

# Google Analytics

You can gather usage statistics for your ViziApps mobile app just like you use [Google Analytics](#) for your website. You will need to set up your Google Analytics account defined with a new mobile app, and then add some custom JavaScript code to your app design. Note that Google Analytics is subject to certain [usage and volume limits](#).

## Setup Google Universal Analytics

1. Log in to your [Google Analytics Account](#)
2. Click on **Admin** in the top menu bar
3. In the **PROPERTY** list, click on **Create New Property**
4. Select **Mobile app** under **What would you like to track?**
5. Enter your **App Name** (which you probably want to be similar to your mobile app's name), **Industry Category**, and **Reporting Time Zone**.
6. Click on **Get Tracking ID**.
7. Copy the **Tracking ID** on the **Tracking Code** page that is shown.

## Universal Analytics Sample Code

1. Open your app design in ViziApps Studio, select the Custom HTML Header, and enter the following code.
2. You must replace 'UA-XXXX-Y' with your Universal Analytics **Tracking ID**.
3. In the appStartup() function, enter the names of any pages and buttons that you want to track. Google Analytics uses the term *screens* whereas ViziApps uses the term *pages*, but these are the same.

```
<script
src="https://s3.amazonaws.com/viziapps/apps/Studio/google_analytics1.0.js"></s
cript>
<script>
// ! I M P O R T A N T ! ENTER YOUR OWN GOOGLE ANALYTICS TRACKING ID:
var gua = new v.googleAnalytics( 'UA-XXXX-Y' );

// Set the event handlers for measurement reporting, after the app is ready
function appStartup()
{
    // enter the names of any pages you want tracked, such as:
    gua.pages( 'SecondPage, ThirdPage' );

    // enter the names of any buttons you want tracked, such as:
    gua.buttons( 'helpButton, loadDataButton, clearDataButton' );
}
```

```
}  
$(document).on( "DOMContentLoaded", appStartup );  
</script>
```

You can use the wildcard, as in `gua.pages('*')` to measure all page views, and `gua.buttons('*')` to measure all button presses. This may be more useful in app testing but be careful in production as use of the wildcard may produce too many nuisance measurements and cause you to exceed Google's limits.

## View Real-Time Results

It may take a day for usage statistics to accumulate. However, you can see immediate results from page views and events such as button taps in the Real-time display. 

1. Log in to your [Google Analytics Account](#)
2. Click on **Home**, then select your new Mobile App
3. Select the **Real-time** menu, and then **Screens**, or **Events**
4. You can test your app using App Preview or on a device.

## Page View Measurements

Instead of automating page view measurements with the `gua.pages()` function, you can programmatically issue any page view measurement with the `gua.screen(screen_name)` function. For example, you may only want to report a page view after a data query has succeeded rather than each time the page shows, or even report a different string than the actual page name:

```
gua.screen( 'Data Query Page Successful' );
```

## Event Measurements

The `gua.buttons()` function is just a specific format of event measurement. You can use the `gua.event()` function to report measurements on a wide variety of events in your app, in this format: **`gua.event( Category, Action, Label, Value )`**

<b>Category</b>	Required	The string name you supply for the group of objects you want to track.
<b>Action</b>	Required	A string that is uniquely paired with each category, and commonly used to define the type of user interaction for the web object.
<b>Label</b>	Optional	An optional string to provide additional dimensions to the event data.
<b>Value</b>	Optional	An integer that you can use to provide numerical data about the user event.

You can read a more complete description of event tracking in [Google Analytics Event Tracker Guide](#). Here are some examples:

```
// different actions within a category:
gua.event( 'Videos', 'Play', 'Gone With the Wind' );
gua.event( 'Videos', 'Pause', 'Gone With the Wind' );
gua.event( 'Videos', 'Stop', 'Gone With the Wind' );

// different labels for an action:
gua.event( 'Videos', 'Select', 'Gone With the Wind' );
gua.event( 'Videos', 'Select', 'Huckleberry Finn' );

// applying integer values:
gua.event( 'Videos', 'Download Minutes', 'Gone With the Wind', 7 );
gua.event( 'Videos', 'Download Minutes', 'Huckleberry Finn', 9 );
```

## Custom Measurements

There are some Universal Analytics measurements that are not handled by page views and events. You can implement any of these using the format specified in the [Google Analytics Measurement Protocol](#). You would use this function: **`gua.measure( hit_type, measurement_set )`**

The **`hit_type`** is a string as defined in the [Parameter Reference](#). And the **`measurement_set`** is a JavaScript object that lists a set of measurement codes and their values. As an example of a transaction hit for [Ecommerce Tracking](#):

```
var measurementSet = {
  ti: '12345',          // transaction ID. Required.
  ta: 'westernWear',   // Transaction affiliation.
  tr: '50.00',         // Transaction revenue.
  ts: '32.00',         // Transaction shipping.
  tt: '12.00',         // Transaction tax.
  cu: 'EUR'            // Currency code.
};
gua.measure( 'transaction', measurementSet );
```

From:

<https://viziapps.com/dokuwiki/> - **ViziApps Help Wiki**

Permanent link:

[https://viziapps.com/dokuwiki/google\\_analytics](https://viziapps.com/dokuwiki/google_analytics)

Last update: **2015/01/14 12:04**

