



D. Antonio Jesús Rueda Ruiz y D. Carlos Porcel Gallego pertenecientes al  
Departamento de Informática de la Universidad de Jaén.

### INFORMAN

Que la memoria titulada “Sistema de recomendación para grupos de amigos en Facebook” ha sido realizada por **D. José Antonio Martínez Santiago** con DNI 26498048-R bajo nuestra dirección y se presenta como memoria del Proyecto Fin de Carrera para optar a la titulación de Ingeniería Informática.

Jaén, Septiembre de 2012

VºBº

Fdo: **Antonio Jesús Rueda Ruiz**

Fdo: **Carlos Porcel Gallego**

## **Agradecimientos**

Mi mayor agradecimiento es para mis padres, por todo su esfuerzo y apoyo a lo largo de estos años.

También agradecer a mi hermana, abuelos, y en general, a toda mi familia por estar siempre ahí.

A todos los amigos que me han apoyado, y que sin ellos, todo esto hubiera sido mucho más difícil.

A mis compañeros y profesores, que han hecho inolvidable esta etapa universitaria.

Gracias a todos.



# Contenido

---

1. Introducción .....	9
1.1 Introducción al proyecto .....	10
1.2 Propósito y objetivos del proyecto .....	12
1.3 Diferentes alternativas y elección .....	13
1.4 Estructura del proyecto.....	16
2. Redes sociales.....	19
2.1 Introducción.....	20
2.2 Por qué integrar la aplicación en Facebook .....	23
3. Desarrollo de aplicaciones en Facebook.....	27
3.1 Facebook Developers .....	28
3.2 Aplicaciones en Facebook.....	31
3.3 Graph API y permisos.....	36
3.4 PHP SDK.....	42
3.5 Registrando una aplicación .....	44
4. Procesos de Ingeniería del Software .....	51
4.1 Introducción.....	52
4.2 Análisis de requisitos.....	52
4.1.1 Requisitos funcionales.....	53

4.1.2 Requisitos no funcionales .....	55
4.3 Análisis del sistema .....	56
4.3.1 Casos de uso .....	56
4.3.2 Escenarios .....	72
4.3.3 Modelo del dominio .....	80
4.4 Diseño del sistema.....	81
4.4.1 Diagrama de clases de diseño .....	81
4.4.3 Diagramas de interacción .....	84
4.4.3 Diseño de los datos .....	89
4.4.4 Cálculo de la relevancia .....	98
4.4.5 Diseño de la interfaz .....	100
4.5 Implementación.....	133
4.5.1 Arquitectura de la aplicación .....	133
4.5.2 Lenguajes y tecnologías utilizadas.....	133
4.5.3 Herramientas de desarrollo utilizadas .....	138
4.6 Pruebas .....	140
4.6.1 Test de prueba .....	140
4.6.2 Resultados obtenidos.....	150

4.6.3 Validación .....	152
4.6.4 Visualización en distintos navegadores.....	155
4.7 Instalación de la aplicación .....	158
5. Conclusiones.....	159
5.1. Conclusiones .....	160
5.2 Desarrollos futuros .....	161
Referencias y bibliografía.....	163
Anexo I: Manual de instalación .....	165
A.1 Instalación en el servidor .....	166
A.2 Instalación en el cliente.....	173
Anexo II: Manual de usuario.....	177





# 1. Introducción

---

## 1.1 Introducción al proyecto

A diario y en cualquier lugar del mundo, las personas se recomiendan entre sí contenidos de ocio, como pueden ser películas, música, libros, etc. Generalmente suele ser de manera oral, tras una experiencia positiva del individuo que realiza la recomendación con dicho contenido de ocio.

Estas recomendaciones suelen venir de tu círculo de personas, en el cual con unas existe una afinidad mayor que con otra, por lo que es posible que con algunas personas se compartan gustos en común mientras que con otras no. Esto nos lleva a que, cuando recibimos una recomendación de una película de parte de una persona, tengamos una intuición sobre si nos gustará o no en función de los gustos que conocemos de esa persona. Además, si esa persona nos ha recomendado previamente algo que nos ha gustado realmente, el grado de fiabilidad de esa persona habrá aumentado, mientras que si nos recomendó algo que no fue de nuestro agrado, ese grado de fiabilidad decrementará. Por tanto, no todas las recomendaciones son iguales, depende de que persona nos la haga.

En el caso de recibir varias recomendaciones por parte de tu círculo de amistades, ¿a cuál darle más importancia? Si es entre dos personas, es posible que la decisión sea fácil. Pero si tenemos más de dos es posible que no sea un acto tan trivial. Además, utilizar la experiencia pasada con esas personas sería interesante para intentar predecir cuál recomendación será más de nuestro agrado.

Otro problema muy común es que recibamos una recomendación de alguien en mitad de una conversación, pero luego la olvidamos si no hemos tenido ocasión de anotarla. Sería interesante disponer de una herramienta que nos permitiera tener registradas esas recomendaciones. Pero no solo almacenarlas, sino tener una completa gestión de ellas.

Esta gestión pasaría por el almacenamiento de las recomendaciones, valoraciones dadas (si ha gustado o no), ordenación de recomendaciones según la afinidad que tengamos con la persona que la ha realizado, etc. Sería una manera moderna de utilizar el tradicional boca a boca para recomendar contenidos de ocio.

Para dar solución a lo expuesto anteriormente, este proyecto se basará en la gestión de recomendaciones de contenidos de ocio, más concretamente se centra en la gestión de las tradicionales recomendaciones o sugerencias realizadas por tus amigos o familiares en la que se recomienda un contenido de ocio a través de una aplicación web.

Dicha aplicación estará integrada en una red social, Facebook en este caso, ya que el ambiente social de la aplicación hace idónea la integración en una red social en la que estarán recogidos gran número de los amigos, compañeros de trabajo o familiares del usuario. A través de ella, los usuarios de la aplicación podrán interactuar con sus contactos ofreciendo (y recibiendo) recomendaciones de contenidos de ocio, como pueden ser cine, música, libros, etc.

Y la aplicación no solo se centrará en la recogida y gestión de dichas sugerencias, sino que además se implementarán mecanismos de retroalimentación que permitan valorar una sugerencia realizada. Con esta información, se podrá calcular el grado de afinidad de dos usuarios según las valoraciones previas de ítems de ocio que implique a dichos usuarios.

Por tanto, además de gestionar las recomendaciones recibidas por parte de otros usuarios, las ordenará según el grado de afinidad con ese usuario, ya que representa de algún modo la fiabilidad o confianza en que la nueva recomendación será del agrado por parte del usuario en base a la experiencia previa.

La idea que da forma a este proyecto surgió por parte de un usuario ajeno a la universidad. Este usuario tenía un problema: solía recibir muchas recomendaciones de contenido de ocio, especialmente cine y música, por parte de sus amigos. Esto implicaba dos cosas: por un lado el poder gestionar esas recomendaciones, que a veces caen en el olvido si no son anotadas. Por otro lado, tener un mínimo de confianza sobre si la recomendación será o no de su agrado debido a que la persona que le ha realizado esa recomendación puede tener gustos diferentes.

Por tanto, surge la necesidad de trasladar una tarea que se ha realizado desde siempre, como es la de recomendar a un conocido un contenido de ocio, pero con una herramienta que nos permita gestionar eficientemente y nos prediga cual de esas recomendaciones es mas fiable.

El usuario podrá indicar el grado de satisfacción con una recomendación recibida, de manera que se alimenta el sistema con información que se utilizará para establecer el grado de afinidad entre dos amigos y predecir si es probable que un nuevo ítem que nos ha sido recomendado será de nuestro agrado o no en base a las recomendaciones previas que nos ha dado ese usuario.

## **1.2 Propósito y objetivos del proyecto**

El propósito del proyecto es el desarrollo de una aplicación web funcional e integrada en una red social para la gestión de recomendaciones o sugerencias de contenidos de ocio entre usuarios. La red social en la que estará integrada será Facebook, empleando las herramientas que dicha plataforma ofrece a los desarrolladores y trasladando un concepto clásico como es el de la recomendación a un amigo de un contenido de ocio a un entorno como es una red social y utilizando los datos almacenados para establecer una medida de confianza entre usuarios que permita predecir el grado de afinidad entre usuarios.

