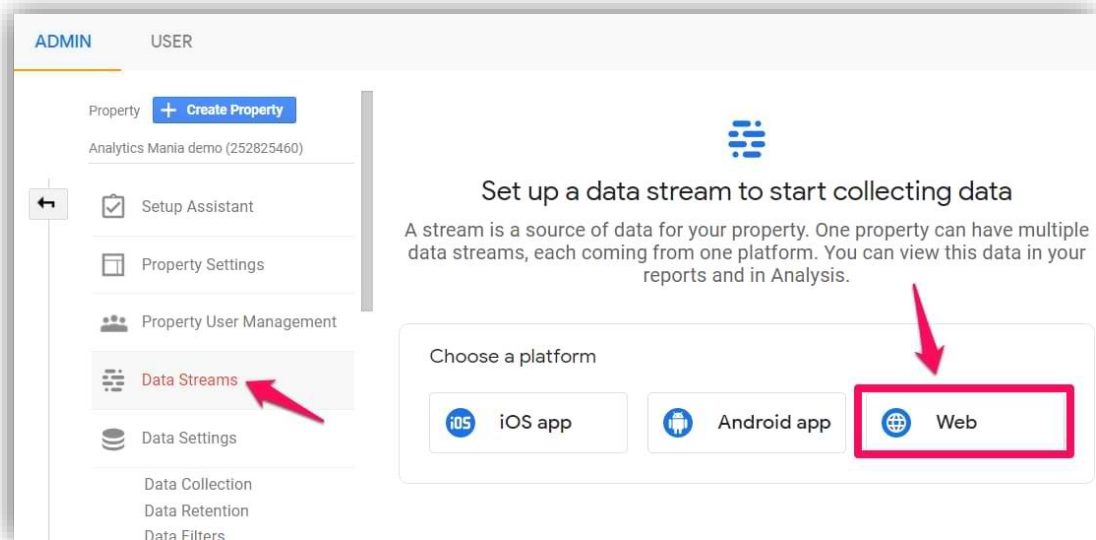


Then enter the name of your property. It might be the name of your website, of your company, a brand, etc. Choose your company's country, reporting time zone, and the main currency that your business operates in.



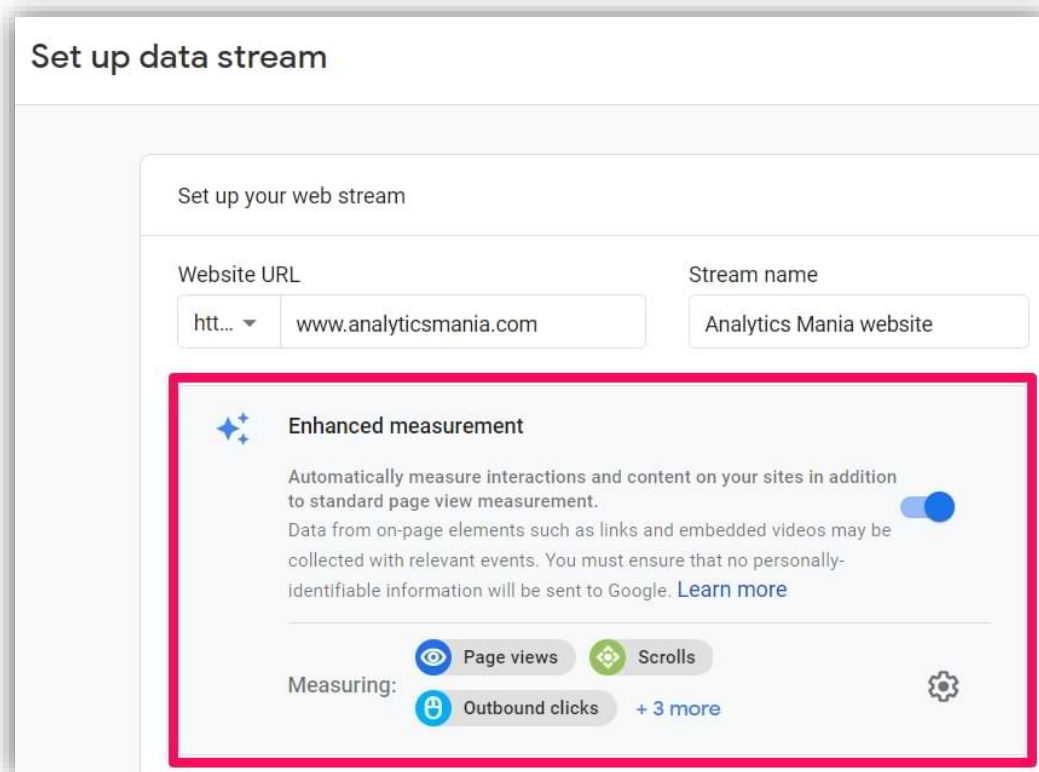
Then press *Next*, answer several questions, click *Create* and your new property will be ready. The next step to complete is to configure your first data stream. It is a data source from which events will be sent to your Google Analytics 4 property. You can have multiple data sources in a single property. For example, 3 web properties, 1 for Android app, and 1 for an iOS app.

In this e-book, I will focus on a **Web** stream. Select it.



Then enter the URL of your website (for example, <https://www.mywebsite.com>). Note, that the protocol (*https*) is already selected. After that, enter the name of your website.

When you create a web data stream in Google Analytics 4, you have an option to enable/disable Enhanced Measurement. It is designed to help marketers get as many events in the reports as possible without the need to cooperate with developers or configure them in Google Tag Manager.



Set up data stream

Set up your web stream


Website URL:

Stream name:

Enhanced measurement

Automatically measure interactions and content on your sites in addition to standard page view measurement.

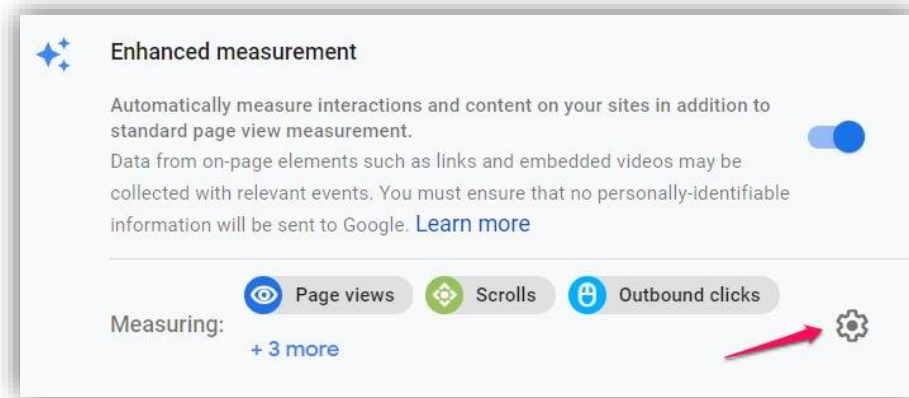
Data from on-page elements such as links and embedded videos may be collected with relevant events. You must ensure that no personally-identifiable information will be sent to Google. [Learn more](#)

Measuring: Page views Scrolls Outbound clicks + 3 more 

By default, this feature is enabled and will automatically track the following events:

- Page view (event name: *page_view*)
- Scroll (event name: *scroll*)
- Outbound link click (event name: *click* with the parameter *outbound: true*)
- Site search (event name: *view_search_results*)
- Video Engagement (events: *video_start*, *video_progress*, *video_complete*)
- File Download (event name: *file_download*)

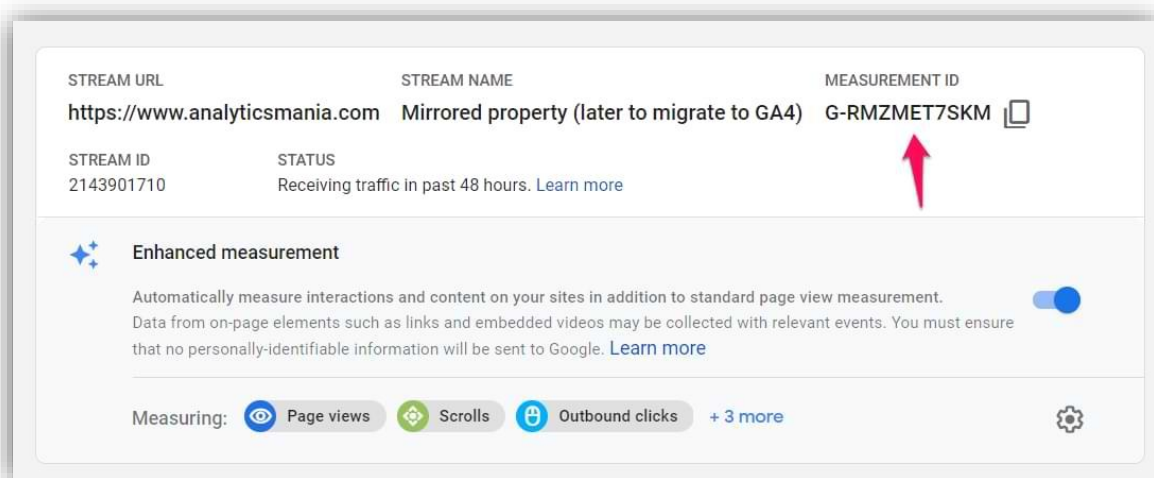
If you want, you can disable/enable events individually. You can do that by clicking the gear icon in the *Enhanced Measurement* section and then clicking toggle buttons.



When you're ready, press the **Create stream** button.

#1.3. Install Google Analytics 4 with Google Tag Manager

When you create a data stream (web), you will see a Measurement ID.



At the top-right corner, you will see the tracking ID of your new property. Copy it. We'll need to use it in Google Tag Manager.

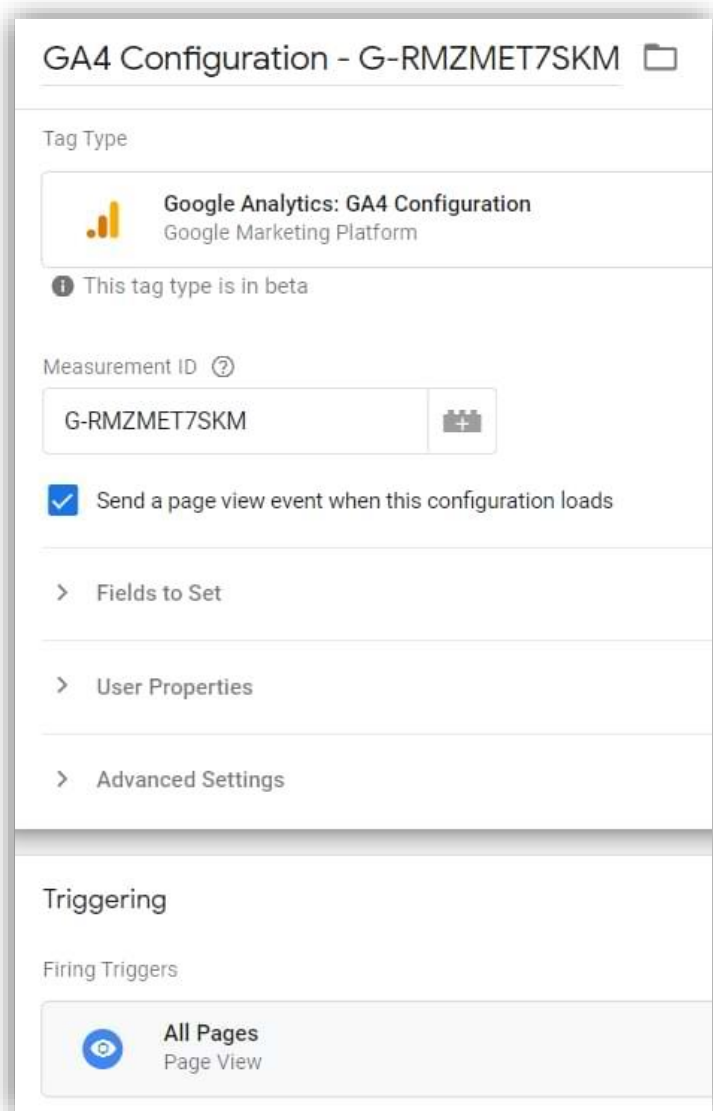
By the way, if you ever get lost in the GA4 interface and you need to revisit this page, go to *Admin > Data Streams >* choose your most recently created stream, and then you will return to the same window that I have displayed in the screenshot above.

Then go to your Google Tag Manager container (if you haven't worked with it, [here's a tutorial on how to get started](#)) > *Tags > New* and choose GA4 configuration.

In the Measurement ID field, enter the ID that you copied in the GA4 interface. Keep the *Send a page view event when this configuration loads* if you want to automatically track pageviews.

Usually, that's ok keep it enabled but in some situations, e.g. on single-page applications, it is more recommended to disable the pageview checkbox and create just a standalone configuration tag. You can [read more about that here](#).

In the Triggering section, select *All Pages* and then name the tag, e.g. *GA4 - Pageview*.

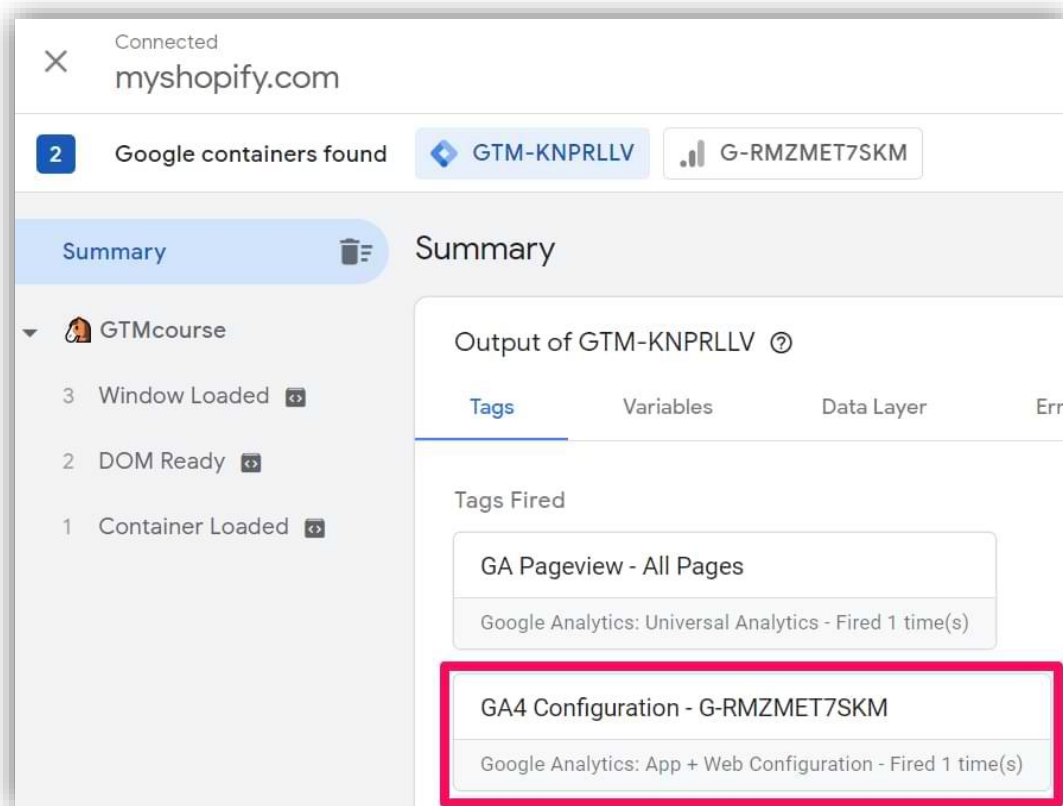


#1.4. DebugView in Google Analytics 4

Please memorize this chapter of the e-book. I will refer to DebugView multiple times throughout the entire book because it's an essential part of making sure that your data was properly sent to Google Analytics 4.

Enable the [GTM preview mode](#) to test your changes.

Once you enable the preview mode, you should see the new GA4 tag among tags that fired.

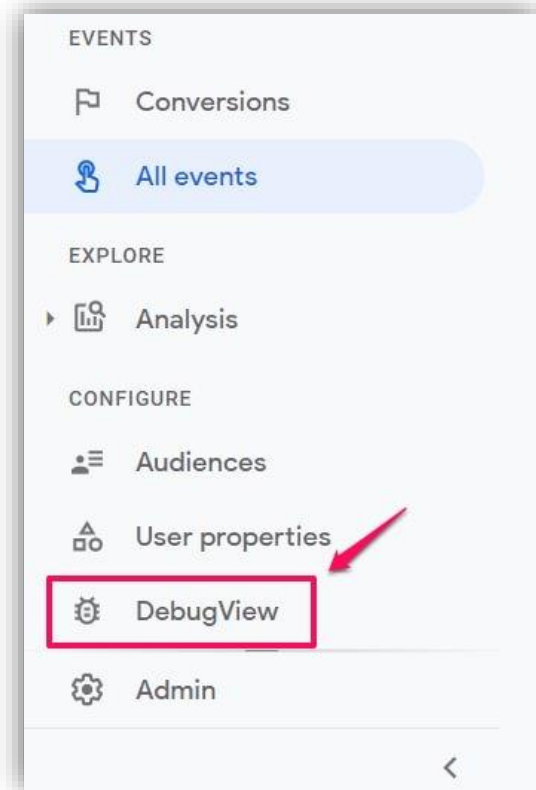


Once you configure your events, it's time to test them. The primary feature built for debugging GA 4 data is the *DebugView* section. You can find it at the bottom left corner of your GA4 interface. Click it.

Most likely, you will not see any data there (yet). That's because that data in the *DebugView* is visible ONLY when the debug mode in GA4 is enabled. Do not mix this with the GTM Preview and Debug mode. They are two different beasts.

To enable the debug mode in GA4, you have several options (any of them will work):

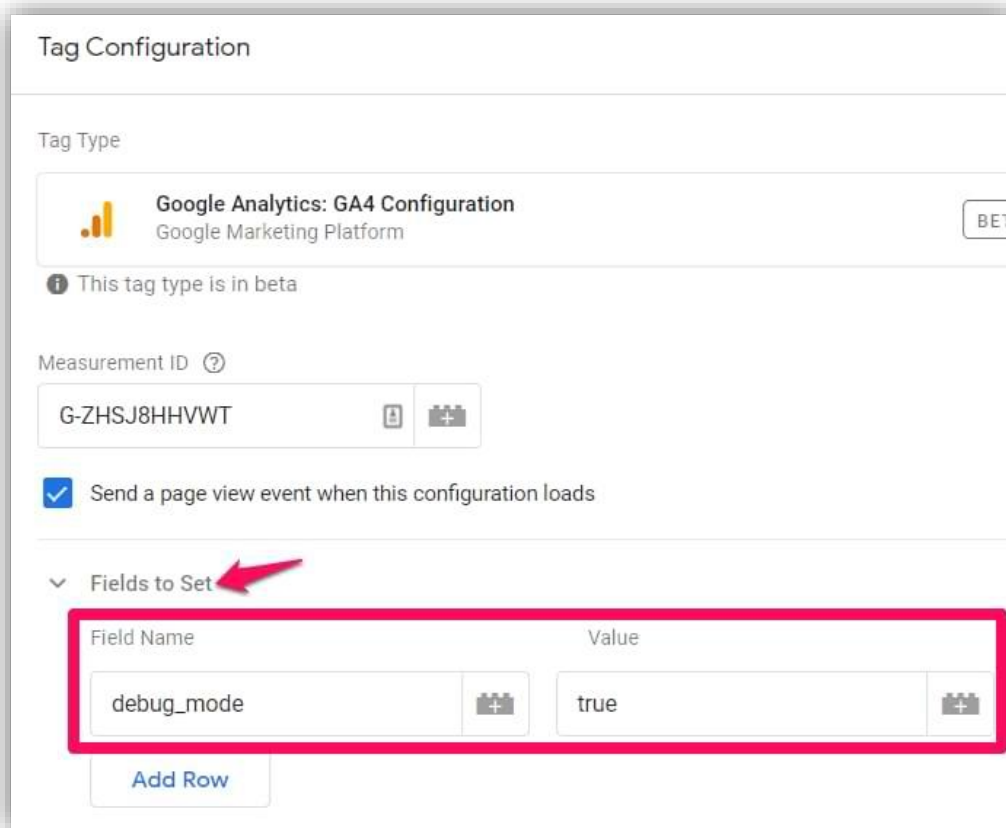
- Enable the GA debugger Chrome extension
- OR send a *debug_mode* parameter together with an event
- OR have enabled Google Tag Manager's Preview mode on a page that you're debugging



In the case of the first option, [install the extension here](#), and then click its icon (so that you can see the **ON** ribbon). From this moment, you will start seeing your events coming into the DebugView.

