



ST2228

Web and Mobile Analytics

TOPIC 6: MOBILE ANALYTICS PART II

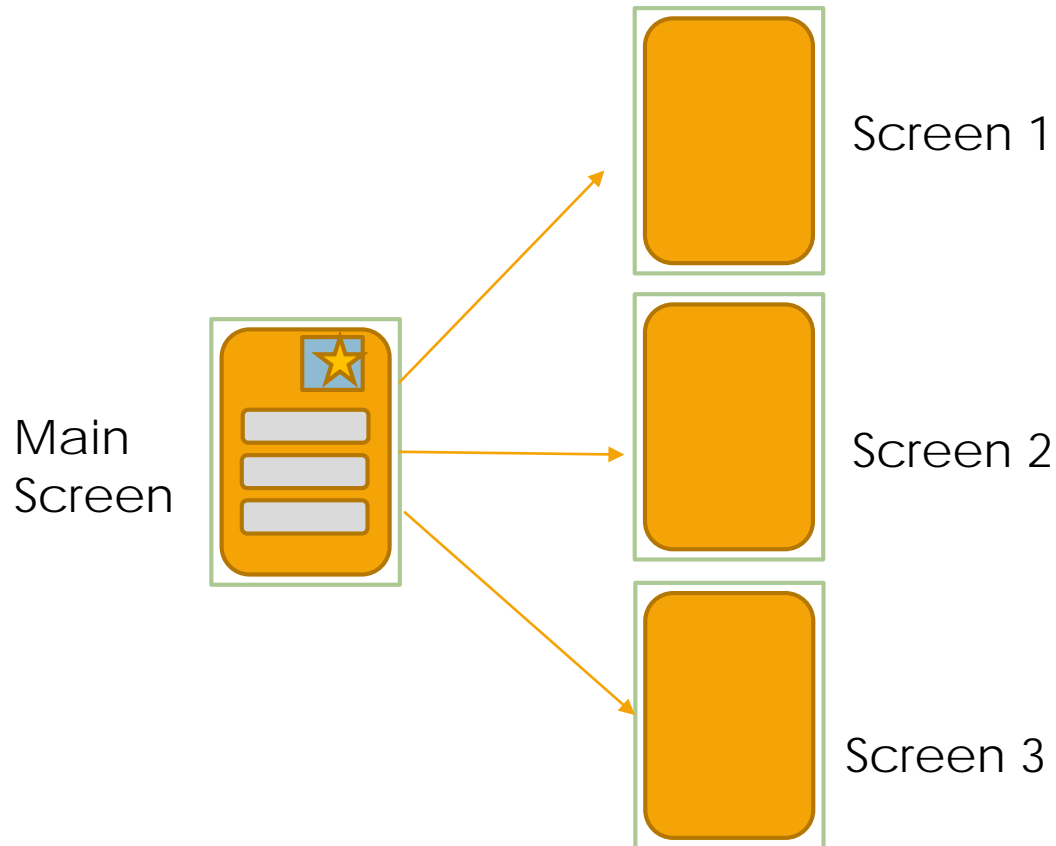
Learning Objectives

- ▶ Create Mobile App with PhoneGap
- ▶ Add the GA account to track the mobile app
- ▶ Adding the PhoneGap GA plug-in
- ▶ Testing instrumented mobile app with the Google Analytics account

PhoneGap

DEVELOPING A SIMPLE
TEST APP USING
PHONEGAP

Simple test app



[About](#)[Developer](#)[Community](#)[Apps](#)[Support](#)

Install PhoneGap

Install

To Install 3.2: Ensure that you have [NodeJS](#) installed, then open your commandline and run the following:

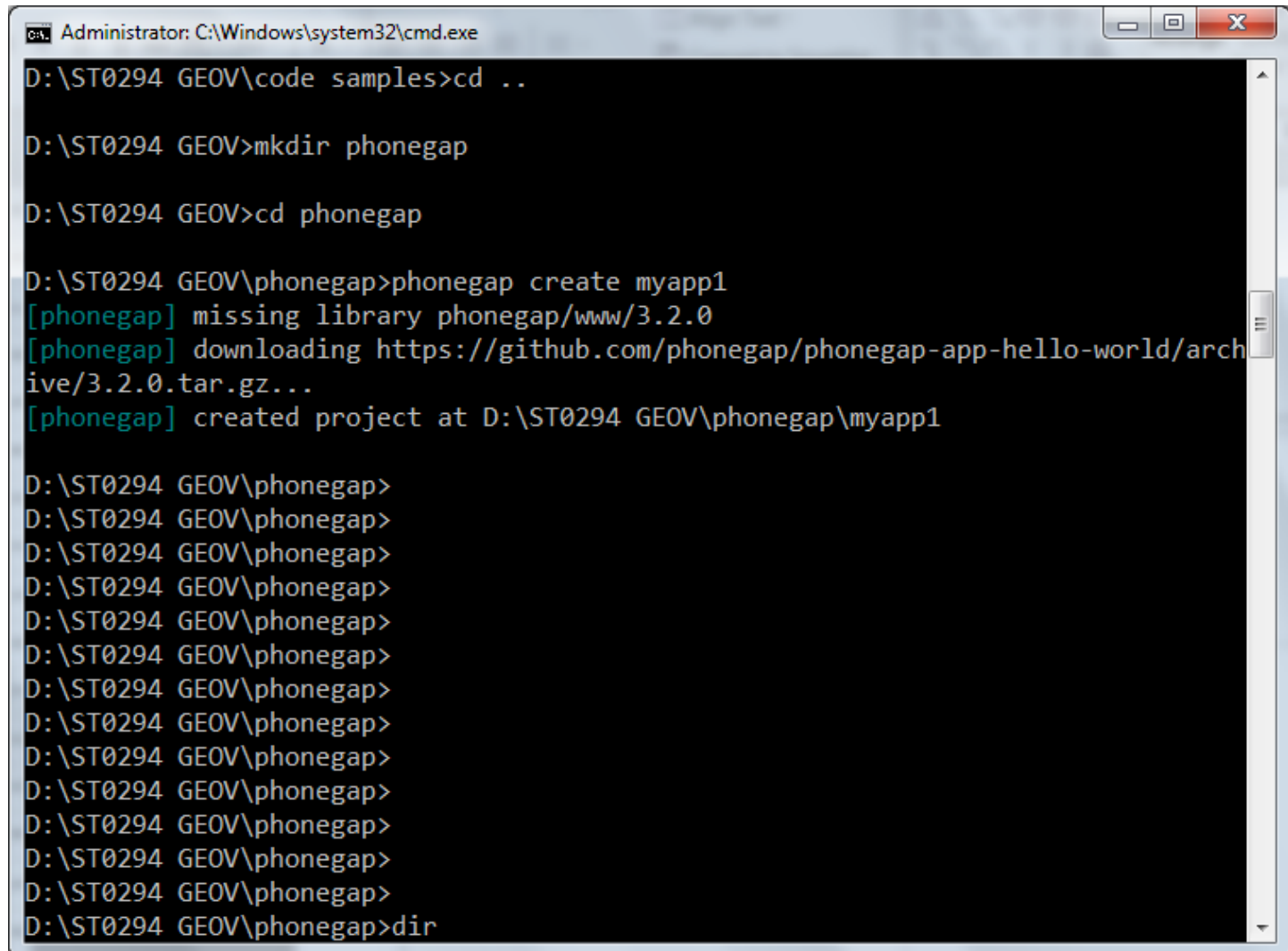
```
C:\> npm install -g phonegap
```

Once installation completes, you can invoke `phonegap` on command line for further help.