Para la lograr éste propósito, se marcan los siguientes objetivos:

- Estudio de la red social Facebook, y más concretamente, de su API para desarrolladores.
- Desarrollo de una aplicación web integrada en Facebook que permita enviar y recibir recomendaciones entre amigos de la red social.
- Creación de una interfaz intuitiva, usable y que motive al usuario a usarla y a compartir la aplicación con sus contactos en la red social.
- Estudio de diferentes medidas de confianza para establecer el grado de afinidad entre usuarios en base a la información obtenida en el pasado y la implementación de la misma.
- Establecer un mecanismo que permita identificar si un contenido a recomendar ya ha sido recomendado previamente por otros usuarios y sugerirlo cuando otro usuario quiera recomendarlo.
- Memoria detallando las actividades realizadas en este proyecto

### **1.3 Diferentes alternativas y elección**

Una vez planteado el problema, se analizó si actualmente hay alguna alternativa que de cobertura a esa necesidad. Una idea similar que se podría utilizar es la de adaptar un sistema de recomendación clásico.

Un sistema de recomendación es un tipo de filtro de información que presenta al usuario contenido que predice que será de su agrado, en función de la información que tenga sobre el usuario. Básicamente, pueden haber de dos tipos: basados en contenido (se compara entre el contenido del ítem a recomendar y el perfil del usuario) o basados en colaboración (presenta al usuario contenidos que han sido del agrado de usuarios con un perfil similar al

del usuario). En definitiva, es el propio sistema el que genera la recomendación y se la envía al usuario.

Por tanto, había dos alternativas a escoger:

- **Opción nº 1:** Los usuarios envían recomendaciones en varias categorías (pelis, música, libros, etc.), a sus amigos. Estas recomendaciones se ordenan en un ranking en función de la afinidad con cada amigo. Las valoraciones posteriores que el usuario destinatario realizarías tras ver la peli, leer el libro etc. actualizarían el nivel de afinidad con el amigo en cuestión según la valoración dada al ítem recomendado.
- **Opción nº 2:** Un enfoque tradicional como sistema de recomendación. El sistema es el que genera las recomendaciones y se las ofrece al usuario. Existiría una base de datos con los contenidos de ocio y los usuarios que tuvieran instalada la aplicación valorarían cada ítem. Internamente el sistema crearía perfiles de usuarios con gustos semejantes (no tendrían que ser amigos) y te proporcionaría nuevas sugerencias a partir de las películas que sean bien valoradas por usuarios compatibles a ti. Esto sería similar a FilmAffinity, sólo que integrada en Facebook.

La primera opción es más sencilla de implementar, ya que no requiere de un motor de recomendación que genere las recomendaciones a los usuarios según las valoraciones ofrecidas por estos a los contenidos de ocio de la base de datos. No requiere una base de datos previa de películas, música, etc. a priori, si no que los usuarios hacen la sugerencia y se almacena. En resumen, sería una forma moderna de hacer una cosa que se ha hecho desde siempre: recomendar a nuestros amigos o familiares la visión de una película o la lectura de un libro.

La segunda opción es un modelo ya estudiado y probado que es conocido que ofrece unos resultados aceptables. Al no utilizar solo las relaciones entre amigos, puede elaborar un grafo de afinidad mucho más complejo. Además, esta

probado que el modelo funciona en sitios como FilmAffinity. Sin embargo, aporta menos originalidad ya que sería trasladar un sistema ya existente e integrarlo en una red social como Facebook. Además, conlleva un coste de mantenimiento, ya que hay que añadir las nuevas películas, discos de música, libros, etc... que vayan saliendo al mercado a la base de datos. Además, es posible que el hecho de que un usuario necesite valorar previamente ciertos ítems para que el motor de recomendación le genere una recomendación ajustada a su perfil puede hacer que el usuario pierda el interés antes de utilizar realmente la aplicación.

La siguiente tabla muestra un resumen de las consideraciones tenidas en cuenta a la hora de escoger un enfoque u otro en el desarrollo de la aplicación:

Opción nº 1	Opción nº2
La aplicación es más sencilla.	La aplicación es más compleja al necesitar un motor de recomendación que genere las recomendaciones.
No requiere que los contenidos a recomendar estén previamente registrados en el sistema.	Requiere una base de datos con los contenidos a recomendar. Esto implicaría un mantenimiento extra para añadir nuevos libros, películas, etc.
No requiere acción inicial por parte del usuario, podría enviar o recibir una recomendación desde el primer momento que esté registrado en la aplicación.	Requiere un esfuerzo por parte del usuario para solucionar el arranque en frío del motor de recomendación valorando un conjunto de ítems inicialmente.
Es una forma moderna y original de adaptar algo que se ha hecho toda la vida. Los usuarios utilizarían la	Es menos original ya que es trasladar un sistema ya probado a una aplicación para una red social.

aplicación de manera natural.	
El grafo de usuarios afines es más reducido debido a que se limita a los amigos de tu red social. Inicialmente el grado de confianza de las recomendaciones no será bueno.	El grafo de usuarios afines es más grande al tener en cuenta usuarios sin ninguna relación entre sí, el grafo de afinidades sería mucho más complejo.

**Tabla 1 Comparación de alternativas**

Una vez estudiadas a fondo ambas opciones, nos vamos a decantar por la primera opción. Mientras que la segunda opción solo consiste en trasladar un sistema ya probado y funcionando a una red social, la primera opción se trata de una manera novedosa de hacer algo que la gente lleva haciendo toda la vida, recomendarle cosas a sus amigos.

## 1.4 Estructura del proyecto

En este primer capítulo se ha introducido de una manera breve el proyecto a realizar, especificando el propósito y los objetivos del mismo y una comparación entre las diferentes alternativas planteadas.

En el segundo capítulo se hablará de las redes sociales en la actualidad, ofreciendo una visión general de ellas y explicando por qué se ha decidido integrar la aplicación en Facebook.

El tercer capítulo se centrará en los aspectos de Facebook para desarrolladores, realizando un estudio general de la plataforma que Facebook ofrece de apoyo a los desarrolladores (Facebook Developers) y explicando el proceso para dar de alta la aplicación del proyecto en Facebook.

El capítulo cuarto es el capítulo central y repasará todo el proceso de Ingeniería del Software llevado a cabo: Análisis, diseño e implementación del



sistema, así como las pruebas realizadas para garantizar que la aplicación cumpla con los requisitos especificados.

En el quinto capítulo se expondrán las conclusiones obtenidas del desarrollo de éste proyecto, mientras que en el sexto veremos unas posibles líneas de ampliación futuras del mismo.

Finalmente, se puede consultar la lista de bibliografía empleada en el desarrollo de la aplicación y la redacción de ésta memoria.

Unidos a éste documento se encuentran dos anexos: el manual de instalación que indica como instalar la aplicación paso a paso y el manual de usuario con ayuda acerca del funcionamiento de la aplicación



## **2. Redes sociales**

---

## 2.1 Introducción

En la actualidad, hay muy pocas personas que desconozcan nombres como Facebook o Twitter. Las redes sociales se han convertido en un fenómeno global que ha revolucionado las relaciones personales a través de internet, especialmente a través de la llegada de la web 2.0 y su popularización. A diferencia de la web 1.0, en la que el usuario solo era un consumidor de información a través de la lectura de sitios web, en la web 2.0 el usuario se convierte en un generador de contenidos, introduciendo información.

Es por ello, que desde algunos sitios llaman a la web 2.0 como la web social. Por tanto, los sitios y aplicaciones web 2.0 tienen en común la participación activa del usuario compartiendo información con otros usuarios. Supone un cambio de mentalidad en lo que hasta ahora había sido la web. Las personas ahora interaccionan con otras en la red, permitiendo establecer relaciones entre ellos y la creación de redes personales. En portales como foros, portales de video, juegos online, etc..., los usuarios pueden comunicarse con otros usuarios por medio de mensajes, comentarios. Esto establece grupos de usuarios con algún interés común, ya sea ver videos musicales o jugar al mismo juego de rol online.