The other option is to send a *debug_mode* parameter with every event that you wish to see in the *DebugView*.

In Google Tag Manager, you can do that by opening the GA4 Configuration tag and adding the following parameter:



If the *debug_mode* parameter is set to true, the event will be seen in *DebugView*. If you are using a hardcoded gtag.js, [here are the instructions](#) on how to include the *debug_mode* parameter.

But since we are working with Google Tag Manager, having the [GTM Preview mode](#) enabled automatically sets the *debug_mode* to *true*, thus the data will start appearing in the GA4 *DebugView*.

When you start seeing data in the *DebugView*, you can click on every individual event and then a list of parameters will be displayed.

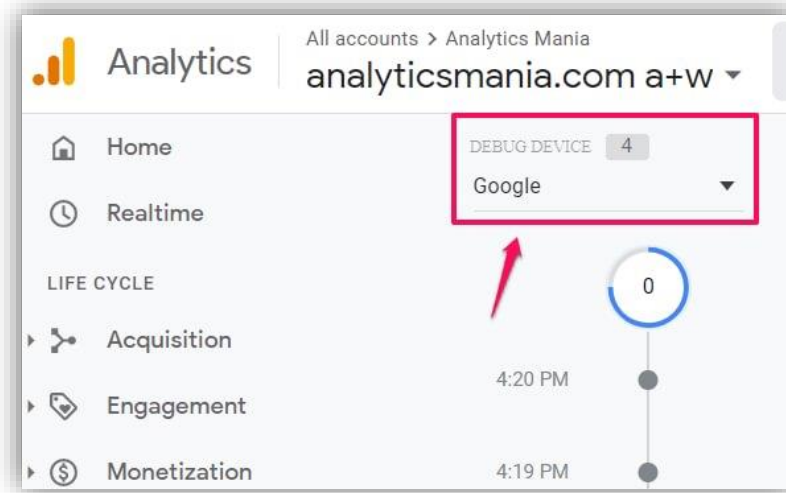
Click on that parameter to see the value that was received by GA4. Now that is some granular debugging!

The screenshot displays the Google Analytics DebugView interface. On the left, a vertical timeline shows a sequence of events: 'user_engagement' at 10:31:09 PM, 'page_view' at 10:31:10 PM, a 50s delay, 'page_view' at 10:32:00 PM, a 31s delay, 'menu_click' at 10:32:31 PM, 'page_view' at 10:32:32 PM, 'scroll' at 10:32:33 PM, and a 4s delay. A blue circle with '0 new' is at the top. On the right, the details for the selected 'page_view' event are shown, including 'User properties' and parameters like 'ga_session_number', 'medium', 'page_location', 'page_referrer', 'page_title', and 'source'. Two red arrows point to the 'page_view' event in the timeline and the 'page_location' parameter in the details panel. Text annotations in red with white outlines read: '1. Click on the event' and '2. Click on a parameter to see its value'.

However, I have noticed some delays between the event actually happening on a website and then appearing in *DebugView*. Sometimes, I have to wait for several minutes until the data come in. That's a bit unfortunate and hopefully, the team behind Google Analytics 4 will improve this in the future.

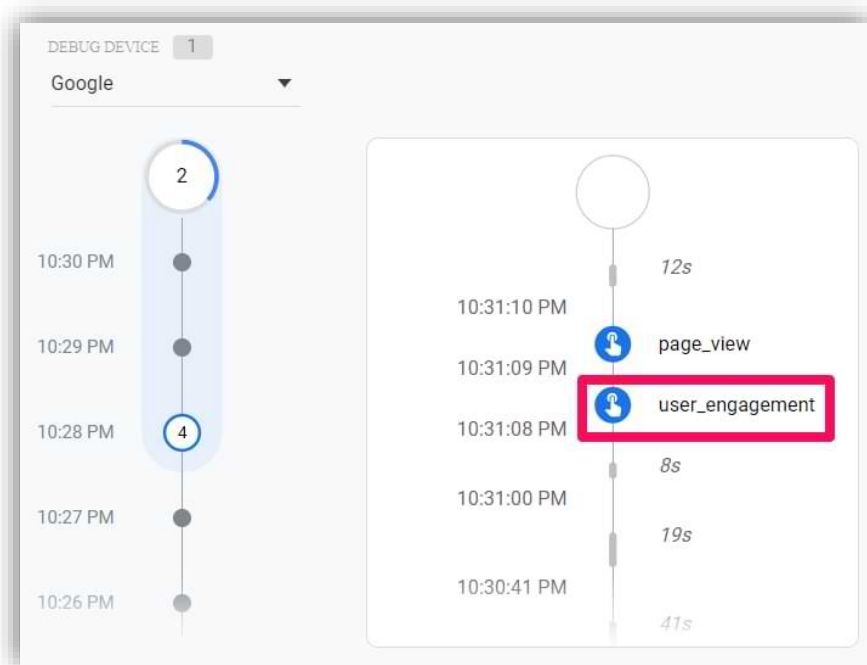
Sometimes, refreshing the *DebugView's* page helps too.

Also, make sure that you have selected the correct Debug device in the top left corner.



If multiple visitors have enabled the debug view (e.g. they all have enabled the Chrome extension), you will see multiple devices there and it might require some time to find yourself. This especially applies to my blog, when many of my readers have enabled the GA Debugger Extension and I have to guess which device (out of the other 15) is mine :)

Anyway, once you start seeing data in *DebugView*, things will look like this.

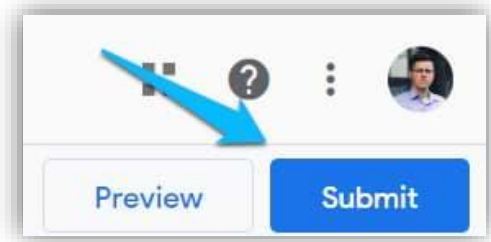


#1.5. Publish your changes in Google Tag Manager

Once you made sure that the data is coming in and it is displayed properly, you should submit your GA4 changes in the GTM container and publish it.

You can do that by clicking the SUBMIT button in the top right corner and then complete all the other steps that the user interface asks you to do.

After that, you should soon start seeing the new data coming in your real-time reports as well.

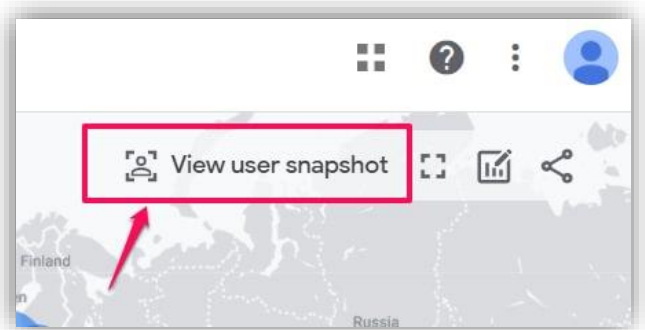


#1.6. Google Analytics 4 Realtime Reports

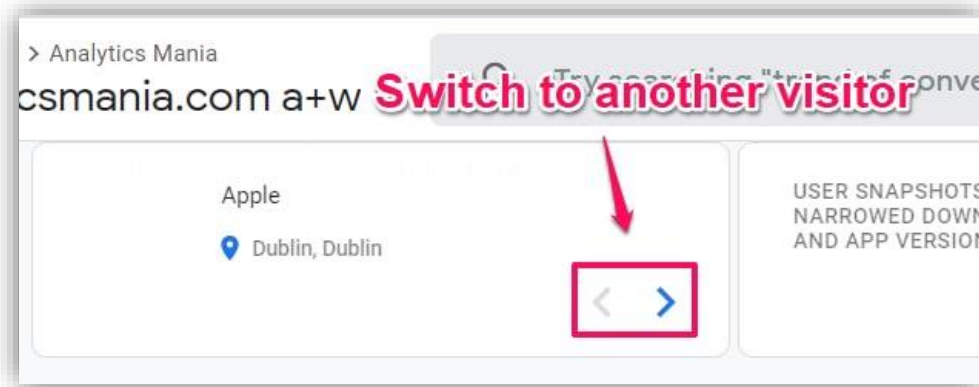
On the left sidebar of the Google Analytics 4 interface, go to *Realtime*. This is where you will see the data coming into your reports. Unlike in the previous version (Universal Analytics), the new report offers you capabilities to see the data on a much more granular level.

First, you will see a map and a bunch of cards with traffic sources, most popular events, the number of users in the last 30 minutes (by the way, Universal Analytics real-time report shows the number of users in the last 5 minutes).

You can also take a look at the snapshot of an individual user. You can do that by clicking the **View user snapshot** button in the top right corner.



Then, you will see a stream of all events of that particular user, you can click on them (just like in the *DebugView*) to see things on a more granular level. If you wish to look at another user/visitor, you can click the button here:



To exit the snapshot, simply press the **Exit snapshot** button in the top right corner.

#1.7. Frequently asked questions

Here are some additional questions that you might have and I wanted to address.

What if I already have Universal Analytics installed on my site? Please refer to my blog post on [how to upgrade to Google Analytics 4](#). But a short version is that you can use both GA versions at the same time. In the long run, it will be recommended to keep only Google Analytics 4 (but that recommendation will be reasonable probably just in late 2021).

What if I already have Google Analytics 4 tracking code hardcoded in the website's source code. I would highly recommend implementing GA4 (or any other JavaScript-based marketing/analytics tool) via Google Tag Manager because of its flexibility. [Click here to learn more about GTM benefits](#). But a short version goes like this: your developers should remove the hardcoded tracking code from your website and you should implement Google Analytics 4 via Google Tag Manager. My [Google Tag](#)

[Manager Course for Beginners](#) covers the process and tips on how to migrate hardcoded tools to Google Tag Manager.

If I already have Universal Analytics implemented in Google Tag Manager, should I create a new GTM container for Google Analytics 4? No, you don't need to create a new GTM container just for Google Analytics 4. Use your existing Google Tag Manager container for the new GA version as well. Google Analytics 4 requires completely different GA tags but they can definitely co-exist with Universal Analytics tags in the same container.

=====

After you have installed Google Analytics 4 on your website, the next step is to implement event tracking. They are the backbone of your GA setup. Every interaction that you want to measure is an event.

And when we move towards conversion tracking, you will learn that a conversion in GA 4 is just an event with some VIP status.

Chapter II

Track events with GA4

#2.1. Bye-bye to Event Category, Action, and Label

Compared to Universal Analytics, a.k.a. UA (the previous Google Analytics version), the data model in Google Analytics 4 is much more flexible. This means that there are fewer restrictions and required fields/parameters compared to the UA.

The most notable difference regarding events is 4 parameters that you could send together with every event to Universal Analytics (excluding custom dimensions):

- Event Category (*required*)
- Event Action (*required*)
- Event Label (*optional*)
- Event Value (*optional*)

And when you open the Universal Analytics event reports, you'd first see the Event Category. Then you can click it and drill down deeper (to see action and then label).

In Google Analytics 4, the naming convention is much more flexible and it solely depends on the person who is implementing the tracking setup.

In GA4, there is a parameter called *Event Name*, and then everything else depends on what you have planned. If you wish, you can send no additional parameters. Or you can send 4 additional parameters that better describe the context of that event. For example, let's say that someone filled in the "Request demo" form on your website. You could send the following event:

- Event name: *request_demo*
- **product_name**: *amazing product*
- **company_size**: *99-250*
- **company_industry**: *internet and telecom*
- **annual_revenue**: *50-100M*

The last 4 items in the list above are custom parameters that you can send together with an event. However, keep in mind that [there are some limits](#) regarding the

number of custom parameters that you can send with a single event. I will mention them additionally later in this e-book.

While you can come up with any events you want, there are some recommendations that you should follow. But first, let's take a look at 4 categories of events in GA4.

If you are not sure how to name/structure your events, I will give you some hints later in this e-book + a spreadsheet template. But first, let's learn about various types/categories of events in Google Analytics 4 and how you can create/modify them. I believe that it's important to have a good understanding of events in GA4 before we start thinking about the structure.

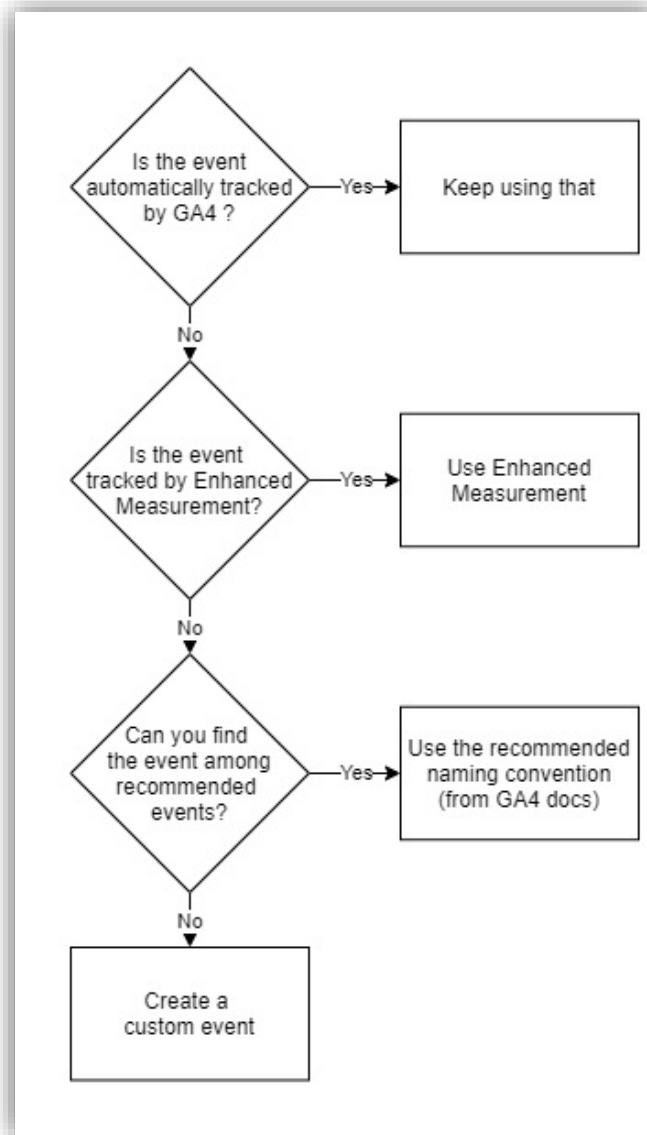
#2.2. 4 categories of events in Google Analytics 4

In GA4, events can be split into [4 categories](#):

- Automatically collected events
- Enhanced Measurement events
- Recommended events
- Custom events

The process of how you should plan the structure of your events (names and parameters) is by doing the following:

1. Check if the event that you want to track is among the **automatically collected events**
2. If not, then check if your event is mentioned among the **Enhanced Measurement** events (e.g. scroll or file_download)
3. If not, then **check the recommended events** and their naming convention
4. If none of the above categories apply to your events, **create a custom event with an event that you want**



In the upcoming 4 chapters of this e-book, I will explain where to look and how to find the category of your event. And once you get more familiar with events (and how to configure them), I will share some tips (and a sample spreadsheet) that can help you think about the structure of your events and a naming convention.

#2.2.1. Category #1. Automatically captured events

Since I am mostly focusing on web tracking, the same principle will apply to this guide. There are certain events that GA4 tracks automatically. You can find a [full list here](#). On that list, you will find both app and web events. In fact, the list of web events is a bit longer than I'd expect. Enhanced Measurement events are also listed there.

For now, let's temporarily ignore them and focus only on those events that are automatically captured and should not be added to any other event category. The **list of website events** goes like this:

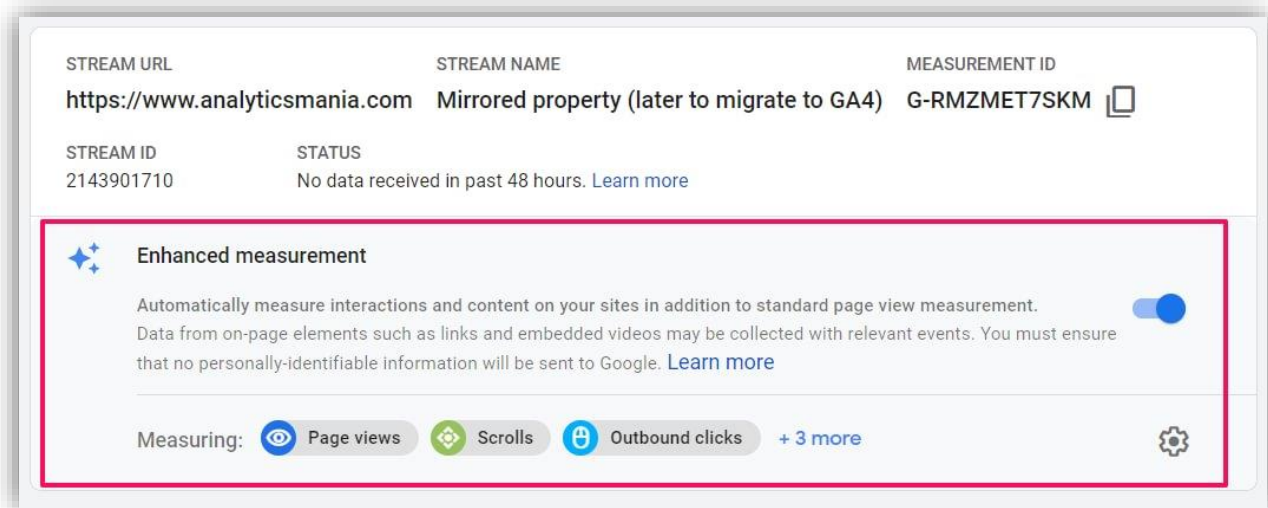
- **first_visit**. This is the first time a user visits a website or launches an app.
- **session_start**. When a user engages the app or website and starts a new session
- **user_engagement**. This one is a bit more mysterious. I could not find 100% accurate information about this event. But it looks like this event is fired 10 seconds after the visitor has stayed on a page. But it also might periodically fire while the app is in the foreground. I will update this part once I find more information.

I didn't mention the `page_view` here because I added it to the *Enhanced Measurement* section. Also, if you want to see the app events, [you can check them here](#).

#2.2.2. Category #2. Events tracked via Enhanced Measurement

When you configure a web data stream in Google Analytics 4, you have an option to use Enhanced Measurement. It is designed to help marketers get as many events in the reports as possible without the need to cooperate with developers or configure in Google Tag Manager.

