



Facebook Pixel and Conversions API

User Guide

The [Facebook Pixel](#) is a piece of JavaScript code for your website that enables you to measure, optimize and build audiences for your ad campaigns. Using the Facebook Pixel, you can leverage the actions people take on your website across devices to inform more effective Facebook advertising campaigns.

The [Facebook Conversions API \(Facebook Server-Side API\)](#) (for web) allows advertisers to send web events from their servers directly to Facebook. Server-side events are linked to a pixel and are processed like browser pixel events. This means that server-side events are used in measurement, reporting, and optimization in the same way as browser pixel events.

Apptrian Facebook Pixel and Conversions API extension for Magento adds Facebook Pixel with Dynamic Ads code on CMS pages (including the home page), category pages, product pages, catalog search pages, catalog advanced search pages, customer registration page, checkout page (default Magento one but also any other without any coding, only by typing page handle (full action name) into our extension config.), and checkout success page. The extension supports Advanced Matching (if the customer is logged in) and has the ability to add custom parameters.

Optionally Facebook Pixel block can be added to any other Magento page or any other extension page (blogs, form pages, etc.) via our extension config by typing page handle (full action name). The Facebook Pixel allows you to place a single pixel on your entire website to report conversions, build audiences and get rich insights into how people use your website.

Conversion measurement lets you track actions people take after viewing your Facebook Ads across multiple devices, including mobile phones, tablets, and desktop computers. By creating a Facebook Pixel and adding it to the pages of your website where conversions happen, like the checkout page, you will see who converts as a result of your Facebook Ads. The pixel will continue to monitor the actions people take after clicking on your ad. You can see which device they saw the ad on and which device they ultimately converted on.

Account & Pricing

To use this extension, a free [Facebook Business Manager](#) account is required.

NOTE: Facebook Advertising service is not free; additional charges apply to users running Facebook Ad Campaigns.

Features

Facebook Pixel

- An easy way to enable or disable Facebook Pixel.
- Advanced Matching (if the customer is logged in).
- Works out of the box just type your Facebook Pixel ID (or comma-separated IDs for multipixel setups) in our extension config.
- Option to type page handles (full action names) where you want Facebook Pixel. Using this option Facebook Pixel can be added to any other Magento page or any other extension page (blogs, form pages, etc.).
- Option to enable or disable the Facebook Pixel base code.
- Option to enable or disable the Facebook Pixel noscript tag.
- Individual options to enable or disable AddToCart, AddToWishlist, CompleteRegistration, InitiateCheckout, PageView, Purchase, Search, and ViewContent events.
- Option to fire PageView event with other events: CompleteRegistration, InitiateCheckout, Purchase, Search, and ViewContent.
- Option to move parameters outside contents.
- Options to detect selected SKUs for all product types including bundle, configurable, and grouped products. **WARNING! ***

* **WARNING!** Facebook Pixel (client-side) detection of selected SKUs feature is heavily dependent on the theme you are using and its customization. It is impossible to predict all possible themes and customizations. Because of this detection will not work on every theme. This is why extension default configuration comes with this feature turned off. You can turn it on, try it out if it is working for you and if it does leave it on, otherwise turn it off. If you are a developer you will be pleased to know that everything is conveniently located in **code.phtml** file. To make it work, in some cases only the adjustment of few jQuery selectors is needed in others complete rewrite of the detection code is needed. Because of this, **we do not give any warranty for the detection feature, nor it is covered by our support service.**

Unlike Facebook Pixel client-side detection, the Facebook Server-Side detection is not dependent on the theme you are using and its customization. Therefore if you want to use detection, we recommend you to use server-side detection and events.

Facebook Conversions API (Facebook Server-Side API)

- An easy way to enable or disable Facebook Conversions API (Facebook Server-Side API).
- Advanced Matching (if the customer is logged in).
- Option to type your Facebook Access Token in our extension config.
- Option to type your Facebook API Version in our extension config.
- Option to type your Test Event Code in our extension config.
- Option to log server-side fired events in the Magento log file.
- Individual options to enable or disable AddToCart, AddToWishlist, CompleteRegistration, InitiateCheckout, PageView, Purchase, Search, and ViewContent events. * **Notice**
- Option to fire PageView event with other events: CompleteRegistration, InitiateCheckout, Purchase, Search, and ViewContent.
- Option to move parameters outside contents.
- Options to detect selected SKUs for all product types including bundle, configurable, and grouped products.