Una red social se podría definir como un conjunto de personas unidas entre si por alguna relación o interés en común, como puede ser la de amistad. No es un concepto nuevo, ya que en la vida real estamos conectados con una serie de personas: amigos, familia, compañeros de trabajo, etc. Una red social traslada los actos cotidianos en las relaciones sociales a un sistema informático, lo cual crea un nuevo modelo de interacción entre personas que hasta ahora era imposible. Hasta ahora, para comunicar algo a los amigos, como la terminación de tus estudios, había que llamar telefónicamente o bien quedar en persona

para transmitir esa información. En una red social, basta con crear una nueva publicación que será compartida en tiempo real con todos tus amigos.



**Figura 1** Redes sociales

Hoy en día es raro el usuario de internet que no dispone de cuenta en algún tipo de red social, ya sea Facebook, Youtube o Twitter. De hecho, los usuarios consumen cada vez más redes sociales diferentes y es por ello que estas se están interconectando: publicar nuestra última entrada de Twitter en el estado de Tuenti, o compartir el último video que hemos visto en Youtube a Facebook.

En las redes sociales no solo se realiza la comunicación con otros usuarios, sino que se expone de una manera abierta y en tiempo real nuestros gustos o intereses con todos los usuarios con los que tengamos alguna relación en la red social

El surgimiento de las redes sociales se remonta apenas una década atrás, tras el enorme crecimiento de usuarios conectados a la red a partir del año

2000. En 2002, se lanza la red social **Friendster**, alcanzando tres millones de usuario en poco más de tres meses. Poco tiempo después, comienzan a surgir un gran conjunto de redes sociales. En el 2003 se publican, entre otras, **Facebook**, **LinkedIn**, **MySpace**, **Hi5**, **Xing**... En 2005, MySpace se convierte en la red social más importante de Estados Unidos, y se lanza el portal de almacenamiento de videos online **Youtube**. Un año más tarde, nace otra las grandes redes sociales actuales: **Twitter**. Además, en España se crea una red social orientada al segmento más joven de población: **Tuenti** y Facebook comienza a crecer en número de usuarios, hasta que en el año 2008 supera a MySpace, y en el año 2009 alcanza los 400 millones de usuarios frente a los escasos 60 de MySpace. En 2010 nace la red social que más crecimiento experimenta a día de hoy: **Pinterest**. Sitios como Facebook, LinkedIn, Twitter o Youtube siguen aumentando el número de usuarios. En 2011, LinkedIn se convierte en la segunda red social más importante en Estados Unidos. Google, tras varios intentos fallidos de entrar en el mercado de las redes sociales con Orkut o Buzz, lanza Google+.



Figura 2 Redes sociales populares

Actualmente, Facebook se acerca a los mil millones de usuarios, Twitter supera los 200 millones, Google+ registra 62 millones. La red social española Tuenti alcanza los 13 millones de usuarios.

Se puede realizar una clasificación a grandes rasgos en las redes sociales:

- **Redes sociales horizontales:** Están dirigidas a un público genérico sin una temática concreta. La función principal es crear un perfil de usuario y compartir contenido con su lista de contactos. En este tipo se encuadran Facebook o Tuenti.
- **Redes sociales verticales:** Están especializadas en alguna temática concreta, pudiendo adaptarse a esa temática de un modo que una red social horizontal no podría, ofreciendo contenidos o actividades más concretos. Por tanto, pueden ser de tipos muy variados, como compartir fotografías (Flickr), musicales (Last.fm), deportivas (Timpik).

Sería imposible contabilizar el número de redes sociales que hay en la actualidad, pues prácticamente a diario surgen nuevas redes sociales, la mayoría de ellas verticales ante la imposibilidad de competir con gigantes como Facebook. Sin duda, han revolucionado la forma en la que gestionamos nuestras relaciones personales.

## 2.2 Por qué integrar la aplicación en Facebook

Facebook es una empresa creada por Mark Zuckerberg y es actualmente la mayor red social del mundo. Aunque en un principio era un sitio para estudiantes de la Universidad de Harvard, actualmente está abierto a cualquier persona del mundo.



Figura 3 Logotipo de Facebook

Una de las estrategias que han ayudado al gran crecimiento de Facebook es la de abrir la plataforma a desarrolladores externos, lo que permite que terceros puedan desarrollar aplicaciones integradas en ellas y hacer negocio gracias a la red social.

Para este proyecto, se han considerado las siguientes ventajas que aporta integrar la aplicación en Facebook, frente a haber realizado una aplicación web con una dirección propia:

- **Sin registro.** La principal y más evidente ventaja es que los usuarios que posean cuenta en Facebook no tienen por qué registrarse en la aplicación, lo que les ahorra el proceso de rellenar el formulario, validar su correo, etc. Esto puede marcar la diferencia entre conseguir un usuario o perderlo debido a la ansiedad e impaciencia de los usuarios frente a los formularios de registro.
- **Obtener tráfico web desde Facebook:** El tráfico web que se puede obtener es muy grande y de bastante calidad, ya que suelen ser recomendaciones o invitaciones que se realizan unos contactos a otros. Además, la aplicación queda indexada en el centro de aplicaciones de Facebook y cualquiera podría encontrarla desde el buscador.



- **Retorno de usuarios.** Debido a que los usuarios de Facebook entran a la red social prácticamente a diario, es probable que el usuario vea de nuevo la aplicación y continúe usándola. De otra manera, se suele dar el caso de que cae en el olvido, y aunque llegara a ser de utilidad para el usuario, éste deja de usarla.
- **Obtener más información sobre el usuario:** Aunque el usuario rellene un formulario de registro, éste debería ser demasiado extenso para recoger toda la información que se podría obtener con Facebook sin necesidad de que la introduzca el usuario. Además, el usuario podría buscar los contactos de su red social que ya utilizan la aplicación, mientras que en una aplicación externa sería más complicado, teniendo que volver a agregarlos.



# **3. Desarrollo de aplicaciones en Facebook**

---

### 3.1 Facebook Developers

El primer paso para el desarrollo de la aplicación será estudiar los conceptos básicos y diferentes posibilidades que tenemos al desarrollar una aplicación por la red social Facebook.

En este capítulo comenzaremos dando una visión general del portal de documentación para el desarrollo de aplicaciones web en Facebook, llamado Facebook Developers. Esta es la documentación oficial de Facebook para el desarrollo de aplicaciones y sitios webs conectados con Facebook.



Figura 4 Pantalla de bienvenida en Facebook Developers

Facebook Developers es en sí misma una aplicación de Facebook, por lo que para poder registrar nuestra aplicación debemos conceder permisos a la aplicación para acceder a nuestra información básica como si de una aplicación más se tratara.



Figura 5 Solicitud de permiso de la aplicación Desarrollador

Además, para poder registrar una aplicación, tu cuenta personal de Facebook debe estar verificada, es decir, se debe introducir un número de teléfono o una tarjeta de crédito para validar la cuenta. Facebook comprueba que el usuario es real. Es necesario para algunas funcionalidades en la red social, entre ellas el añadir nuevas aplicaciones como se ha comentado anteriormente. Lo más sencillo es introducir un número de teléfono válido.



**Figura 6 Crear aplicación sin tener la cuenta validada**

Una vez introducido el teléfono, nos enviarán un mensaje con un código de validación, el cual una vez introducido nos validará la cuenta y podremos registrar nuestra aplicación. El proceso de registro y configuración de la aplicación se verá un poco más adelante en el capítulo *3.4 Registrando nuestra aplicación*.

Como se puede observar en la página principal de Developers, Facebook distingue tres grandes tipos de aplicaciones integradas en su red social:

- **Sitios web:** Facebook nos ofrece una serie de herramientas para hacer los sitios webs “sociales”, es decir, conectarlos con Facebook a través de diferentes plugins sociales como el conocido botón “Me gusta”, simplificar el proceso de registro permitiendo al usuario registrarse con su cuenta de Facebook, etc.
- **Aplicaciones móviles:** Gracias a ello podremos dar un toque social a aplicaciones de iOS (iPhone, iPad) o Android, además de a los sitios web para móviles. La funcionalidad más utilizada es el registro en la aplicación a través de la cuenta de Facebook, de manera similar a los

sitios web. Otras redes sociales como Pinterest o Timpik permiten al usuario iniciar sesión con Facebook en sus aplicaciones móviles.

