



Escola d'Enginyeria de Telecomunicació i
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TRABAJO DE FINAL DE CARRERA

TÍTULO DEL TFC: Estudio del desarrollo de juegos online como aplicaciones Facebook

TITULACIÓN: Ingeniería Técnica de Telecomunicaciones, especialidad en Telemática

AUTOR: Sara Díaz Segovia

DIRECTOR: Sergio Machado Sánchez

FECHA: 17 de diciembre de 2012

Título: Estudio del desarrollo de juegos online como aplicaciones Facebook

Autor: Sara Díaz Segovia

Director: Sergio Machado Sánchez

Fecha: 17 de diciembre de 2012

Resumen

A día de hoy, Facebook se ha introducido en casi todos los aspectos de la vida cotidiana y no tan cotidiana de cualquier persona con acceso a la red de redes.

Además, se ha convertido en una herramienta de valoración de imagen de cada usuario de la red social, usada sobre todo por empresas, ya que la imagen que da un trabajador en su vida privada puede ser asociada a la empresa por el público en general.

Incluso, las aplicaciones se han convertido en un punto importante de esta plataforma ya que, en la actualidad no sólo se desarrollan aplicaciones con la finalidad de entretener a las personas sino que las empresas utilizan estas aplicaciones para llegar a más público y ofrecer sus servicios.

El objetivo principal de este trabajo es ver cómo se integra una aplicación en Facebook, así como todos los datos que se pueden manejar en cuanto al usuario se refiere y de cómo la información referente al uso de la aplicación se puede controlar.

Title: Development's Research of online games as Facebook applications

Author: Sara Díaz Segovia

Director: Sergio Machado Sánchez

Date: December, 17th 2012

Overview

Nowadays, Facebook has become a part of almost all day-to-day life's sides, and not so daily, of any person with access to the Internet.

Besides, it has turned into an image assessment tool of each user in the social network, used specially by the companies, so that the image given by a worker in its private life can be associated to the company mostly by the public.

Even the applications have become an important point of this platform since, nowadays, not only applications in order to entertain people are developed but companies use these applications to reach a wider audience and offer their services.

The main purpose of this study is seeing how an application is integrated in Facebook, as well as the data that can be managed as regard as the user and how the information referred to the use of the application can be controlled.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1. VISIÓN GENERAL	2
1.1. Desarrollo en Facebook.....	2
1.2. Tecnologías usadas	3
1.2.1. Apache.....	4
1.2.2. MySQL.....	4
1.2.3. PHP	4
1.2.4. Javascript.....	5
1.2.5. CSS	5
CAPITULO 2. DESARROLLO DE APLICACIONES EN FACEBOOK.....	6
2.1. Tipos de integración	6
2.1.1. Integración en páginas web.....	6
2.1.2. Aplicaciones para móvil.....	6
2.1.3. Aplicaciones en Facebook.....	6
2.2. Integración de Aplicaciones en Facebook.....	7
2.3 Qué se necesita para crear una aplicación en Facebook	8
2.4. API de Facebook.....	10
2.4.1. Información sobre la aplicación	11
2.4.2. Información que se puede obtener del usuario que usa la aplicación	12
CAPÍTULO 3. APLICACIÓN SNAKE PARA FACEBOOK.....	16
3.1. Aplicación base: ¡Hola Mundo!	16
3.3 Guardar los datos del usuario: Uso de la Base de Datos	20
3.4 Publicar en el muro del usuario	23
3.5. Enviar invitaciones a amigos	24
3.6. Juego SNAKE.....	26
3.7. Ranking.....	29
3.8. Achievements (trofeos).....	30
3.9. Aplicación completa.....	31
CAPITULO 4. CONCLUSIONES	33
REFERENCIAS.....	34

Índice de figuras

- Fig 1.1** Tecnologías usadas y su relación en la red.
- Fig 2.1** Separación de la aplicación en iframes
- Fig 2.2** Cuadro de diálogo para crear aplicación.
- Fig 2.3** Información básica de la aplicación.
- Fig 2.4** Información de la aplicación.
- Fig 3.1** Cuadro de dialogo con los permisos que pide la aplicación
- Fig 3.2** Resultado de la aplicación.
- Fig 3.3** Permisos extras añadidos a la configuración.
- Fig 3.4** Cuadro de diálogo para autorizar los datos que usará la aplicación.
- Fig 3.5** Datos del usuario mostrados al iniciar la aplicación.
- Fig 3.6** Creación de la base de datos.
- Fig 3.7** Datos del usuario almacenados en la base de datos.
- Fig 3.8** Comprobación de usuario ya registrado en base de datos.
- Fig 3.9** Introducción satisfactoria de los datos en la tabla.
- Fig 3.10** Introducción del campo para poder publicar en nombre del usuario.
- Fig 3.11** Post publicado en el muro del usuario que usa la aplicación por primera vez.
- Fig. 3.12** Cuadro de diálogo con los amigos a los que poder invitar para usar la aplicación.
- Fig. 3.13** Solicitud de uso de la aplicación en la página del usuario al que se ha enviado la invitación.
- Fig 3.14** Procesos que se siguen en el juego Snake.
- Fig 3.15** Área del juego.
- Fig 3.16** Ubicación de la comida y la serpiente dentro del área de juego.
- Fig 3.17** Puntuaciones guardadas en la Base de datos.
- Fig 3.18** Diagrama de la aplicación completa.

Índice de tablas

Tabla 2.1 Campos que se pueden obtener del objeto *application*.

Tabla 2.2 Información básica que se obtiene del usuario.

Tabla 2.3 Información adicional del usuario obtenida con permisos.

Tabla 3.1 Campos que contiene la tabla *users* de la base de datos.

Tabla 3.2 Campos que contiene la tabla *ranking*.

INTRODUCCIÓN

Objetivos y motivación

En la actualidad el concepto de aplicación y entretenimiento están presentes en el día a día de la sociedad. Un gran número de dispositivos electrónicos, como teléfonos móviles, tabletas electrónicas, reproductores de música ofrecen la posibilidad de poder utilizar aplicaciones.

Por eso, la motivación de este trabajo es desarrollar un juego típico, como es el *snake*, dentro de una de las plataformas más utilizadas a día de hoy, como es la red social Facebook.

Esta red social ha sustituido las relaciones sociales tradicionales, de manera que tus gustos, preferencias, ideologías quedan reflejadas en tu perfil de usuario, facilitando tanto a amigos como a empresas a tener una visión sobre tu persona.

Además del propio juego, se pretende añadir parte de la funcionalidad que la propia plataforma proporciona mediante la API de Facebook. Por ejemplo, obtener los datos del usuario, interactuar con los amigos pudiendo enviar invitaciones, ver las puntuaciones obtenidas del resto de usuarios, etc.

Estructura del TFC

Este trabajo está compuesto por esta introducción, cuatro capítulos y bibliografía. A continuación se detalla una breve descripción de lo contenido en cada uno de los capítulos.

Capítulo 1: Visión general

En este capítulo se explica brevemente en qué consiste el desarrollo en general en la plataforma Facebook y las tecnologías que se usan para el desarrollo de la aplicación (Apache, MySQL, PHP, Javascript y CSS).

Capítulo 2: Desarrollo de aplicaciones en Facebook

Este capítulo se centra en diferenciar las diferentes formas de integrar aplicaciones (en tu propia web, en Facebook o en móviles).

Capítulo 3: Aplicación Snake para Facebook

Este capítulo consta de todas las fases por las que se ha pasado a la hora de desarrollar la aplicación final. Estas fases incluyen: mostrar los datos del usuario, invitar a amigos a usar la aplicación, publicar en el muro, guardar datos en la base de datos, obtención del ranking y el propio juego *snake*. También aparece una futura ampliación de la aplicación (Achievements).

Capítulo 4: Conclusiones

Se exponen las principales conclusiones a las que se ha llegado después de la realización del trabajo y las dificultades encontradas. Además, incluye un breve estudio de ambientalización del trabajo.

CAPÍTULO 1. VISIÓN GENERAL

1.1. Desarrollo en Facebook

Facebook cuenta, a día de hoy, con más de mil millones de usuarios. A parte de una red social, Facebook es también una plataforma para el desarrollo de juegos y aplicaciones que amplían la experiencia de usuario dentro de la red social. Facebook permite crear aplicaciones bajo la propia interfaz del portal a la vez que ofrece las herramientas necesarias para poder llevar a cabo su objetivo e interactuar con la comunidad.

La expansión de esta red social ha llegado a tal nivel que, incluso cualquier tipo de empresa puede ofrecer servicios mediante sus propias aplicaciones integradas en la Facebook.