Usage

```
$ phonegap create my-app  
$ cd my-app  
$ phonegap run android
```

Learn more at docs.phonegap.com



```
C:\> Administrator: C:\Windows\system32\cmd.exe

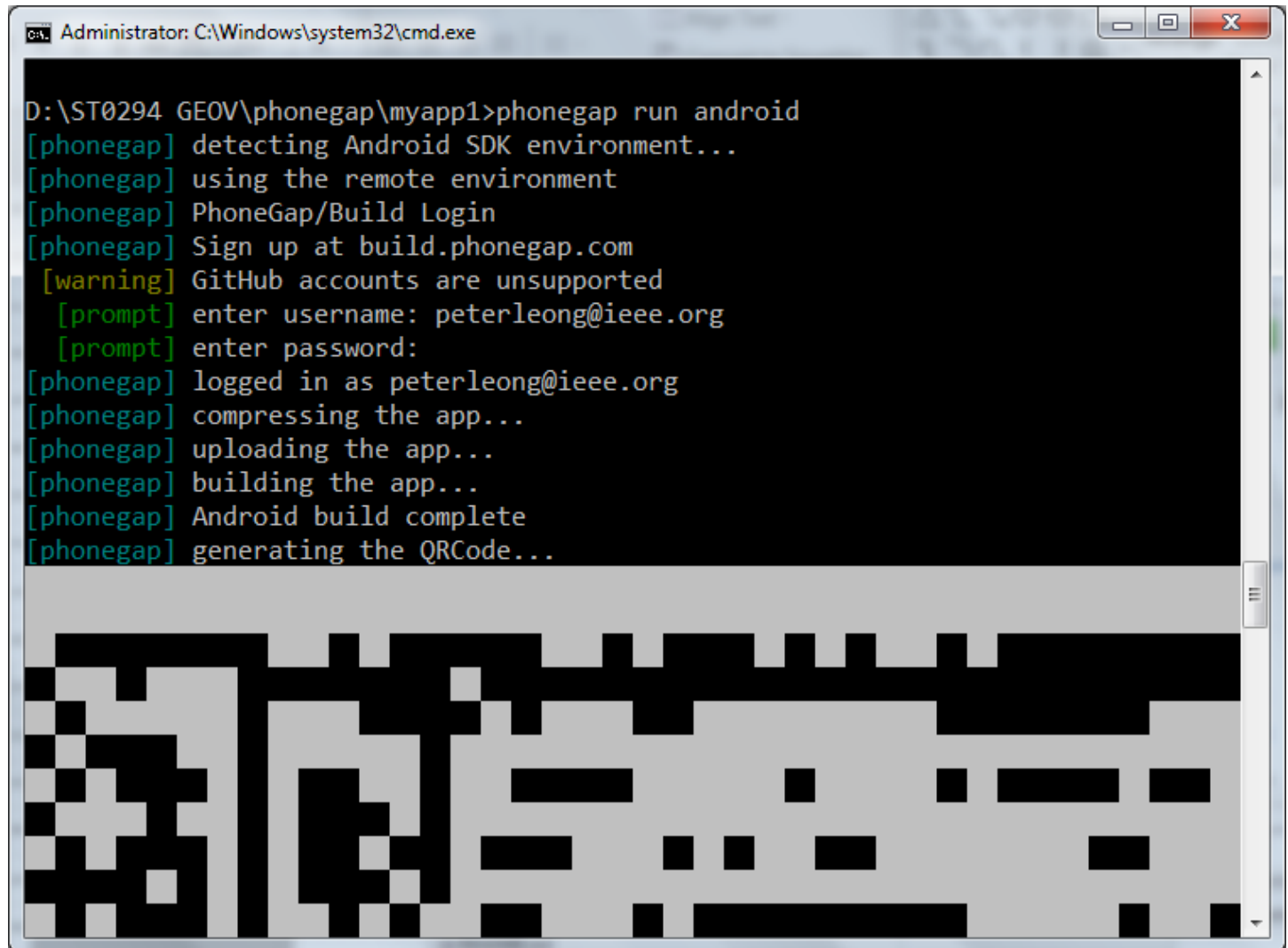
D:\ST0294 GE0V\code samples>cd ..

D:\ST0294 GE0V>mkdir phonegap

D:\ST0294 GE0V>cd phonegap

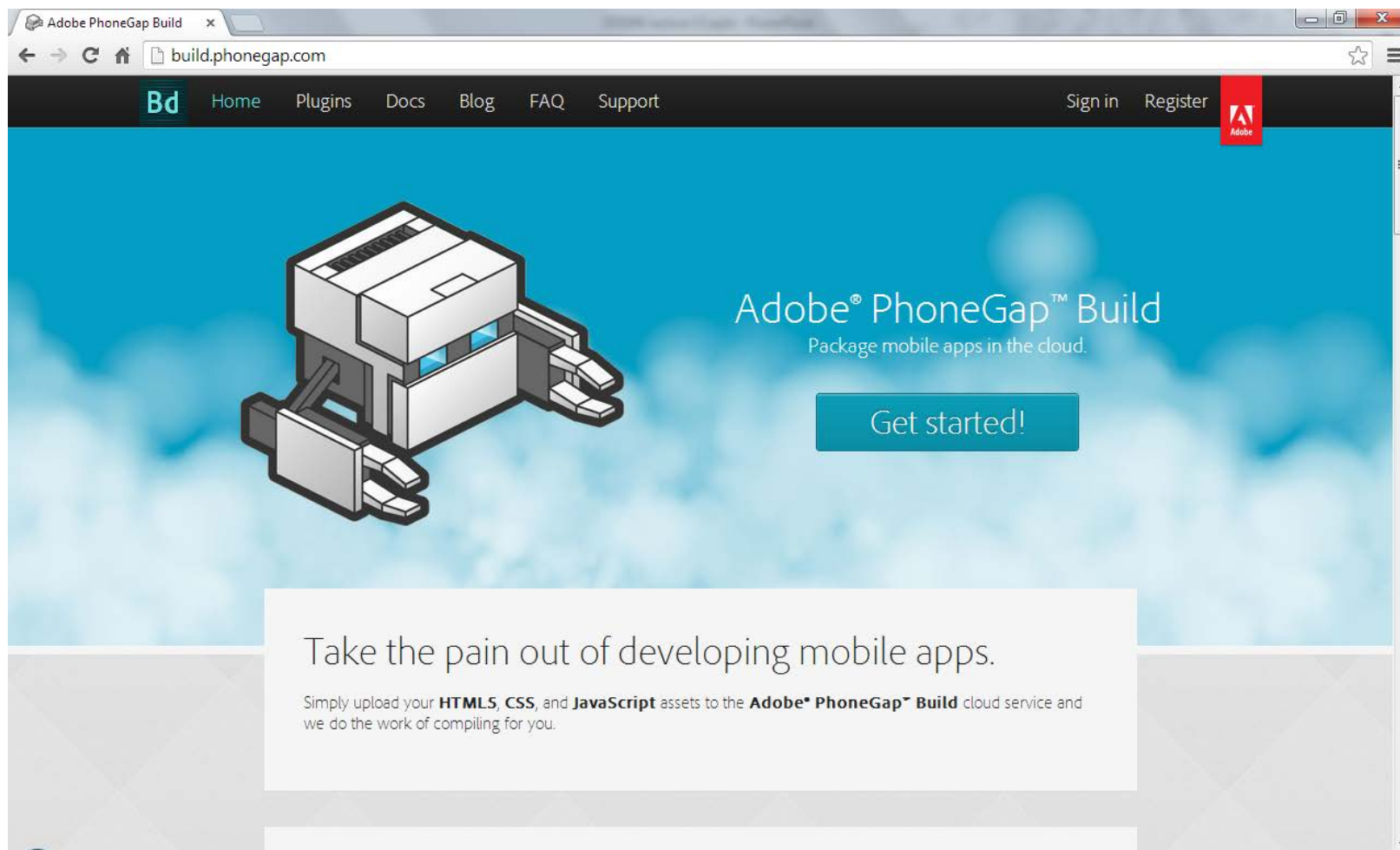
D:\ST0294 GE0V\phonegap>phonegap create myapp1
[phonegap] missing library phonegap/www/3.2.0
[phonegap] downloading https://github.com/phonegap/phonegap-app-hello-world/archive/3.2.0.tar.gz...
[phonegap] created project at D:\ST0294 GE0V\phonegap\myapp1

D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>
D:\ST0294 GE0V\phonegap>dir
```



```
Administrator: C:\Windows\system32\cmd.exe

D:\ST0294 GEOV\phonegap\myapp1>phonegap run android
[phonegap] detecting Android SDK environment...
[phonegap] using the remote environment
[phonegap] PhoneGap/Build Login
[phonegap] Sign up at build.phonegap.com
[warning] GitHub accounts are unsupported
[prompt] enter username: peterleong@ieee.org
[prompt] enter password:
[phonegap] logged in as peterleong@ieee.org
[phonegap] compressing the app...
[phonegap] uploading the app...
[phonegap] building the app...
[phonegap] Android build complete
[phonegap] generating the QRCode...
```