- **Aplicaciones en Facebook:** Son aplicaciones completamente integradas en Facebook, mostrándose en una zona delimitada dentro de Facebook conocida como *Canvas*. Es el tipo de aplicación que se va a desarrollar en este proyecto, por lo que vamos a profundizar y analizar todas las herramientas de integración que nos ofrece.

## 3.2 Aplicaciones en Facebook

Como hemos mencionado anteriormente, las aplicaciones integradas en Facebook nos ofrecen la posibilidad de integrar muchos aspectos de la red social en nuestra aplicación, como publicaciones o notificaciones a usuarios. Están a disposición del desarrollador todas las herramientas que ofrece Facebook como los Plugins Sociales, la Graph API y los diálogos con el usuario. Veremos con más profundidad en que consiste cada uno, junto a otras herramientas, en la siguiente sección.

Una aplicación en Facebook consta de tres conceptos básicos: la página canvas, los canales sociales y las estadísticas de la aplicación.

La **página canvas** es el contexto en el que son cargadas las aplicaciones de Facebook. Las aplicaciones son construidas en cualquier lenguaje que soporte la programación web, como pueden ser PHP, Java, Python, etc. Las aplicaciones se alojan en un servidor externo, y se indica a Facebook la dirección http desde la cual deben cargar la aplicación en el canvas.

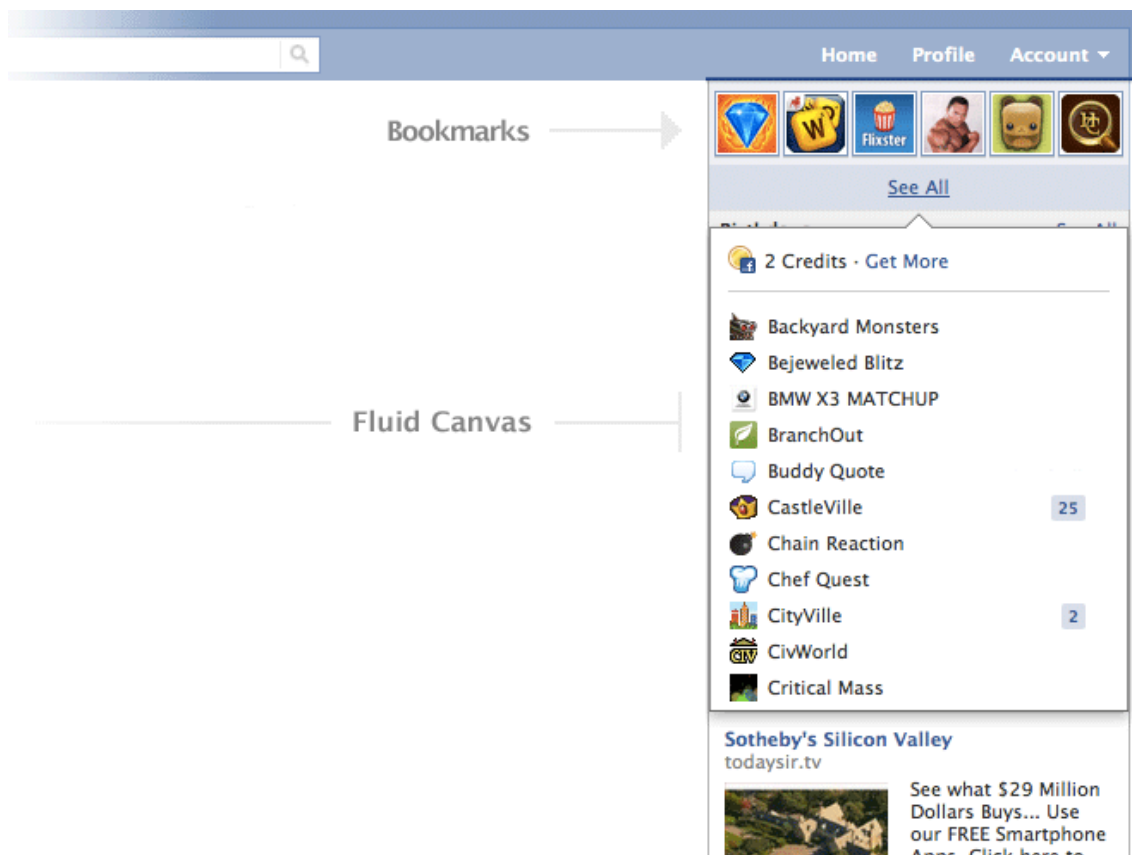


Figura 7 Canvas de una aplicación Facebook

La dirección se carga en un iframe dentro de la misma página, utilizando su propia hoja de estilos y código JavaScript. Por defecto, la anchura de la página canvas es de 760px, pero se puede indicar que la obtenga dinámica, en la cual la aplicación quedará alineada a la izquierda y utilizará toda la anchura y altura de la que disponga el navegador del usuario.

Los **canales sociales** son la principal herramienta para conseguir más usuarios en nuestra aplicación. Facebook provee varios canales que permiten al usuario descubrir la aplicación gracias a la actividad que generan los usuarios de la aplicación. Los canales que nos ofrece Facebook son:

- **Bookmarks** (Marcadores): Permiten al usuario volver a utilizar una aplicación utilizada anteriormente. Cuando un usuario comienza a



utilizar una nueva aplicación en Facebook, son automáticamente creados. Aparecen en dos lugares: la columna izquierda de la página principal del usuario y en la parte superior derecha en el canvas de una aplicación. Disponen de un contador con el número de solicitudes pendientes que tiene el usuario en la aplicación, y ofrecen algunas funcionalidades como agregar aplicaciones a favoritos, editar ajustes referentes a esa aplicación o eliminarla de los marcadores.

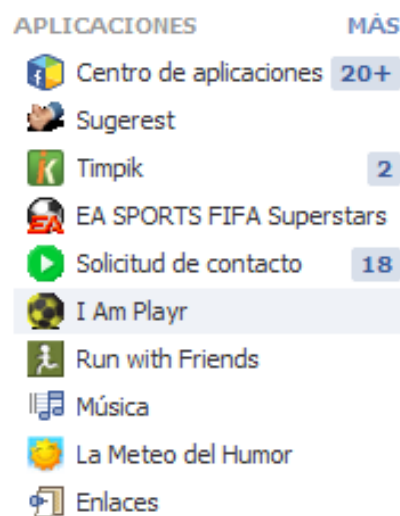


Figura 9 Marcadores en Facebook

- **Notifications** (Notificaciones): Se utilizan para notificar sobre cambios relevantes en el contenido de la aplicación, enviando una notificación a los usuarios para los cuales sea relevante ese cambio. Es también una manera de conseguir que los usuarios inviten a sus amigos a utilizar la aplicación. Los usuarios pueden notificar a sus amigos sobre algún cambio en la aplicación o indicarles que hagan una determinada acción (aceptar un regalo, ayudarles a completar un objetivo...). Esto motiva a

los usuarios amigos a comenzar a utilizar también la aplicación. Hay dos tipos de notificaciones que pueden ser enviadas desde la aplicación:

- **Generadas por el usuario:** Son confirmadas por el usuario mediante un cuadro de diálogo. Una vez confirmadas y enviadas, actualizan automáticamente el contador de notificaciones que hemos visto previamente en los marcadores.
  - **Generadas por la aplicación:** Son generadas y enviadas por la aplicación sin pedir confirmación del usuario. Para ello, el usuario debe haber autorizado previamente a la aplicación. Un uso común es enviar automáticamente una notificación a un usuario que ha pasado un tiempo inactivo en la aplicación a modo de recordatorio para conseguir que vuelva a utilizar la aplicación.
- **Newsfeed stories:** Se trata de historias publicadas en el feed del usuario, por lo que serán visibles para sus amigos. Se pueden distinguir los siguientes tipos:
    - **Discovery stories:** Facebook genera la historia cuando el usuario comienza a utilizar una nueva aplicación. Esta historia queda registrada en el feed del usuario, por lo que sus amigos la verán en novedades, permitiéndoles descubrir aplicaciones que sus amigos ya están usando. En los ajustes de la aplicación se puede configurar para que se generen por defecto o no.



