



SEPLAN

Secretaría Técnica de
Planeación y Evaluación

SECRETARÍA TÉCNICA DE PLANEACIÓN Y EVALUACIÓN

Investigación y codificación de ejemplo con API's para interacción con redes sociales

ELABORADO POR:

Ing. Victor Geovanni Barbosa Reyes

REVISADO POR:

NOMBRE APELLIDO

MÉRIDA YUCATÁN, 18 DE OCTUBRE DE 2018

HOJA DE CONTROL DE DOCUMENTO

PROYECTO			
API's para interacción con redes sociales			
FECHA	REVISIÓN	ELABORACIÓN	NOTAS
18/10/2018	1.0	Ing. Victor Giovanni Barbosa Reyes	

TABLA DE CONTENIDO

1. INTRODUCCIÓN.....	1
2. DESARROLLO	1
2.1. API de Facebook	1
2.2. API de Twitter	4
2.3. API de Google.....	5
3. CONCLUSIONES.....	7
4. REFERENCIAS.....	8

1. INTRODUCCIÓN

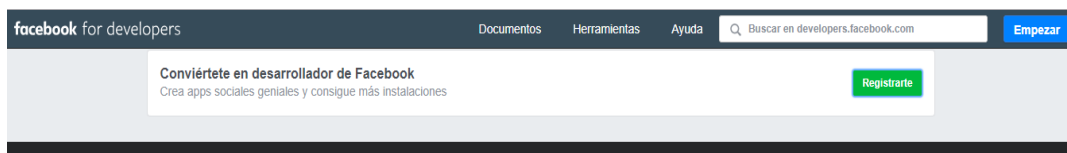
El uso de API's de las redes sociales ha crecido mucho desde que Google lanzó la primera versión de su API de Youtube en mayo de 2008, y desde que Facebook lanzó la primera versión de su API Graph en abril de 2010. Hoy, estas API te dan la oportunidad de buscar en las plataformas de redes sociales, así como de realizar inicios de sesión, siempre y cuando el usuario acepte los permisos para compartir su información.

2. DESARROLLO

El objetivo principal de implementar estas herramientas es proveer a los usuarios otro medio de acceso al sitio de “Consulta ciudadana”, y a su vez obtener la información básica de los mismos.

2.1. API de Facebook

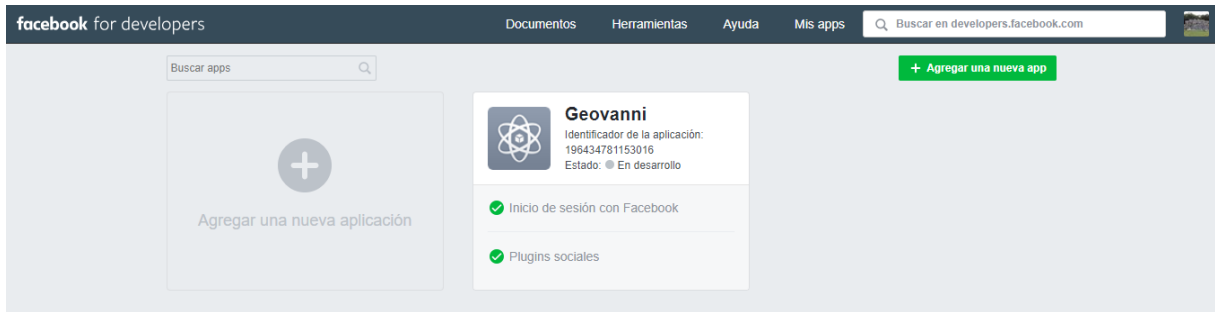
El inicio de sesión con Facebook puede ser implementado utilizando el SDK de Facebook o de manera manual. En este caso nos enfocaremos en el inicio con el SDK.