* **Notice** Since version 4.4.0 you can send both the Facebook Pixel event and Facebook Conversions API event simultaneously. The server-side events are not visible in the Facebook Pixel Helper extension for Chrome. You can see server-side events in your Facebook Business Manager.

Category (Category Page)

- Option to type page handles (full action names) where you want category-related code.
- Option to type the event name you want to use for category tracking.
- Options to add custom parameters to your category-related event with parameters to attributes mapping. Example: `google_product_category=google_product_category|name=name`. The format is simple `param1=attribute1|param2=attribute2` Facebook Pixel custom parameter and Magento category attribute are connected by = sign and pairs are separated by | sign.

Product (Product Page - AddToCart, AddToWishlist, ViewContent Events)

- Option to type page handles (full action names) where you want product-related code.
- Option to select SKU you want to use for bundle products. Options: Product SKU as (id), Children SKUs as (ids), Children SKUs as (ids) and Product SKU as (item_group_id)
- Option for content_type to use product_group instead of product for bundle products.
- Option to select SKU you want to use for configurable products. Options: Product SKU as (id), Child SKU as (id), Child SKU as (id) and Product SKU as (item_group_id)
- Option for content_type to use product_group instead of product for configurable products.
- Option to select SKU you want to use for downloadable products. Options: Product SKU as (id), Product SKU as (id) and Parent SKU as (item_group_id)
- Option to select SKU you want to use for grouped products. Options: Product SKU as (id), Children SKUs as (ids), Children SKUs as (ids) and Product SKU as (item_group_id)
- Option for content_type to use product_group instead of product for grouped products.
- Option to select SKU you want to use for simple products. Options: Product SKU as (id), Product SKU as (id) and Parent SKU as (item_group_id)
- Option to select SKU you want to use for virtual products. Options: Product SKU as (id), Product SKU as (id) and Parent SKU as (item_group_id)
- Option to use content_ids instead of contents for the ViewContent event.
- Option to type the Magento product attribute code you want to use for the global content_name parameter.
- Option to type the Magento product attribute code you want to use for the global content_category parameter.
- Options to add custom parameters to your product-related events with parameters to attributes mapping. Example: `google_product_category=google_product_category|color=color`. The format is simple `param1=attribute1|param2=attribute2` Facebook Pixel custom parameter and Magento product attribute are connected by = sign and pairs are separated by | sign.

Quote (Checkout Page - InitiateCheckout Event)

- Option to type page handles (full action names) where you want quote-related code.
- Option to select SKU you want to use for bundle products. Options: Product SKU as (id), Children SKUs as (ids), Children SKUs as (ids) and Product SKU as (item_group_id)
- Option to select SKU you want to use for configurable products. Options: Product SKU as (id), Child SKU as (id), Child SKU as (id) and Product SKU as (item_group_id)
- Option to select SKU you want to use for downloadable products. Options: Product SKU as (id),

Product SKU as (id) and Parent SKU as (item_group_id)

- Option to select SKU you want to use for grouped products. Options: Children SKUs as (id)s, Children SKUs as (id)s and Product SKU as (item_group_id)
- Option to select SKU you want to use for simple products. Options: Product SKU as (id), Product SKU as (id) and Parent SKU as (item_group_id)
- Option to select SKU you want to use for virtual products. Options: Product SKU as (id), Product SKU as (id) and Parent SKU as (item_group_id)
- Option to type parameter name if you want to use Magento quote ID.
- Options to add custom parameters to your quote-related event with parameters to attributes mapping. Example: google_product_category=google_product_category|color=color. The format is simple param1=attribute1|param2=attribute2 Facebook Pixel custom parameter and Magento product attribute are connected by = sign and pairs are separated by | sign.