Figura 10 Activar Social Discovery

- **Pulishing stories:** Además de las historias generadas por Facebook, la aplicación puede publicar historias sobre lo que ellos están haciendo en la aplicación. Por ejemplo, podría publicar en el feed de un amigo que ha recibido una nueva recomendación proveniente de otro amigo. Estas historias aparecen en el perfil del usuario y en el feed de sus amigos, incluyendo en ellas un enlace a la aplicación.
- **Búsqueda:** Cuando un usuario realiza una búsqueda en la barra de navegación superior, se muestran aplicaciones coincidentes con esa búsqueda. Para que una aplicación aparezca en el índice de resultados debe tener al menos 10 usuarios activos mensuales. Una vez alcanzada esta cifra, se pondrá en cola para ser añadida al índice. El índice se reconstruye cada 3 o 4 semanas, por lo que es posible que aunque la aplicación cumpla el requisito no aparezca hasta pasadas unas semanas.
- **Página:** Es posible crear también una página en Facebook asociada a la aplicación. Su creación se realiza a través de la página de administración de la aplicación. Esta página se diferencia de una página tradicional en algunos aspectos, como poseer el botón de “Ir a la aplicación”, además de algunos enlaces como pueden ser bloquear la aplicación, reportarla o contactar con el desarrollador.

Finalmente, las estadísticas de Facebook son ofrecidas por una herramienta llamada Insights. Nos permiten obtener una gran cantidad de información de la actividad de los usuarios en la aplicación. A través de esta herramienta se puede ver el número de usuarios autorizados, el número de

usuarios activos, las publicaciones que han enviado a través de la aplicación, etc. Además de ello, provee diagnóstico de errores en la aplicación en las llamadas a la API.

### 3.3 Graph API y permisos

El núcleo de Facebook es un grafo social en el que están interconectada la información de la red social. La Graph API presenta una manera muy simple y consistente de obtener información de este grafo social. La información se representa como objetos en el grafo (personas, fotos, eventos, etc.) y conexiones entre estos objetos (relaciones de amistad, fotos etiquetadas, etc.).

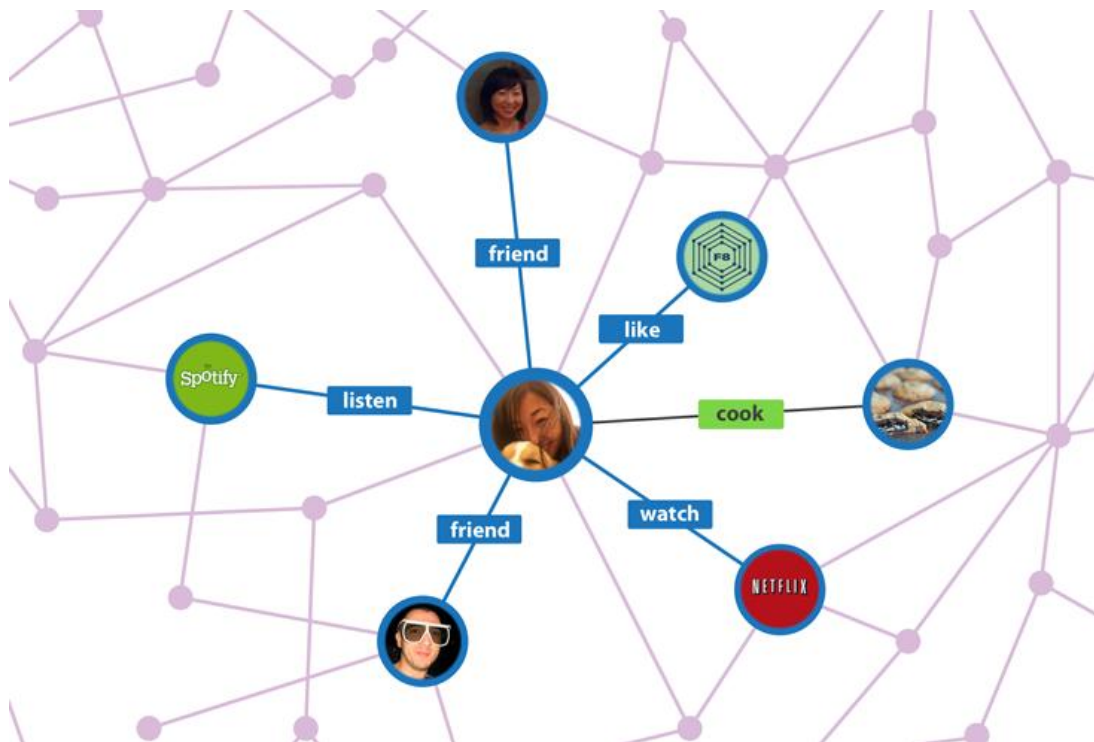


Figura 11 Graph API de Facebook

Todos los objetos tienen un identificador asociado, a través de él se puede acceder a las propiedades del objeto a través de una llamada HTTPS a la Graph API. Las llamadas a la API devuelve el conjunto de información del objeto solicitado en formato JSON. La estructura de una llamada a la Graph API sería la siguiente: <https://graph.facebook.com/ID>

Usuarios y páginas también pueden ser identificados por un nombre único además de por su identificador numérico. Por ejemplo, si realizamos una petición a la API del objeto de la página oficial de Coca Cola en Facebook nos devolvería el siguiente JSON con la información relativa a su página:

```
{
  "name": "Coca-Cola",
  "is_published": true,
  "website": "http://www.coca-cola.com",
  "username": "coca-cola",
  "founded": "1886",
  "description": "Created in 1886 in Atlanta, Georgia, by Dr.
John S. Pemberton, Coca-Cola was first offered as a fountain
beverage at Jacob's Pharmacy by mixing Coca-Cola syrup with
carbonated water. \n\nCoca-Cola was patented in 1887, registered
as a trademark in 1893 and by 1895 it was being sold in every
state and territory in the United States. In 1899, The Coca-Cola
Company began franchised bottling operations in the United
States. \n\nCoca-Cola might owe its origins to the United
States, but its popularity has made it truly universal. Today,
you can find Coca-Cola in virtually every part of the
world.\n\nCoca-Cola Page House Rules: http://CokeURL.com/q28a",
  "about": "The Coca-Cola Facebook Page is a collection of your
stories showing how people from around the world have helped
make Coke into what it is today.",
  "location": {
    "latitude": 19.2,
    "longitude": -96.1333
  },
  "checkins": 115,
  "talking_about_count": 1174115,
  "category": "Food/beverages",
  "id": "40796308305",
  "link": "https://www.facebook.com/coca-cola",
  "likes": 49177296,
  "cover": {
    "cover_id": "10151829640053306",
    "source": "http://a8.sphotos.ak.fbcdn.net/hphotos-ak-
ash3/s720x720/529413_10151829640053306_446360541_n.jpg",
    "offset_y": 0
  }
}
```

```
}  
}
```

Como hemos indicado anteriormente, los objetos están conectados unos con otros por medio de relaciones. La estructura de las llamadas es la siguiente: [https://graph.facebook.com/ID/TIPO\\_CONEXION](https://graph.facebook.com/ID/TIPO_CONEXION). El tipo de conexión puede ser, por ejemplo, friends (amigos) o music (música).

La Graph API permite acceder a toda la información pública de un objeto con las llamadas que hemos visto anteriormente. La información pública de un usuario serían datos como su nombre o imagen de perfil. Pero para obtener información adicional de un usuario, como podría ser sus amigos o sus grupos, es necesario obtener el denominado *access token*. Por tanto, si se quiere obtener información sobre el usuario, como su dirección de correo electrónico, es necesario pedir permiso previamente al usuario.

