

Google Analytics Unity SDK

Anyone can comment

[Short Overview](#)

[Released Apps with the plugin](#)

[Setup](#)

[How To Use](#)

[Web Player](#)

[PlayMaker Actions](#)

[How to Get Support](#)

Short Overview

You can easily implement all power of Google Analytics to your game.

Plugin has clean and easy to use implementation of Google Analytics [Measurement Protocol](#). Also you can find starting forward use example inside.

As a developer, you can extend the functionality of the Google Analytics service — from customizing data collection for web and mobile, to programmatically accessing Analytics view (profile) and reporting data, to building your own Analytics solution. The Google Analytics developer platform provides access to the resources used to collect, configure, and report on user interactions to your online content.

Released Apps with the plugin

Setup

Get started with Analytics

Step 1: Sign up for an account.

Visit the [Google Analytics website](#) to sign up for a new account. Click the *Access Google Analytics* button and follow the on-screen instructions.

Step 2: Set up account properties.

Google Analytics is flexible and can support different account configurations, but your setup affects how data appears in your reports. To help plan your setup, consult the articles in our [overview of accounts, users, properties, and views](#).

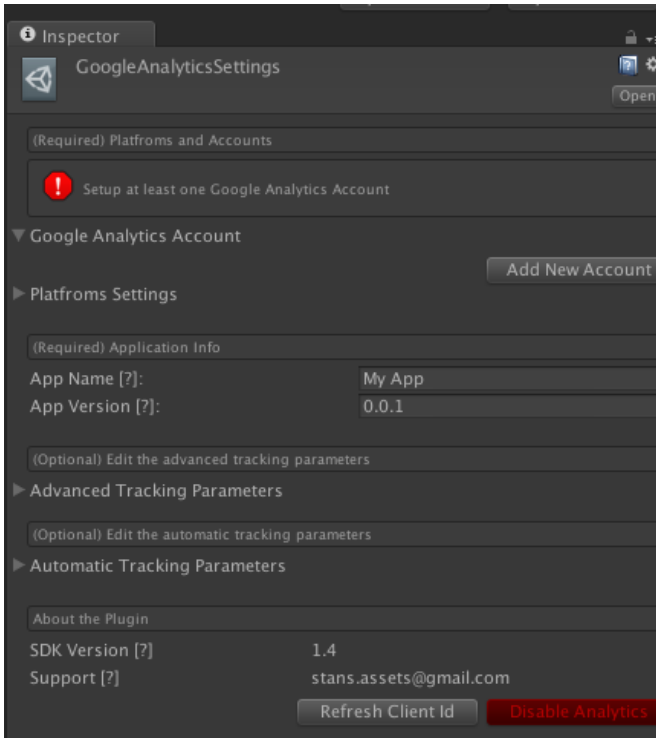
If you're setting up Google Analytics to track an app, consult our [best practices for mobile-app analytics setup](#).

Step 3: Set up your tracking code.

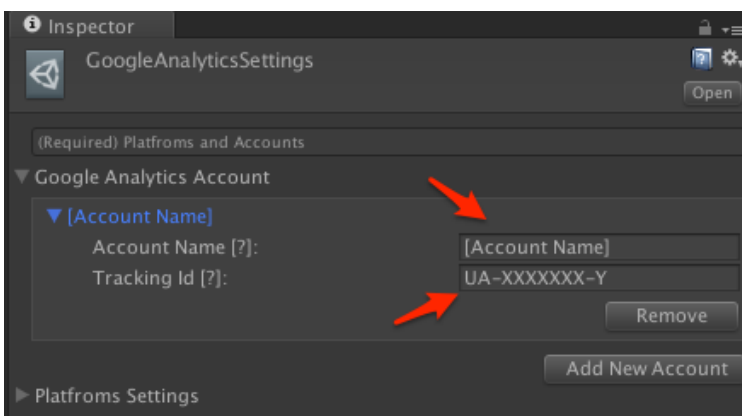
Include the Analytics tracking ID and code in your website or mobile app in order to collect and send usage data to your Analytics account. You can track multiple properties in the same account, but every property must be set up individually. Follow different processes to [set up web tracking](#) and to [set up mobile app tracking](#).

Plugin Setup

1) Go to the Window → Google Analytics → Edit Settings



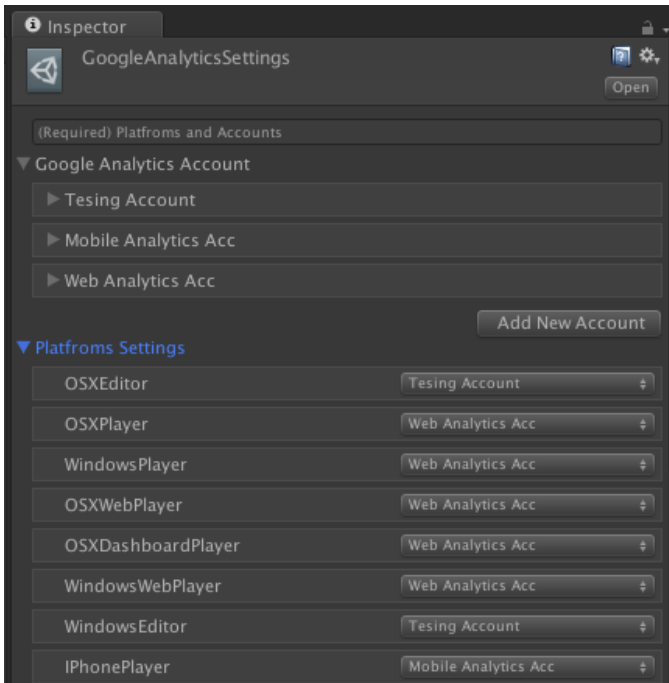
This will open analytics settings window. Where you set your profile data by pressing **Add New Account** button.



