Send Push Notification via GCM

Purpose:

This document purpose is to make the readers to understand, how to send push notification using Google Cloud Messaging for an Android PhoneGap application

Step 1

* Create a project in <https://console.developers.google.com/home>
* Will get a sender ID or Project Id and Server API key

Step 2

* Download the following plugin <https://github.com/phonegap/phonegap-plugin-push> from <https://github.com/>
* Copy the src file from plugin to your root folder
* Copy the **push.js** notification from downloaded plugin and paste to the **app/www/**

Step 3

* Do the following configuration to your **config.xml**  file

|  |
| --- |
| <preference name="android-build-tool" value="gradle" />  <plugin name="phonegap-plugin-push" source="npm" />  <feature name="Notification">  <param name="android-package" value="org.apache.cordova.dialogs.Notification" />  </feature> |

* Do the following configuration to your **AndroidManifest.xml** file

|  |
| --- |
| <!-- For Push Notification GCM Access -->  <uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />  <uses-permission android:name="android.permission.GET\_ACCOUNTS" />  <uses-permission android:name="android.permission.WAKE\_LOCK" />  <uses-permission android:name="com.google.android.c2dm.permission.RECEIVE" />  <permission android:name="$PACKAGE\_NAME.permission.C2D\_MESSAGE" android:protectionLevel="signature" />  <uses-permission android:name="$PACKAGE\_NAME.permission.C2D\_MESSAGE" />  <!-- activity, receiver and service tags to the application section. -->  <activity android:name="com.plugin.gcm.PushHandlerActivity"/>  <receiver android:name="com.plugin.gcm.CordovaGCMBroadcastReceiver" android:permission="com.google.android.c2dm.permission.SEND" >  <intent-filter>  <action android:name="com.google.android.c2dm.intent.RECEIVE" />  <action android:name="com.google.android.c2dm.intent.REGISTRATION" />  <category android:name="$PACKAGE\_NAME" />  </intent-filter>  </receiver>  <service android:name="com.plugin.gcm.GCMIntentService" />  <activity ... android:launchMode="singleTop"> |

Step 4

* Include the **cordova.js, push.js** to the **pushNotification.html** file
* Write the following javascript code inside the **onDeviceReady**  function

|  |
| --- |
| function onDeviceReady() {  var push = PushNotification.init({  android: {  senderID: "864987576725" //Sender Id or Project ID  },  ios: {  alert: "true",  badge: "true",  sound: "true"  },  windows: {}  });  push.on('registration', function(data) { alert(data.registrationId);  localStorage.setItem("registrationId",data.registrationId);  });    }  document.addEventListener('deviceready', onDeviceReady, true); |

What is registration ID?

* Registration ID is a unique identifier of each device that is registered with GCM.
* It has 256 characters

Still now we have registered our mobile to the GCM and now we have the registration id. So far it is a good progress. And now we have to send the notification from the server.

Step 5

* For Example PHP XAMPP server

|  |
| --- |
| $to = c4gqeT8Hxac:APA91bHjBPwAceW0Z08zfwGpEnvG2qGAGcPq6bFqMTrtUZ1cxH7qj\_r7h9ts9kJ1bPWJMlZnDIma2uRp-KIgRiiIybMaaDBli9JdhUme6ouyS-CObvskbV-jGeJhVl6kwEY\_XgSJIzIa; //registration ID of the device    $title = "From:Sathish”;  $message = “Hai How are you dude !!! ?”;  // API access key from Google API's Console  // replace API  //define( 'API\_ACCESS\_KEY', 'AIzaSyCtDCehTVdcYJQVQa\_tJ6cuUnYI\_6uWXKc');  if (!defined('API\_ACCESS\_KEY')) define('API\_ACCESS\_KEY', 'AIzaSyCtDCehTVdcYJQVQa\_tJ6cuUnYI\_6uWXKc');  $registrationIds = array($to);  $msg = array  (  'message' => $message,  'title' => $title,  'vibrate' => 1,  'sound' => 'beep.wav'  // you can also add images, additionalData  );  $fields = array  (  'registration\_ids' => $registrationIds,  'data' => $msg  );  $headers = array  (  'Authorization: key=' . API\_ACCESS\_KEY,  'Content-Type: application/json'  );  $ch = curl\_init();  curl\_setopt( $ch,CURLOPT\_URL, 'https://gcm-http.googleapis.com/gcm/send' );  curl\_setopt( $ch,CURLOPT\_POST, true );  curl\_setopt( $ch,CURLOPT\_HTTPHEADER, $headers );  curl\_setopt( $ch,CURLOPT\_RETURNTRANSFER, true );  curl\_setopt( $ch,CURLOPT\_SSL\_VERIFYPEER, false );  curl\_setopt( $ch,CURLOPT\_POSTFIELDS, json\_encode( $fields ) );  $result = curl\_exec($ch );  curl\_close( $ch );  echo $result; |

* You would get the $result like the following if the message is success

|  |
| --- |
| {"multicast\_id":5355530895910547183,"success":1,"failure":0,"canonical\_ids":0,"results":[{"message\_id":"0:1454915047488700%5e98c713f9fd7ecd"}]} |

* Now you would have received the notification from the Google Cloud Messaging Server
* If the server API key is wrong then you may get the invalidRegistration ID error or miss matched API key
* Above is the just a notification with no action. You can add action buttons in the notification as per your need