Antes de llevar a cabo la implementación se deberá generar un identificador de la aplicación ingresando en el siguiente enlace <https://developers.facebook.com/apps/>

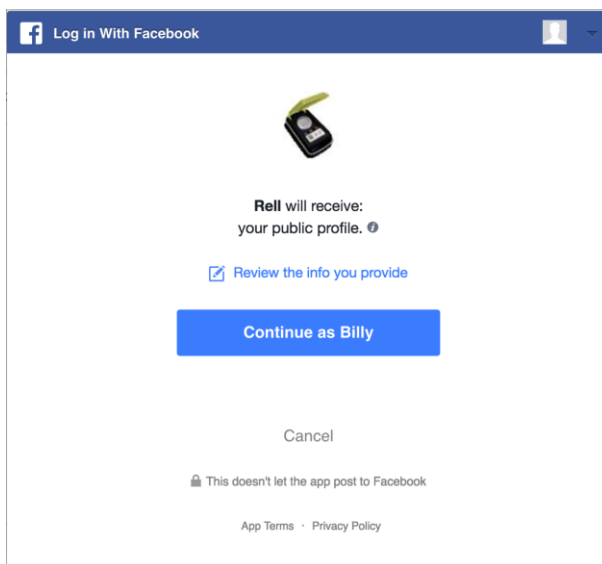


Después de la creación de la cuenta, se deberá generar una app para el inicio de sesión, la cual proporcionará un identificador que posteriormente será utilizado en la implementación del código javascript.



Una vez hecho todo lo anterior se procederá a la integración en el sitio. Lo primero será comprobar si el usuario ha iniciado sesión con anterioridad en la página con una función llamada `FB.getLoginStatus`, la cual devuelve un parámetro correspondiente al estado del usuario.

Si la persona no ha iniciado sesión se puede utilizar el cuadro de diálogo para pedirle que lo haga. De igual manera para cerrar sesión se usará la misma ventana.





Ejemplo de implementación de la API

```
4 <title>Facebook Login JavaScript Example</title>
5 <meta charset="UTF-8">
6 </head>
7 <body>
8 <script>
9 // This is called with the results from from FB.getLoginStatus().
10 function statusChangeCallback(response) {
11   console.log('statusChangeCallback');
12   console.log(response);
13
14   if (response.status === 'connected') {
15     testAPI();
16   } else {
17     document.getElementById('status').innerHTML = 'Please log ' +
18       'into this app.';
19   }
20 }
21
22 function checkLoginState() {
23   FB.getLoginStatus(function(response) {
24     statusChangeCallback(response);
25   });
26 }
27
28 window.fbAsyncInit = function() {
29   FB.init({
30     appId      : '196434781153016',
31     cookie     : true,  // enable cookies to allow the server to access
32                       // the session
33     xfbml      : true,  // parse social plugins on this page
34     version    : 'v2.8' // use graph api version 2.8
35   });
36
37   FB.getLoginStatus(function(response) {
38     statusChangeCallback(response);
39   });
40 };
41
42 // Load the SDK asynchronously
43 (function(d, s, id) {
44   var js, fjs = d.getElementsByTagName(s)[0];
45   if (d.getElementById(id)) return;
46   js = d.createElement(s); js.id = id;
47   js.src = "https://connect.facebook.net/en_US/sdk.js";
48   fjs.parentNode.insertBefore(js, fjs);
49 }(document, 'script', 'facebook-jssdk'));
50
51 // Here we run a very simple test of the Graph API after login is
52 // successful. See statusChangeCallback() for when this call is made.
53 function testAPI() {
54   console.log('Welcome! Fetching your information.... ');
55   FB.api('/me', function(response) {
56     console.log('Successful login for: ' + response.name + 'First name: ' + response.first_name);
57     document.getElementById('status').innerHTML =
58       'Thanks for logging in, ' + response.name + '!';
59   });
60 }
61 </script>
62
63 <!--
64 Below we include the Login Button social plugin. This button uses
65 the JavaScript SDK to present a graphical Login button that triggers
66 the FB.login() function when clicked.
67 -->
68
69 <fb:login-button scope="public_profile,email" onlogin="checkLoginState();">
70 </fb:login-button>
71
72
73
74
75
```


2.2. API de Twitter

Cuenta pendiente de aprobación

 Developer Use cases Products Docs More

consulta ciudadana 

Apps



Application under review

Your developer account application needs to be approved before you can create apps. Once it is approved, you can create an app here and get an API key in order to authenticate and integrate with most Twitter developer products.