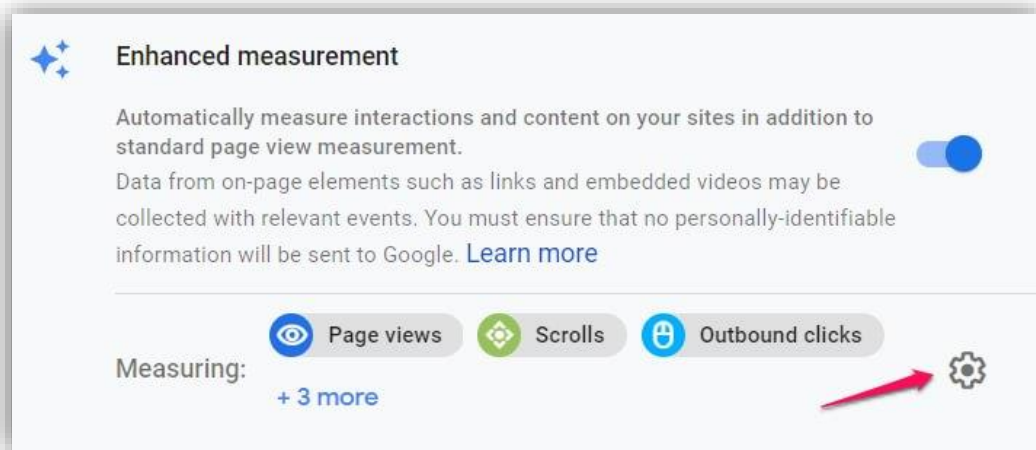
If you go to *Admin > Data Streams >* Select the web data stream, the following window will open where you will see a section called *Enhanced Measurement*.



By default, this feature is enabled and will automatically track the following events:

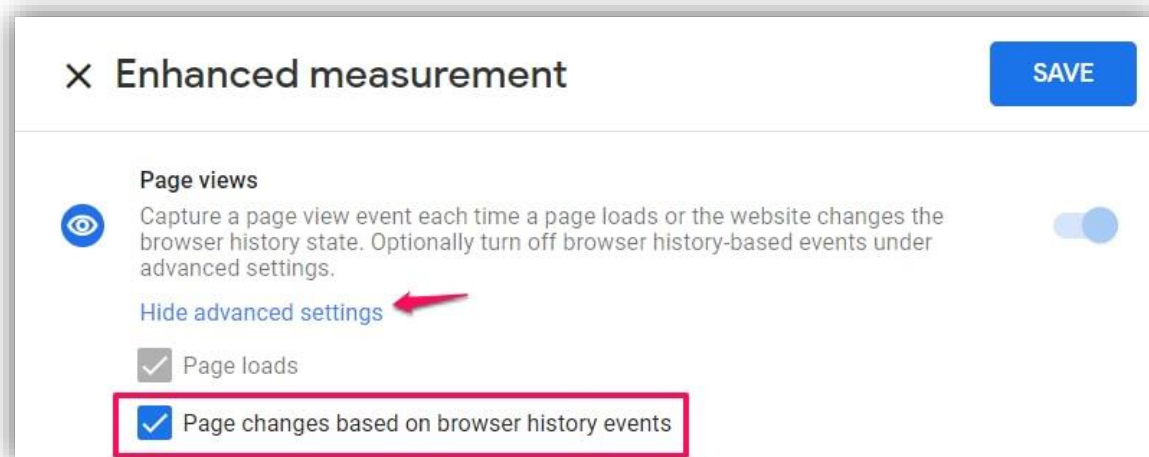
- **Page view** (event name: *page_view*)
- **Scroll** (event name: *scroll*)
- **Outbound link click** (event name: *click* with the parameter *outbound: true*)
- **Site search** (event name: *view_search_results*)
- **Youtube Video Engagement** (events: *video_start*, *video_progress*, *video_complete*)
- **File Download** (event name: *file_download*)

If you want, you can disable/enable events individually and you can do that by clicking the gear icon in the *Enhanced Measurement* section and then clicking toggle buttons.



Also, some events can be customized additionally. Let's take a quick look at each event and their respective settings.

page_view. This event is sent to GA4 when a new page loads or the URL of the page



changes without reloading the page (a.k.a. history change events)

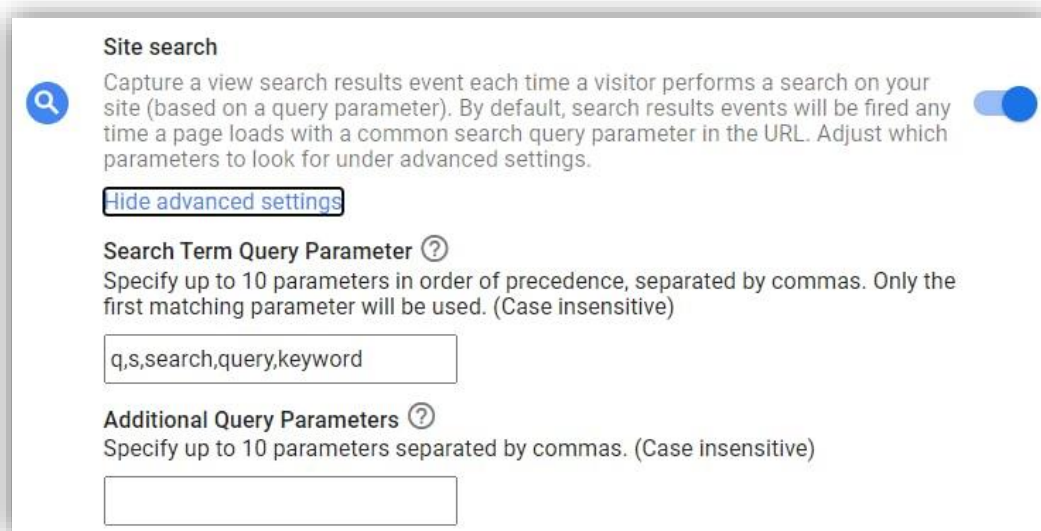
scroll. This event is sent to GA4 once per page when a visitor scrolls below the 90% threshold of the page height.

click. This is an event for the outbound link click (when a clicked link redirects a visitor to another domain. Together with this event, an additional parameter *outbound* (with value "true") is sent. The event with this parameter

means that there was an outbound link click. Together with this event, a bunch of other parameters is sent: *link_classes*, *link_domain*, *link_id*, *link_url*.

If your business operates on several domains, you can go to *Tagging Settings* (of that very same event stream) > *Configure your domains* and then enter the domains of your business. All the domains that are listed there will not trigger the outbound link click event.

view_search_results. This event will be sent to Google Analytics 4 if the page loads and the URL of the page contains a query parameter, such as *q*, *s*, *search*, *query*, *keyword*. But if you want, you can include more of them (up to 10).



Site search

Capture a view search results event each time a visitor performs a search on your site (based on a query parameter). By default, search results events will be fired any time a page loads with a common search query parameter in the URL. Adjust which parameters to look for under advanced settings.

[Hide advanced settings](#)

Search Term Query Parameter ⓘ

Specify up to 10 parameters in order of precedence, separated by commas. Only the first matching parameter will be used. (Case insensitive)

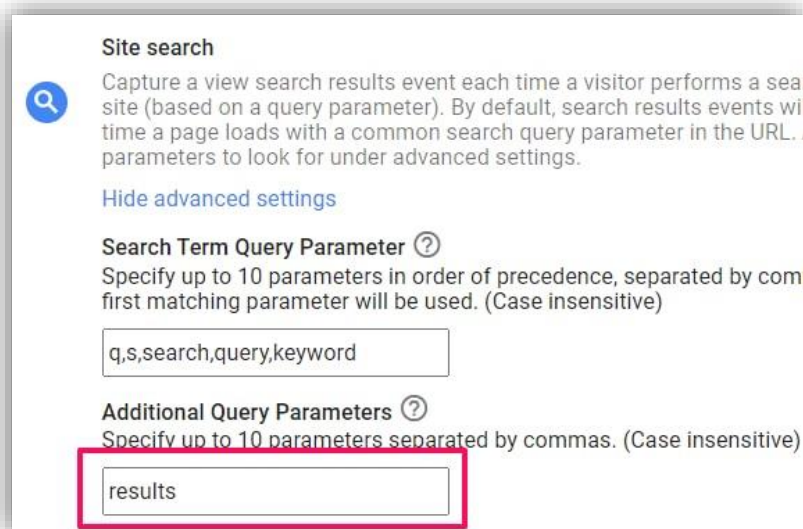
q,s,search,query,keyword

Additional Query Parameters ⓘ

Specify up to 10 parameters separated by commas. (Case insensitive)

If the address of your website's search results page looks like this: *https://www.yourwebsite.com/search?key=my+search+term*, then you should enter the word "key" (without quotation marks in the settings of the search event). If the URL of the search results contains more useful parameters, you can include them as well.

For example, if the web address of your search results page looks like this: *https://www.yourwebsite.com/search?key=search+term&results=50*, you could include the "results" (without quotation marks) in the 2nd field of the configuration. Then this parameter will be automatically tracked by GA4.



video_start, video_progress, video_complete. Enhanced Measurement is capable of tracking interactions of the embedded Youtube video on your website and then sends "video_..." events to GA4. But those embedded Youtube video players must contain the `?enablejsapi=1` parameter in their URL. And that is not always available on websites. Also, there are many other reasons why [Youtube video tracking might not work out of the box](#), hence you will definitely need some tweaks.

Because of those nuances, I'd say that this automatic video tracking will not work in as many cases as you might think, hence I'd personally vouch for [doing that with GTM](#).

file_download. This event is sent to GA4 when a link is clicked and that link contains one of the following file extensions: `.pdf, .xls, .xlsx, .doc, .docx, .txt, .rtf, .csv, .exe, .key, .pps, .ppt, .pptx, .7z, .pkg, .rar, .gz, .zip, .avi, .mov, .mp4, .mpe, .mpeg, .wmv, .mid, .midi, .mp3, .wav, .wma`. In other words, if the link opens/downloads a file. Currently, it is not possible to update the list with other extensions in the GA4 interface.

Once you select which events you wish to track, hit *Save* in the top right corner and then make sure that Enhanced Measurement is enabled (the toggle is blue).

From that moment, your Google Analytics 4 will start automatically tracking the events. See chapter #2.6. of this e-book to learn how to check if your data is coming in properly.

#2.2.3. Category #3. Recommended events

As I have previously mentioned in this e-book, when you are trying to pick the name (and the parameters) for your event, first take a look at automatically tracked events (maybe those events are already being tracked), then check the Enhanced Measurement events. If none of those events cover your case, take a look at the list of recommended events. Google has published several pages for different industries:

- [Generic \(for all properties\)](#)
- [Retail/Ecommerce](#)
- [Jobs, education, real estate](#)
- [Travel](#)
- [Games](#)

Check every one of those lists and see if any of the events match your need. For example, if you want to track when a user logs in, then you will find the "login" event in the list of "All properties".

Event	Trigger...	Parameters
earn_virtual_currency	when a user has earned virtual currency	virtual_currency_name, value
join_group	when a user joins a group Allows you to track the popularity of various clans or user groups.	group_id
login	when a user logs in.	method
purchase	when a user completes a purchase	transaction_id, value, currency, tax, shipping, items, coupon
refund	when a user receives a refund	transaction_id value

Even though Google Analytics 4 data model is very flexible (and you can use a different event name for login, for example, *logged_in*), Google recommends that you implement their recommend events when it makes sense to you. That should help Google Analytics' reports better understand your data and apply it in their Machine Learning capabilities. However, I still don't know what are those capabilities (at least in late 2020).

Also, for the majority of those recommended events, Google also recommends some parameters. Speaking of the *login* event, there is a *method* parameter. If users can log in to your website via *email*, *google login*, *facebook login*, etc., then it might make sense to track the method as well.

join_group	when a user joins a group Allows you to track the popularity of various clans or user groups.	group_id
login	when a user logs in.	method
purchase	when a user completes a purchase	transaction_id, value, currency, tax, shipping, items_coupon

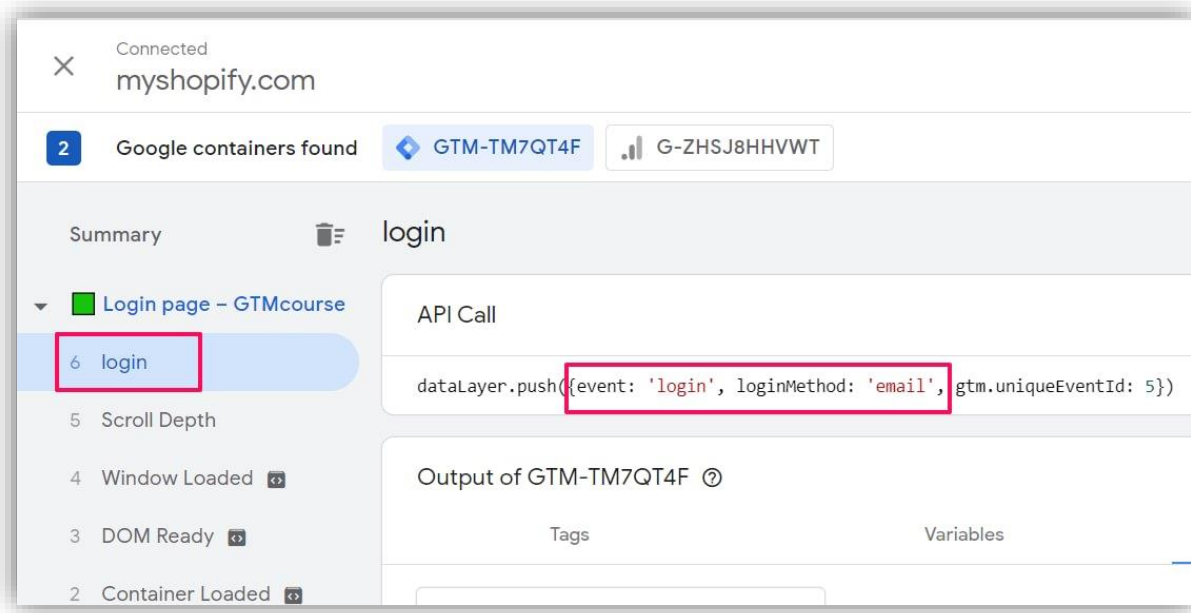
#2.2.3.1. Example of a Recommended Event: login tracking

Let's continue with the aforementioned login example. Let's say that I am working on a website where visitors can log in. I want to track the exact moment when users/visitors do so and which login method are they using. I have asked a developer to activate the following [dataLayer.push](#) code when a user logs in to his/her account.

```
<script>
window.dataLayer = window.dataLayer || [];
window.dataLayer.push({
  'event' : 'login',
  'loginMethod' : 'email' // this should be replaced by your developer
});
</script>
```

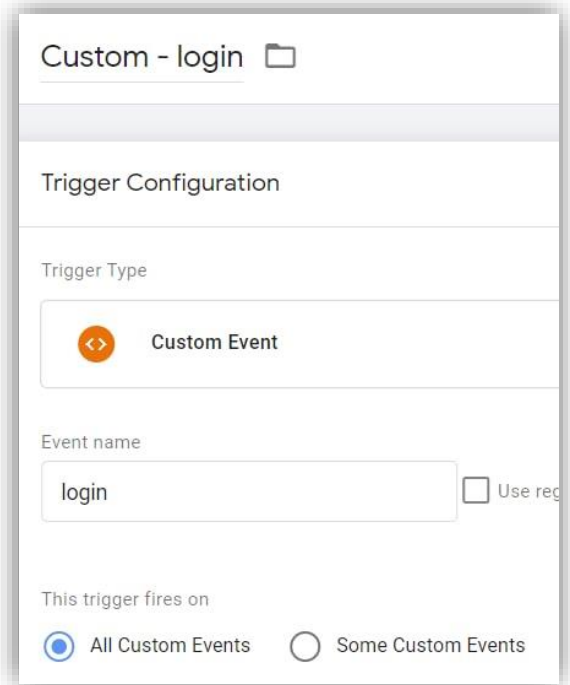
The value of the *loginMethod* should be replaced with the actual login method that a user used. That is the developer's job to write custom code that replaces it.

When a developer implements that code and I successfully login, in [GTM's preview mode](#), I should see the following Data Layer event (see the screenshot below). Now, keep in mind that this event a GTM (or Data Layer) event. It is not sent to GA4 (or any other platform) yet. Right now, it is just a collection of data points that are at our disposal, meaning that we can send the data further to other tools like GA or FB pixel (or any other analytics/marketing platform that we use).

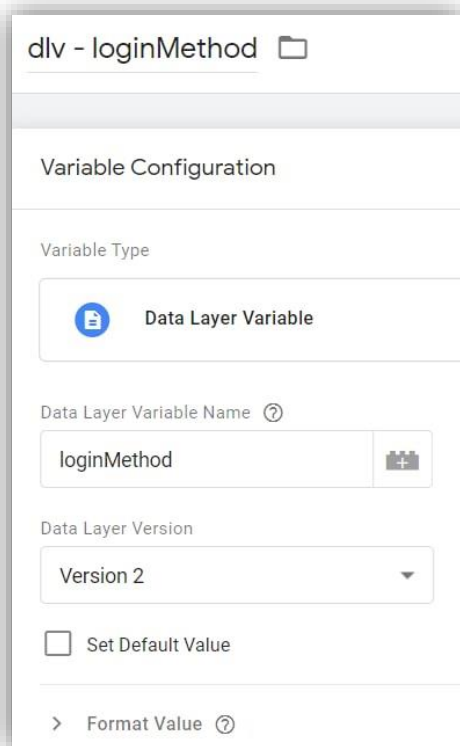


Now, let's create a [Data Layer Variable](#) that will access the value of the *loginMethod* and also a [Custom Event Trigger](#) for the *login* event in the Data Layer.

In Google Tag Manager, go to *Triggers > New > Custom Event*, enter the following settings, and press *Save*.



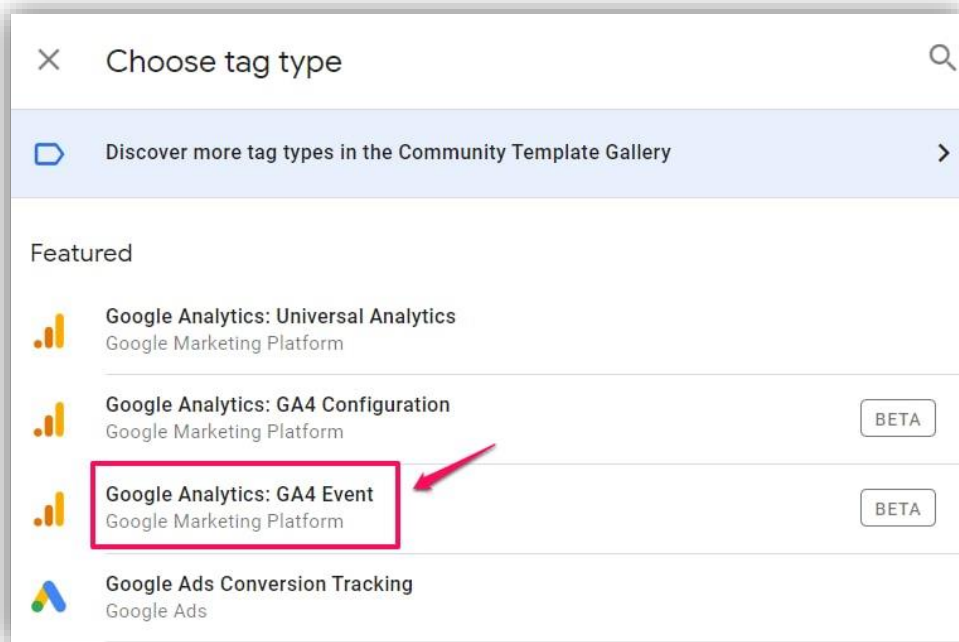
Then go to *Variables > User-defined variables > New > Data Layer Variable* and enter the following settings (see the screenshot below).



I entered the *loginMethod* because that is exactly what a developer pushed to the [Data Layer](#). If your parameter's name is different, then enter that name. Important: variable names are case-sensitive.

After you create a variable and a trigger, it's time to create a Google Analytics 4 event tag. With it, we will be able to send the event data to GA4.

In Google Tag Manager, go to *Tags > New > Google Analytics: GA4 Event*.



Before you do this, I expect that you are already tracking pageviews and you already have a GA4 configuration tag (well, because have you already completed Chapter I of this book).