Si se accede al centro de aplicaciones que cada usuario dispone en su menú, se puede obtener una lista con todos los tipos en los que Facebook clasifica las aplicaciones:

- **Libros:** aplicaciones para encontrar, compartir, valorar y comprar libros.
- **Negocio:** aplicaciones para ayudar a las personas a encontrar o compartir puestos de trabajo, aprender sobre profesiones, etc.
- **Comunicación:** aplicaciones que permiten a las personas interactuar y tener conversaciones con otras personas por chat, video, actualizaciones de estado, etc.
- **Educación:** aplicaciones que ayudan a las personas a aprender nuevas habilidades a través de cursos, comprando libros de texto, encontrando tutores, etc.
- **Ocio:** aplicaciones para encontrar espectáculos, valorar películas, etc.
- **Fashion:** aplicaciones para compartir ropa, tendencias y estilos.
- **Finanzas:** aplicaciones para aprender acerca del dinero, del gasto y de las inversiones.
- **Comer y beber:** aplicaciones para compartir recetas, descubrir nuevos restaurantes, anotar donde has comido, etc.
- **Juegos:** aplicaciones de juegos estructurados. Los juegos suelen ser interactivos, con objetivos, reglas y desafíos.
- **Salud y Fitness:** aplicaciones que ayudan a las personas con su salud, bienestar o fitness.
- **Estilo de vida:** aplicaciones relacionadas con tendencias de estilo de vida y hábitos.
- **Local:** aplicaciones que permiten encontrar lugares cerca del usuario.
- **Música:** aplicaciones para escuchar música, descubrir nuevas bandas, crear listas de reproducción, etc.
- **Navegación:** aplicaciones que ayudan a llegar adonde se quiere ir, proporcionando mapas, direcciones, etc.

- **Noticias:** aplicaciones para leer noticias, compartir historias importantes, etc.
- **Aplicaciones para Páginas:** aplicaciones que ayudan a las empresas a gestionar o mejorar sus páginas de Facebook.
- **Foto:** aplicaciones para visualizar, compartir y editar fotografías y obras de arte.
- **Productividad:** aplicaciones para organizar y administrar el tiempo de las personas.
- **Referencia:** aplicaciones que proporcionan información completa o complementaria sobre una variedad de temas.
- **Compras:** aplicaciones para comprar ropa, artículos para el hogar, joyería, etc.
- **Deportes:** aplicaciones para ver, seguir y aprender acerca de los deportes.
- **Viajar:** aplicaciones para descubrir lugares de todo el mundo.
- **Utilidades:** aplicaciones útiles que hacen que las actividades cotidianas, como el seguimiento de la vida de la batería o la búsqueda en la web, más fácil.
- **Video:** aplicaciones para crear, buscar y ver vídeos, trailers, clips cortos, etc.
- **Tiempo:** aplicaciones para el seguimiento y la planificación de todo el tiempo.

1.2. Tecnologías usadas

Dentro de todas las posibilidades para el desarrollo de aplicaciones web, se ha optado por la arquitectura AMP (Apache, MySQL y PHP), cuyas tecnologías se detallan a continuación.

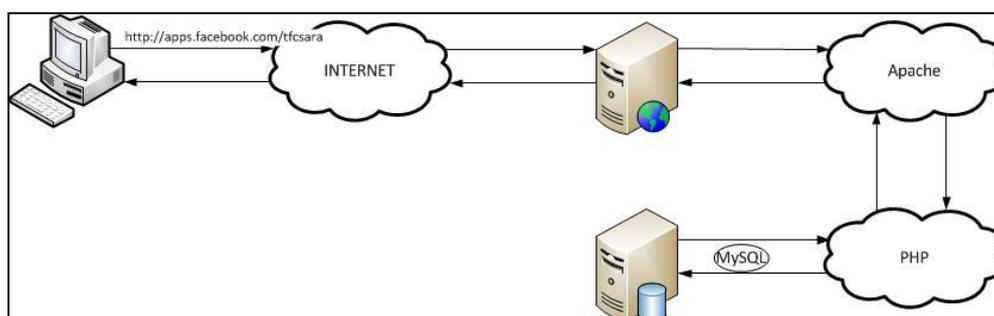


Fig 1.1 Tecnologías usadas y su relación en la red.

Como característica importante destacable de utilizar esta arquitectura es que los elementos que la componen son *open source*, esto es, programas que son de código abierto y por lo general, gratuitos.

Para completar la funcionalidad de las aplicaciones y más concretamente en este caso, se han utilizado, además, los lenguajes Javascript y CSS.

1.2.1. Apache

El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Apache se usa, principalmente, para enviar páginas web estáticas y dinámicas en la World Wide Web (WWW).

Apache se usa, principalmente, para enviar páginas web estáticas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web.

1.2.2. MySQL

MySQL (My Structure Query Language) es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.

MySQL es muy utilizado en aplicaciones web, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

1.2.3. PHP

PHP (*Hypertext Pre-processor*) es un lenguaje de programación interpretado (Lenguaje de alto rendimiento), diseñado originalmente para la creación de páginas web dinámicas. Se usa, principalmente, para la interpretación del lado del servidor (*server-side scripting*) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

1.2.4. Javascript

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (*client-side*), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, aunque existe una forma de JavaScript del lado del servidor (*Server-side JavaScript* o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

1.2.5. CSS

El nombre hojas de estilo en cascada viene del inglés *Cascading Style Sheets*, del que toma sus siglas. CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML2 (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

La información de estilo se puede adjuntar como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "<style>".

CAPITULO 2. DESARROLLO DE APLICACIONES EN FACEBOOK

2.1. Tipos de integración

Facebook, a través de su plataforma, permite a los desarrolladores poder crear aplicaciones dentro de su propio sistema ya sea a través de un ordenador personal como de un dispositivo móvil o proporcionar diferentes funcionalidades propias de la infraestructura de Facebook para aquellos usuarios que dispongan de su propia página web.

2.1.1. Integración en páginas web

Cualquier página web tiene la opción de utilizar utilidades propias de Facebook de forma que permiten que el sitio web sea más social.

Al incluir las funcionalidades que requiera cada caso en particular se obtienen diferentes ventajas que se detallan a continuación:

- Crear un servicio más personalizado para los usuarios que navegan por la página web.
- Tener experiencia social a través de los *plug-in's* sociales. Un ejemplo de ello es el botón "Me gusta".
- Simplificar el registro y proceso de inicio de sesión de los usuarios utilizando el botón "Inicio de sesión" y el "*plug-in* de Registro".

2.1.2. Aplicaciones para móvil

La Plataforma de Facebook permite crear aplicaciones para diferentes dispositivos móviles, incluso los que usan como sistema operativo iOS (iPhone y iPad) o Android. Además, se puede implementar cualquier tipo de aplicación web para móvil usando tecnologías web como HTML5, Javascript o CSS.

2.1.3. Aplicaciones en Facebook

Implementar una aplicación en Facebook da la oportunidad de integrarse profundamente en la experiencia de usuario básico. Usando las funcionalidades nativas de Facebook, como *Solicitudes* y *Marcadores* se puede crear un lugar social ideal para los usuarios.

2.2. Integración de Aplicaciones en Facebook

Al integrar una aplicación en Facebook, se pueden diferenciar dos partes tal y como se muestra en la figura 2.1:



Fig 2.1 Separación de la aplicación en iframes.

La zona bordeada de amarillo es la que corresponde al contenedor de Facebook. Es decir, es toda aquella información que carga la plataforma Facebook. Se puede observar diferentes tipos de información como juegos que pueden interesar al usuario, páginas recomendadas (que tienen relación con los intereses de los amigos del usuario, etc.) o aspectos de la página que el usuario puede valorar, así como los menús típicos comunes que aparecen en parte superior.

La zona bordeada de negro corresponde al iframe propio de la aplicación, es decir, son los datos que se cargan desde un servidor web externo donde está alojada la aplicación que se está ejecutando.

Un *iframe* (por *inline frame* o *marco incorporado*) es un elemento HTML que permite insertar o incrustar un documento HTML dentro de un documento HTML principal.

En este caso, se trata de un contenido web que se ejecuta en un servidor. Durante todo el desarrollo se utilizará un servidor local, es decir, un servidor que se ejecuta en la propia máquina donde también se ejecuta el proceso navegador (Chrome, Firefox, Explorer, etc.).

2.3 Qué se necesita para crear una aplicación en Facebook

Para poder desarrollar una aplicación en Facebook es necesario seguir el circuito establecido por la plataforma. Los pasos previos a la codificación que se deben seguir son los que se detallan a continuación:

- Dar de alta como desarrollador al usuario que creará la aplicación

En este paso se requiere la validación de la cuenta de usuario de Facebook. Existen varias formas de hacerla:

- Mediante el número de teléfono móvil: Se debe introducir el número de teléfono para poder recibir un código de confirmación con el fin de validar la cuenta.

Para el ámbito español, sólo es efectivo si el número de teléfono corresponde al Operador Movistar.

- Mediante tarjeta de crédito.

- Dar de alta la aplicación.

Lo primero que se debe hacer es ir a la página web de alta de aplicaciones en Facebook. La URL se muestra a continuación:

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

En la página, además de poder crear y editar aplicaciones, se puede encontrar de toda la documentación necesaria para poder llevar a cabo una aplicación.