Adobe PhoneGap Build

build.phonegap.com/apps

Bd Apps Plugins Docs Blog FAQ Support

Feedback


Your apps + new app



HelloWorld

Hello World sample application that responds to the deviceready event.


App ID	Version	PhoneGap
670923	1.0.0	2.9.0
Owned by peterleongkw@gmail.com		
Last built (S) 6 minutes		
Source zip package		



private

Update code

Rebuild all



Part of Adobe Creative Cloud

Your PhoneGap Build plan includes a Creative Cloud free membership. Sign in with an AdobeID to receive 2GB of storage access to Adobe Edge Tools & Services and more.

Get started

Language: English

Powered by Adobe® PhoneGap™

Copyright © 2013 Adobe Systems Incorporated. All rights reserved.


PhoneGap is a trademark of Adobe. You may not use PHONEGAP or GAP in the title of any application built using PhoneGap or the

Adobe PhoneGap Build

build.phonegap.com/apps/670923/builds

HelloWorld

Hello World sample application that responds to the deviceready event.



Install

Update code Rebuild all

Builds Plugins Collaborators Settings

App ID	Version	PhoneGap	Owned by	Last built (S)	Source
670923	1.0.0	2.9.0	peterleongkv@gmail.com	8 minutes	zip package

iOS No key selected Rebuild Error

Android No key selected Rebuild apk

Windows No key selected Rebuild OTA Install

Rebuild ipk Rebuild wgz

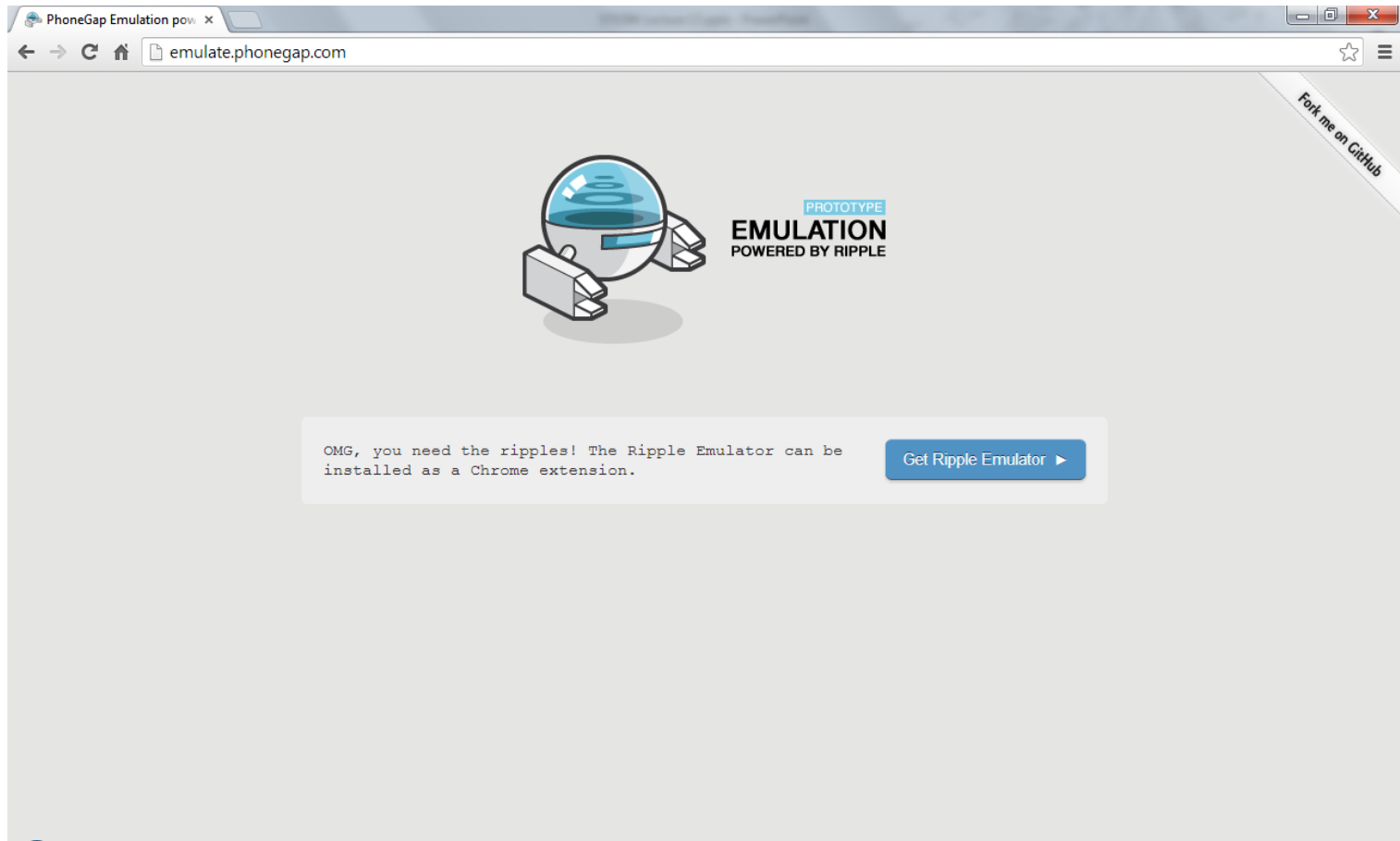
title

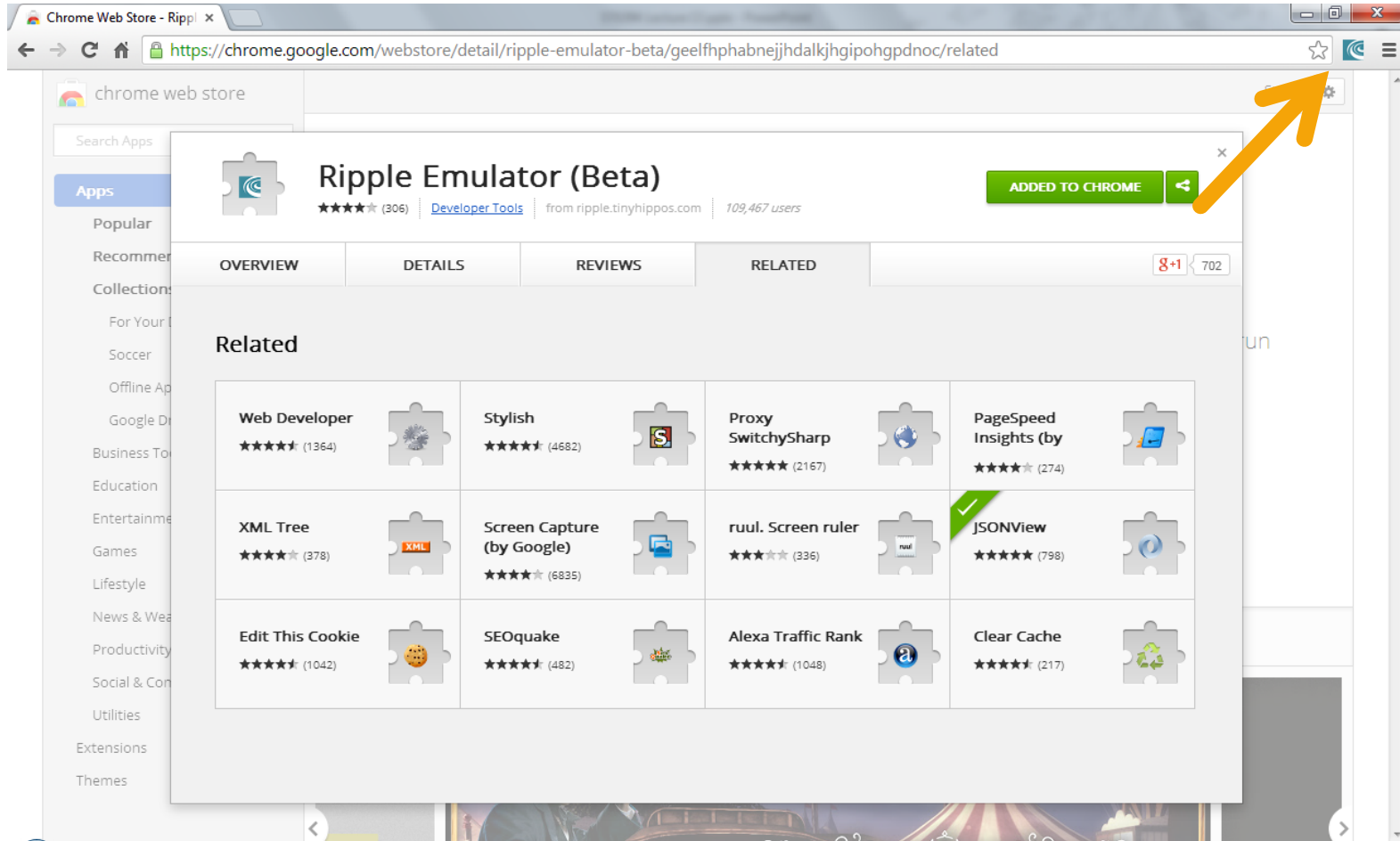
certificate (p12) file Choose File No file chosen

