

Crear una aplicación con Facebook Graph Api, en modo IFrame | SEO Mallorca

Y después del post sobre [como crear una aplicación con la nueva Facebook Graph api en modo FBML](#) le toca al turno a la aplicación del modo IFrame, que sin duda poco a poco va teniendo más sentido dado que Facebook va a terminar por eliminar la posibilidad de crear aplicaciones FBML en su sistema. La verdad es que este post lleva escrito un tiempo pero por problemas entre mi Facebook y mi hosting no funcionaba bien el ejemplo. Espero que a día de hoy esto esté resuelto y pido disculpas si os provoca problemas el ejemplo al visitarlo.

El modo IFrame en las aplicaciones de Facebook se caracteriza porque la aplicación pasa a estar realmente en tu propia página web y Facebook simplemente muestra un IFrame (un marco) en su sistema con el que cargar tu web. Facebook, con sus últimos cambios pasa a recomendar este sistema para que los desarrolladores se pongan las pilas y creen sus aplicaciones al mismo tiempo como aplicaciones de Facebook y como webs que usan Facebook como base. Así pues, esta pequeña explicación irá en este sentido: la posibilidad de crear una aplicación que valga a la vez para ser cargada en Facebook o en nuestra propia web.

Antes de seguir dejar también claro que voy a centrarme en mostrar la parte de servidor de la aplicación. Las buenas aplicaciones IFrame suelen valerse de un SDK de servidor y al mismo tiempo incorporan el SDK javascript para las acciones en la parte de cliente: etiquetas FBML, acciones de login, solicitudes, etc. Como solo vamos a tratar con la parte del servidor nos veremos forzados a no usar etiquetas FBML y tratar el login de usuario sin ventanas modales ni popUps.

Si quieres aprender un poco sobre el SDK javascript de Facebook puedes consultar este mismo blog para ver un [ejemplo de uso del SDK javascript de Facebook](#).

1. Configurando nuestra aplicación en el panel de Facebook

Empezamos como siempre (puedes ver los pasos para crear tu aplicación en el [tutorial de Facebook](#) de este blog): Con un usuario de Facebook creamos nuestra nueva aplicación dándonos de alta en la aplicación de desarrolladores. A partir de ahí configuramos nuestra aplicación como sigue:

Ahora pasamos a descargar el SDK de servidor que queramos usar para nuestro desarrollo. La lista de SDK's disponibles la podemos encontrar en [este enlace](#). Para los ejemplos de este artículo usamos la [facebook PHP SDK](#). Nos la bajamos y la incluimos -con un "include()" - en el flujo de todas las páginas que vaya a ir cargando la aplicación.

A partir de aquí ya podemos empezar a escribir el código PHP de nuestra aplicación. Este lo subiremos a la carpeta que hemos indicado como "canvas URL" y como "Site URL" que es donde Facebook va a ir a buscar la interacción con nuestra aplicación.

2. Creando el objeto \$facebook

Para crear nuestro objeto facebook, basta con incluir el facebook PHP SDK al principio de nuestro código y llamar a la clase "Facebook". A esta le pasaremos 3 parametros en un array numérico:

- appId, es el id numérico que le da Facebook a nuestra aplicación.
- secret, es el código secreto (que no debe ser publicado) que nos permite identificar las conexiones.
- cookie, es un valor con el que indicar si se aceptan sesiones por cookie o no. Por defecto es mejor marcarlo a "true" para que una vez cargada la sesión de usuario esta se maneje por cookies.

Así pues nuestro código quedará así:

3. Creando la sesión de usuario en Facebook con Facebook Graph Api y el PHP SDK

Una vez tenemos la clase iniciada tenemos que identificar al usuario actual en ella para poder rescatar e interactuar con sus datos. Con la sesión iniciada la clase de Facebook podrá hacer las consultas a Facebook incluyendo un código especial -llamado token- en las peticiones. Este código es la base de la conexión con Facebook -llamada OAuth2.0- y sin él nuestro desarrollo no accederá a ningún dato del usuario.