Order (Checkout Success Page - Purchase Event)

- Option to type page handles (full action names) where you want order-related code.
- Option to select SKU you want to use for bundle products. Options: Product SKU as (id), Children SKUs as (id)s, Children SKUs as (id)s and Product SKU as (item_group_id)
- Option to select SKU you want to use for configurable products. Options: Product SKU as (id), Child SKU as (id), Child SKU as (id) and Product SKU as (item_group_id)
- Option to select SKU you want to use for downloadable products. Options: Product SKU as (id), Product SKU as (id) and Parent SKU as (item_group_id)
- Option to select SKU you want to use for grouped products. Options: Children SKUs as (id)s, Children SKUs as (id)s and Product SKU as (item_group_id)
- Option to select SKU you want to use for simple products. Options: Product SKU as (id), Product SKU as (id) and Parent SKU as (item_group_id)
- Option to select SKU you want to use for virtual products. Options: Product SKU as (id), Product SKU as (id) and Parent SKU as (item_group_id)
- Option to type parameter name if you want to use Magento order ID.
- Option to type parameter name if you want to use Magento order increment ID.
- Option to type parameter name if you want to use Magento quote ID.
- Options to add custom parameters to your order-related event with parameters to attributes mapping. Example: google_product_category=google_product_category|color=color. The format is simple param1=attribute1|param2=attribute2 Facebook Pixel custom parameter and Magento product attribute are connected by = sign and pairs are separated by | sign.

Search (Search Result Page - Search Event)

- Option to type page handles (full action names) where you want search-related code.
- Option to type the event name you want to use for search tracking.
- Option to type the parameter name you want to use for search tracking.
- Option to type request parameters you want to include for search tracking.

Data Processing Options

- An easy way to enable or disable Data Processing Options.
- Option to type Data Processing Options as a comma-separated list.
- Option to type Data Processing Options Country
- Option to type Data Processing Options State

Miscellaneous

- Option to select Facebook product identifier. Magento product SKU (Recommended) or Magento product ID.
- Option to enable or disable customer external_id (Magento customer ID).

Installation Instructions

There are several ways you can install any Magento extension. Our extension is no different. We will show you three ways to install the extension, but you must not mix them. Choose one and stick to it.

If you do not know how to install an extension or you wish a professional to do it for you, we offer an additional installation service for a small fee.

Installation via file uploading (If you purchased the extension on our site)

If you purchased the extension on our site then unpack .zip file from /Magento2/InstallByUploadingFiles/ directory inside your Magento root.

- Run following commands on Magento root directory:

```
php bin/magento maintenance:enable
php bin/magento module:enable --clear-static-content Apptrian_FacebookPixel
php bin/magento setup:upgrade
php bin/magento cache:flush
php bin/magento setup:static-content:deploy
php bin/magento maintenance:disable
php bin/magento cache:flush
```

Installation via Composer (If you purchased the extension on our site)

If you purchased the extension on our site then copy .zip file from /Magento2/InstallWithComposer/ directory inside your MAGENTO_ROOT/vendor/apptrian/packages/ directory (create directory if does not exist).

- Run following commands on Magento root directory:

```
php bin/magento maintenance:enable
composer config repositories.apptrianartifacts artifact $(pwd)/vendor/apptrian/packages
composer require apptrian/facebook-pixel
php bin/magento module:enable --clear-static-content Apptrian_FacebookPixel
php bin/magento setup:upgrade
php bin/magento cache:flush
php bin/magento setup:static-content:deploy
php bin/magento maintenance:disable
php bin/magento cache:flush
```

Installation via Composer (If you purchased the extension on Magento Marketplace)

If you purchased the extension on Magento Marketplace then you must use Composer. Magento Marketplace does not allow extension downloads.

- Run following commands on Magento root directory:

```
php bin/magento maintenance:enable
composer require apptrian/facebook-pixel
php bin/magento module:enable --clear-static-content Apptrian_FacebookPixel
php bin/magento setup:upgrade
php bin/magento cache:flush
php bin/magento setup:static-content:deploy
php bin/magento maintenance:disable
php bin/magento cache:flush
```

Please make sure your Magento public and private keys are in your Magento root auth.json file. Usually, people install Magento with one Magento Marketplace account (public and private keys) and then purchase an extension with another Magento Marketplace account (different public and private keys). Check the Magento Marketplace account you used to purchase the extension and make sure it's public and private keys are in the Magento root auth.json file. **WARNING!** if you already have keys there, be very careful because maybe some other extensions are purchased with another Magento Marketplace account.