Una vez se ha accedido a la URL especificada anteriormente y se ha pulsado en el botón de “Crear nueva aplicación”, aparecerá el cuadro de diálogo que se muestra en la Fig. 2.2 donde se debe introducir la nomenclatura de la aplicación en Facebook. A continuación, se muestra la figura con dicha información:



Fig 2.2 Cuadro de diálogo para crear aplicación.

Como dato obligatorio, hay que introducir el nombre de la aplicación (*App Name*) y de forma opcional (ya que se puede modificar a posteriori), el espacio de nombres de la aplicación, que se comentará más adelante.

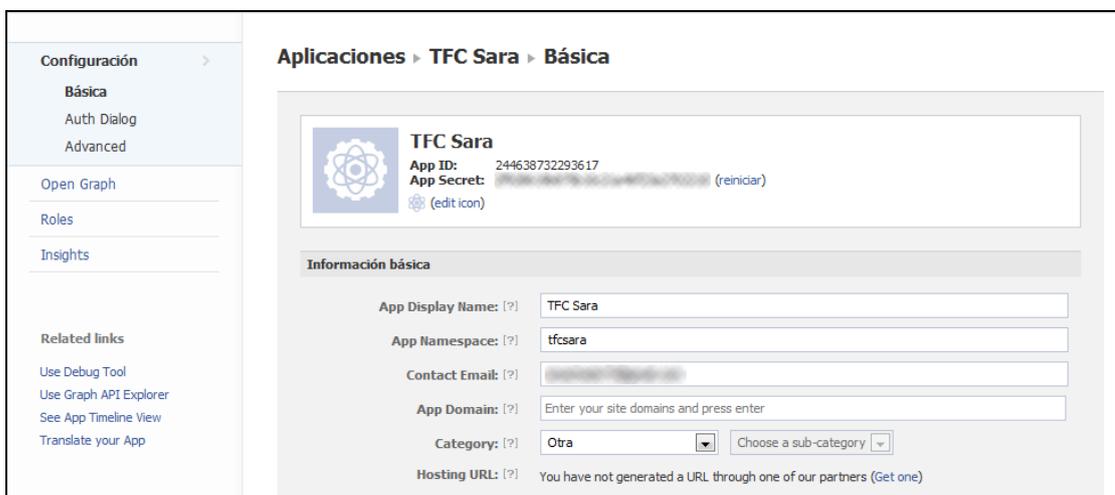
Además, si se quiere alojar la aplicación en un servidor web con un de los grupos colaboradores de Facebook, hay que marcar la casilla de “*Web Hosting*”.

Al pulsar en Continuar, aparecerá la página de configuración de la aplicación. En este paso, hay que configurar diferentes aspectos para poder visualizar la aplicación. Además, se generan dos claves: la *App ID/API Key* y el *Código secreto de la aplicación*, que servirán para poder acceder a la aplicación y se deberán añadir en el código fuente de la misma.

En este punto, se completarán junto con los datos de nomenclatura, el resto de información básica de la aplicación. Por ejemplo:

- El nombre para mostrar de la aplicación.
- El espacio de nombres de la aplicación
- El correo de la persona de contacto.
- La categoría de la aplicación.

En la figura 2.3, se muestran los datos anteriormente citados.



The screenshot displays the 'Basic' configuration page for a Facebook application named 'TFC Sara'. The interface includes a left-hand navigation menu with options like 'Configuración', 'Básica', 'Auth Dialog', 'Advanced', 'Open Graph', 'Roles', 'Insights', and 'Related links'. The main content area shows the application's name, logo, and generated credentials: App ID (244638732293617) and App Secret (with a 'reiniciar' link). Below this, the 'Información básica' section contains several input fields: 'App Display Name' (set to 'TFC Sara'), 'App Namespace' (set to 'tfc Sara'), 'Contact Email' (with a masked email address), 'App Domain' (with a placeholder 'Enter your site domains and press enter'), 'Category' (set to 'Otra'), and 'Hosting URL' (with a note that no URL has been generated).

Fig 2.3 Información básica de la aplicación.

A continuación hay que seleccionar el tipo de aplicación que se va a crear. En este caso, tal y como se ha comentado en el apartado 2.2.3, se elaborará una aplicación en Facebook, por lo tanto, la selección será “*App on Facebook*”.

Una vez se esté en la selección correcta, se tienen que especificar algunos aspectos correspondientes a la ubicación de la aplicación, es decir, desde donde se va leer el código que contiene dicha aplicación.

En este caso, dado que se trabajará desde un servidor local, la URL donde se ejecuta la aplicación es la de nuestra propia máquina, la dirección de *localhost*, *localhost* o <http://127.0.0.1> seguido del nombre del fichero inicial de la aplicación. Toda la información contenida en este fichero es lo que se cargará en el *canvas* de la aplicación. Esta información se debe especificar en el recuadro de *Canvas URL*.

Además, hay que especificar la URL segura (https) de la página principal de la aplicación ya que es un campo obligatorio.

Select how your app integrates with Facebook	
<input checked="" type="checkbox"/> App on Facebook	
Canvas URL: [?]	<input type="text" value="http://127.0.0.1:8080/holamundo.php/"/>
Secure Canvas URL: [?]	<input type="text" value="https://127.0.0.1:8080/"/>
Canvas Page: [?]	<input type="text" value="http://apps.facebook.com/tfcsara"/>
<input checked="" type="checkbox"/> Website	Log in to my website using Facebook.
<input checked="" type="checkbox"/> Mobile Web	Bookmark my web app on Facebook mobile.
<input checked="" type="checkbox"/> Native iOS App	Publish from my iOS app to Facebook.
<input checked="" type="checkbox"/> Native Android App	Publish from my Android app to Facebook.
<input checked="" type="checkbox"/> Pestaña de la página	Build a custom tab for Facebook Pages.
<input type="button" value="Guardar cambios"/>	

Fig 2.4 Información de la aplicación.

Como se puede observar en la figura 2.4, al editar la URL, aparece la dirección real de la aplicación en Facebook: *El Canvas Page*. Esta dirección contiene el espacio de nombres de la aplicación descrita anteriormente.

2.4. API de Facebook

Una API (Interfaz de Programación de Aplicaciones) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

La API de Facebook, por lo tanto, incluye todos los métodos que permitirán desarrollar las aplicaciones.

2.4.1. Información sobre la aplicación

La API de Facebook permite recuperar cualquier información o datos acerca de la aplicación. No obstante, alguna de esta información se ha tenido que especificar en la configuración de la misma.

En el objeto **application** quedan reflejados estos campos que se muestran en la Tabla 2.1.

Tabla 2.1 Campos que se pueden obtener del objeto *application*.

Nombre	Descripción
app_id	El identificador de la aplicación.
api_key	La clave API de la aplicación.
namespace	El nombre que sigue a apps.facebook.com/XXXX.
display_name	El nombre de la aplicación.
icon_url	La URL que hace referencia a la imagen del icono de la aplicación.
logo_url	La URL que hace referencia a la imagen del logo de la aplicación.
company_name	El nombre de la compañía que ha creado la aplicación.
developers	Lista de usuarios de Facebook desarrolladores de la aplicación.
description	Descripción de la aplicación, proporcionada por el desarrollador.
daily_active_users	Número de usuarios que han usado la aplicación en el último día.
weekly_active_users	Número de usuarios que han usado la aplicación en los últimos siete días.
monthly_active_users	Número de usuarios que han usado la aplicación en los últimos treinta días.
category	La categoría de la aplicación.
subcategory	La subcategoría de la aplicación.
is_facebook_app	Si la aplicación está desarrollada por Facebook.
restriction_info	Devuelve restricciones demográficas para la aplicación.
app_domains	Dominios y subdominios que la aplicación puede usar.
auth_dialog_description	La descripción de la aplicación que aparece en el Auth Dialog.
auth_dialog_headline	Una línea de descripción que aparece en la cabecera del Auth Dialog.
auth_dialog_perms_explanation	Texto que explica por qué una aplicación necesita permisos adicionales.
canvas_fluid_height	Indica si la aplicación usa un valor de altura fluido o fijo para el Canvas.

canvas_fluid_width	Indica si la aplicación usa un valor de anchura fluido o fijo para el Canvas.
canvas_url	La URL no segura en la que se carga el contenido del canvas de la aplicación.
contact_email	Email para contactar con los desarrolladores.
created_time	Marca temporal de UNIX que indica cuando se creó la aplicación.
creator_uid	ID del usuario creador de la aplicación
iphone_app_store_id	ID de la aplicación en la iPhone App Store.
hosting_url	Sitio web creado con uno de los colaboradores de hosting.
mobile_web_url	URL a la que los usuarios de Móvil son redirigidos cuando usan la aplicación.
mobile_web_url	URL a la que los usuarios de móviles son dirigidos cuando usan la aplicación.
secure_canvas_url	URL segura desde la que el contenido del Canvas de la aplicación se carga.
user_support_email	E-mail de contacto principal para esta aplicación.