Lo ideal es solicitar al usuario únicamente permisos para información que vaya a utilizar la aplicación realmente, intentando reducirlos al máximo ya que el usuario puede ser reacio a conceder demasiados privilegios a la aplicación sobre su información personal. Por defecto, la aplicación solo puede leer la información básica del usuario. Para poder leer cualquier otra información o realizar publicaciones, debemos solicitar que el usuario conceda permisos a nuestra aplicación.

La información básica contiene la siguiente información: identificador, nombre, foto, género e idioma. Si el usuario ha especificado alguna información pública, como su lista de amigos, también será posible acceder a ella sin necesidad de permisos adicionales.

Podemos distinguir dos tipos de permisos:

- **Permisos básicos:** No son revocables por el usuario. Se solicitan en la primera página del cuadro de dialogo en la autenticación. Si es la primera autenticación del usuario en la aplicación, debe conceder esos permisos para poder continuar y autorizar a la aplicación.
- **Permisos extendidos:** Son presentados al usuario en la segunda página del cuadro de diálogo de autenticación. Se pueden conceder o rechazar individualmente cada uno de ellos. Es por tanto que la aplicación no debería dar por supuesto que el usuario ha aceptado todos los permisos. Con aceptar los permisos básicos el usuario ya podrá acceder a la aplicación, por lo que hay que comprobar si ha aceptado los permisos extendidos.

Una vez vistos ambos tipos de permisos, vamos a describir cada permiso. La primera columna solicita acceder a la propia información del usuario de la aplicación; mientras que la segunda columna el permiso autoriza a acceder a la información de mis amigos.

La lista de permisos básicos es la siguiente:

<b>Usuario</b>	<b>Amigos</b>	<b>Descripción</b>
user_about_me	friends_about_me	Da acceso a la sección Sobre mí del perfil del usuario
user_activities	friends_activities	Provee acceso a la lista de actividades del usuario
user_birthday	friends_birthday	Provee acceso al cumpleaños del usuario
user_checkins	friends_checkins	Da acceso de lectura a los checkins que ha hecho el usuario
user_education_history	friends_education_history	Da acceso a la educación del usuario.
user_events	friends_events	Da acceso a la lista de

---

		eventos a los que ha sido invitado el usuario
user_groups	friends_groups	Da acceso a la lista de grupos de los que forma parte el usuario
user_hometown	friends_hometown	Da acceso al lugar de nacimiento del usuario
user_interests	friends_interests	Da acceso a la lista de intereses del usuario
user_likes	friends_likes	Da acceso a las paginas que el usuario ha indicado que le gustan
user_location	friends_location	Da acceso a la actual ubicación del usuario
user_notes	friends_notes	Da acceso a las notas del usuario
user_photos	friends_photos	Da acceso a las fotos que ha subido el usuario y a las que ha sido etiquetado
user_questions	friends_questions	Da acceso a las preguntas que ha respondida
user_relationships	friends_relationships	Provee acceso a las relaciones familiares y personales del usuario
user_relationship_details	friends_relationship_details	Provee acceso a las preferencias de las relaciones del usuario
user_religion_politics	friends_religion_politics	Provee acceso a la información religiosa y política del usuario
user_status	friends_status	Provee acceso a los mensajes de estado del usuario
user_subscriptions	friends_subscriptions	Da acceso a los subscriptores y suscripciones del usuario
user_videos	friends_videos	Da acceso a los videos que ha subido el usuario o en los que ha sido etiquetado

---



user_website	friends_website	Provee acceso a los sitios webs del usuario
user_work_history	friends_work_history	Da acceso al historial de trabajos del usuario
email	N/A	Provee acceso al email del usuario.

**Tabla 2 Permisos básicos en Facebook**

La lista de permisos extendidos es:

<b>Permiso</b>	<b>Descripcion</b>
read_friendlists	Da acceso a la lista personalizada de amigos que haya creado el usuario.
read_insights	Da acceso a leer las estadísticas de Insight de una página o grupo.
read_mailbox	Da acceso a leer los mensajes de entrada del usuario
read_requests	Permite leer las solicitudes de amistad que tiene el usuario
read_stream	Da acceso a la lista de entradas en el feed del usuario y permite realizar búsquedas de entradas.
xmpp_login	Proporciona a las aplicaciones que se integran con el chat de Facebook la posibilidad de acceder los usuarios.
ads_management	Ofrece la posibilidad de administrar los anuncios y llamar a la API de Facebook Ads en nombre de un usuario.
create_event	Permite a la aplicación crear y modificar eventos pertenecientes al usuario
manage_friendlists	Permite a la aplicación editar o crear listas de amigos del usuario
manage_notifications	Permite leer las notificaciones del usuario y marcarlas como leídas.
user_online_presence	Permite a la aplicación conocer si el usuario está conectado o desconectado.
friends_online_presence	Permite a la aplicación conocer si los amigos del usuario están conectados o desconectados.
publish_checkins	Permite a la aplicación hacer checkins en nombre del usuario.
publish_stream	Permite a la aplicación para publicar contenido, comentarios y “me gusta” en la

---

	biografía del usuario y la de sus amigos.
rsvp_event	Permite a la aplicación confirmar asistencia a eventos en nombre del usuario.

---

**Tabla 3** Permisos extendidos en Facebook

### 3.4 PHP SDK

El SDK de Facebook para PHP nos provee de un conjunto de funcionalidades del lado de servidor para interactuar con la API de Facebook. Puede utilizarse conjuntamente con el SDK de Javascript, utilizado normalmente en el cliente.

Para utilizar el SDK basta con incluirlo en el fichero PHP en el que se quiera utilizar e inicializarlo:

```
require_once("facebook.php");  
  
$config = array();  
  
$config['appId'] = 'IDENTIFICADOR DE APLICACIÓN';  
  
$config['secret'] = 'CÓDIGO SECRETO DE APLICACIÓN';  
  
$facebook = new Facebook($config);
```

Una vez inicializado el objeto Facebook, podremos realizar llamadas a este objeto para comunicarnos con Facebook. Los métodos más importantes de este objeto son:

- **getUser:** El método devuelve el identificador de Facebook asociado al usuario que está actualmente logueado, o devuelve o si no hay ninguno conectado.
- **getLoginUrl:** Genera un enlace en el que el usuario puede pulsar para iniciar sesión en Facebook, aceptar los permisos de la aplicación si es necesario y ser redirigido nuevamente a la aplicación
- **getLogoutUrl:** Genera un enlace a través del cual el usuario puede pulsar y cerrar sesión en Facebook, para ser redirigido después de regreso a la aplicación
- **getLoginStatusUrl:** Genera un enlace en función del estado actual del usuario (si está conectado para desconectarse, o si está desconectado para conectarse)
- **Api:** Es quizás el método más importante o utilizado del SDK de PHP. Permite realizar una llamada a la Graph API de Facebook. Previamente es recomendable comprobar que existe un usuario con la sesión iniciada en la aplicación. Para obtener la información personal del usuario, bastaría con realizar la consulta:

```
$facebook->api('/me','GET');
```

Para publicar en el muro del usuario desde la aplicación:

```
$facebook->api('/me/feed', 'POST',  
  
array(  
  
    'message' => 'Texto del mensaje'  
  
));
```

### 3.5 Registrando una aplicación

Para poder integrar una aplicación con Facebook, tenemos que registrarla previamente en Facebook Developers. Para ello, desde la página principal de Facebook Developers podremos acceder a la pestaña aplicaciones del menú superior izquierdo:

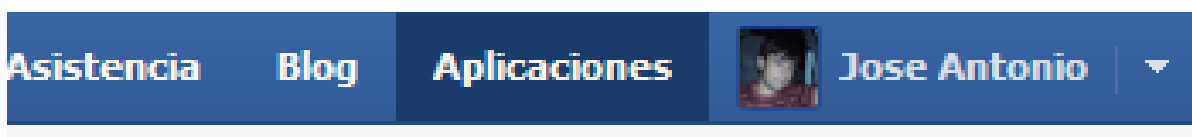


Figura 12 Menú de Facebook Developers

En esta pantalla se mostrarán todas las aplicaciones que hayamos creado o de las que nos hayan dado permiso de administración. Para crear una nueva aplicación pulsamos en el botón superior derecho “+ Create New App”.