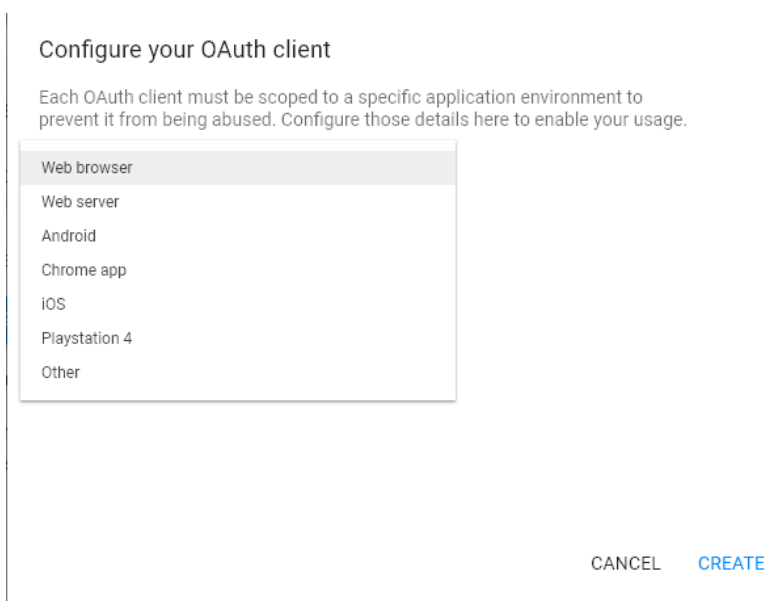
2.3. API de Google

Google+ da la opción a los desarrolladores de utilizar su API para que los usuarios puedan acceder a los sitios mediante el inicio de sesión con su cuenta.

El primer paso para integrar el inicio de sesión de google en el sitio es crear un ID cliente, el cual es necesario para acceder a esa función.



Se debe seleccionar la herramienta que se usará para acceder al inicio de sesión y la URL del sitio al que se redireccionará.



Una vez creado el ID se deberá hacer lo siguiente:

- Referenciar la librería de Google para integrar el módulo de inicio de sesión.
- Especificar el ID del cliente.
- Añadir el botón de inicio de sesión.

Cuando el usuario ha iniciado sesión con su cuenta de Google+ se puede recuperar su información básica (ID, nombre, URL de perfil y correo) mediante el método `getBasicProfile()`.

```
function onSignIn(googleUser) {  
  var profile = googleUser.getBasicProfile();  
  console.log('ID: ' + profile.getId()); // Do not send to your backend! Use an ID token instead.  
  console.log('Name: ' + profile.getName());  
  console.log('Image URL: ' + profile.getImageUrl());  
  console.log('Email: ' + profile.getEmail()); // This is null if the 'email' scope is not present.  
}
```

También se puede habilitar la opción de que el usuario pueda cerrar sesión sin necesidad de usar la ventana de cierre de Google, creando una función que haga referencia al método `GoogleAuth.signOut()` mediante un evento “onclick” en algún botón o una liga.

```
<a href="#" onclick="signOut();">Sign out</a>  
<script>  
  function signOut() {  
    var auth2 = gapi.auth2.getAuthInstance();  
    auth2.signOut().then(function () {  
      console.log('User signed out.');    });  
  }  
</script>
```

3. CONCLUSIONES

Realizando un análisis de la implementación de las tres API's en el sitio nos encontramos con que la más sencilla es la de Google+, ya que únicamente se requiere generar el ID, referenciar la librería en la página y crear el botón de inicio de sesión.

En el caso de Facebook (que se requieren unos pasos más que en la de Google+) se realizaron algunas pruebas con los códigos proporcionados en la documentación. Se logró implementar un inicio básico de sesión obteniendo únicamente el nombre de la cuenta del usuario. La librería ofrece códigos para la obtención de más información básica (esta información no requiere de una revisión de la aplicación por parte del equipo de Facebook Developer) pero aún no se ha tenido éxito en su implementación.

En cuanto a la investigación con twitter no se pudo avanzar mucho puesto que requiere aplicar para obtener un ID de desarrollador. Ya se hizo el llenado del formulario pero estamos a la espera de una respuesta del equipo encargado de la validación.

4. REFERENCIAS

API Facebook

<https://developers.facebook.com/docs/facebook-login/web>

<https://developers.facebook.com/apps/>

<https://developers.facebook.com/docs/facebook-login/manually-build-a-login-flow>

API Twitter

<https://developer.twitter.com/en/docs/basics/developer-portal/guides/apps>

<https://developer.twitter.com/en/docs/twitter-for-websites/log-in-with-twitter/guides/implementing-sign-in-with-twitter>

API Google+

https://www.youtube.com/watch?time_continue=2&v=Oy5F9h5JqEU

<https://developers.google.com/identity/sign-in/web/sign-in>

<https://developers.google.com/identity/sign-in/web/people>