2.4.2. Información que se puede obtener del usuario que usa la aplicación

Dependiendo de los permisos de los que disponga la aplicación, se podrá obtener más o menos información acerca del usuario que añade la aplicación en cuestión.

Cuando un usuario autoriza a la aplicación y no se han solicitado permisos adicionales, esta tiene acceso únicamente a la información básica del usuario. Por defecto, esta información básica incluye algunas propiedades del objeto "User", y son los que se detallan en la Tabla 2.2.

Tabla 2.2 Información básica que se obtiene del usuario.

id	El identificador de Facebook del usuario
name	El nombre completo del usuario
first_name	El nombre del usuario
last_name	El apellido del usuario
gender	El género del usuario: <i>female</i> (femenino) o <i>male</i> (masculino)
link	La URL del perfil del usuario en Facebook
username	El nombre de usuario en Facebook

No obstante, cuantos más datos públicos haya hecho el usuario, de más información se podrá disponer.

Si se desea obtener más información acerca del usuario, es necesario explicitarlo en la configuración de los permisos de la aplicación, concretamente en la sección de “Extended Permissions”.

La tabla 2.3 tiene con toda la información que se puede obtener del usuario con los permisos correspondientes añadidos a la configuración.

Tabla 2.3 Información adicional del usuario obtenida con permisos.

languages	Las lenguas del usuario
third_party_id	Identificador anónimo pero único para el usuario
installed	Especifica si el usuario ha instalado la aplicación asociada al token de acceso a la aplicación que se usa para hacer la solicitud.
updated_time	Última vez que se ha actualizado el perfil del usuario; cambios en <i>languages</i> , <i>link</i> , <i>timezone</i> , <i>verified</i> , <i>interested_in</i> , <i>favorite_athletes</i> , <i>favorite_teams</i> , y <i>video_upload_limits</i> no se reflejan en este campo
verified	Estado de la verificación de la cuenta del usuario (true o false)
bio	La biografía del usuario
birthday	El cumpleaños del usuario
education	Lista de la historia de la educación del usuario
email	Dirección de correo de contacto garantizado por el usuario
hometown	Ciudad natal del usuario
interested_in	Los géneros en los que el usuario está interesado
location	Ciudad actual del usuario
political	Punto de vista político del usuario
quotes	Las citas favoritas del usuario
relationship_status	El estado de la relación del usuario: Single (Soltero) , In a relationship (En una relación), Engaged (Prometido), Married (Casado), It's complicated (Es complicado), In an open relationship (En una relación abierta), Widowed (Viudo), Separated (Separado), Divorced (Divorciado), In a civil unión (En una unión civil), In a domestic partnership (Pareja de hecho)

religi3n	La religi3n del usuario
significant_other	Otros datos significativos del usuario
video_upload_limits	El tama1o del archivo de video y la longitud del archive del video que el usuario puede subir
website	La URL del sitio web personal del usuario
work	Una lista del historial de trabajo del usuario
read_friendlist	Permite el acceso a cualquier lista de amigos que el usuario ha creado. Todos los usuarios tienen datos b1sicos y este permiso garantiza el acceso a las listas de amigos del usuario y s3lo se debe solicitar si la aplicaci3n usa las listas de amigos.
read_insights	Permite acceso de lectura de datos estad3sticos de p1ginas, aplicaciones y dominios que el usuario posee.
read_mailbox	Proporciona la habilidad de leer la Bandeja de Entrada de Facebook de un usuario.
read_requests	Proporciona acceso de lectura a las solicitudes de amistad del usuario.
read_stream	Proporciona acceso a todos los posts publicados en "3ltimas Noticias" y permite a la aplicaci3n realizar b3squedas en la secci3n "3ltimas Noticias" del usuario.
xmpp_login	Permite a las aplicaciones integradas con el Chat de Facebook la habilidad de identificar usuarios.
ads_management	Proporciona la habilidad de administrar anuncios y usar la API de Facebook Ads en nombre de un usuario.
create_event	Permite a tu aplicaci3n crear y modificar eventos en nombre del usuario.
manage_friendlists	Permite a tu aplicaci3n crear y editar la lista de amigos del usuario
manage_notifications	Permite a tu aplicaci3n leer notificaciones y marcarlas como le3das.
user_online_presence	Permite acceso a la presencia online/offline del usuario.
friends_online_presence	Proporciona acceso a la presencia online/offline de los amigos del usuario.
publish_checkins	Permite a tu aplicaci3n realizar confirmaciones en nombre del usuario.
publish_stream	Permite a tu aplicaci3n postear contenido, comentarios y "Me gusta" en cualquier publicaci3n del usuario o de los

	amigos del usuario. Con este permiso, se puede publicar contenido en la sección "Últimas Noticias" de un usuario en cualquier momento.
rsvp_event	Permite a tu aplicación RSVP en eventos en nombre del usuario.

CAPÍTULO 3. APLICACIÓN SNAKE PARA FACEBOOK

En este capítulo, además de explicar cómo se ha integrado el propio juego de snake en Facebook, se detallan las diferentes fases o aplicaciones intermedias por las que se han tenido que pasar para lograr una la aplicación final.

3.1. Aplicación base: ¡Hola Mundo!

La primera toma de contacto con las aplicaciones en Facebook ha sido crear una aplicación base con un mensaje simple.

Antes de poder ejecutar la aplicación, hay que indicar en la URL del lienzo, la ruta del archivo que contiene el código que mostrará por pantalla ¡Hola Mundo!. Es el siguiente:

```
http://127.0.0.1:8080/holaMundo.php
```

```
<?php
    print("¡Hola Mundo!");
?>
```

Para poder acceder a la aplicación hay que escribir en el navegador web la siguiente dirección que dará acceso a la aplicación:

<http://apps.facebook.com/tfcsara/>

Lo primero que vemos es una página como la que se muestra en la Figura 3.1.



Fig 3.1 Cuadro de dialogo con los permisos que pide la aplicación.

Esta página sólo aparece la primera vez que se intenta acceder a la aplicación, (el resto de veces que se use, irá directamente a la aplicación).

Tal y como se aprecia en la figura 3.1, se puede controlar quién puede ver la actividad relacionada de esta aplicación en Facebook y muestra qué información recibe la aplicación en referencia al usuario que la añade. En este caso, dado que no se han añadido permisos extras a la configuración, sólo recibe la información básica del usuario.

Una vez se ha pulsado en el botón “*Ir a la aplicación*”, se obtiene el resultado mostrado en la Figura 3.2.

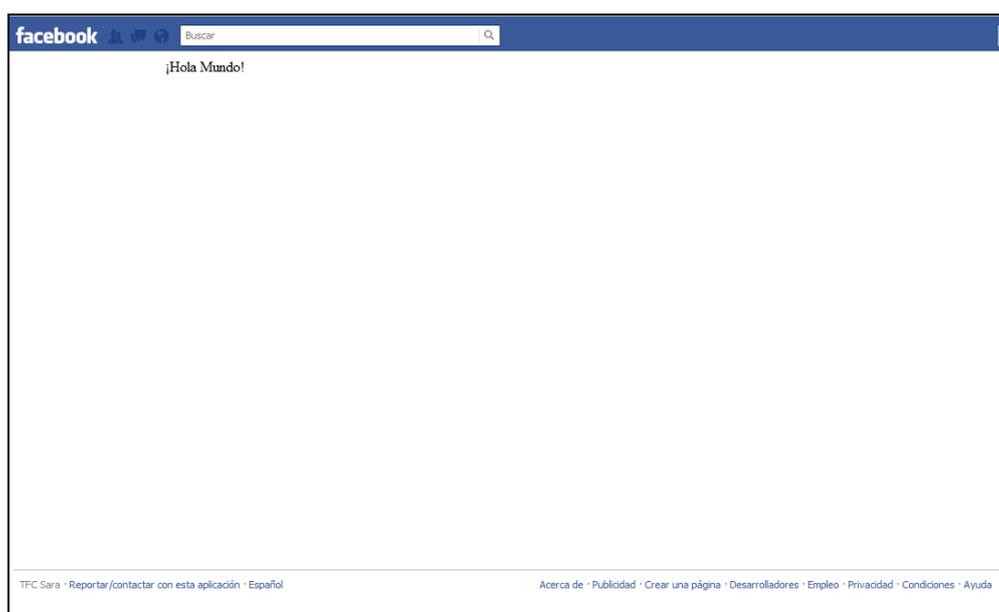


Fig 3.2 Resultado de la aplicación.

3.2. Mostrar datos del usuario

A pesar de la abundante información que se puede extraer del usuario (ver Tablas 2.2 y 2.3), para llevar a cabo la aplicación final, serán de utilidad únicamente los datos que se citan a continuación.

- Información básica.
- Cumpleaños.
- Localización.

En esta segunda fase de la aplicación, en la que deben aparecer en pantalla diferentes datos del usuario que la usa, se ha tenido en cuenta que la información a la que se accede no es toda básica.