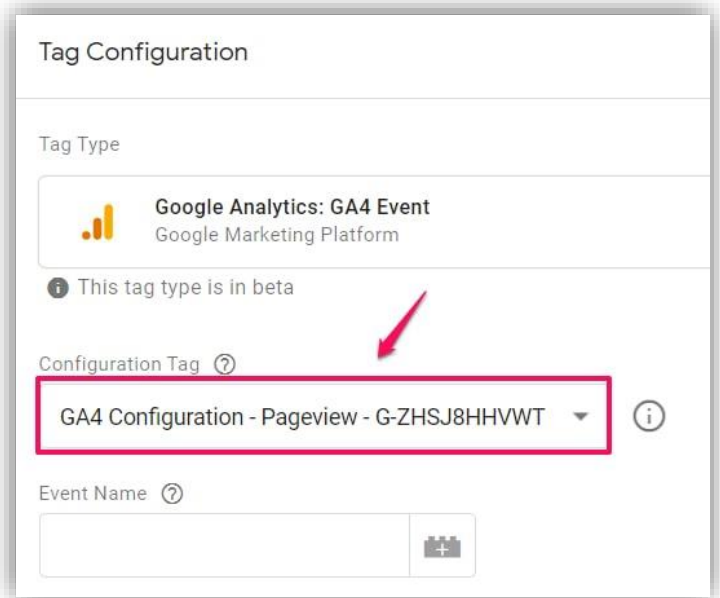
If you haven't then go back to Chapter I and complete all the steps explained there. They are very important for event tracking.

So, when you create a new GA4 event tag, you will need to:

- Define the GA property ID (also known as a Measurement ID)
- And then event name + parameters

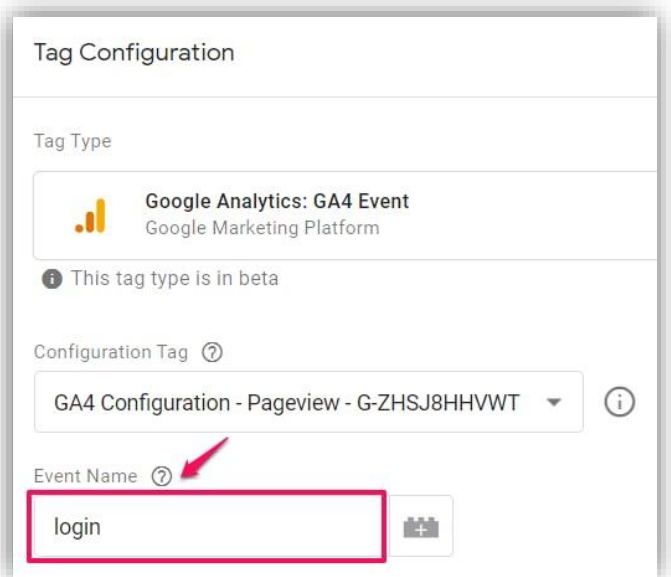
In order to avoid manual work and setting up all the fields/customizations (such as GA Measurement ID) in every tag, you should select your main configuration tag in the event tag.

If you are familiar with Universal Analytics and you used to manage this in Google Tag Manager, the concept of the Configuration Tag is pretty close to the [GA Settings Variable](#). That tag can contain a lot of settings/configurations. If you use that tag in your other GA4 tags, they will inherit the changes.



But at the same time, you can still configure additional settings in your event tag. If a particular parameter/field is configured in both the event tag and in the configuration tag, the event tag's field gets a higher priority.

Now, let's continue the configuration of an event. Enter the event name. Since we are tracking the login event and the name "login" is recommended by Google, this is exactly what we are going to enter.



And then let's enter an additional parameter, the login method. To do that, expand the *Event Parameters* section, click *Add Row*, and then enter *method* in the *Parameter Name* field and insert the previously created Data Layer Variable in the *Value* field. You can insert the variable by clicking the button next to that field.

Tag Type

Google Analytics: GA4 Event
Google Marketing Platform

This tag type is in beta

Configuration Tag
GA4 Configuration - Pageview - G-ZHSJ8HHVWT

Event Name
login

Event Parameters

Parameter Name	Value
method	{{dlv - loginMethod}}

Add Row

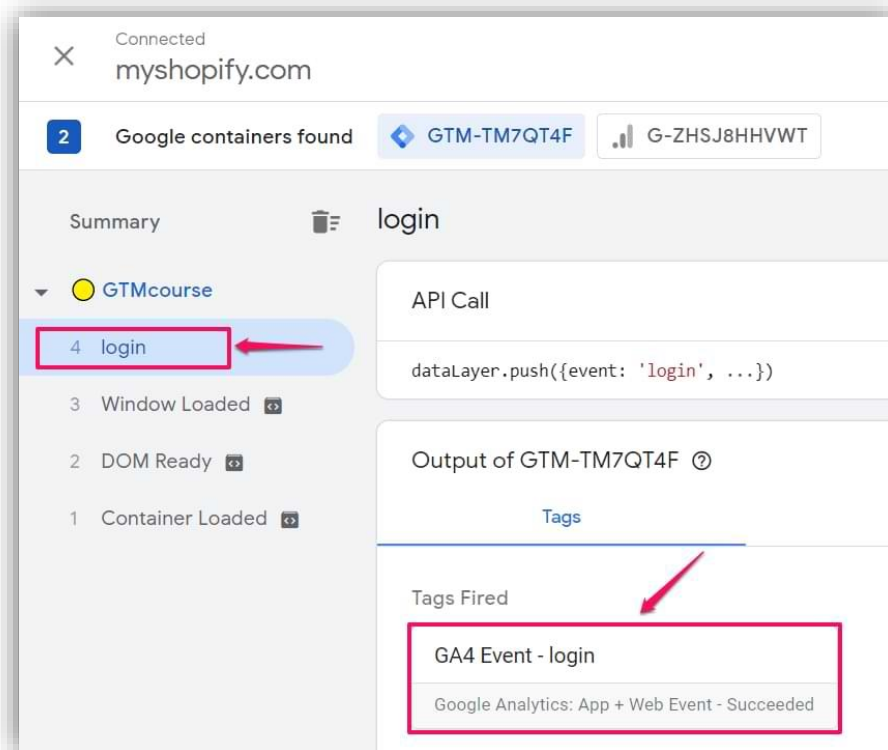
Why did I enter the *method* in the *Parameter Name* field? Because I saw that in the list of [Recommended Events](#).

join_group	when a user joins a group Allows you to track the popularity of various clans or user groups.	group_id
login	when a user logs in.	method
purchase	when a user completes a purchase	transaction_id, value, currency, tax, shipping, items, coupon

What about custom parameters? What if you also want to pass the user's pricing plan or something else? Sure, you can do that. But I will explain custom

parameters/dimensions in the next chapter (#2.2.4). All you need to know right now is that it's completely possible to send custom parameters with recommended events.

Now, it's time to test. Enable the [Preview mode in Google Tag Manager](#), log in to your website and check if your GA4 Event Tag fired on that event. To do that, you will need to click on the *login* event in the Preview mode's left side and then check if the tag fired.



If you indeed see that the tag has fired, that's a good start. Now, you will need to go to Google Analytics 4 and check the DebugView. You will learn more about that in chapter #2.6. of this e-book.

By the way, if you want to implement sales (a.k.a. Ecommerce tracking), then you will also need to use recommended events and follow Google's strict requirements. [Learn more here.](#)

#2.2.4. Category #4. Custom events

Eventually, we have reached the last category of events in Google Analytics 4. If you want to send an event and it is not mentioned among the automatically tracked events, Enhanced Measurement, or Recommended events, then you can create custom events.

The configuration of Custom events is pretty much identical to recommended events. The only difference is that you will need to come up with your own event names.

For example, if you want to track form submissions of a contact form, the name of the event can be any of these:

- form_submission
- contact_form
- contact_form_submission

Or anything else.

GA4 is completely flexible here. However, [there are some limitations](#) when it comes to the maximum length of an event name. Also, there is a [limit of 500 unique event names per GA4 property](#). If you reach it, Google says that you won't be able to track any other new events. And at the moment, you cannot delete unused events (which is unfortunate). Therefore, be careful about creating too many unique event names.

Maybe in the future, this situation/limitation will change. Who knows?

So, speaking of the custom event tracking, let's look at the example where we want to measure clicks on menu links on a website.

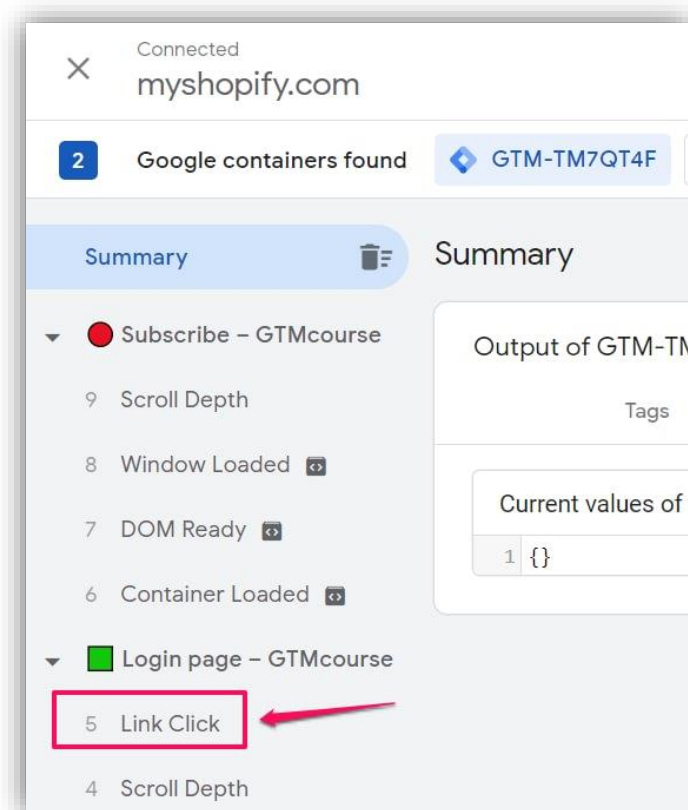
#2.2.4.1. Example of a Custom Event in GA4: Menu Link Click

Note: menu link click tracking trigger conditions differ on most websites (because of different click classes, IDs, etc.). Try to do your best to apply this example.

On a demo website, I have several menu links that I want to track.



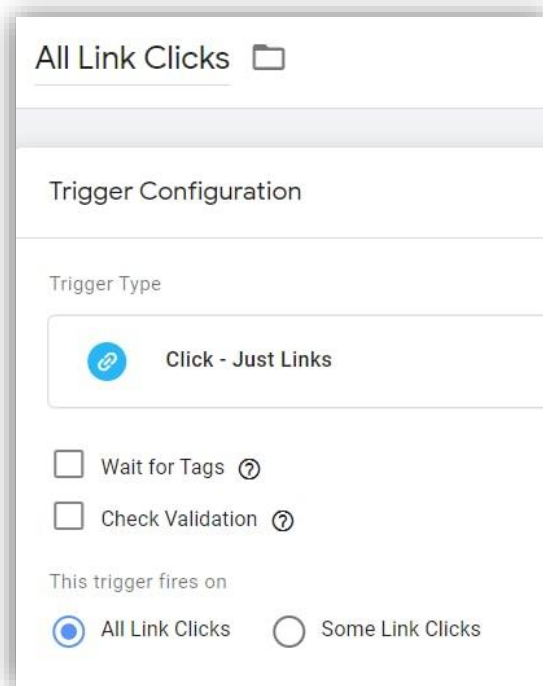
First, I need to create a trigger with correct conditions that distinguish any click from a menu link click. Enable [Google Tag Manager's Preview mode](#) and try clicking any of the menu links on a website. Once you do the first link click, you should see the *Link Click* event in the Preview mode's left sidebar.



If you don't see it, you should:

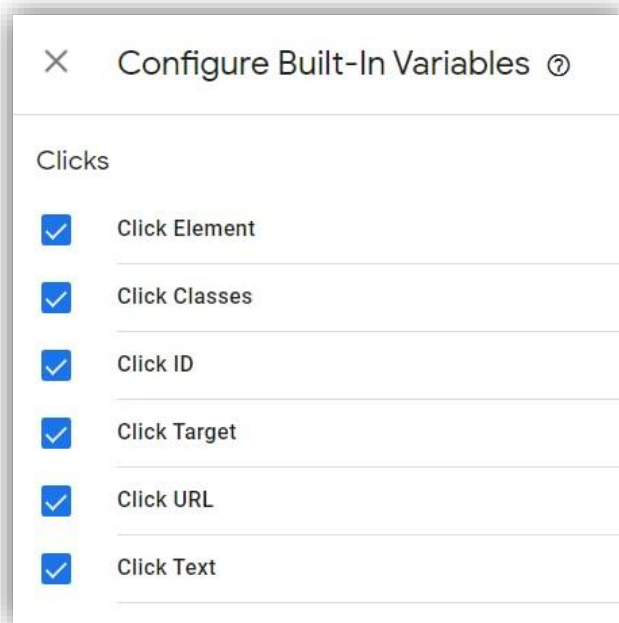
- have at least one Just Links GTM trigger activated on a page
- or have "File download" or "Outbound links" tracking enabled in your Enhanced Measurement Settings

Let's say, that you don't use Enhanced Measurement in GA4. Go to *Google Tag Manager > Triggers > New > Just Links*. Keep the trigger set on "All link clicks" and save the trigger.

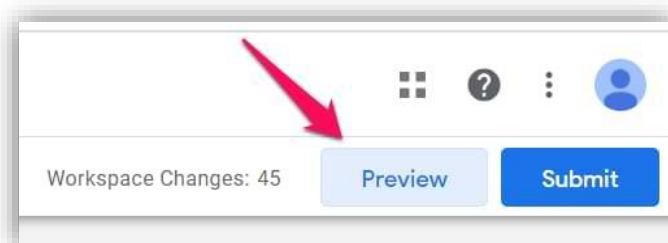


By creating this trigger, we will enable the link-tracking functionality in Google Tag Manager.

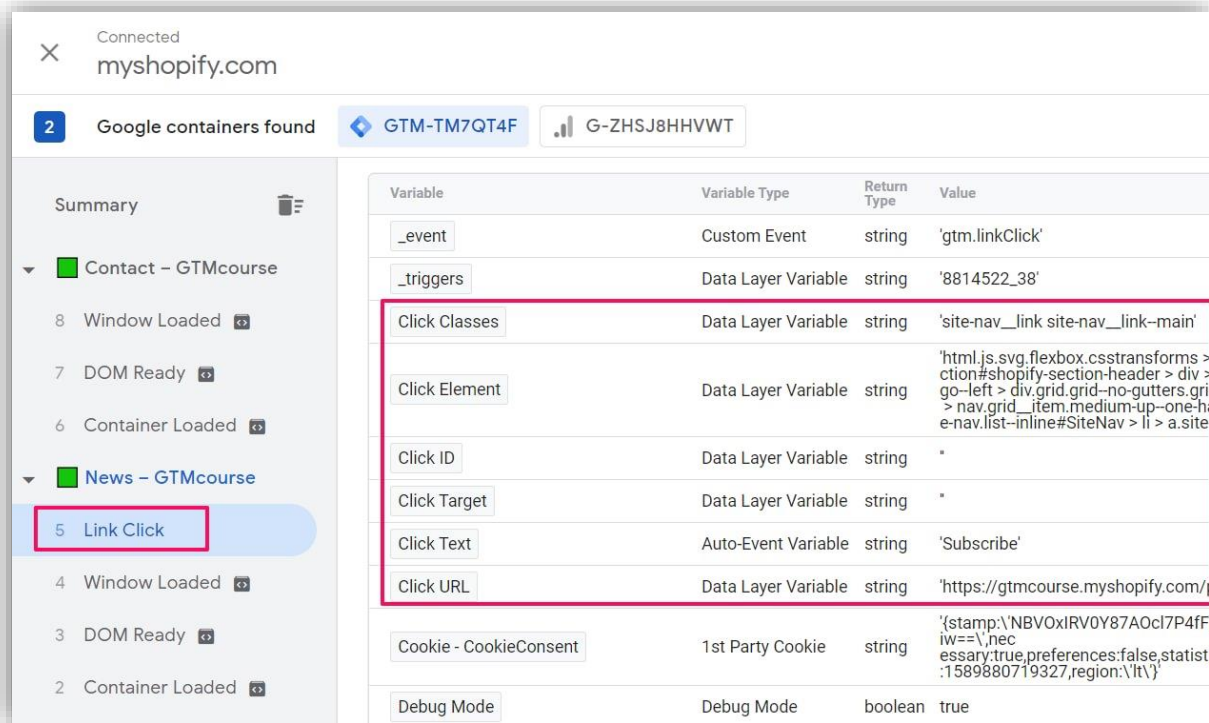
Then go to *Variables > Configure* (in the "Built-in Variables" section) and enable all Click-related variables.



Refresh the preview mode (by clicking the Preview button once again).

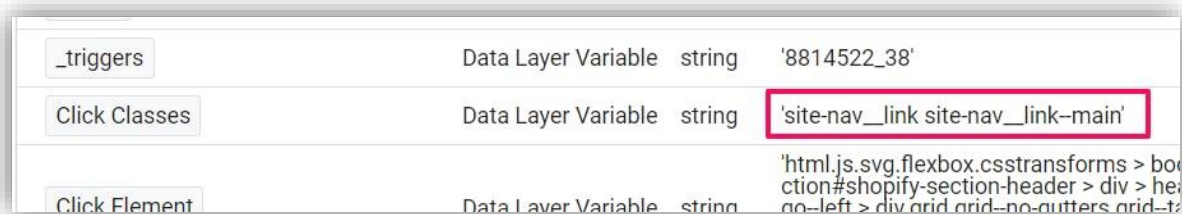


Then go to your website and click any of the menu links. Actually, click at least two of them. Return to the preview mode and you should start seeing *Link Click* events in the preview mode. Click the first *Link Click* event and go to the Variables tab of the preview mode.




Then click on the second *Link Click* event. I am currently looking for some variable that I could use to distinguish a menu link click. I don't want to fire my GA4 event tag on any link click. I want to fire it precisely on a menu link click.

After taking a closer look, I see that both links contain the same *Click Classes* value, `site-nav__link site-nav__link--main`. By the way, `nav` means "navigation" in this context. Great! I will use this in my Just Links trigger.




Go back to the list of triggers in your GTM container and click the previously created trigger for All Link Clicks. We'll edit it and add a condition: *Click Classes* contains `site-nav__link--main`


Link click - menu item 

Trigger Configuration

Trigger Type

Click - Just Links



Wait for Tags 

Check Validation 

This trigger fires on

All Link Clicks Some Link Clicks

Fire this trigger when an Event occurs and all of these conditions are true

Click Classes  contains  site-nav_link-main

Save the trigger. Now, it's time to create a Google Analytics 4 event tag. Go to *Tags > New > Google Analytics: GA4 Event*. Select your existing GA 4 Configuration tag and then enter the event name. Its value is up to you. Any of the following options are perfectly fine:

- *menu_click*
- *menu*
- *menu_item_click*
- *menu click*
- etc.