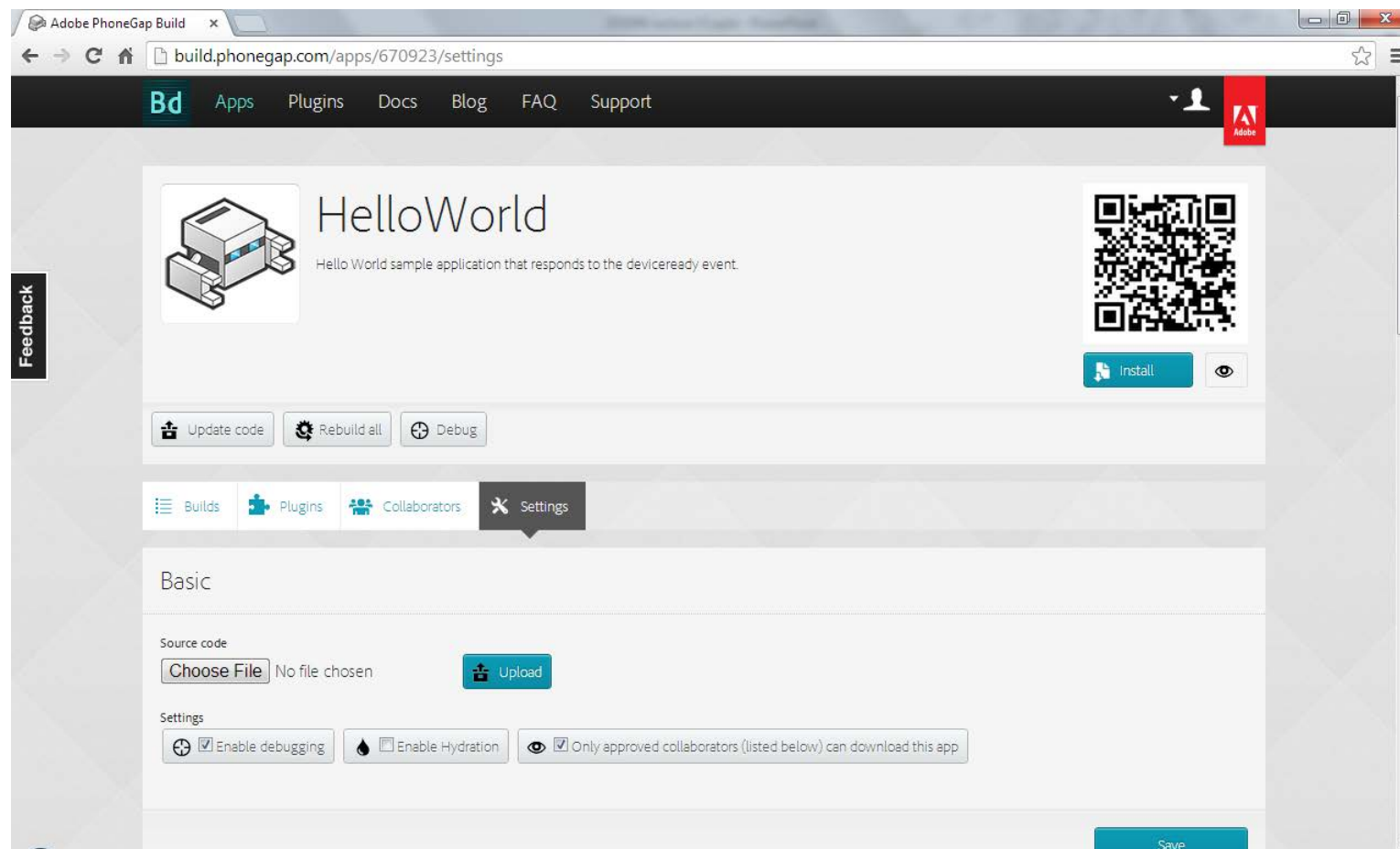
provisioning profile Choose File No file chosen

Powered by Adobe® PhoneGap™

Copyright © 2013 Adobe Systems Incorporated. All rights reserved.








Adobe PhoneGap Build


build.phonegap.com/apps

Bd Apps Plugins Docs Blog FAQ Support

Your apps + new app


 **HelloWorld**
Hello World sample application that responds to the deviceready event.

App ID 670923	Version 1.0.0	PhoneGap 2.9.0
Owned by peterleongkw@gmail.com		
Last built (9) 1 minute		
Source zip package		