Para ello, hay que añadir al apartado “*User & Friend Permissions*” (en la sección de configuración de los permisos) los ítems siguientes:

- email
- user_birthday
- location

En la imagen que se muestra en la Figura 3.3, se pueden ver los tres campos a los que accederá la aplicación:



Fig 3.3 Permisos extras añadidos a la configuración.

De esta manera, al acceder por primera vez a la aplicación, aparecerá el diálogo mostrado en la Figura 3.4.



Fig 3.4 Cuadro de diálogo para autorizar los datos que usará la aplicación.

Podemos observar, que la aplicación, ahora, recibe la información básica del usuario, además de su correo electrónico, su fecha de nacimiento y su ubicación.

Asimismo, se puede tener control de quién puede ver los posts que la aplicación puede hacer en el muro del usuario.

El siguiente paso es pulsar en el botón “Ir a la aplicación”, que mostrará diferente información acerca del usuario, tal y como se muestra en la Figura 3.5.



Fig 3.5 Datos del usuario mostrados al iniciar la aplicación.

3.3 Guardar los datos del usuario: Uso de la Base de Datos

En esta fase, se va a proceder a el almacenamiento de los datos del usuario para su posterior manipulación en una base de datos, por lo tanto, lo primero que se debe hacer es crear una base de datos.

Para ello, el servidor web Apache instalado se ha incorporado un administrador de base de datos, como phpMyAdmin, que facilita las tareas de creación y administración de base de datos.

Cuando se haya accedido, mediante el navegador web a la página de administración anteriormente citada, hay que crear una nueva base de datos (**fbapp**) en la pestaña Base de datos, tal y como se muestra en la Figura 3.6. Esta será la base de datos de nuestra aplicación:



Fig 3.6 Creación de la base de datos.

A continuación, se procederá a la creación de la tabla *users*. Esta tabla contendrá la información del usuario recogida mediante la API de Facebook. Esta tabla tendrá tantos campos como datos del usuario se quiera obtener. En este caso, la tabla tendrá los campos que se detallan en la Tabla 3.1.

Tabla 3.1 Campos que contiene la tabla *users* de la base de datos.

CAMPO	TIPO	PRIMARY KEY ¹	DESCRIPCIÓN
ID	BIGINT ²	SI	Identificador del usuario
NAME	VARCHAR ³ (longitud 255)	NO	Nombre completo del usuario
FIRST_NAME	VARCHAR (longitud 255)	NO	Nombre del usuario
LAST_NAME	VARCHAR (longitud 255)	NO	Apellido del usuario
GENDER	VARCHAR (longitud 255)	NO	Sexo del usuario
LOCALE	VARCHAR (longitud 255)	NO	Ciudad del usuario
EMAIL	VARCHAR (longitud 255)	NO	E-mail del usuario
BIRTHDAY	VARCHAR (longitud 255)	NO	Cumpleaños de usuario

Ahora, al entrar en la aplicación, esta mostrará los datos del usuario que la está usando y, además, insertará en la base de datos la información que ha recogido del usuario. Por ejemplo, la figura 3.7 muestra el contenido de la tabla *users* tras la instalación de la aplicación por parte del usuario “TFC Sara”.

The screenshot shows the phpMyAdmin interface with the 'usuarios' table selected. The SQL query is: `SELECT * FROM 'usuarios' LIMIT 0, 30`. The table structure is as follows:

ID	NAME	FIRST_NAME	LAST_NAME	GENDER	LOCALE	EMAIL	BIRTHDAY
100002894186773	Tfc Sara	Tfc	Sara	female	es_ES	ts.sara@gmail.com	02/25/1986

Fig 3.7 Datos del usuario almacenados en la base de datos.

No obstante, antes de introducir los datos en la base de datos, la aplicación comprueba que el usuario que está haciendo uso de la aplicación, no haya

¹ Una *primary key* (clave primaria) de una tabla relacional identifica de forma única cada registro en la tabla. Puede ser cualquier atributo que garantice ser único.

² BIGINT: Tipo de dato INTEGER (número entero) con valor desde -2^{63} hasta $2^{63}-1$.

³ VARCHAR: Tipo de dato de longitud variable con un máximo de 8000 caracteres ANSI.

hecho uso de la aplicación anteriormente mediante la comprobación del ID en la base de datos. Si ya existe, mostrará sus datos pero aparecerá el mensaje “El usuario ya existe en la Base de Datos”. Ver Figura 3.8.



Fig 3.8 Comprobación de usuario ya registrado en base de datos.

Si el usuario no existe, la aplicación mostrará sus datos y confirmará que la información se ha guardado con éxito en la base de datos. Ver Figura 3.9.

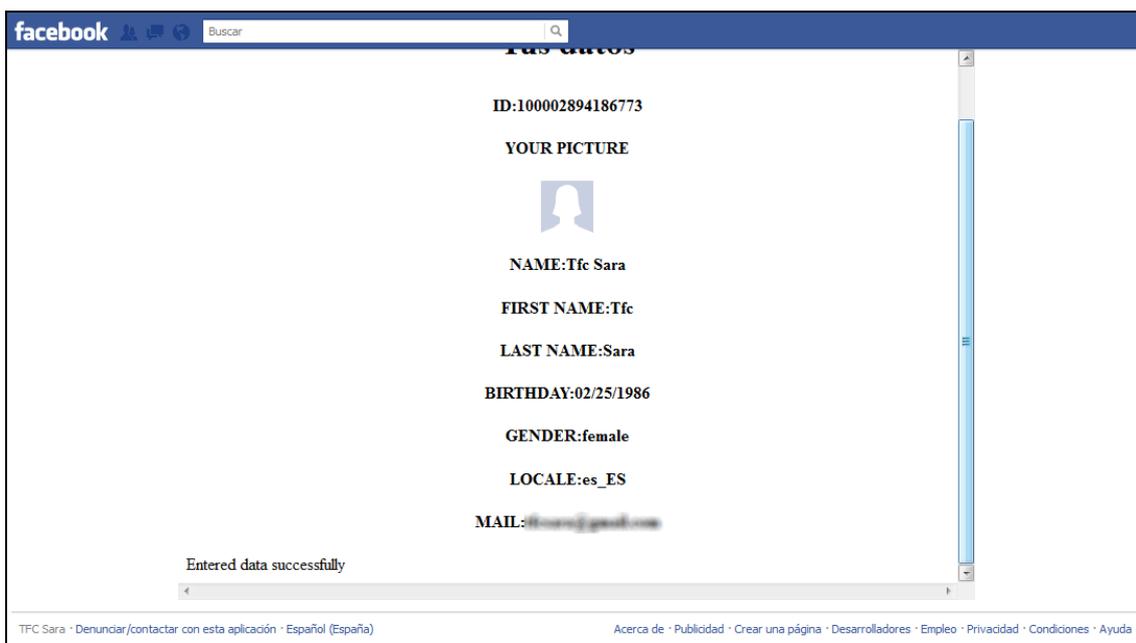


Fig 3.9 Introducción satisfactoria de los datos en la tabla.

3.4 Publicar en el muro del usuario

Para que la aplicación pueda escribir en el muro del usuario en cuestión, es necesario añadir un permiso extra. En este caso, hay que añadir en la configuración de los permisos de la aplicación, en el apartado “Extended Permission”, el ítem *publish_stream* tal y como se muestra en la figura 3.10:

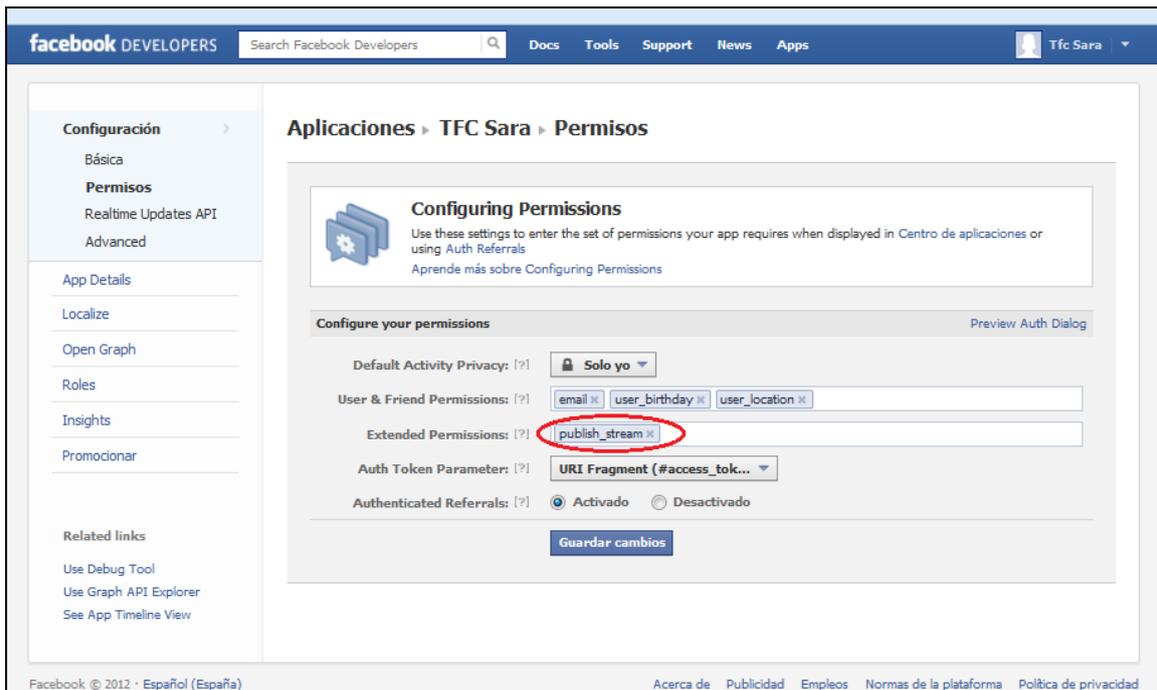


Fig 3.10 Introducción del campo para poder publicar en nombre del usuario.