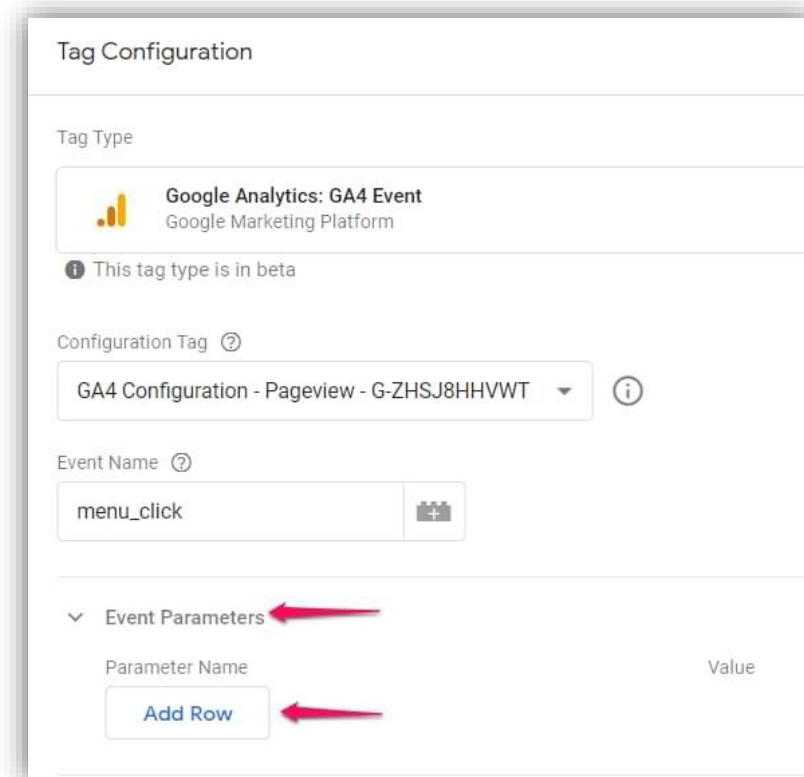
These are just some of the examples. Since we are working with a custom event, we can name it whatever we want (as long as we follow [the limitations of name length](#), for example).

In my case, I used *menu_click* event name. But this time, sending just the event name is not very useful. You'd probably want to know which exact menu items are clicked more often. How about we also send the name of the menu item and the URL?

We could do that by sending additional event parameters. Even though you could use parameters that are already mentioned in the documentation of the Enhanced Measurement (like *link_url* and *link_text*), let's create two custom parameters (because I just want to show you how to configure them).

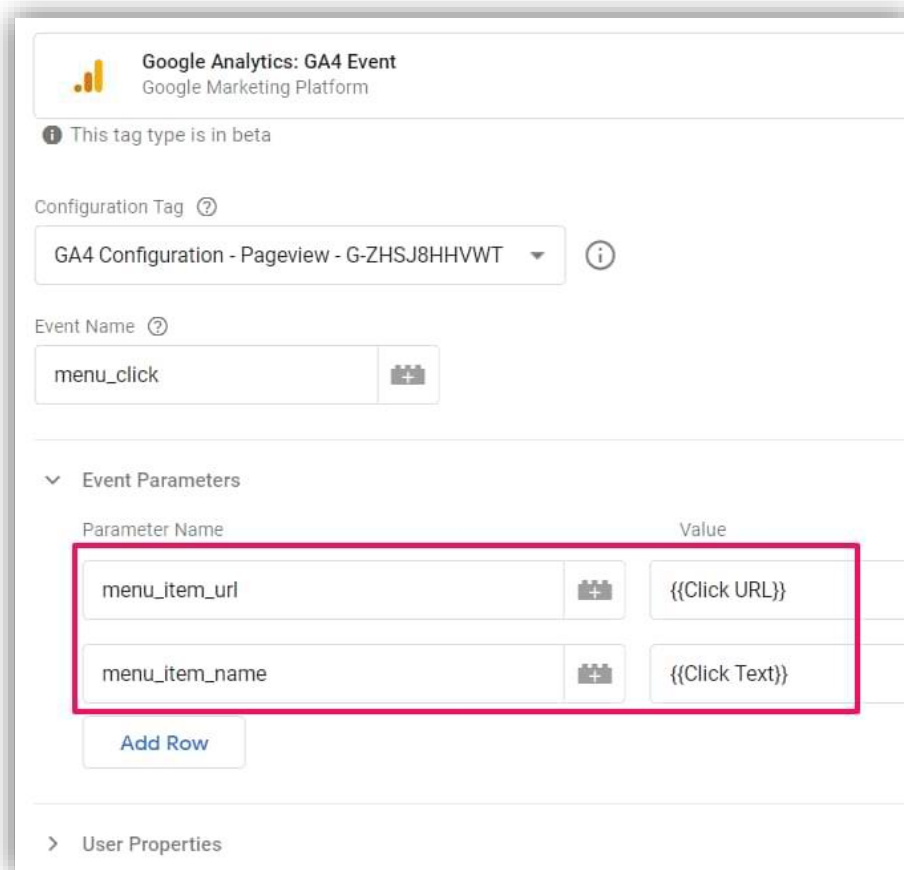
I am going to pass two parameters, *menu_item_url* and *menu_item_name*. I made these parameter names up and you can do it too. That's what the flexibility of the data model offers.

To send custom parameters, you must expand the *Event Parameters* section in the Google Analytics 4 event tag and then click *Add Row*.



Enter the name of the first parameter. In my case, that is *menu_item_url*. In the *Value* field, I will insert a variable that returns a clicked URL. Luckily, GTM already offers such a variable, it's called *Click URL*. Click the **Insert Variable** button and then select the {{Click URL}} variable.

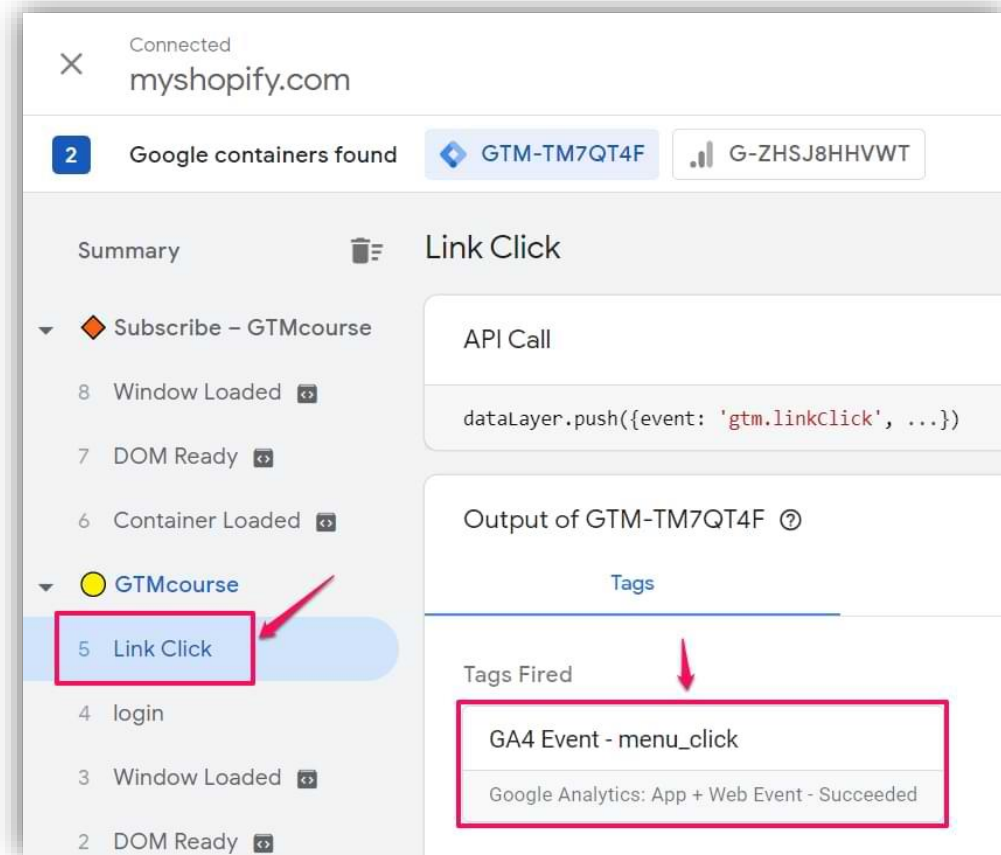
Let's add another parameter, *menu_item_name*. For that purpose, GTM offers another built-in variable, {{Click Text}}. Insert it as well. Since visitors of your website can translate your page, {{Click Text}} value will also change, hence you will see a more diverse list of collected values in your GA4 reports.



If you want to send more parameters, you are free to do that. You can send up to [25 custom parameters with a single event](#).

Assign the previously created *Just Links* trigger to this tag. Save the tag. Refresh the GTM Preview mode and click a couple of your menu items. Go back to the Preview

mode and click those *Link Click* events. Check if the GA4 event tags for Menu Link Clicks have fired.



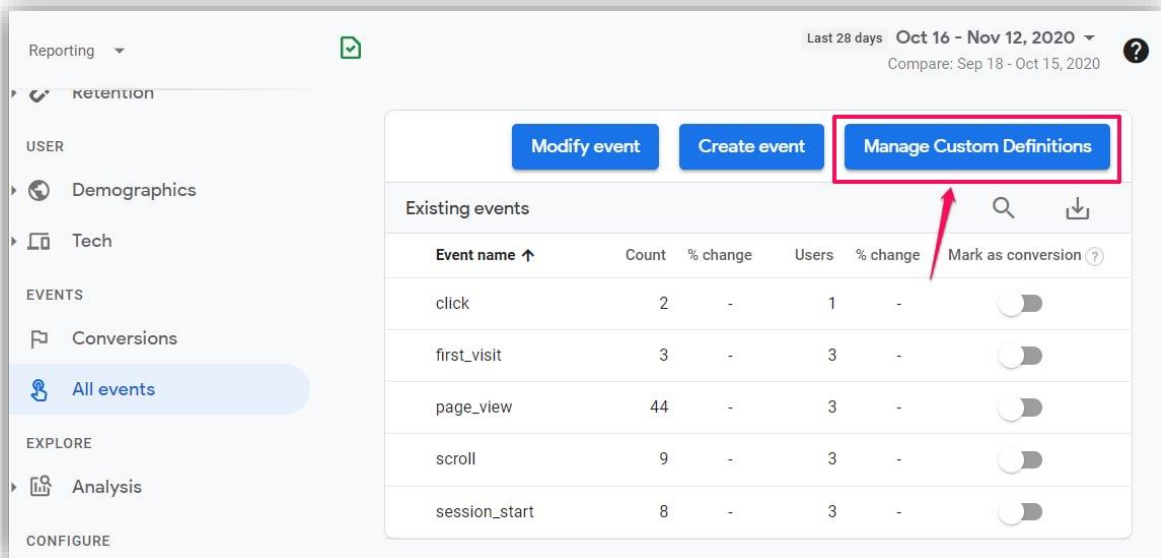
Now, you will need to go to Google Analytics 4 and check the *DebugView*. You will learn more about that in chapter #2.5. of this e-book.

IMPORTANT: even though you will see your custom parameters in Real-time reports and *DebugView* in GA4, you will not be able to see them in other GA4 reports. **Unless you register those parameters as custom dimensions** in Google Analytics. Read the next chapter of this e-book.

#2.2.4.2. IMPORTANT: Register custom definitions

This applies to any custom parameter that you send to Google Analytics 4. If you want to see/use them in things like Funnel reports, Exploration, see their reporting cards in standard reports, etc., you must register custom parameters in the GA interface.

In Google Analytics 4, go to *All Events* and then click *Manage Custom Definitions*. Since we sent 2 custom parameters with the menu link click, we must register them both here.



The screenshot shows the Google Analytics 4 interface for 'All Events'. The 'Manage Custom Definitions' button is highlighted with a red box and a red arrow. Below it is a table of existing events.

Event name ↑	Count	% change	Users	% change	Mark as conversion ?
click	2	-	1	-	<input type="checkbox"/>
first_visit	3	-	3	-	<input type="checkbox"/>
page_view	44	-	3	-	<input type="checkbox"/>
scroll	9	-	3	-	<input type="checkbox"/>
session_start	8	-	3	-	<input type="checkbox"/>

Click the *Create Custom Dimensions* button and then enter the name of the first parameter that you sent. In my case, that is *menu_item_url*. Save it.

Custom Definitions > Dimensions

Create custom dimension

Event parameter name
menu_item_name

Custom dimension name
This is the name you'll see in reports, so use a descriptive name.
menu_item_name

Enter your custom dimension here

Then register a second parameter (in my case, that was *menu_item_name*) and save it. By the way, I sometimes use the words "custom parameter" and "custom dimension" interchangeably. In general, custom metrics and custom dimensions are both grouped as custom parameters.

And now we wait. Within the next 24 hours, the custom parameters will start appearing in your Google Analytics 4 reports.

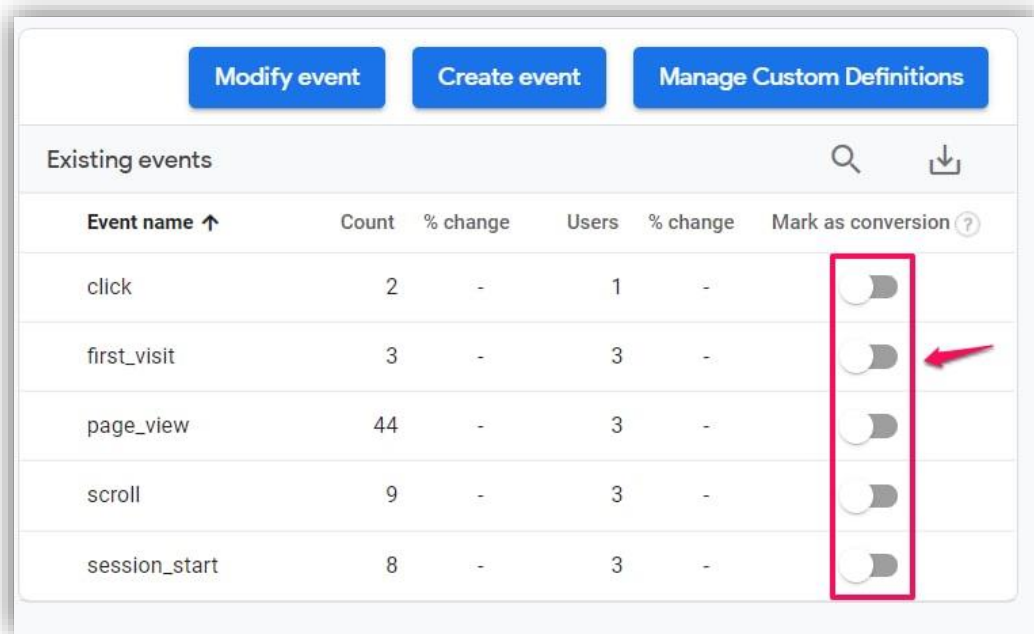
=====

Sometimes, you might want/need to create/edit events directly from the Google Analytics 4 interface. Well, now you can. Read the following two chapters to learn more about these features.

=====

#2.3. Useful Feature: Create new events (GA4 interface)

The process of how conversions (formerly known as *goals*) are created is different in GA4 (compared to Universal Analytics). It's enough just to click a toggle next to an event in the *All Events* list.

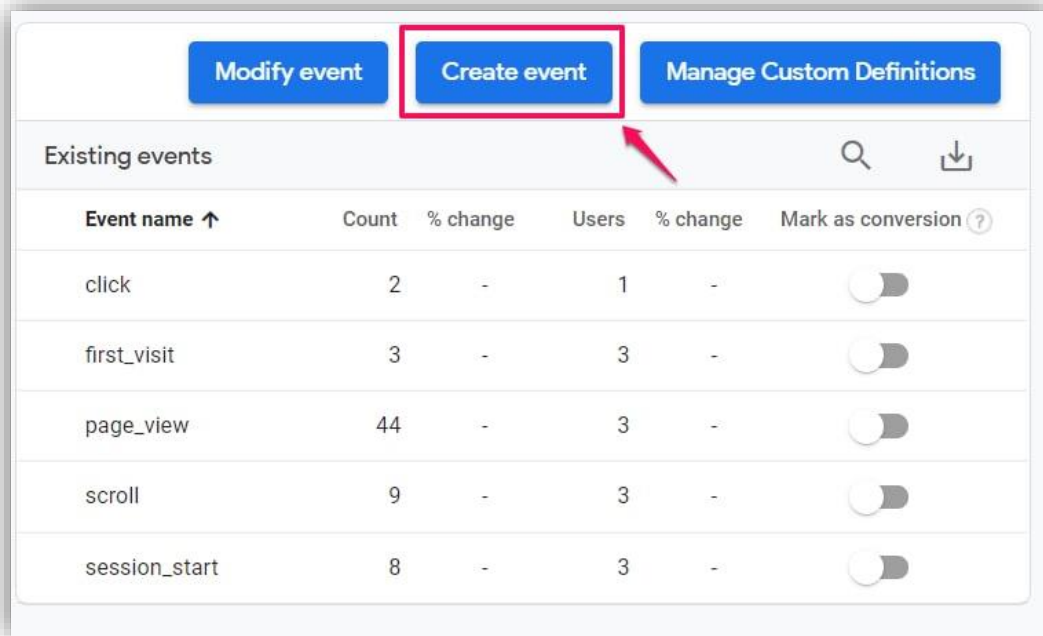


But there is one problem. If you toggle the event and mark it as conversion, ALL of those events will be treated as conversions (from this point moving forward). But what if you want to have only certain events be marked as a conversion.

Example: you have a "Thank you" page to which users are redirected when they subscribe to a newsletter. Let's say that the URL is <https://www.mywebsite.com/thank-you/>. If I mark *page_view* event as a conversion, ANY pageview will become a conversion. How can I separate only those pageviews that happened on the */thank-you/* page?

You can either send a dedicated event (with a different name) from Google Tag Manager/Gtag.js, or you can use the *Create Event* feature in the GA4 interface.

This feature allows you to create a new event based on other incoming events. On the *All Events* page, click *Create Event* and then click *Create*.



Then you will need to enter the name of the custom event. Once again, you can name it whatever you want. Just make sure that the name clearly communicates what it means. *thankyou_page_visit* might be a good option.

Then let's move on to the *Matching Conditions* section. Here we must tell GA4 what kind of event are we looking for. When that particular event will be spotted, then our *thankyou_page_visit* should be created as well.

In my case, I need to enter the following conditions:

- *event_name* **equals** *page_view*
- *page_location* **contains** */thank-you/*

Create event

Analytics Mania
 G-W6QB81P6Y6

Configuration

Custom event name ?

Matching conditions

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

Parameter	Operator	Value
<input style="width: 100%;" type="text" value="event_name"/>	<input style="width: 100%;" type="text" value="equals"/>	<input style="width: 100%;" type="text" value="page_view"/>
<input style="width: 100%;" type="text" value="page_location"/>	<input style="width: 100%;" type="text" value="contains"/>	<input style="width: 100%;" type="text" value="/thank-you/"/>

If you want to copy all the parameters from the *page_view* event to the new event, keep the checkbox *Copy parameters from the source event* enabled.

If some of the parameter's name is incorrect and you want to fix it as well, you can *Add Modification* in the *Parameter Configuration* section. For example, if an event contains the parameter *pricingPlan* but you want it to be *pricing_plan*, you can introduce a new field (and reuse its value) while removing the incorrect parameter (by leaving the *New Value* empty).

Matching conditions

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

Parameter	Operator	Value
event_name	equals	page_view
page_location	contains	/thank-you/

[Add condition](#)

Parameter configuration

Copy parameters from the source event

Modify parameters [?](#)

Parameter	New value
pricing_plan	[[pricingPlan]]
[[pricingPlan]]	Example: [[source_parameter_name]] or 1234

[Add modification](#)

Will insert the value of the "pricingPlan" parameter

Empty field will remove the parameter

Take a closer look at [[pricingPlan]] in the screenshot above. Double square brackets mean that GA4 will reuse the value of the parameter *pricingPlan* in that event.