Empezamos por una declaración muy simple...

Esto informará en "\$fbSession" de si la sesión ha sido iniciada o no. Sin embargo, esto no crea la sesión por si sola. Al estar trabajando en nuestra propia página la sesión tiene que venir dada o bien por URL o bien por cookies. La forma más corriente de conseguir esta sesión es mediante el SDK Javascript que lanza directamente un popUp de login, sin embargo como hemos dicho que vamos a intentar prescindir de este, lo que haremos será redirigir al usuario hacia el sistema de Login de Facebook para que este nos lo devuelva con las variables de sesión.

Aquí vamos a tener que diferenciar 2 formas distintas de hacerlo puesto que estamos creando una aplicación que funciona al mismo tiempo dentro y fuera de Facebook la forma de dirigir al usuario no será la misma. Así que antes de nada deberemos detectar en nuestro sistema si el usuario está viendo nuestra aplicación a través de Facebook o directamente en nuestra web. Un ejemplo de como hacerlo sería mediante el Rererrer:

Otra forma sería examinando si facebook nos ha enviado la variable "fb_sig_in_iframe", que envía siempre (queramos o no) cuando carga a través de iframe una app:

3.1 Cargando la sesión de usuario desde dentro de Facebook

Solucionado esto, pasemos a ver como logar al usuario cuando este lanza nuestra aplicación. El SDK de PHP nos permite usar el método “getLoginUrl” para crear urls de login a las que dirigir al usuario. Este recibe varios posibles parámetros a través de un array asociativo. Veamos cuales necesitamos.

- canvas y fbconnect: nos indica si la forma de cargar la aplicación.
- next: nos indica la url a la que volverá el usuario cuando termine de identificarse
- cancel_url: En el caso de pedir permisos al usuario y este negarlos, esta es la url a la que se devolverá al usuario cuando termine de identificarse
- req_perms: Nos permite pedir permisos especiales al usuario (lo veremos más tarde)

Una vez tengamos la url a la que dirigir al usuario, con javascript le redirigiremos a esta directamente. El usuario ya está en facebook y acaba de pedir entrar en una aplicación con lo que no le sorprenderá que directamente se le identifique.

3.2 Cargando la sesión de usuario desde el acceso directo a nuestra web

Este caso es bastante parecido, pero con algunos pequeños cambios:

- Para empezar la url de vuelta de “next” y “cancel_url” será la de nuestro servidor en vez de la de la app, puesto que el usuario quería entrar en nuestro dominio y por tanto no tiene mucho sentido que acabe dentro de facebook.
- Aparte de esto, si bien es posible crear un redirect (php o javascript) al usuario directamente esto también podría asustarle... Imagina que estás en una web y al entrar en una sección de golpe te pide permisos de Facebook, da miedo, ¿no crees? Por eso creo que es mucho mejor mostrar al usuario una página donde se le explique que va a ser conectado con Facebook y al final de esta un link para conectarse.

4. Pidiendo permisos al usuario con Facebook Graph Api y el PHP SDK

Los permisos son accesos a información o acciones que los usuarios dan a nuestras aplicaciones. Por defecto no podemos hacer mucho más que consultar el nombre del usuario y algunos de sus datos. Pero podemos preguntarle a este si nos permite hacer otras cosas con su cuenta. Estos permisos van desde acciones simples como permitir a la aplicación ver las fotos del usuario hasta controlar totalmente la cuenta del usuario en cualquier momento desde dentro y fuera de Facebook. Es preciso examinar que permisos vamos a necesitar para que nuestra aplicación haga lo que nosotros deseamos y pedirle al usuario tan solo estos permisos... de otra forma podemos perder demasiados usuarios en el proceso.

Los distintos tipos de permisos, así como las acciones que nos permite hacer los usuarios al aceptarlos quedan descritos en [esta página](#).