Al introducir este ítem, la aplicación podrá publicar posts en el muro del usuario. Los posts que publique se tienen que predefinir en el código fuente de la aplicación. En este caso, la API de Facebook permite definir un mensaje (“Ha empezado a usar TFC Sara”) accediendo al muro del usuario de la aplicación. Ver Figura 3.11.



Fig 3.11 Post publicado en el muro del usuario que usa la aplicación por primera vez.

3.5. Enviar invitaciones a amigos

El siguiente paso es poder enviar invitaciones a los amigos del usuario que aún no la hayan instalado.

Para llevar a cabo esta funcionalidad, hay que saber, primero de todo, qué amigos tienen instalada la aplicación. Para ello, Facebook proporciona la posibilidad de poder recuperar la lista de todos los usuarios de la plataforma que tienen la aplicación instalada, haciendo uso del objeto FQL (Facebook Query Language). El objeto permite, mediante sentencias SQL, la consulta de las tablas de usuarios de Facebook.

De esta manera, se recuperan, inicialmente, los identificadores de usuarios que tienen la aplicación instalada. Una vez recuperada la lista, se consulta qué usuarios de la lista son amigos del usuario que está utilizando la aplicación. FQL devuelve el listado y mediante la función de la llamada a la API *sendRequestViaMultiFriendSelector()*, que muestra automáticamente un cuadro de diálogo con los amigos del usuario a los que enviar la invitación, de los cuales se han excluido los que ya tienen la aplicación instalada.

La figura que 3.12, muestra el cuadro de diálogo con los amigos a los que poder enviar la invitación.

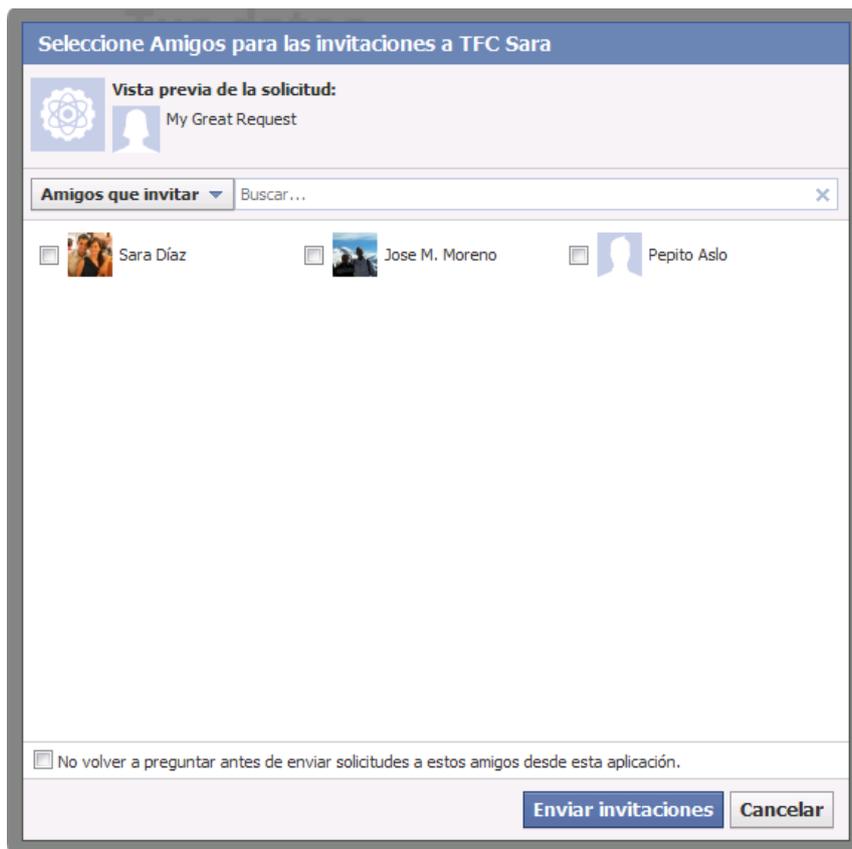


Fig. 3.12 Cuadro de diálogo con los amigos a los que poder invitar para usar la aplicación.

Al enviar la invitación a los amigos seleccionados, estos reciben una nueva solicitud en su página de Facebook conforme se les invita a usar TFC Sara, tal y como se muestra en la figura 3.13.

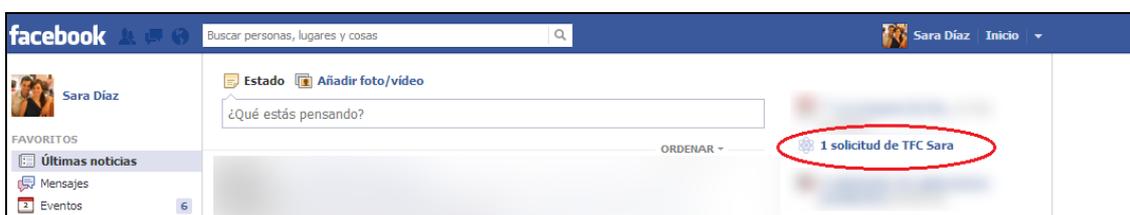


Fig. 3.13 Solicitud de uso de la aplicación en la página del usuario al que se ha enviado la invitación.

3.6. Juego SNAKE

Este apartado se centra no solo en el propio juego de la serpiente sino que además incluye otras funcionalidades, algunas implementadas y otras para implementaciones futuras.

En la Figura 3.14, se muestra un diagrama con los pasos que se siguen al cargar la página de la aplicación.

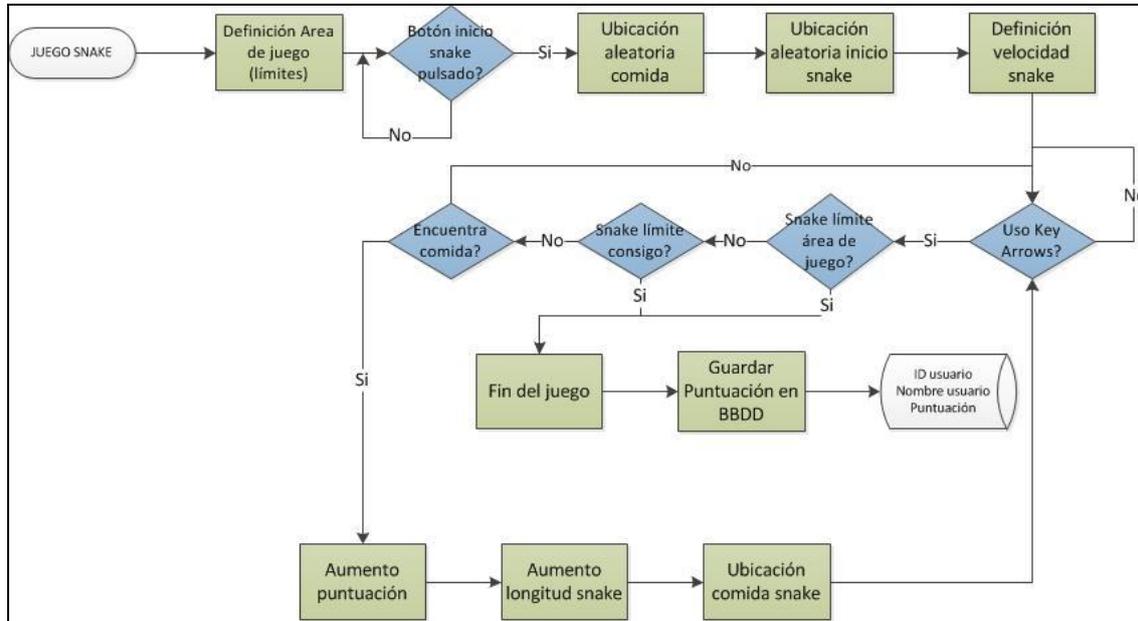


Fig 3.14 Procesos que se siguen en el juego Snake.

Al acceder al juego se define, primero de todo, el Área de juego (Ver Fig. 3.15), es decir los márgenes o límites en los cuales se moverá la serpiente una vez se inicie el juego. En este caso se definen dos constantes *rect_h* (altura) y *rect_w* (anchura) del área de juego.