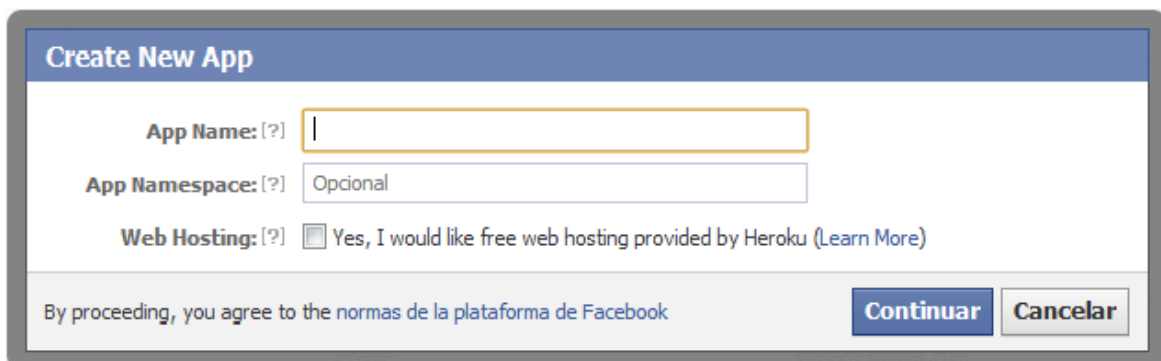


Figura 13 Crear nueva aplicación

A continuación se nos mostrara el formulario de creación de una nueva aplicación, los campos que hay que completar son:

- **App Name:** Es el nombre de la aplicación, que debe ser una cadena comprendida entre los 3 y los 32 caracteres. No se puede utilizar ninguna marca registrada u otros derechos de un tercero, ya que en ese caso Facebook podría eliminar la aplicación

- **App Namespace:** Es utilizado para la URL que genera Facebook para la aplicación (<http://apps.facebook.com/namespace>). Además, también se utiliza para definir las acciones y objetos personalizados en Open Graph. Solo puede contener letras minúsculas, guiones y guiones bajos.
- **Web Hosting:** En caso de activar esta casilla, se obtendrá un alojamiento gratuito para la aplicación de parte de Heroku, un servicio externo a Facebook que provee de alojamiento gratuito para las aplicaciones en la red social.



The image shows a screenshot of the Facebook 'Create New App' form. The form has a blue header with the text 'Create New App'. Below the header, there are three input fields: 'App Name: [?]' with a text input box, 'App Namespace: [?]' with a text input box containing the word 'Opcional', and 'Web Hosting: [?]' with a checkbox and the text 'Yes, I would like free web hosting provided by Heroku (Learn More)'. At the bottom of the form, there is a line of text: 'By proceeding, you agree to the normas de la plataforma de Facebook'. To the right of this text are two buttons: 'Continuar' (highlighted in blue) and 'Cancelar'.

Figura 14 Formulario de nueva aplicación

Una vez creada la aplicación, se nos mostrará la pantalla principal de para la edición de la misma. Como hemos visto, solo hemos introducido los datos básicos de la aplicación: Nombre e identificador en el espacio de nombres de aplicaciones.

Aplicaciones > Aplicación de prueba > Básica



The screenshot shows the configuration page for a Facebook application named 'Aplicación de prueba'. At the top, there is a logo placeholder and the application name. Below this, the 'App ID' is 410697405653223 and the 'App Secret' is 7c741a1826593a670e6028a58a8c85ba, with a '(reiniciar)' link. There is an '(edit icon)' link next to a small square icon. The main section is titled 'Información básica' and contains several fields: 'Display Name' (Aplicación de prueba), 'Namespace' (appdeprueba-pfc), 'Contact Email' (app@prueba.com), 'App Domains' (a text input with a placeholder 'Enter your site domains and press enter'), 'Category' (a dropdown menu set to 'Otra' with a 'Choose a sub-category' dropdown next to it), 'Hosting URL' (a text input with a placeholder 'You have not generated a URL through one of our partners (Get one)'), and 'Sandbox Mode' (radio buttons for 'Activado' and 'Desactivado', with 'Desactivado' selected).

Figura 15 Configuración de la aplicación

Por tanto, tendremos que editar la aplicación para decir a Facebook el tipo de aplicación, el logo, permisos necesarios, etc.

Un dato importante a observar es el identificador que ha generado a la aplicación (APP ID) y el token secreto de la misma (APP Secret). Estos datos nos servirán para después realizar llamadas a la API de Facebook desde el código de nuestra aplicación, ya que es la manera de verificar que el código ejecutado es realmente el de la aplicación.

En la parte superior, se puede editar tanto la imagen de la aplicación como el icono que aparecerá en el navegador mientras usamos la aplicación en Facebook.

En la parte inferior se escoge como se integra la aplicación con Facebook, es decir, que tipo de aplicación será realmente.



Figura 16 Tipos de integración con Facebook

Las opciones son:

- **Website with Facebook Login:** Se utiliza para integrar el login con la cuenta de Facebook en cualquier sitio web tradicional.
- **App on Facebook:** Se utiliza la aplicación dentro de Facebook.com.
- **Mobile web:** Añade un marcador a la aplicación móvili desde Facebook mobile.
- **Native iOS App:** Permite publicar desde una aplicación nativa de iOS hacia Facebook.
- **Native Android App:** Permite publicar desde una aplicación nativa de Android hacia Facebook.
- **Pestaña de la página:** Genera una pestaña personalizada en una página de Facebook.

Para la aplicación de este proyecto, se va a utilizar *App on Facebook*, ya que nuestra aplicación estará totalmente integrada y solo se podrá utilizar desde el interior de la red social.

Website with Facebook Login Log in to my website using Facebook.

App on Facebook

Canvas Page: [?] http://apps.facebook.com/appdeprueba-pfc

Canvas URL: [?]

Secure Canvas URL: [?]

Canvas Width: [?]  Fluid  Fixed (760px)

Canvas Height: [?]  Fluid  Fixed at  px

Mobile Web Bookmark my web app on Facebook mobile.

Figura 17 Configuración del tipo Aplicación en Facebook

Al seleccionar esta opción, nos aparece un formulario en el que tendremos que introducir la URL desde la cual Facebook cargará nuestra aplicación. Además, desde el 1 de octubre de 2011, es obligatorio introducir también una dirección segura, bajo el protocolo https. También se puede especificar si el ancho y el alto serán dinámicos (*Fluid*) o estarán ajustados a un tamaño fijo.

Por último, vamos a definir los permisos que solicita la aplicación. En el menú izquierdo, dentro de configuración se escoge la opción permisos.



**Configuring Permissions**  
Use these settings to enter the set of permissions your app requires when displayed in Centro de aplicaciones or using Auth Referrals  
[Aprende más sobre Configuring Permissions](#)

**Configure your permissions** [Preview Auth Dialog](#)

Default Activity Privacy: [?] \* Ninguna (opción predet... ▼

User & Friend Permissions: [?] Select permissions to request from users

Extended Permissions: [?] Select permissions to request from users

Auth Token Parameter: [?] Query String (?code=...) ▼

Authenticated Referrals: [?]  Activado  Desactivado

[Guardar cambios](#)

**Figura 18** Selección de permisos de la aplicación

En *user & friend permissions* se escogen los permisos básicos que va a solicitar la aplicación. En *Extended permissions* se introducen los permisos extendidos que necesita, recordando que estos pueden ser revocados individualmente por el usuario.

Hay muchas más opciones a la hora de configurar la aplicación en Facebook, pero no se va a profundizar más debido a que se ha visto lo necesario para la configuración de esta aplicación.