Como hemos visto al identificar al usuario podemos usar la variable "req_perms" para que se soliciten los permisos al usuario en ese mismo momento. Nada tan simple como añadir separados por comas los distintos permisos que necesitamos y Facebook se encargará de solicitarlos -pero solo si no los tenemos ya-.

El problema con los permisos no viene tanto por pedirlos como por controlar si los tenemos antes de hacer nada que no nos esté prohibido. Pero eso lo veremos dentro de unos pocos párrafos...

5. Solicitando datos a la Facebook Graph Api con el PHP SDK

La nueva Facebook Graph Api ha simplificado mucho la forma de hacer peticiones. Básicamente tenemos que ir formando los nodos sobre los que queremos ir accediendo en forma de url. Con esta uri montada llamaremos al método "api()" que nos devolverá un array con la información solicitada. La

[documentación](#) de la Graph api es bastante clara con lo que no deberías tener serios problemas para entenderla. Por ejemplo:

A partir de esta información yo puedo ir anidando una petición tras otra hasta encontrar lo que busco. Primero podría sacar mi lista de amigos y de ahí, con su id, solicitar cualquiera de sus datos (siempre que tenga permiso para acceder al él).

En el tutorial para [crear aplicaciones de Facebook en modo FBML](#), lo que hicimos fue sacar la información del usuario y las fotos de sus amigos. En este ejemplo, puesto que como no hemos usado el SDK Javascript no disponemos de FBML simplemente sacaremos el nombre del usuario y el de sus amigos (nada de fotos), si quieres ver el mismo ejemplo con FBML mira dicho tutorial...

6. Creando un mensaje en el muro del usuario que usa nuestra aplicación

Ahora veremos como publicar contenido en el muro de los usuarios. Pero antes recordemos el tema de las excepciones: No se puede publicar en el muro de los usuarios si estos no nos han dado permiso para hacerlo. Por este motivo antes de hacer este tipo de acciones es necesario comprobar si nuestra aplicación tiene los permisos necesarios para hacerlo.

Desgraciadamente, la parte de Facebook Graph api no nos deja consultar los permisos de un usuario por lo que la forma más cómoda que tenemos para hacerlo en cualquier momento es usando la API FQL. Esta es una parte de la api que nos permite consultar datos más rebuscados a modo de sentencias SQL. Por suerte para nosotros el SDK PHP también nos permite lanzar este tipo de consultas. Para ello solo tenemos que lanzar la llamada al método `api()` como un array en el que le indicamos el método "fql.query" y la query que queremos lanzar. Veamos un ejemplo de como solicitar el nombre del usuario de esta forma:

Ligeramente más difícil, ¿verdad? Es por esto que yo

recomendaría usar para la mayor parte de los datos la Graph Api y solo cuando se necesiten datos secundarios, del sistema o con relaciones complejas usar FQL.

Podéis ver un detalle de todos los tipos de datos accesibles con FQL -y comprobar que son muchos más que con la Graph API- en [este enlace](#).

Volviendo al tema que nos ocupa, los permisos, veamos como rescatarlos con FQL:

Ahora sí! Ya estamos preparados para intentar publicar en el muro del usuario con Graph Api. Para hacer la publicación usaremos también el método “api()” solo que en este caso en vez de incluirle una sola variable tendremos que pasarle 3:

Veamos como quedaría entonces la comprobación de permisos junto con la publicación en el muro del usuario:

Y se acabó

Con esto ya puedes empezar a dar forma a tus aplicaciones IFrame en Facebook. Si quieres ver la aplicación de este tutorial puedes hacerlo visitando...

Recuerda que en realidad son la misma aplicación ☐

Nota: los servidores de OVH (donde está alojado este blog) están teniendo serios problemas con las llamadas a la api de facebook 😞 , por ese motivo es posible que de vez en cuando aparezca un error al cargar la aplicación. En caso de que suceda, por favor, recarga la página un par de veces hasta que desaparezca.

El código

También puedes copiar todo el código de la aplicación para crear la tuya propia a partir de aquí:

Comparte este artículo:

[Twttear](#)

This content was originally published [here](#).