iOS Android Windows BlackBerry HP i

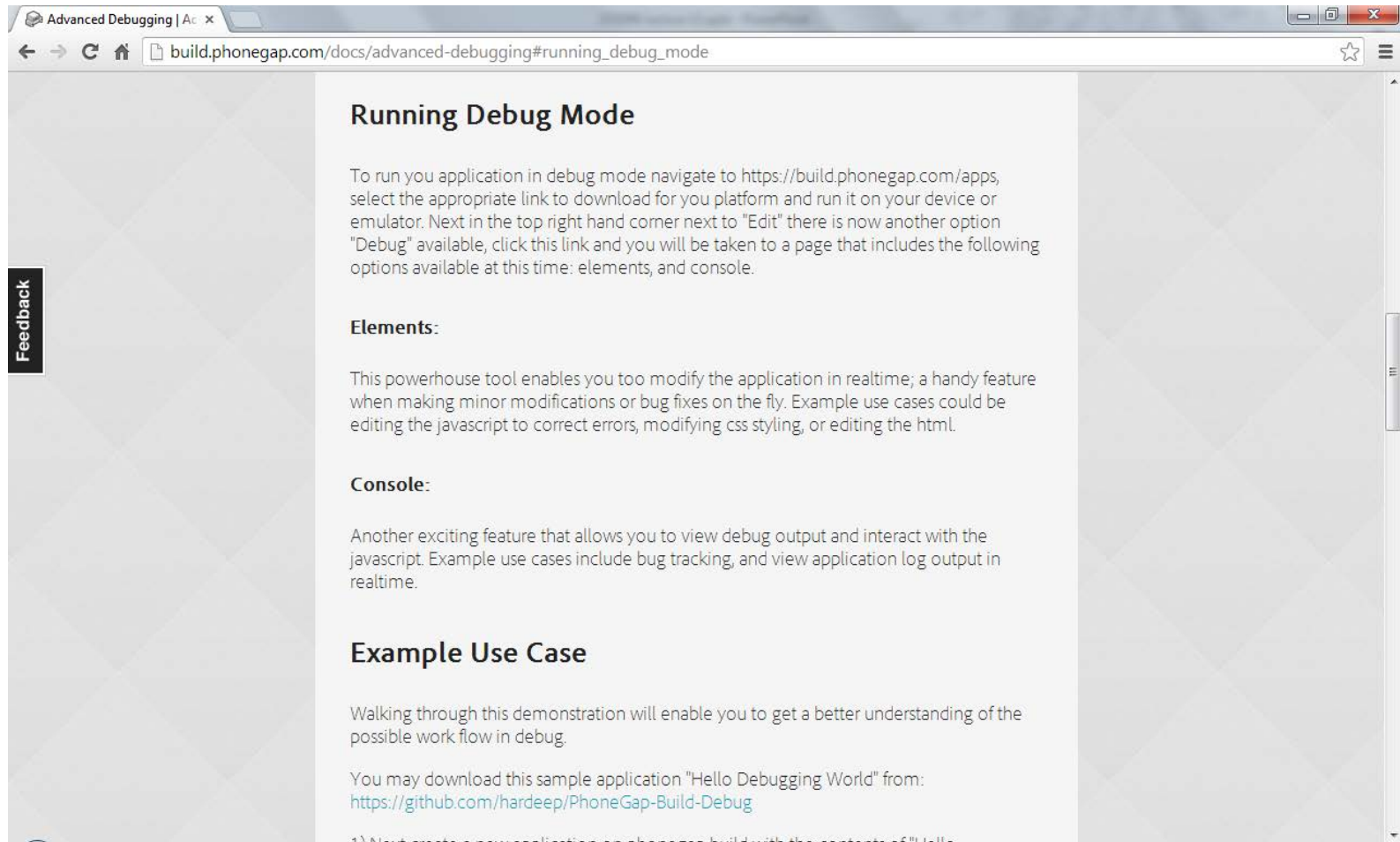
Update code Rebuild all Debug

 **Part of Adobe Creative Cloud**
Your PhoneGap Build plan includes a Creative Cloud free membership. Sign in with an AdobeID to receive 2GB of storage access to Adobe Edge Tools & Services and more. [Get started](#)

Language: English

Powered by Adobe® PhoneGap™

Copyright © 2013 Adobe Systems Incorporated. All rights reserved.
PhoneGap is a trademark of Adobe. You may not use PHONEGAP or GAP in the title of any application built using PhoneGap or the



Advanced Debugging | Ac x

build.phonegap.com/docs/advanced-debugging#running_debug_mode

Running Debug Mode

To run your application in debug mode navigate to <https://build.phonegap.com/apps>, select the appropriate link to download for your platform and run it on your device or emulator. Next in the top right hand corner next to "Edit" there is now another option "Debug" available, click this link and you will be taken to a page that includes the following options available at this time: elements, and console.

Elements:

This powerhouse tool enables you too modify the application in realtime; a handy feature when making minor modifications or bug fixes on the fly. Example use cases could be editing the javascript to correct errors, modifying css styling, or editing the html.

Console:

Another exciting feature that allows you to view debug output and interact with the javascript. Example use cases include bug tracking, and view application log output in realtime.

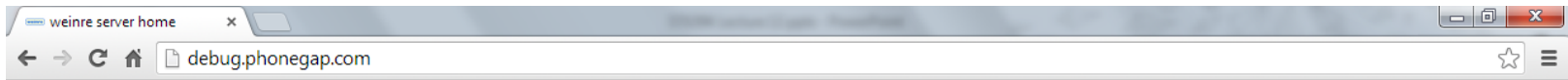
Example Use Case

Walking through this demonstration will enable you to get a better understanding of the possible work flow in debug.

You may download this sample application "Hello Debugging World" from:
<https://github.com/hardeep/PhoneGap-Build-Debug>

1) Next create a new application on phonegap build with the contents of "Hello

Feedback



Getting Started



Step 1: select your guid (or just use the one we've randomly assigned)

Step 2: inject this script into your index.html

```
<script src="http://debug.phonegap.com/target/target-script-min.js#anonymous"></script>
```

Step 3: click this link to start debugging

<http://debug.phonegap.com/client/#anonymous>

powered by 

GA for mobile

ADDING THE GOOGLE
ANALYTICS ACCOUNT
TO TEST APP

Create new account (Admin tab)

The screenshot shows the Google Analytics Admin interface for the 'My Outdoor Store' account. The interface is divided into three main sections: ACCOUNT, PROPERTY, and VIEW.

ACCOUNT Section:

- Account dropdown: My Outdoor Store
- Search bar: Search
- Account list:

Account Name	Account ID
My Outdoor Store	49343288
peterleong	9585691