Configuration

Our extension works out of the box. The only thing you need to do is type your Facebook Pixel ID and Access Token in our extension configuration. To do this, log in to your Magento Admin and go to:

Stores > Configuration > Apptrian Extensions > Facebook Pixel and Conversions API > Facebook Pixel (Client-Side Settings) > Facebook Pixel ID

Stores > Configuration > Apptrian Extensions > Facebook Pixel and Conversions API > Facebook Conversions API (Server-Side Settings) > Facebook Access Token

See FAQ on how to get Access Token.

All options are self-explanatory and have comments and tooltips. Make sure you read them.

After changing options, make sure you refresh your Magento cache.

WARNING! You need to remove all other implementations of Facebook Pixel and use only our extension.

- Remove all other extensions that fire Facebook Pixel.
- Remove all theme customizations that fire Facebook Pixel.
- If you are using GTM (Google Tag Manager), make sure it is not firing Facebook Pixel events.
- If you have set "Event Triggers" in "Facebook Event Setup Tool", you need to remove them all and then disable the "Track Events Automatically Without Code" option.

FAQ

Q: How do I get Facebook Access Token?

A: There are two ways of getting your Access Token:

- Via Events Manager (Recommended)
- Using Your Own App

To get Access Token via Events Manager (Recommended), go to Facebook Business Manager and Events Manager. Choose the pixel you want to implement. Select the **Settings** tab. Find the Conversions API section and click on **Create Access Token** under **Set up manually**, and follow the instructions pop up. Once you

have your token (save it to your computer because you will not be able to see it again). The Events Manager gives you access to a new tab called **Implementation**. Navigate to the **Implementation** tab and follow the instructions displayed on the screen. This way, Facebook automatically creates a Conversions API App and Conversions API System User for you. There is no need to go through the app review or request any permissions.

Q: Facebook Conversions API (Facebook Server-Side API) does not work?

A: Facebook deprecated v6.0 of the Server-Side API. Go to our extension config and for the Facebook API Version option type v12.0 or newer, then refresh your Magento cache (System > Cache Management) and you are done.

Q: In Facebook Event Manager, there are errors "(X) Event Missing Deduplication Parameter"?

A: This error is caused by using other Facebook Pixel solutions in parallel with ours. You need to remove all other implementations of Facebook Pixel and use only our extension. That means:

- Remove all other extensions that fire Facebook Pixel.
- Remove all theme customizations that fire Facebook Pixel.
- If you are using GTM (Google Tag Manager), make sure it is not firing Facebook Pixel events.
- If you have set "Event Triggers" in "Facebook Event Setup Tool", you need to remove them all and then disable the "Track Events Automatically Without Code" option.

Q: Is it possible to add the Facebook Pixel to other Magento pages or extension pages?

A: Yes it is. In our extension config just use the option "Add Facebook Pixel To" and type your page handle (full action name). For example, if you want to add Facebook Pixel to a blog page and that blog page handle is "blog_post_view", you would type this into "Add Facebook Pixel To" at the end. Page handle depends on a particular extension this is just an example. Do not forget to Refresh your Magento Cache.

Q: When I check my product pages with the "Facebook Pixel Helper" extension for Chrome the AddToCart and AddToWishlist events have orange triangles with warnings?

A: The "Facebook Pixel Helper" extension for Chrome does not understand JavaScript/jQuery that is why on page load you have orange triangles for AddToCart and AddToWishlist events. If you read the warnings of the "Facebook Pixel Helper" there is an explanation: "We detected event code but the pixel has not activated for this event, so no information was sent to Facebook. This could be due to an error in the code, but could also occur if the pixel fires on a dynamic event such as a button click." AddToCart and AddToWishlist are dynamic events triggered by a button click. After you click a button they will become green. This has nothing to do with our extension nor Facebook Pixel, it is the behavior of the "Facebook Pixel Helper" extension for Chrome. In short, this is normal.