Under Account data data fill your **Account Name** and **Tracking Id**.

Note: First created account will be treated as default acc

Note: You can add more accounts and connect platform to account using **Platforms** tab.



Feel in other string under required setting tab, and you can get started to use plugin API.

- **App Name.** For your own use and organization.
- **App Version.** Your application build version.

Feel free to modify and experiment with another settings marked as “Optional”

How to use

Initialization

There is two ways to initialize google analytics.

1) Simply add Google Analytics Gameobject on the your initial(menu) scene.

You may use

Window → Google Analytics → Create Analytics GameObject

2) You can initialize analytics from code

```
GoogleAnalytics.StartTracking();
```

WARNING: This call will be ignored if you already have Google Analytics game object on your scene, like in the example scene

WARNING: if you do not have Google Analytics game object and you skip **StartTracking** call, GoogleAnalytics game object will be initialized on first **GoogleAnalytics.Client** call.

Data reporting

Some data is tracked automatically after initialization. You can of course disable it under plugin settings window. By default automatic tracking is enabled for:

- Exceptions
- System Info
- Levels

Additional data can be sent from any part of your code. You can send common hit types described in [Measurement Protocol Developer Guide](#), using *GoogleAnalytics.Client* class functions

- `SendPageHit`
- `SendEventHit`
- `SendTransactionHit`
- `SendItemHit`
- `SendSocialHit`
- `SendExceptionHit`
- `SendUserTimingHit`
- `SendScreenHit`

You may also construct own Hits. To do this use.

GoogleAnalytics.Client.CreateHit function

Then add [Measurement Protocol](#) data using correspondent function and sent hit using

GoogleAnalytics.Client.Send function.

Web Player

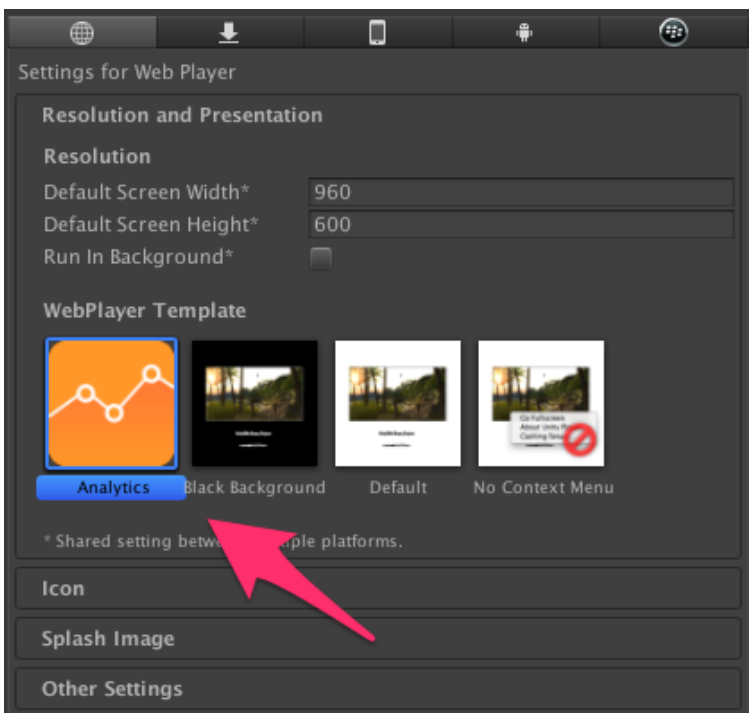
Due to [Security Sandbox of the Webplayer](#) we not able to send data directly from Unity Web Player on a domain other than the one hosting your .unity3d file.

That's why plugin use additional java script for sending the data.

And that's why we not able to send data from the editor while we in the web player mode. Because in the web player mode Editor will emulate this restriction.

Here is small guide how to enable analytics data sending from the Web Player.

One the [Player Settings](#) before building and choose **Analytics** template in the Web Player settings.



In case you want to use your own template, you need to connect the **GoogleAnalyticsUnityPlugin.js** script to your Web Player template. With can be founded at: **Assets/WebPlayerTemplates/Analytics/js/GoogleAnalyticsUnityPlugin.js**

PlayMaker Actions

The plugin now contains playmaker actions.

The actions scripts can be found in the zip archive at:

Assets/Extensions/GoogleAnalytics/Addons/PlayMakerActions

You can simply unzip it to the same folder and Google Analytics actions will appear under playmaker actions menu.

You always welcome on the [PlayaMker Actions Forum Thread](#) to request new actions or report a bug.

The current actions list is:

- GA_EventHit
- GA_ItemHit
- GA_PageHit
- GA_ScreenHit
- GA_SocialHit
- GA_TransactionHit
- GA_UserTimingHit