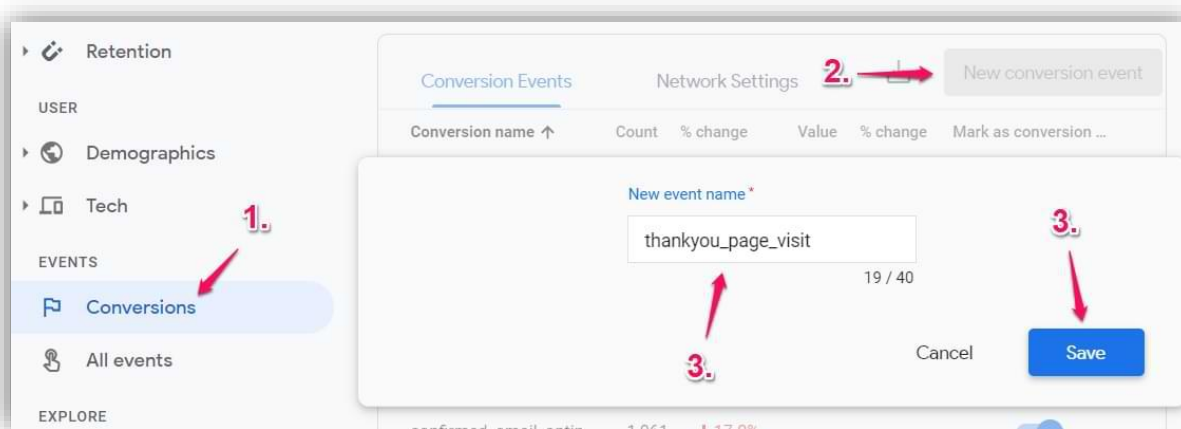
Once you save the changes, you will be able to view them in the Real-time reports and the *DebugView* of GA4. You will learn more about that in chapter #2.5. of this e-book.

Also, when you create that new event in GA4 (and you want that to become a conversion), don't forget to mark that event as a conversion in the All Events report.

Event name ↑	Count	% change	Users	% change	Mark as conversion ?
thankyou_page_visit	6,228	↑ 13.2%	4,427	↑ 13.2%	<input checked="" type="checkbox"/>
confirmed_email_optin	1,063	↓ 17.7%	809	↓ 15.9%	<input type="checkbox"/>

If you don't like to wait for up to 24 hours (because that *thankyou_page_visit* will not appear in the list of all your events immediately), you can create a new conversion immediately. Click the *Conversions* section on the left sidebar of your GA4 interface.

Then click *New conversion event* and enter the name of the event that you have just created. Click *Save*.



That way you will mark the newly created *thankyou_page_visit* event as a conversion (without the need to wait for 24 hours until it appears on the "All events" page).

#2.4. Another useful feature: Modify Events (in GA4 interface)

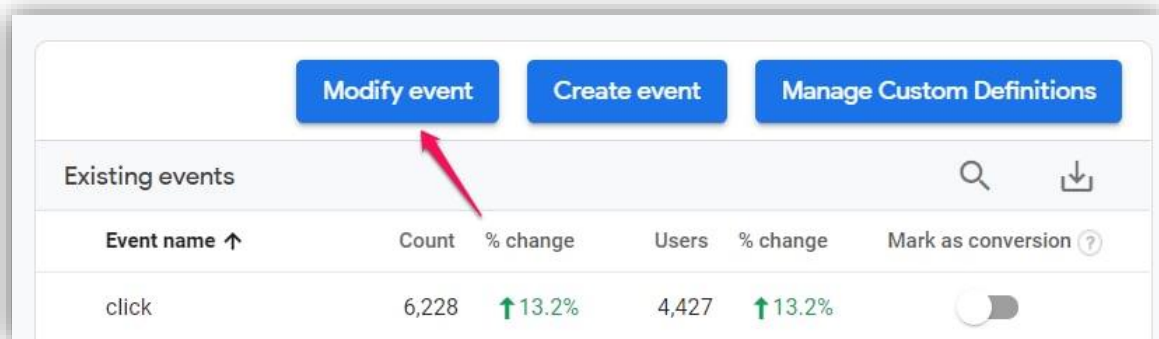
If you don't want to create new events (like in the previous chapter) but instead, you just want to fix a typo in the existing events, you can use the *Modify Events* feature in the Google Analytics 4 interface.

In my *All Events* report, I see two *page_view* events.

- *page_view*
- *page_view?undefined*

I have no idea where that *page_view?undefined* came from but I don't want to see it in my reports. Even though the count of this event is low, I'd like to fix it.

I can do that by going to the *All Events* report and clicking *Modify Event*.



Then click *Create*.

- Enter the name for the fix (this is for internal use)
- Then enter the matching condition (in my case, that's *event name equals page_view?undefined*)
- And then enter the needed modifications (in my case, that's *event_name* and *page_view*)

If GA4 spots an event with the name *page_view?undefined*, it will change its name to *page_view*.

Modify event

Analytics Mania
 G-W6QB81P6Y6

Configuration

Modification name ?

Fix for page_view?undefined

Matching conditions

Modify events that match ALL of the following conditions

Parameter	Operator	Value
event_name	equals ▼	page_view?undefined

Add condition

Modify parameters ?

Add, remove, or edit parameters, including event_name

Parameter	New value
event_name	page_view

Add modification

As you can see, the interface and the flow here is quite similar to the *Create Event* feature that I have described in the previous chapter of this e-book.

Once you save the changes, you will be able to view them in the Real-time reports and the *DebugView* of GA4. You will learn more about that in the next chapter of this e-book.

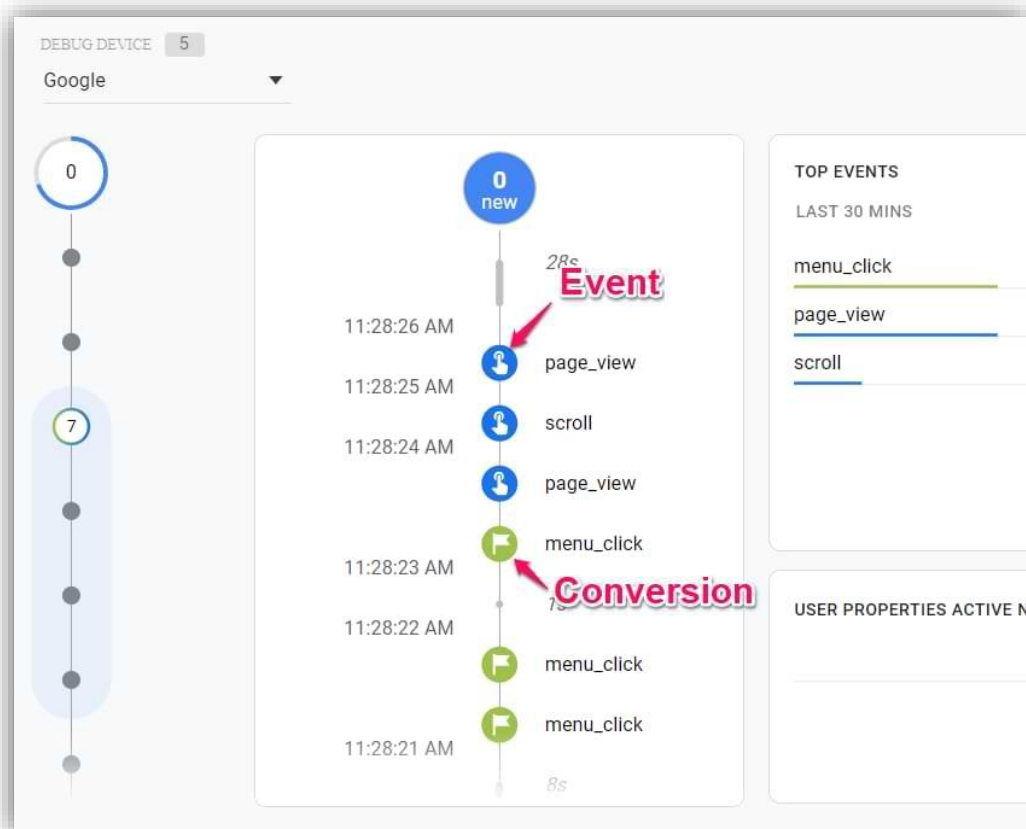
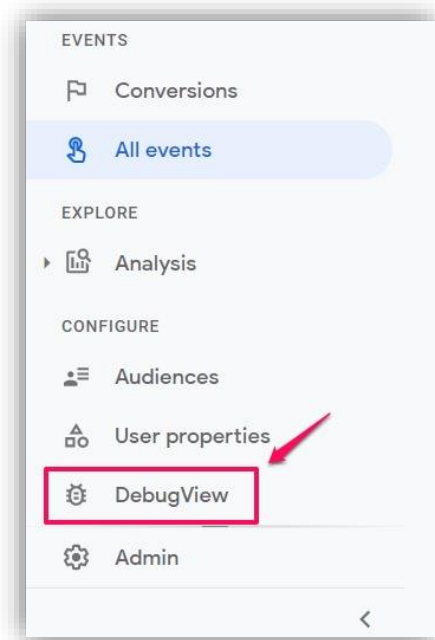
By the way, keep in mind that you can create up to 50 modifications per property and that this feature does not apply to historic data. This modification will apply only to new incoming events.

#2.5. Test your events in GA4

Once you configure your events, it's time to test them. The primary feature built for debugging GA 4 data is the *DebugView* section. You can find it at the bottom left corner of your GA4 interface. Click it.

If you need a refresher on how *DebugView* works, feel free to re-visit chapter #1.4. of this e-book.

When you start seeing data in the DebugView, things might look like this:



Events are marked with blue icons, conversions with green. Please ignore the fact that I treat *menu_click* events as conversions. This is just for demonstration purposes :)

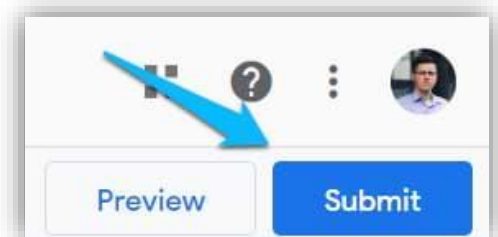
If you click the event, you will see the parameters that were sent together with an event. Click on the parameter to see its value.

The screenshot shows the Google Analytics interface. On the left, a vertical timeline displays a sequence of events: a 'new' session at 11:28:12 AM, followed by 'page_view' events at 11:28:13 AM, 11:28:21 AM, and 11:28:24 AM. A 'menu_click' event is highlighted in green at 11:28:23 AM, with a red arrow pointing to it and the text '1. Click here'. To the right, a detailed view of the 'menu_click' event is shown. The 'Parameters' tab is active, displaying a list of parameters and their values. A red arrow points to the 'menu_item_name' parameter, which has the value 'GTM RECIPES', with the text '2. Click the parameter' above it. Another red arrow points to the value 'GTM RECIPES', with the text '3. Here's the value of the parameter' below it.

Parameters	User properties
▶ debug_mode	1
▶ engagement_time_msec	1
▶ ga_session_id	1
▶ ga_session_number	1
▼ menu_item_name	1
GTM RECIPES	
▶ menu_item_url	1
▶ page_title	1

Once you made sure that the data is coming in and it is displayed properly, you should submit your GA4 changes in the GTM container and publish it.

You can do that by clicking the SUBMIT button in the top right corner and then complete all the other steps that the user interface asks you to do.



After that, you should soon start seeing the new data coming in your real-time reports as well. Please refer to Chapter #1.6. of this e-book to learn more about Realtime reports.

#2.6. Where can I find events data in GA4 reports?

This e-book focuses mainly on the setup of event tracking in Google Analytics 4. I will not dive deep into how to find insights, raise questions, answer questions, etc. However, I can show you several places where you can find the event data that you have just sent to Google Analytics 4.

Keep in mind that at the moment of writing this e-book (late 2020), the built-in reporting capabilities of Google Analytics 4 are still quite limited. Hopefully, that will change soon. Let's take a look at what we have now. Here are some of the reports:

- *All Events* reports
- *Engagement > Events*
- *Analysis > Analysis Hub*

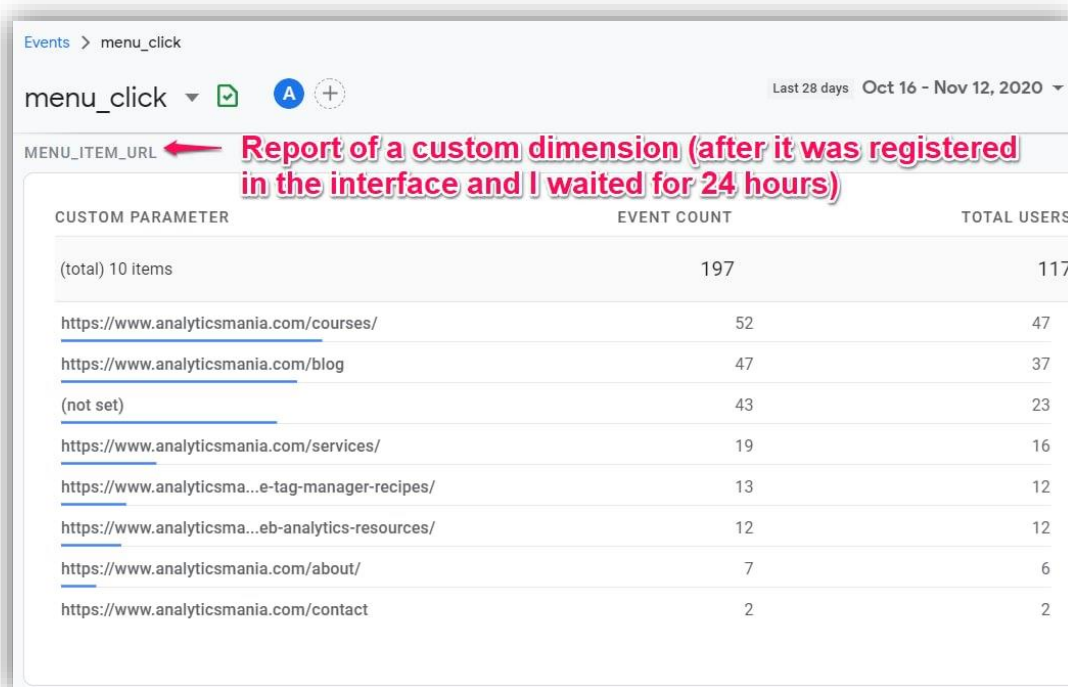
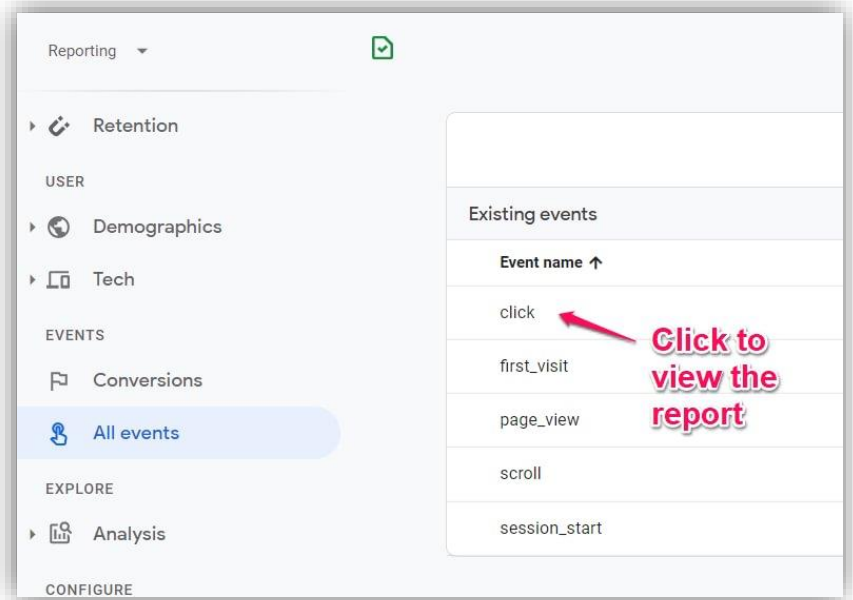
But that is not all. Since GA4 is an event-based analytics platform, every report is affected by events that you send (in one way or another). The aforementioned reports are just the most notable examples.

"All Events" reports

Here you will see a list of all events that are tracked by your Google Analytics 4 property. Click on any event to see a bit more detailed view/reports of that particular event.

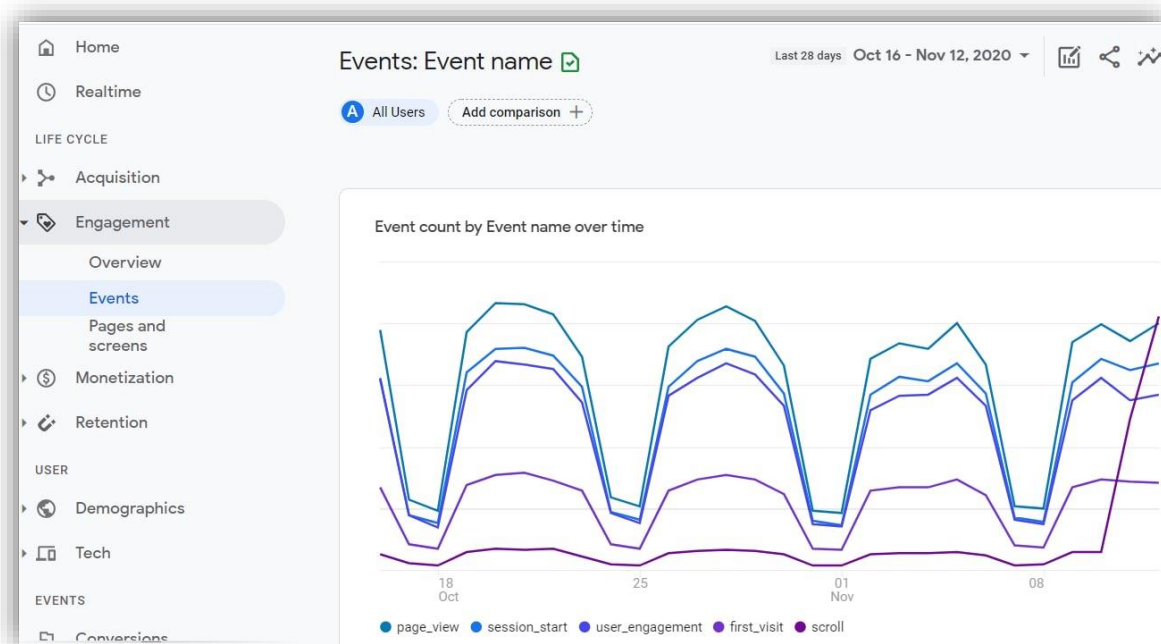
If you are used to reporting tables (that were in Universal Analytics), you will be disappointed. Hopefully, in the future, they will contain more functionality. If you want to see some additional cards about parameters that you sent together with the events, those parameters must be registered as custom definitions in your

GA 4 property. After you do that (and after 24 hours pass), you will start seeing additional cards with the parameter data.



"Engagement → Events" report

This is an overview of all events that were sent to your property + a couple of charts. Below them, you will see a list of events and if you click on any of them, you will be taken to the very same type of report that was mentioned in the "All Events" report section.