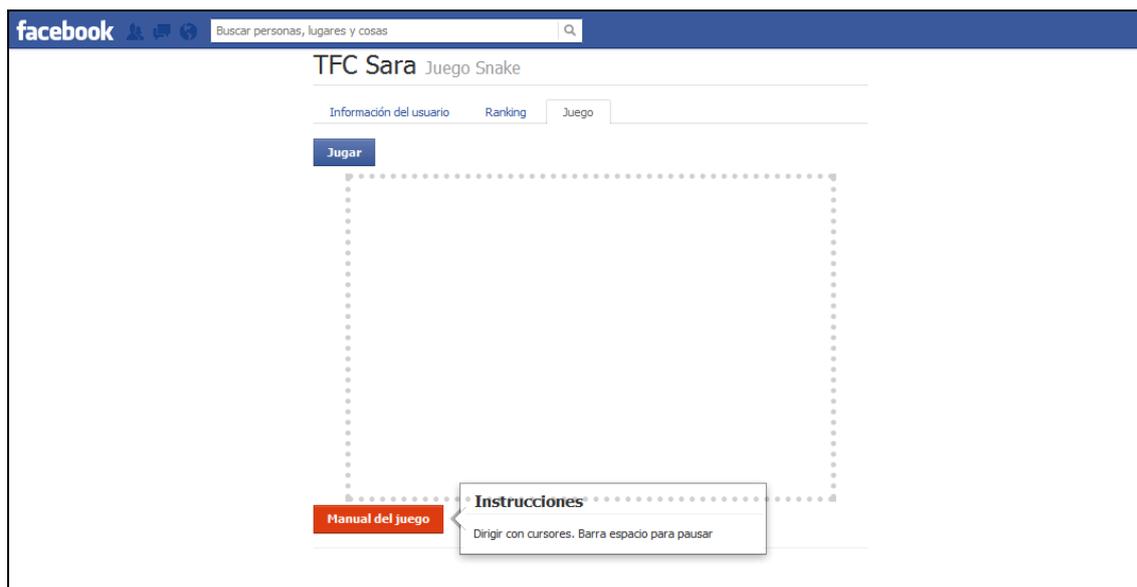


Fig 3.15 Área del juego.

A continuación, se espera a que el usuario inicie el juego pulsando el botón Jugar. De no ser así, el juego nunca empieza.

Cuando el usuario ha decidido empezar a jugar al *snake*, se procede a ubicar la comida de la serpiente. Esta ubicación se hace de forma aleatoria dentro del área de juego anteriormente descrito. Además, la comida tiene por defecto un valor definido. Esta comida será la que ayude a la serpiente a aumentar su longitud y la puntuación del usuario.

El siguiente paso es ubicar la posición de inicio de la serpiente. Este procedimiento también es aleatorio y se hace dentro del área de juego.

En la figura 3.16 se muestra el inicio del juego con la comida y la posición de la serpiente antes de empezar a usar los cursores.

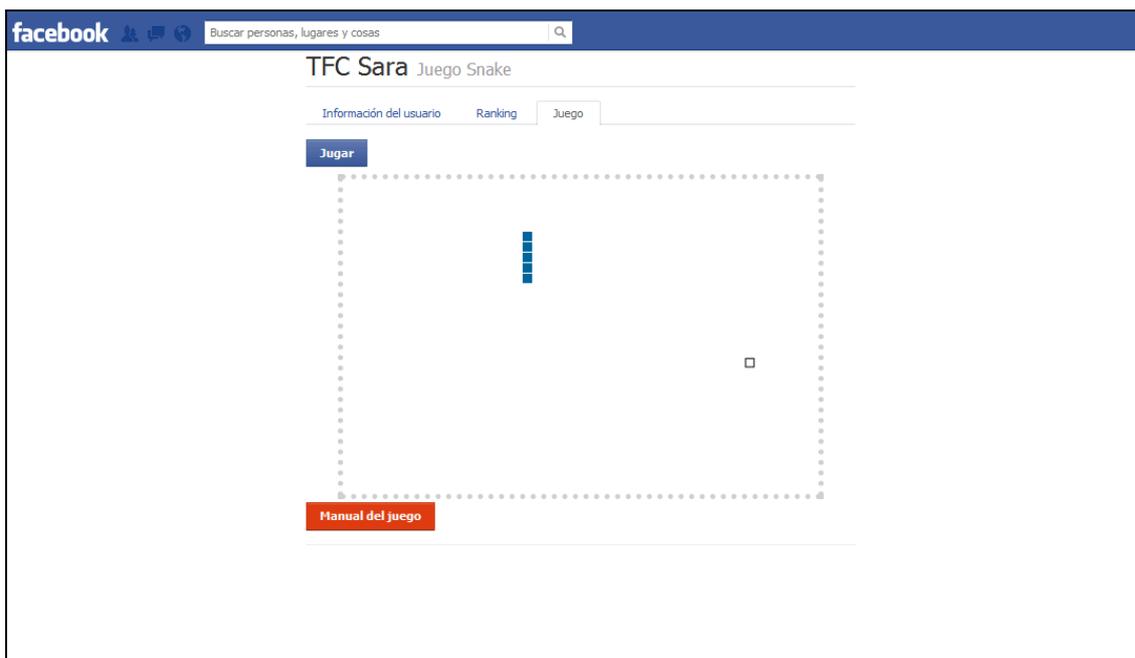


Fig 3.16 Ubicación de la comida y la serpiente dentro del área de juego.

Como último procedimiento previo al propio inicio del juego, se determina la velocidad a la que irá la serpiente por defecto. No obstante, esta velocidad se puede modificar para ir más deprisa pulsando la tecla espaciadora del teclado.

A partir de este punto, el usuario empieza a interactuar con el juego mediante las teclas de dirección del teclado.

Al hacer uso de estas teclas, pueden ocurrir varios escenarios:

- A) La serpiente llega al límite del área de juego, es decir, choca contra las “paredes” lo cual implica la finalización del juego.
- B) La serpiente choque consigo misma y por lo tanto, acabe la partida.
- C) La serpiente encuentre comida y con ello, aumente su puntuación y su longitud.

Tanto en el caso 1) como en el caso 2) el juego finaliza, se muestra la puntuación obtenida por el usuario en la misma página. Esta puntuación se guardará en una base de datos y a posteriori se podrán consultar los diferentes resultados tal y como se comenta en el apartado 3.X.

Para el caso 3), después de haber conseguido la comida, volvería a aparecer otra ubicación aleatoria de la comida. Seguidamente, se pueden volver a usar los cursores para obtener comida nuevamente, chocar con los límites del área de juego o chocar consigo misma y así, sucesivamente.

3.7. Ranking

El Ranking muestra las puntuaciones conseguidas por todos los usuarios de la aplicación.

Se ha creado una tabla llamada *ranking* donde se almacenan las puntuaciones obtenidas por los jugadores. En esta tabla se guardan diferentes datos referentes al usuario en cuestión. Los campos que componen la tabla “ranking” se muestra en la tabla 3.2.

Tabla 3.2 Campos que contiene la tabla *ranking*.

CAMPO	TIPO	PRIMARY KEY	DESCRIPCIÓN
POS	INT ⁴	NO	contador del número de registros almacenados
FB_ID	BIGINT	NO	identificador del usuario
FB_NAME	VARCHAR (longitud 255)	NO	nombre del usuario
GAME_POINTS	INT	NO	puntuación obtenida en el juego

Cada vez que se finaliza una partida, la puntuación obtenida se almacena en la tabla, tal y como se muestra en la figura 3.17.

The screenshot shows the phpMyAdmin interface for a database. The 'ranking' table is selected and displayed in a grid view. The table structure is as follows:

POS	FB_ID	FB_NAME	GAME_POINTS
1	100002894186773	Tfc Sara	47
2	100002894186773	Tfc Sara	21
3	100002894186773	Tfc Sara	0
4	100002894186773	Tfc Sara	33
5	100002894186773	Tfc Sara	93
6	100002894186773	Tfc Sara	10
7	100002894186773	Tfc Sara	0
8	100002894186773	Tfc Sara	0
9	100002894186773	Tfc Sara	0
10	100002894186773	Tfc Sara	0
11	100002894186773	Tfc Sara	0
12	100002894186773	Tfc Sara	0
13	100002894186773	Tfc Sara	0
14	100002894186773	Tfc Sara	168
15	100002894186773	Tfc Sara	229
16	100002894186773	Tfc Sara	798

Fig 3.17 Puntuaciones guardadas en la Base de datos.

Para implementar esta funcionalidad, se han creado dos tipos de ranking, uno general y otro personal.

⁴ INTEGER: tipo de dato que puede representar un subconjunto finito de los números enteros

En el general, aparecerán las 10 mejores puntuaciones que tiene registrado el juego en la base de datos, independientemente del usuario que esté usando la aplicación.