Render.sg	38224952
sandbox	49337198
SP DMIT	22603848

[Create new account](#)

[Change History](#)

PROPERTY Section:

- Property dropdown: My Outdoor Store - mobile data
- Property list:

Property Name	Property ID
Property Settings	
User Management	
Tracking Info	
Remarketing	
Custom Definitions	
Data Import	
Social Settings	

VIEW Section:

- View dropdown: mobile app (master view)
- View list:

View Name	View ID
View Settings	
User Management	
Goals	
Content Grouping	
Filters	
Channel Grouping	

PERSONAL TOOLS & ASSETS

- Segments
- Annotations
- Custom Alerts

Choose mobile app type

Google Analytics

Home Reporting Customization **Admin**

peterleongkw77@gmail.com
My Outdoor Store - mobile data - -
mobile app (master view)

New Account

What would you like to track?

Website Mobile app

Setting up your account

Account Name required
Accounts are the top-most level of organization and contain one or more tracking IDs.

My GAPLugin Test App

Setting up your property

App Name

GAPLugin Test App

Already tracking an app with Google Analytics? You might not need this step!

An existing tracking ID can be reused in multiple app versions, app editions, and across platforms. In some cases, one tracking ID can also be used in multiple apps. Using an existing or new tracking ID affects how data appears in your reports.

Review the [Best Practices for Mobile App Analytics set up](#) to find out if you should get a new tracking ID or use an existing tracking ID.

Industry Category ?

Select One

Reporting Time Zone

United States (GMT-08:00) Pacific Time

Get the tracking id

The screenshot shows the Google Analytics Admin interface in a web browser. The browser's address bar displays the URL: <https://www.google.com/analytics/web/?hl=en#management/Settings/a49886282w81747018p84635548/%3Fm.page%3DTrackingCode/>. The Google Analytics logo is in the top left, and the user's email, peterleongkw77@gmail.com, is in the top right. The main navigation bar includes links for Home, Reporting, Customization, and Admin (which is highlighted). Below this, the page title is 'Administration My GAPLugin Test App / GAPLugin Test App'. On the left, a sidebar lists various settings: PROPERTY (selected), Property Settings, User Management, Tracking Info (expanded), Tracking Code (highlighted in red), Session Settings, Remarketing, Custom Definitions, Data Import, and Social Settings. The main content area shows the 'Tracking ID' as 'UA-49886282-1' and the title 'Mobile App tracking - Download and use the SDK'. It includes a brief explanation of SDKs and provides download links for both Android and iOS, each with a 'Download', 'Download with admob features', and 'Guide' button.