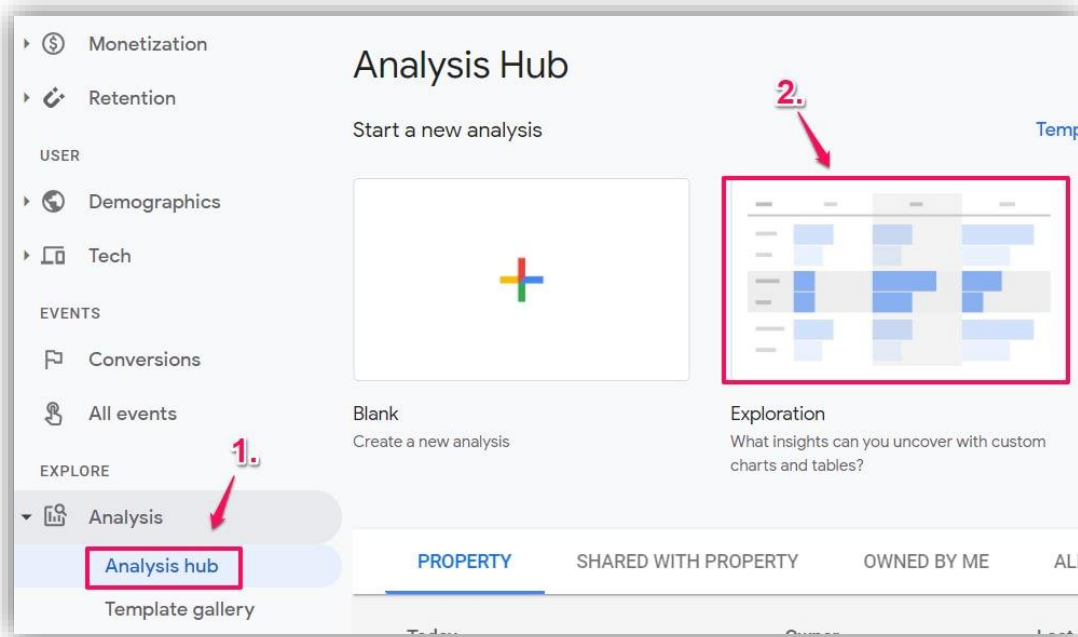
Analysis › Analysis Hub reports

This is the place where you will be able to drill-down into your data. In the Analysis Hub, you will be able to use reports such as Exploration, Funnel Analysis, Pathing, etc.

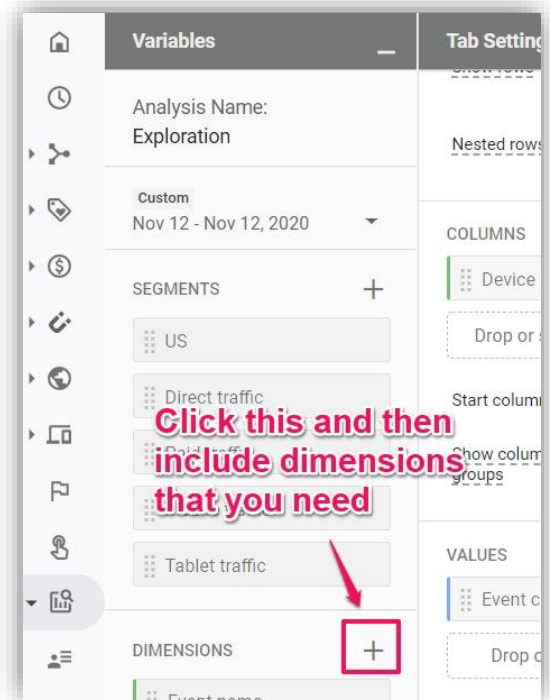
Here is an example of an Exploration report. Keep in mind that you will need to wait for up to 24 hours for the data to come into your reports. So if you don't see the data now, be patient.

Let's say that I want to see how different device categories are interacting with the menu bar of my site. Remember, I set up the menu_click event?

So, let's go to **Analysis > Analysis Hub** and click on the *Exploration* block.

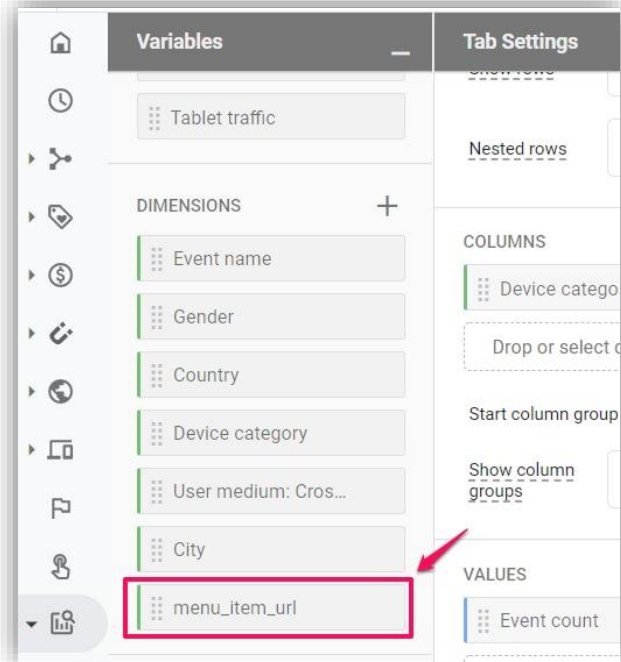


Then in the *Variables* column, I need to include the *menu_item_url* custom dimension because I want to see how many clicks did get each menu URL. Click the Plus icon and then find the *menu_item_url* dimension. Select it and it will be included in the list of possible dimensions in the report.

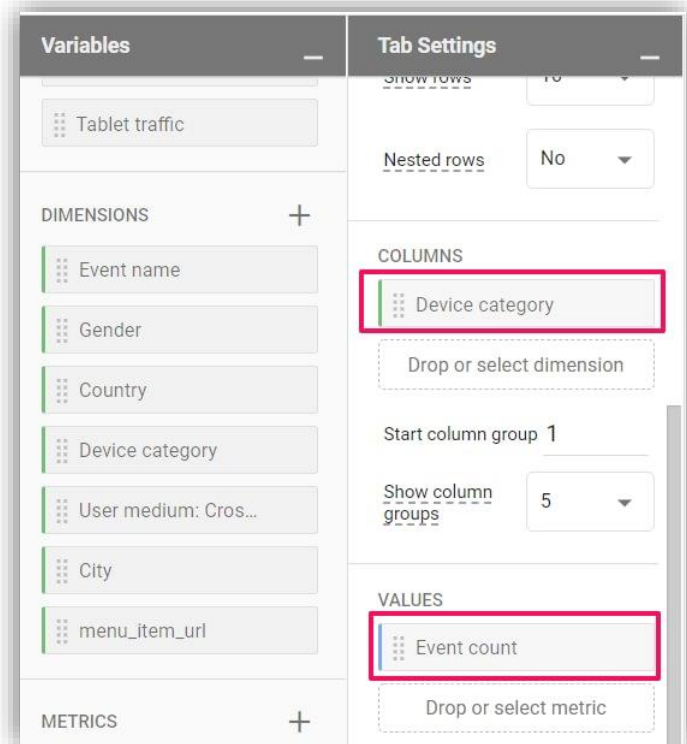


If you CANNOT see the menu_item_url (or whatever custom parameter are you looking for), you have to register them as custom dimensions in GA4 first in *All Events > Manage Custom Definitions* (check chapter 2.2.4.2. in this e-book). And then wait for 24 hours until the data appears in the reports. If the reports don't work properly, wait for even longer.

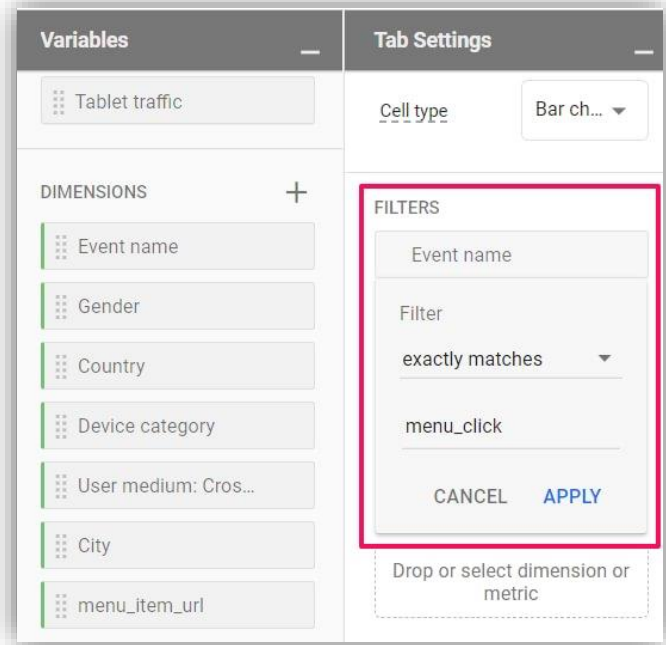
Then let's edit the *Tab settings*. Remove the existing dimensions in the ROWS and include the custom definition that you want to include. In my case, that's *menu_item_url*.



In the COLUMNS section, I used the Device category dimension. In the VALUES section, select the metric that you wish to see. I used the Event Count.



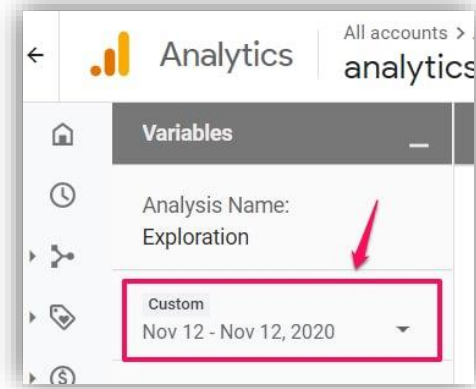
Then you will need to filter only to those events that actually contained the *menu_item_url* dimension. In my case, that is just a *menu_click* event. That's why at the bottom of the *Tab Settings* section, you should enter a filter: *Event name* **exactly matches** *menu_click*.



That's it, your report will now display how different device categories are clicking menu items; which ones are the most popular and how many times were they clicked.

Exploration 1			
Device category	desktop	tablet	Totals
menu_item_url	Event count	Event count	↓ Event count
Totals	96 98.97% of total	1 1.03% of total	97 100% of total
1 https://www.analyticsmania.com/...	35	1	36
2 https://www.analyticsmania.com/...	29	0	29
3 https://www.analyticsmania.com/...	15	0	15
4 https://www.analyticsmania.com/...	8	0	8
5 https://www.analyticsmania.com/...	7	0	7
6 https://www.analyticsmania.com/...	1	0	1
7 https://www.analyticsmania.com/...	1	0	1

Also, don't forget to change the date range if you need it. You can do that in the top-left corner of the Exploration interface.



You can also use event data in things like funnel reports, but let's keep it for another tutorial (hopefully) in the future.

#2.7. How to plan your events in GA4

Note: This chapter will be just an overview of how you should do it.

Speaking of actual planning, I would say that a spreadsheet is your best friend.

- Write down all the events that you want to track and then:
 - Check whether they fall under the categories: automatically collected, enhanced measurement, or recommended.
 - If yes, check their naming convention of event names and parameters (dimensions). If not, then come up with your own values. Just keep in mind that [there are some limitations related to the length](#).
- If you have a huge list of event names, be aware of another limitation. Currently, you can have up to 500 unique events per property. If you are close to that limit in your spreadsheet, maybe it would make sense to combine some events under the same event name and introduce an additional property (currently, the limit of registered custom properties are 50 text

properties and 50 numeric properties. More about registered properties - later in this e-book).

[Here is an example of the spreadsheet with events](#) that you could prepare yourself and then try to pick the right naming convention. You can use it as an example/inspiration to come up with your own spreadsheet. Let's take a quick overview of the spreadsheet.

There are two sheets:

- The first one is for the list of events and what kind of parameters do you want to track together with them
- The second one is a list of parameters with their explanations

The first sheet:

- **In column B**, you can just briefly describe an event in plain English
- **In column A**, you should enter the name of the event that you will use in Google Analytics 4. You should pick this name based on the previously described process: check the automatically tracked events, enhanced measurement, and recommended events. If none of the events match yours, then add a custom name. It looks pretty clean to use this principle to name the event, e.g. *event_name* (all lowercase and connected with an underscore). Event names like "Submitted the Form" will also work, but the all-lower-case-with-underscore looks cleaner (I believe that the term [snake-case](#) applies here).
- **Column C** is for type (is it Automatically collected, Enhanced Measurement, Recommended or Custom?).
- **Column D** is for parameters that you want/plan to track with particular events. I did not include default parameters that are automatically tracked with every event: *language, page_location, page_referrer, page_title, screen_resolution*.

Event name	Definition	Type	Parameters
click	Outbound link click (when a click link does not belong to our company's hostnames)	Enhanced Meas ▾	link_classes link_domain link_id link_url outbound
file_download	File download (link click)	Enhanced Meas ▾	file_extension file_name link_classes link_domain link_id link_text link_url
scroll	Scroll > 90%	Enhanced Meas ▾	percent_scrolled

If you are dealing with mobile apps as well, you could include an additional column "Platform" where you could enter "web" or "Android / iOS".

The second sheet:

- **Column A** is for the parameter name
- **Column B** is for a platform. If you are working just with the website, feel free to remove that column.
- **Column C** is the type (is it Built-in, Recommended, or Custom). Built-in means that it is used by automatically tracked events or Enhanced Measurement. Recommended parameters are for recommended events. Custom parameters are your own unique events.
- **Column D** is for description (in plain English).

Parameter name	Platform	Type	Description
affiliation	Web	Recommend ▾	The store affiliation for the event. Needed for purchase tracking.
annual_revenue	Web	Custom ▾	The range of the annual revenue that a user provided with the form submission, example: 50-100M
company_industry	Web	Custom ▾	The company industry that a user provided with the form submission, example: "telecom and internet"
company_size	Web	Custom ▾	The company size that a user provided with the form submission, example: "99-250"
coupon	Web	Recommend ▾	Coupon associated with the cart/purchase
currency	Web	Recommend ▾	Local currency of the collected price
file_extension	Web	Built-in ▾	The extension of the file (of the link that was clicked). Example: pdf
file_name	Web	Built-in ▾	The name of the file (of the link that was clicked). Example: /files/whitepaper.pdf
inquiry_type	Web	Custom ▾	The value of the "Inquiry type" dropdown field of the contact form. Possible values: "General inquiry", "Question about billing", "Partnership inquiry", "Other inquiry"

IMPORTANT: This spreadsheet is just an example. You don't have to blindly follow it. If you wish, you can take just some parts and adapt it to your needs.

Once you prepare the plan, then you can track events with Google Analytics 4. Don't rush too soon. Otherwise, you might face the consequences in the long run.

#2.8. Event limits in Google Analytics 4

Another thing to keep in mind that there are some limits (and some are gone (at least for now?). I would always recommend for you to refer [to this page](#) when it comes to limits.

Anyway, here are some of the things you should remember now:

- There is no limit for events in total (at least for now). In Universal Analytics (free account), the allowed limit of hit per property was [10 million](#). At the current moment, there is no such limit in Google Analytics 4. But you never know what might happen in the future :)
- You can have up to 500 unique event names per property (Enhanced Measurement events are not counted). So if you have an event called *login* and also *Login*, these will be counted as two unique event names.
- You can have up to 50 custom dimensions and 50 custom metrics registered in a property. The keyword here is **registered**. This means that you have to go to *All Events > Manage custom definitions* and have them configured there. If you send more custom parameters but they are not registered, you're fine.
- You can send up to 25 parameters with an event
- There are some length limits for event names and parameters as well. Both (the event name and the parameter name) must be up to 40-characters-long.

To learn more about the limits, as always, [refer to this page](#).

Chapter III

Conversions in GA4

#3.1. What is a conversion in general?

I realize that some of my readers who read this e-book might be completely new to web analytics and the concept of conversions. If you are one of them, here's a quick introduction. And if you already know what conversions are, feel free to skip to the next chapter of this e-book.

A conversion is an important interaction that you want your visitors/users to complete. Conversions can be split into micro and macro conversions. An example of a micro-conversion can be a newsletter subscription, a download of a whitepaper, etc.

Micro conversions are usually described as conversions that put your visitors/users one step closer to the main (macro) conversion.

Macro-conversions are the most important interactions, such as a purchase.

By tracking conversions, you can better understand what is working for your business and what isn't. For example, you can create a segment of your users who have made a purchase and then try to understand what they are doing, what is their behavior, etc. Also, conversions are used to measure the effectiveness of advertising campaigns and then redistribute your advertising budget.

Hopefully, this super-brief introduction gave you a better understanding of this concept.

Now, let's learn how to actually track conversions with Google Analytics 4.

#3.2. Goodbye, goals. Hello, conversions!

Those who have worked with the previous versions of Google Analytics, for example, Universal Analytics, are familiar with the term "Google Analytics Goal". Basically, that's how GA was calling conversions in the past.

If you want to treat a page view of the order confirmation page as a conversion, you could create a destination goal. Speaking of other types of Google Analytics goals, you could create goals that:

- Were based on events and their parameters
- Or that were based on the duration of the session (if a visitor spends more than X minutes on your website)
- Or that were based on the number of page/screen views per session

2 Goal description

Name

The most important conversion ever

Goal slot ID

Goal Id 9 / Goal Set 2

Type

Destination ex: thanks.html

Duration ex: 5 minutes or more

Pages/Screens per session ex: 3 pages

Event ex: played a video

Smart Goal Smart Goal not available.

Measure the most engaged visits to your website and automatically turn them into conversions to improve your Google Ads bidding. [Learn more](#)

Also, it's worth mentioning that you could create up to 20 goals per GA view. In GA 4, that limit is 30 conversions per property.

Well, with the rise of GA4, the concept of Google Analytics goals is gone. Now, the most important interactions are called *conversions* (this term has been adopted by the industry and other marketing/analytics tools for many years and even GA users were often referring to goals as "conversions").

So in reality, it's a welcome cosmetic change that's been long overdue. However, the name is not the only thing that has changed. Many things are now different in Google Analytics 4 when it comes to conversions:

- The way we have to configure them is different
- The types of conversions are different (no more out-of-the-box destination, session-duration, or number-of-pageviews goals)

It's all about the event now. It's your job to send them and mark the most important ones as conversions.

Speaking of ways how to configure conversions in Google Analytics 4, there are two options (that you can control) and one that's predefined. Let's start with the latter.

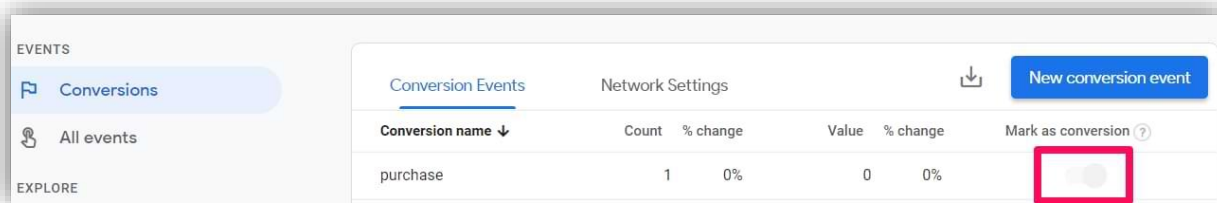
#3.3. Predefined Conversions

There is a bunch of [predefined conversions](#) that you cannot disable:

- `first_open` (applies to mobile applications)
- `app_store_subscription_convert` (applies to mobile applications)
- `app_store_subscription_renew` (applies to mobile applications)
- `in_app_purchase` (applies to mobile applications)
- `purchase` (applies to websites and mobile applications)

Speaking of purchase, here is a guide by Simo Ahava that explains [the entire implementation process](#) in great detail.

If you go to your Google Analytics 4 property and then click *Conversions* (in the left menu), you will see the list of those predefined conversions (they will be displayed there if you have received at least one event of that particular name). If you have only the Web data stream connected to the property, your only default conversion will be *purchase* (and it that cannot be disabled).



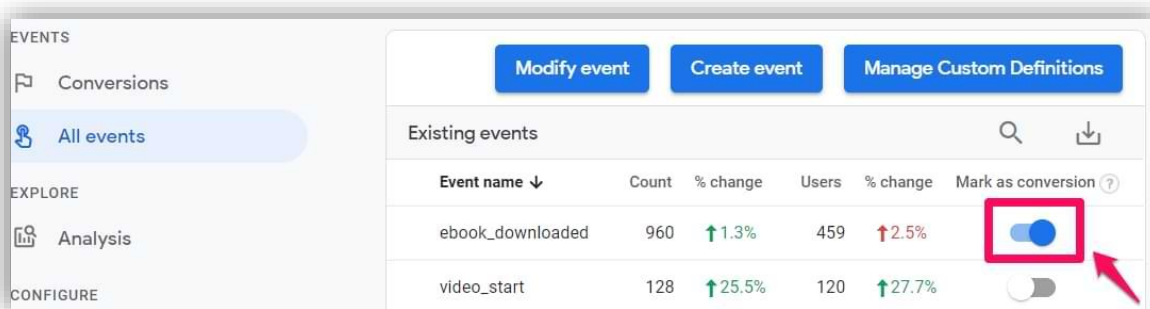
If you have connected any of the app data streams, then you'll most likely see the rest of the aforementioned conversions too. But since there isn't much we can configure, that's all I can say in this chapter.