Para obtener de la base de datos los diez mejores resultados, se realiza una consulta SQL, ordenando las puntuaciones de forma descendente y limitándola a diez posiciones. La *query* usada es la siguiente:

```
$select = "SELECT * FROM $tablename ORDER BY GAME_POINTS DESC LIMIT 10";
```

En el personal, como su nombre indica, aparecerán las 5 mejores puntuaciones del jugador. En este caso, la *query* usada es la siguiente:

```
$select = "SELECT * FROM $tablename WHERE FB_ID = $id ORDER BY GAME_POINTS DESC LIMIT 5";
```

Se puede observar en esta sentencia que se ha tenido en cuenta el id del usuario para obtener las puntuaciones, porque son las únicas que interesa recuperar.

3.8. Achievements (trofeos)

Una de las opciones que Facebook da a los desarrolladores es la implementación de los *Achievements* o trofeos.

Esta utilidad posibilita una aplicación con más completa ya que se puede publicar los logros conseguidos en el muro y por lo tanto, se da más publicidad a la aplicación.

Para poder implementar esta funcionalidad es obligatorio catalogar la aplicación como juego (Game) en el menú de la configuración de la aplicación.

Es el objeto ***game.achievement*** de la API de Facebook el que permite definir los diferentes trofeos o logros del juego. Además, existen una serie de conceptos que se deben tener en cuenta a la hora de configurar esta opción. A continuación se detallan:

- Cada logro tiene puntos asociados.
- En cada juego se obtiene un total de 1000 puntos distribuidos entre todos los logros.
- En cada juego se obtienen, como máximo, 1000 logros.
- Los logros que sean más escasos y tengan puntuaciones más altas, recibirán más distribución. Por ejemplo, los logros que tengan valores menores a 10 puntos, no obtendrán apenas distribución. Las aplicaciones deben aspirar a conseguir entre 50 y 100 logros con puntuaciones repartidas de la siguiente forma: 50 puntos (difícil), 25 puntos (medio) y 10 (fácil).

- Un usuario puede alcanzar un logro en particular de un juego únicamente una vez.
- Es contrario a la política de uso, eliminar un logro, excepto en el caso de logros de prueba.

3.9. Aplicación completa

En el diagrama de la figura 3.18 se muestra el funcionamiento completo de la aplicación creada en Facebook. Todas las fases anteriormente descritas constan a excepción de la aplicación base ¡Hola Mundo!

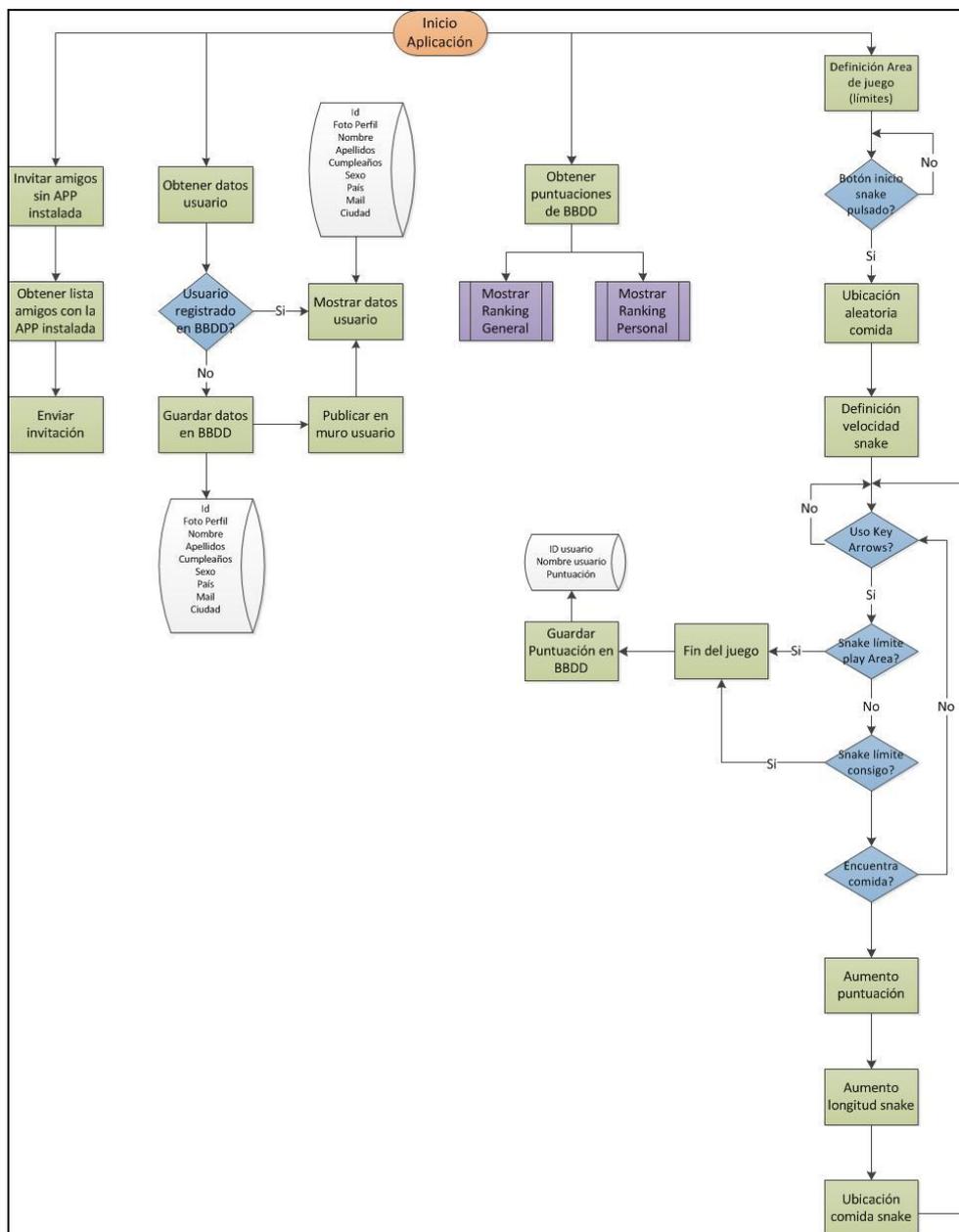


Fig 3.18 Diagrama de la aplicación completa.

El entrar a la aplicación, se puede distinguir perfectamente cada una de las partes ya que se han elaborado pestañas para separar la parte del juego, los rankings, la información del usuario y el botón que permite invitar a los amigos.

CAPITULO 4. CONCLUSIONES

Una vez realizado el desarrollo de la aplicación *snake* integrada en Facebook, podemos llegar a las siguientes conclusiones.

Se necesitan pasos previos para la creación de una aplicación, así como tener instalados una serie de tecnologías para poder desarrollar las aplicaciones. No obstante, cualquier usuario de la plataforma puede crear de forma muy sencilla cualquier tipo de aplicación, ya sea para entretenimiento como para uso laboral.

Mediante las aplicaciones, se puede obtener cualquier tipo de información de los usuarios en Facebook haciendo llamadas a la API de Facebook y accediendo a los campos que se necesiten del usuario en cuestión. No obstante, la información a la que se quiere acceder no siempre es pública. Existen dos tipos: la que no requiere permisos y la que sí. Esta última se regula mediante un cuadro de dialogo donde el usuario puede aceptar o rechazar el acceso a su información. En definitiva, es el usuario el que decide compartir su información.

La integración de un juego no deja de ser igual que en una propia página web, es decir, se carga desde un servidor externo donde está alojada la aplicación, sólo que en este caso, se hace en un espacio definido por la aplicación en Facebook.

Durante la elaboración del trabajo ha ocurrido algún problema a la hora de validar la cuenta de usuario para poder crear una aplicación.

A nivel personal, he adquirido conocimientos básicos de programación en PHP o MySQL, indispensables para la realización del trabajo, con lo cual, además de ver cómo se integran las aplicaciones y toda la información que ello conlleva, ha habido un previo aprendizaje de las tecnologías anteriormente citadas.

En cuanto al estudio de ambientalización referente a este trabajo, este se reduce a la energía consumida por el ordenador personal para la realización de la aplicación.

REFERENCIAS

- [1] <http://developers.facebook.com/docs/appcenter/categories/>
- [2] http://es.wikipedia.org/wiki/Servidor_HTTP_Apache
- [3] <http://es.wikipedia.org/wiki/MySQL>
- [4] <http://es.wikipedia.org/wiki/Php>
- [5] <http://es.wikipedia.org/wiki/JavaScript>
- [6] <http://es.wikipedia.org/wiki/Css>
- [7] <http://developers.facebook.com/docs/guides/web/>
- [8] <http://developers.facebook.com/docs/guides/canvas/>
- [9] <http://developers.facebook.com/mobile/>
- [10] <http://es.wikipedia.org/wiki/Iframe>
- [11] http://es.wikipedia.org/wiki/Interfaz_de_programaci%C3%B3n_de_aplicaciones
- [12] <https://developers.facebook.com/docs/achievements/>