Google Analytics

Home Reporting Customization Admin

peterleongkw77@gmail.com
My Outdoor Store - mobile data -
mobile app (master view)

Administration
My GAPLugin Test App / GAPLugin Test App

PROPERTY
GAPLugin Test App

Property Settings

User Management

Tracking Info

Tracking Code

Session Settings

Remarketing

Custom Definitions

Data Import

Social Settings

Tracking ID
UA-49886282-1

Mobile App tracking - Download and use the SDK

Here are the SDKs available for your app.
You can download the appropriate SDK using the links below, or read about steps for integrating the SDK into your app through our guides.

Google Analytics Android SDK

Download Download with admob features Guide

Google Analytics iOS SDK

Download Download with admob features Guide

Create the master, test, unfiltered views

The screenshot displays the Google Analytics Admin interface for a property named 'My GAPLugin Test App'. The interface is organized into three main columns: ACCOUNT, PROPERTY, and VIEW.

- ACCOUNT Column:** Contains a dropdown menu for 'My GAPLugin Test App' and a list of account-level settings: Account Settings, User Management, AdWords Linking, AdSense Linking, All Filters, and Change History.
- PROPERTY Column:** Contains a dropdown menu for 'GAPLugin Test App' and a list of property-level settings: Property Settings, User Management, Tracking Info, Remarketing, Custom Definitions, Data Import, and Social Settings.
- VIEW Column:** Contains a dropdown menu for 'All Mobile App Data'. The dropdown menu is open, showing a search bar, the selected view 'All Mobile App Data', and a 'Create new view' button. Below the dropdown, there is a list of view-level settings: Goals, Content Grouping, Filters, Channel Grouping, Segments, Annotations, and Custom Alerts.

The top navigation bar includes the Google Analytics logo, the 'Admin' tab, and the user profile 'peterleongkw77@gmail.com' with the property name 'GAPLugin Test App' and the selected view 'All Mobile App Data'.

Google Analytics

Home Reporting Customization Admin

peterleongkw77@gmail.com
GAPlugin Test App --
GAPlugin Test (Test)

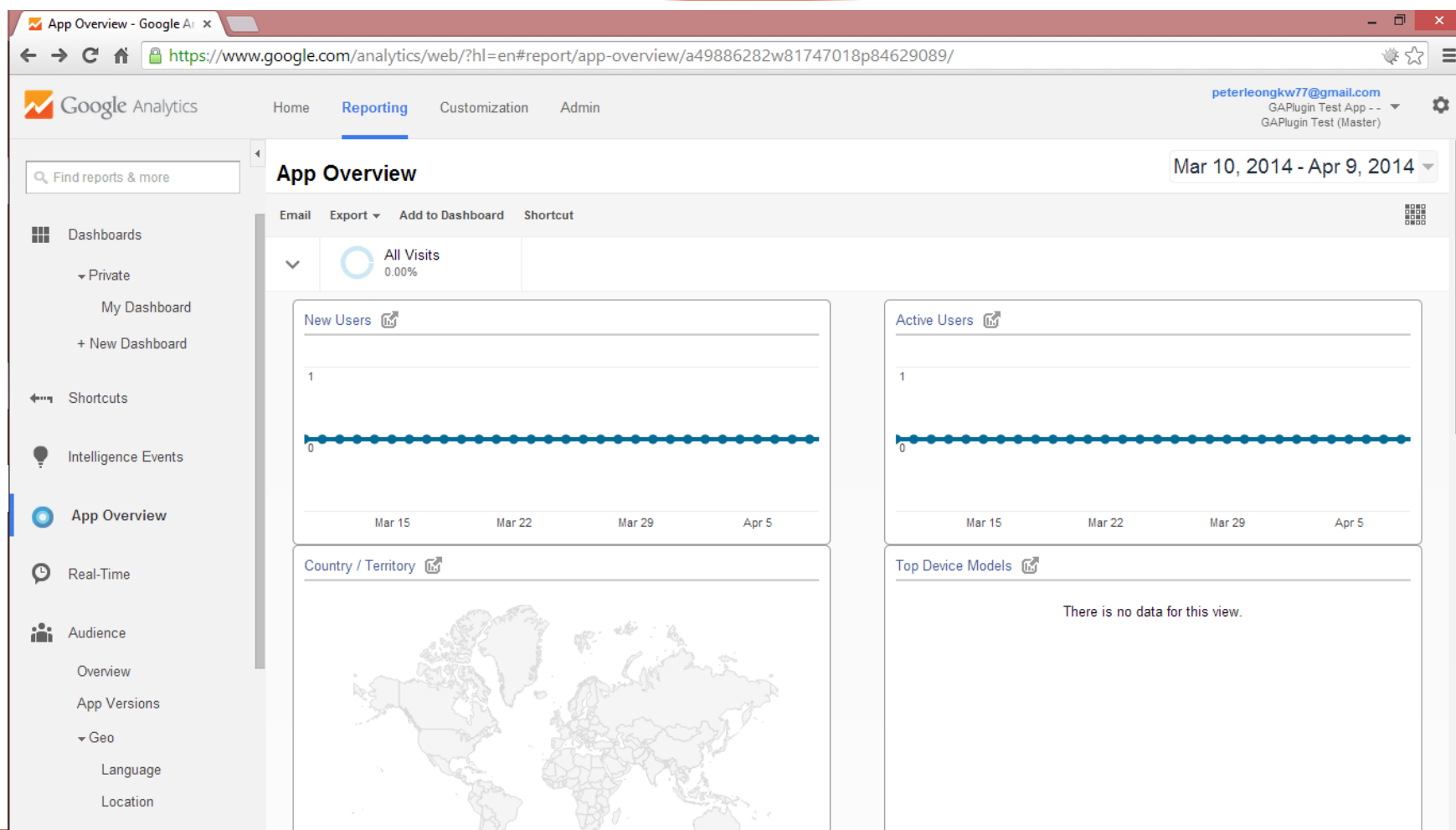
Mar 10, 2014 - Apr 9, 2014

Mode Show All

	Visits	Avg. Visit Duration	Bounce Rate	Goal Conversion Rate
My GAPlugin Test App				
GAPlugin Test App (UA-49886282-1)				
All Mobile App Data	-	-	-	-
GAPlugin Test (Master)	-	-	-	-
GAPlugin Test (Test)	-	-	-	-
GAPlugin Test (Unfiltered)	-	-	-	-
My Outdoor Store				
peterleong				
Render.sg				
sandbox				
SP DMIT				