Now, let's move to the next group of conversions.

#3.4. Conversions that can be switched on

Now, if you want to mark an event as a conversion, you just have to go to the list of *All Events* (on the left sidebar) and then switch the toggle next to the event that's important for your business.

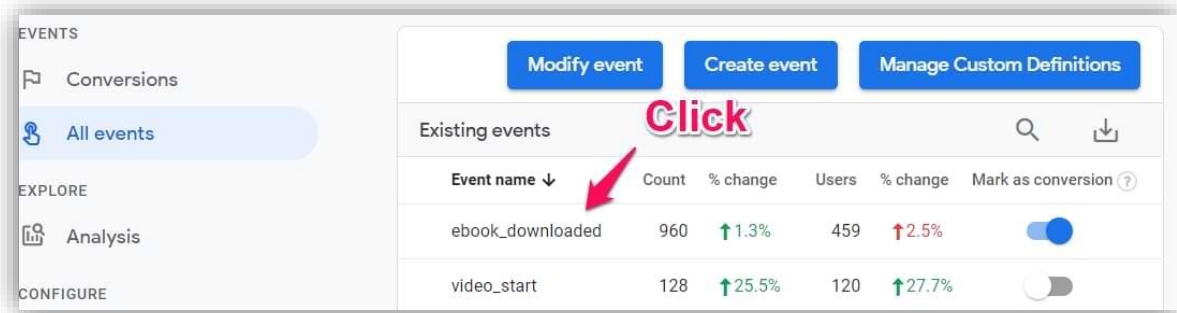
So, for example, if you have an event called *ebook_downloaded*, then you can flip the switch and turn it into a conversion (note: this will apply only to the new data. The events that were collected in the past will not be turned into conversions retroactively).



Alternatively, you can go to *Conversions* (on the left sidebar), then press *New conversion event* and enter the name of the event, for example, *ebook_downloaded*.

There is no difference between entering the name of the event manually and flipping the toggle button in the *All Events* list. Except that if you decide to flip the switch, you will first need to wait until that event appears in the list.

Once you do that, wait for up to 24 hours, and you will start seeing conversion data in the list of all Conversions. Click on the conversion that you wish to inspect closer.



And you will see a report (by default, it's the *Conversions by Event Source*).

If you want to find out about other places where the conversion data is available in the Google Analytics 4 interface, jump to chapter #3.8. of this e-book.

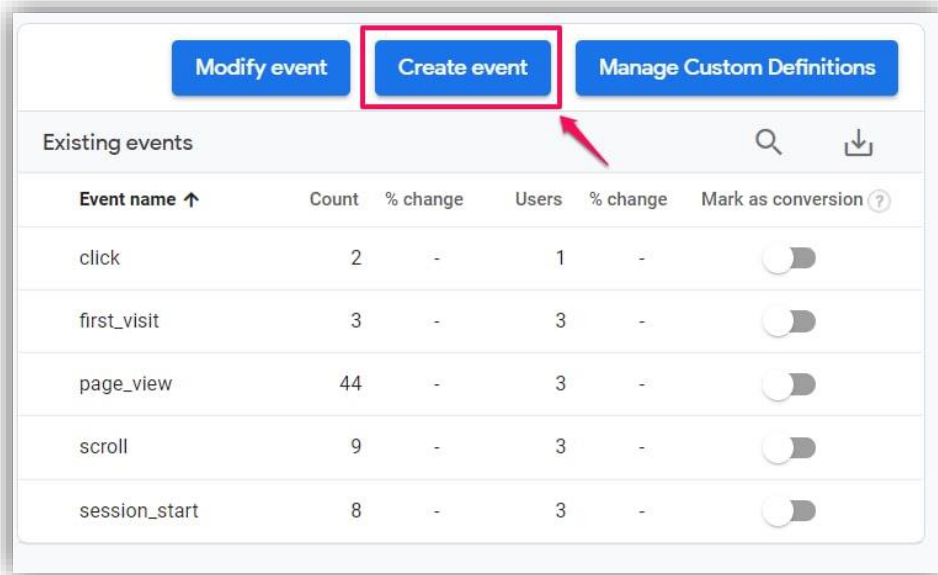
#3.5. Create events and mark them as conversions

I bet that some of my readers have one question now. What if you don't want to mark ALL events (of a certain event name) as conversions?

Example: you have a "Thank you" page to which users are redirected when they subscribe to a newsletter. Let's say that the URL is <https://www.mywebsite.com/thank-you/>. If I mark *page_view* event as a conversion, ANY pageview will become a conversion. How can I separate only those pageviews that happened on the */thank-you/* page?

You can either send a dedicated event (with a different name) from Google Tag Manager/Gtag.js, or you can use the *Create Event* feature in the GA4 interface.

This feature allows you to create a new event based on other incoming events. On the *All Events* page, click *Create Event* and then click *Create*.



Then you will need to enter the name of the custom event. Once again, you can name it whatever you want. Just make sure that the name clearly communicates what it means. *thankyou_page_visit* might be a good option.

Then let's move on to the *Matching Conditions* section. Here we must tell GA4 what kind of event are we looking for. When that particular event will be spotted, then our *thankyou_page_visit* should be created as well.

In my case, I need to enter the following conditions:

- *event_name* **equals** *page_view*
- *page_location* **contains** */thank-you/*

Create event

Analytics Mania
G-W6QB81P6Y6

Configuration

Custom event name ?

Matching conditions

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

Parameter	Operator	Value
<input style="width: 95%; border: none;" type="text" value="event_name"/>	<input style="width: 95%; border: none;" type="text" value="equals"/>	<input style="width: 95%; border: none;" type="text" value="page_view"/>
<input style="width: 95%; border: none;" type="text" value="page_location"/>	<input style="width: 95%; border: none;" type="text" value="contains"/>	<input style="width: 95%; border: none;" type="text" value="/thank-you/"/>

If you want to copy all the parameters from the *page_view* event to the new event, keep the checkbox *Copy parameters from the source event* enabled.

If some of the parameter's name is incorrect and you want to fix it as well, you can *Add Modification* in the *Parameter Configuration* section. For example, if an event contains the parameter *pricingPlan* but you want it to be *pricing_plan*, you can introduce a new field (and reuse its value) while removing the incorrect parameter (by leaving the *New Value* empty).

Matching conditions

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

Parameter	Operator	Value
event_name	equals	page_view
page_location	contains	/thank-you/

[Add condition](#)

Parameter configuration

Copy parameters from the source event

Modify parameters [?](#)

Parameter	New value
pricing_plan	[[pricingPlan]]
[[pricingPlan]]	Example: [[source_parameter_name]] or 1234

[Add modification](#)

Will insert the value of the "pricingPlan" parameter

Empty field will remove the parameter

Take a closer look at [[pricingPlan]] in the screenshot above. Double square brackets mean that GA4 will reuse the value of the parameter *pricingPlan* in that event.

Once you save the changes, you will be able to view them in the Real-time reports and the *DebugView* of GA4.

Also, when you create that new event in GA4 (and you want that to become a conversion), don't forget to mark that event as a conversion in the All Events report.

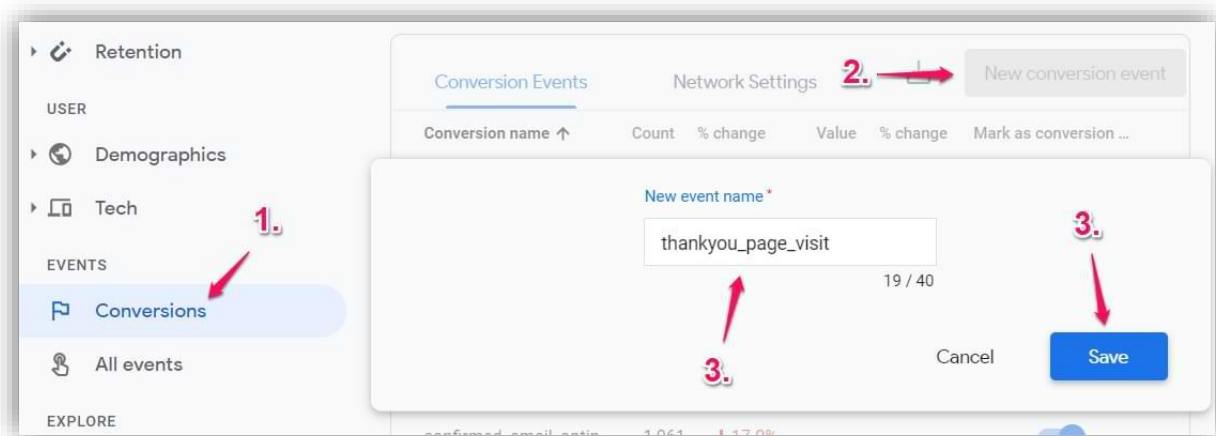
[Modify event](#) [Create event](#) [Manage Custom Definitions](#)

Existing events 🔍 📄

Event name ↑	Count	% change	Users	% change	Mark as conversion ?
thankyou_page_visit	6,228	↑ 13.2%	4,427	↑ 13.2%	<input checked="" type="checkbox"/>
confirmed_email_optin	1,063	↓ 17.7%	809	↓ 15.9%	<input type="checkbox"/>

If you don't like to wait for up to 24 hours (because that *thankyou_page_visit* will not appear in the list of all your events immediately), you can create a new conversion immediately. Click the *Conversions* section on the left sidebar of your GA4 interface.

Then click *New conversion event* and enter the name of the event that you have just created. Click *Save*.



That way, you will mark the newly created *thankyou_page_visit* event as a conversion (without the need to wait for 24 hours until it appears on the "All Events" page).

#3.6. Or try planning your event naming convention better

I'm just continuing the previous chapter of this e-book here.

If we are talking about events that you are sending from the web site's code or Google Tag Manager to GA4, you could just plan your event naming convention better and create more distinct events.

Imagine that you track various form submissions with the event name *form_submission* but you want to treat only certain types of forms as conversions. Let's say that you track contact form submissions, search form

submission, and registration form submission events, but you only want to treat the registration form submission as a conversion.

In that case, you could track 3 separate events:

- *contact_form_submission*
- *search_form_submission*
- *sign_up*

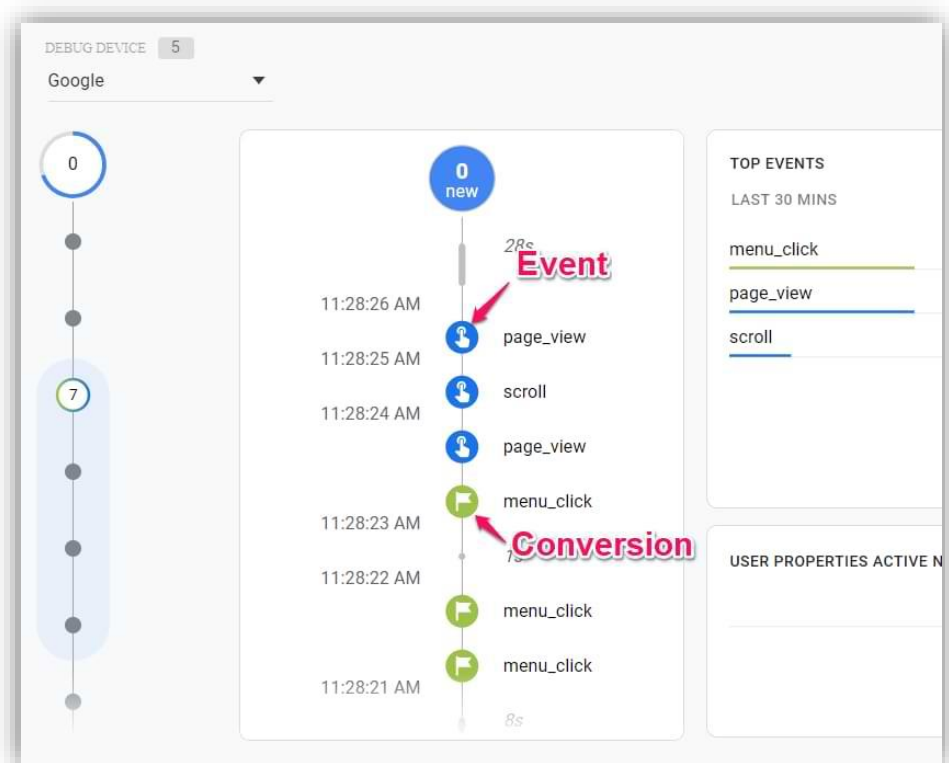
And then mark only the *sign_up* event as a conversion. This, of course, requires more thorough planning upfront but every solid setup starts with one.

#3.7. Check the data in Google Analytics 4 DebugView

Once you configure your conversions, it's time to test them. The primary feature built for debugging GA 4 data is the *DebugView* section. You can find it at the bottom left corner of your GA4 interface. Click it.

If you need a refresher on how the DebugView works, feel free to revisit chapter #1.4. of this e-book.

When the data starts coming into your DebugView, things will look like this.



Events are marked with blue icons, conversions with green.

If you click the event, you will see the parameters that were sent together with an event. Click on the parameter to see its value.

Click to unpause

0 new

28s

11:28:26 AM

11:28:25 AM page_view

11:28:24 AM scroll

11:28:24 AM page_view

11:28:23 AM menu_click

11:28:22 AM menu_click

11:28:21 AM menu_click

11:28:13 AM page_view

11:28:12 AM page_view

2. Click the parameter

1. Click here

3. Here's the value of the parameter

menu_click

Parameters

User properties

debug_mode 1

engagement_time_msec 1

ga_session_id 1

ga_session_number 1

menu_item_name 1

GTM RECIPES

menu_item_url 1

page_title 1

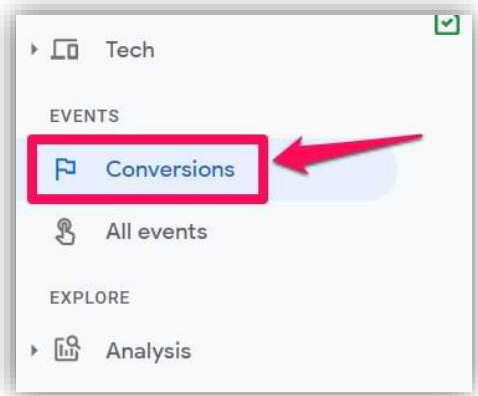
Once you made sure that the data is coming in and it is displayed properly, you should submit your GA4 changes in the GTM container and publish it.

#3.8. Where can I see the conversion data in GA 4 reports?

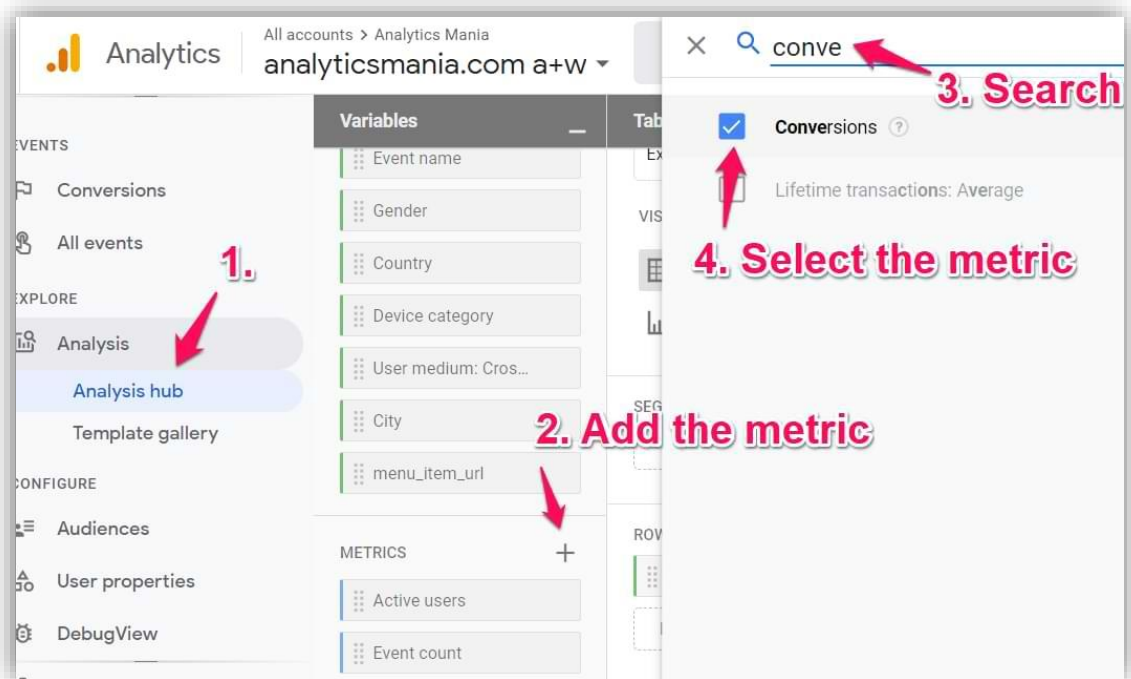
The conversion data is available in various places of the Google Analytics 4 interface. To name a few (but this list is definitely not complete):

- *Conversions* section in the left sidebar menu. You can click on any of those conversions to see how the data changed over time and the main traffic

sources for that particular conversion



- *Acquisition > Traffic Acquisition* and then there is a column *Conversions* in the table
- If you have implemented Ecommerce tracking, the data will appear in the *Monetization* reports
- *Analysis > Analysis hub*. For example, you can include the metric *Conversions* in the Exploration report.



Final words

Google Analytics is a completely different tool (if we compare it to the previous versions of Google Analytics). And this e-book just proves the fact that there's a steep learning curve ahead.

We were just scratching the surface of GA4 capabilities and the e-book is already ~90 pages long.

If you want to learn more about the platform, then keep an eye on analyticsmania.com and [my YouTube channel](#).

Also, if you want to be able to track various interactions and send them to Google Analytics (or any other marketing/analytics tool that you are using), **Google Tag Manager is the best option**. But there's a steep learning curve as well (unless I show you a shorter and more efficient way).

Your Best Way to Learn GTM

I have seen many people trying to learn Google Tag Manager on their own. Even though *some of them* managed to reach the goal, they:

- Spent hundreds of hours
- Got stuck multiple times in the learning process
- Were still lacking some essential GTM knowledge
- Some of them even did costly mistakes because their setup was incorrect

All of this could have been prevented if they decided to get a mentor and learn Google Tag Manager the right way.

Since 2016, I have taught many marketers and analysts to get the most out of GTM. And throughout those years, I have polished a step-by-step system that helps them achieve the wanted result as soon as possible.

If you want to become a GTM power user, check out my [Google Tag Manager Masterclass for Beginners](#). I will teach you the framework that will help you move from *I have no idea about GTM* to *I can climb mountains with GTM*.

My course includes:

- 9 Modules
- Time-saving checklists, templates, etc.
- Many practical tasks and a sandbox website to practice
- Lifetime 24/7 access to the course material
- Free updates
- My feedback
- Complete hand-holding and support
- And so much more!



This is THE GTM course that you need. Stop wasting time while trying to connect puzzle pieces about GTM. I have already done that, and [I can show you everything in my course](#).

This cheat sheet was delivered to you by



Julius Fedorovicius
Founder of [Analytics Mania](#)
julius@analyticsmania.com
[@fedorovicius](#)



www.analyticsmania.com/