This table was generated on 4/10/14 at 5:24:39 PM - Refresh Table

Check the App Overview for the master view



PhoneGap GA Plug-in

ADDING THE PLUG-IN

PhoneGap GA Plug-in

phonegap-build/GAPugin x


GitHub, Inc. [US] <https://github.com/phonegap-build/GAPugin>

GitHub

This repository Search or type a command

Explore Features Enterprise Blog

Sign up Sign in

PUBLIC  **phonegap-build / GAPugin**
forked from bobeast/GAPugin

★ Star 116 Fork 118

Google Analytics Plugin

46 commits 3 branches 7 releases 9 contributors

branch: master GAPugin

This branch is 0 commits ahead and 0 commits behind master Pull Request Compare

bump version

Bob Easterday authored 5 months ago latest commit 340201342d

Example	Update plugin to support GoogleAnalytics SDK v2	11 months ago
spec	put back file I deleted for no discernable reason	10 months ago
src	Fixes for Cordova 3.0	6 months ago
www	Merge pull request #37 from mkuklis/master	5 months ago
.gitignore	Add initial test spec	11 months ago
MIT-LICENSE	Initial Commit	a year ago
README.md	Update readme with better install instructions	6 months ago
plugin.xml	bump version	5 months ago

README.md

Code

Issues 35

Pull Requests 15

Pulse

Graphs

Network

HTTPS clone URL

<https://github.com>

You can clone with HTTPS or Subversion.

Clone in Desktop

Download ZIP

GAPugin

Google Analytics plugin for iOS and Android. This allows you to post usage information to your Google Analytics account.

Preparation:

Before you can begin collecting metrics data, you need to set up a GoogleAnalytics Mobile App account so you can view them. When you do so, you will obtain an app tracking id which we'll use during session initialization. Start by going to the [Google Analytics](#) site and click on the **Create an Account** button. Once signed in, click on the **Admin** button and the **+ New Account** button under the **Accounts** tab. At the top of the resulting tab, select the **App** button in answer to the **What would you like to track?** query. Fill out the form as appropriate. Complete instructions can be found [here](#).

Installation:

local

Add the following feature tag in your config.xml

```
<feature name="GAPugin" >  
  < param name="android-package" value="com.adobe.plugins.GAPugin"/>  
</feature>
```


Installation:

local

Add the following feature tag in your config.xml

```
<feature name="GAPugin" >
  < param name="android-package" value="com.adobe.plugins.GAPugin"/>
</feature>
```

This plugin is based on [plugman](#). to install it to your app, simply execute plugman as follows;

```
plugman install --platform [PLATFORM] --project [TARGET-PATH] --plugin [PLUGIN-PATH]
```

where

[PLATFORM] = ios or android

[TARGET-PATH] = path to folder containing your xcode project

[PLUGIN-PATH] = path to folder containing this plugin

For additional info, take a look at the [Plugman Documentation](#)

PhoneGap Build

To use this plugin with PhoneGap Build, add the following plugin reference to your config.xml

```
<gap:plugin name="com.adobe.plugins.gapugin" />
```

Usage

The plugin creates the object `window.plugins.gaPlugin`

After `onDeviceReady`, create a local var and startup the plugin like so;

```
var gaPlugin;

function onDeviceReady() {
    gaPlugin = window.plugins.gaPlugin;
    gaPlugin.init(successHandler, errorHandler, "UA-12345678-1", 10);
}
```

To get things rolling you need to call `init()` when your device ready function fires. `Init` takes 4 arguments; 1) success - a function that will be called on success 2) fail - a function that will be called on error. 3) id - Your Google Analytics account ID of the form; UA-XXXXXXXX-X This is the account ID you were given when you signed up. 4) period - An integer containing the minimum number of seconds between upload of metrics. When metrics are logged, they are enqueued and are sent out in batches based on this value. You'll want to avoid setting this value too low, to limit the overhead of sending data.

Example:

```
gaPlugin.init(successHandler, errorHandler, "UA-12345678-1", 10);
```

To track an event, call (oddly enough) `trackEvent()`. `trackEvent` takes 6 arguments;

- 1) `resultHandler` - a function that will be called on success
- 2) `errorHandler` - a function that will be called on error.
- 3) `category` - This is the type of event you are sending such as "Button", "Menu", etc.
- 4) `eventAction` - This is the type of event you are sending such as "Click", "Select". etc.
- 5) `eventLabel` - A label that describes the event such as Button title or Menu Item name.
- 6) `eventValue` - An application defined integer value that can mean whatever you want it to mean.

Example:

```
gaPlugin.trackEvent( nativePluginResultHandler, nativePluginErrorHandler, "Button", "Click", "event only", 1);
```

`TrackEvent` covers most of what you need, but there may be cases where you want to pass arbitrary data. `setVariable()` lets you pass values by index (Up to 20, on free accounts). For free accounts, each variable is given an index from 1 - 20. Reusing an existing index simply overwrites the previous value. Passing an index out of range fails silently, with no data sent. The variables will be sent ONLY for the next `trackEvent` or `trackPage`, after which those indexes will be available for reuse. `setVariable()` accepts 4 arguments;

- 1) `resultHandler` - a function that will be called on success
- 2) `errorHandler` - a function that will be called on error.
- 3) `index` - the numerical index representing one of your variable slots (1-20).
- 4) `value` - Arbitrary string data associated with the index.

Example:

```
gaPlugin.setVariable( nativePluginResultHandler, nativePluginErrorHandler, 1, "Purple");
```

Important:

Variable values are assigned to what Google calls Custom Dimensions in the dashboard. Prior to calling `setVariable()` in your client for a particular index, you need to create a slot in the GA dashboard. When you do so, you will be able to assign a name for the dimension, its index, and its scope. More info on creating Custom Dimensions can be found [here](#). The next event or page view you send after `setVariable` will contain your variable at Custom Dimension specified by the index value you passed in the `setVariable` call. This [Example](#) app shows how you might use that next event to specify a label for the variable you just set.

In addition to events and variables, you can also log page visits with `trackPage()`. Unlike variables, however, page hits do not require a subsequent call to `trackEvent()` as they are considered unique events in and of themselves. `trackPage()` takes 3 arguments;

- 1) `resultHandler` - a function that will be called on success
- 2) `errorHandler` - a function that will be called on error.
- 3) `url` - The url of the page hit you are logging.

Example:

```
gaPlugin.trackPage( nativePluginResultHandler, nativePluginErrorHandler, "some.url.com");
```

Finally, when your app shuts down, you'll want to cleanup after yourself by calling `exit()`; `exit()` accepts 2 arguments;

- 1) `resultHandler` - a function that will be called on success
- 2) `errorHandler` - a function that will be called on error.

Example:

```
gaPlugin.exit(nativePluginResultHandler, nativePluginErrorHandler);
```

Testing

TESTING THAT THE DATA
FEED TO GOOGLE
ANALYTICS IS ACTIVE