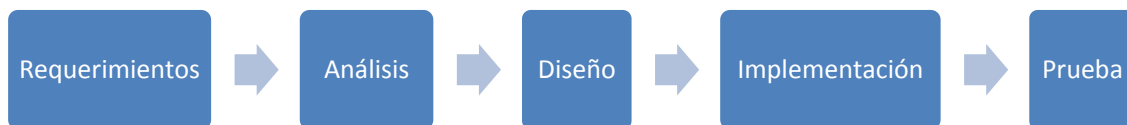
# **4. Procesos de Ingeniería del Software**

---

## 4.1 Introducción

Las actividades que conforman la Ingeniería del Software son las siguientes:

- **Especificación de Requerimientos:** se obtienen el propósito del sistema y las propiedades y restricciones del mismo.
- **Análisis del sistema:** se obtiene un modelo del sistema correcto, completo, consistente, claro y verificable.
- **Diseño del Sistema:** se definen los objetivos del proyecto y las estrategias a seguir para conseguirlos.
- **Implementación:** se traduce el modelo a código fuente.
- **Prueba:** verificar y validar el sistema.



En los puntos siguientes se profundizará en cada una de estas actividades y en como se han llevado a cabo en el ámbito nuestro proyecto.

## 4.2 Análisis de requisitos

En este apartado expondremos los requisitos que se deben tener en cuenta para desarrollar la aplicación web integrada en la red social Facebook.

#### **4.1.1 Requisitos funcionales**

Los requisitos funcionales son las tareas que la aplicación debe realizar. Definen el funcionamiento del sistema. Se define el siguiente conjunto de requisitos funcionales:

- **RF1: Alta de usuarios:** Un usuario debe poder tanto darse de alta aceptando ciertos permisos para autorizar a la aplicación a utilizar su información de Facebook a la aplicación y dar permiso realizar publicaciones en el muro del usuario.
- **RF2: Iniciar sesión en la aplicación:** Un usuario, el cual ha sido previamente registrado, puede iniciar sesión en la aplicación desde su cuenta de Facebook.
- **RF3: Enviar una recomendación:** Un usuario debe poder enviar una recomendación a sus amigos de la red social, con independencia de que estén registrados o no en la aplicación.
- **RF4: Recibir recomendaciones:** Un usuario debe poder recibir las recomendaciones enviadas por sus amigos y ser notificado por ello.
- **RF5: Descartar recomendaciones:** Un usuario debe poder descartar una recomendación recibida.
- **RF6: Valorar recomendación:** Un usuario debe poder valorar en una escala de uno a cinco el ítem que ha sido recomendado donde cinco es mayor satisfacción y uno es menor satisfacción.
- **RF7: Modificar valoración:** Un usuario debe poder modificar la valoración inicial dada a una recomendación por parte de un usuario.
- **RF8: Consultar ayuda:** Un usuario debe poder consultar ayuda acerca del funcionamiento de la aplicación

- **RF9: Ver afinidad global con usuarios:** Un usuario debe poder ver la afinidad global que tiene con sus amigos en función de la información almacenada de recomendaciones pasadas.
- **RF10: Ver afinidad específica con usuarios:** Un usuario debe poder ver la afinidad concreta de una categoría que tiene con sus amigos en función de la información almacenada de recomendaciones pasadas.
- **RF11: Calcular relevancia:** El sistema debe poder calcular la relevancia que tendrá una recomendación para un usuario en función de varios factores como la afinidad del usuario con su amigo o la valoración que éste último le dio al ítem enviado.
- **RF12: Ver listado de recomendaciones recibidas:** Un usuario debe poder ver el listado de las recomendaciones que ha recibido, ordenado por la relevancia calculada por el sistema de que serán de su agrado.
- **RF13: Ver listado de recomendaciones enviadas:** Un usuario debe poder ver el listado de las recomendaciones que ha enviado a sus amigos y están pendientes de valorar por ellos.
- **RF14: Consultar historial de recomendaciones:** Un usuario debe poder consultar el historial completo de recomendaciones enviadas y recibidas que ya fueron valoradas.
- **RF15: Invitar amigos:** Un usuario debe poder enviar invitaciones a sus amigos en la red social para que comiencen a utilizar la aplicación.
- **RF16: Publicar en la biografía:** La aplicación debe publicar en la biografía de los usuarios cuando reciban una nueva recomendación y cuando se den de alta en la aplicación.

- **RF17: Ver perfil de un amigo.** La aplicación debe poder mostrar el perfil de un amigo en la aplicación, mostrando información sobre afinidad e intercambios de recomendaciones entre el usuario y su amigo.
- **RF18: Contactar con el administrador.** El usuario debe poder contactar con el administrador de la aplicación, a través de un formulario en el que pueda indicar fallos, sugerencias o consultas sobre la misma

#### ***4.1.2 Requisitos no funcionales***

Los requerimientos no funcionales son las restricciones para las funcionalidades ofrecidas por el sistema. Por ejemplo: fiabilidad, portabilidad, usabilidad, tiempo de respuesta, etc.

- **Arquitectura:**
  - La arquitectura de la aplicación es el modelo cliente/servidor.
  - La aplicación debe ser una aplicación web y el usuario debe poder acceder a ella a través de un navegador.
  - La aplicación debe cumplir con los estándares web y su funcionamiento aplicación debe ser independiente del navegador utilizado
- **Requisitos del servidor:**
  - **Velocidad:** el servidor debe tener una capacidad de procesamiento suficiente para la ejecución de la aplicación en un tiempo de respuesta aceptable y de una manera fiable
  - **Memoria:** el servidor debe tener la suficiente memoria RAM para la ejecución de la aplicación.
  - **Almacenamiento:** la base de datos debe tener una capacidad suficiente para que la aplicación sea escalable.

- **Facilidad de uso:**
  - El usuario debe poder familiarizarse con la aplicación de una manera intuitiva.
  - El usuario debe poder predecir el siguiente estado de la aplicación tras la realización de una acción.
  - La interfaz debe ser consistente, para la realización de una tarea se deben seguir los mismos pasos.
- **Disponibilidad:**
  - La aplicación debe estar accesible a través de Internet
  - La aplicación debe estar disponible las 24 horas y los 7 días de la semana.
- **Seguridad:**
  - Solo podrán acceder a la aplicación los usuarios previamente registrados. Para que un usuario pueda hacer uso de la aplicación debe aceptar los permisos que requiere la aplicación solicitados por Facebook.

## 4.3 Análisis del sistema

### 4.3.1 Casos de uso

Durante la fase de análisis identificamos las necesidades de usuarios y clientes como requisitos. Los requisitos funcionales se expresan como casos de usos (Ivar Jacobson, El proceso Unificado de Desarrollo de Software, 2001) y los demás requisitos o bien se adjuntan a los casos de uso a los que afectan, o bien se guardan en una lista aparte o se describen de alguna otra forma.



Un caso de uso especifica una secuencia de acciones, incluyendo variantes, que el sistema puede llevar a cabo, y que producen un resultado observable de valor para un actor concreto.

Los casos de uso describen como se realiza una tarea de manera exacta.

Por lo tanto, es necesario determinar quiénes son los actores participantes en cada uno de los casos de uso. Un actor modela una entidad externa que se comunica con el sistema, es decir, es un tipo de usuario del sistema. Un actor, al igual que un caso de uso, debe tener un nombre único y puede tener una descripción asociada.

En nuestro sistema contamos con los dos actores siguientes:

- **Usuario no registrado:** Se trata de un usuario que accede por primera vez a la aplicación y aún no se encuentra registrado en la base de datos.
- **Usuario registrado:** Se trata de un usuario que ya ha accedido en ocasiones anteriores a la aplicación y se encuentra registrado en la misma.
- **Facebook:** Se trata de la API de Facebook. Interactúa con el sistema recibiendo solicitudes de llamadas a la API o devolviendo información solicitada

Una vez definidos cuales van a ser los actores del sistema, es el momento de crear los distintos casos de uso. A la hora de realizar esta acción es importante que cada uno de los requerimientos funcionales ya definidos aparezca en al menos uno de los casos de uso, aunque puede haber nuevos casos de uso en los que no aparezcan ningún requerimiento, ya que estamos en una fase de refinamiento del sistema donde queremos construir un modelo detallado del mismo.

Un paso previo a la creación y descripción de los distintos casos de uso es la obtención de los diversos diagramas de casos de uso de nuestro sistema. El primero es un diagrama frontera es decir, un diagrama que describe completamente la funcionalidad de un sistema:

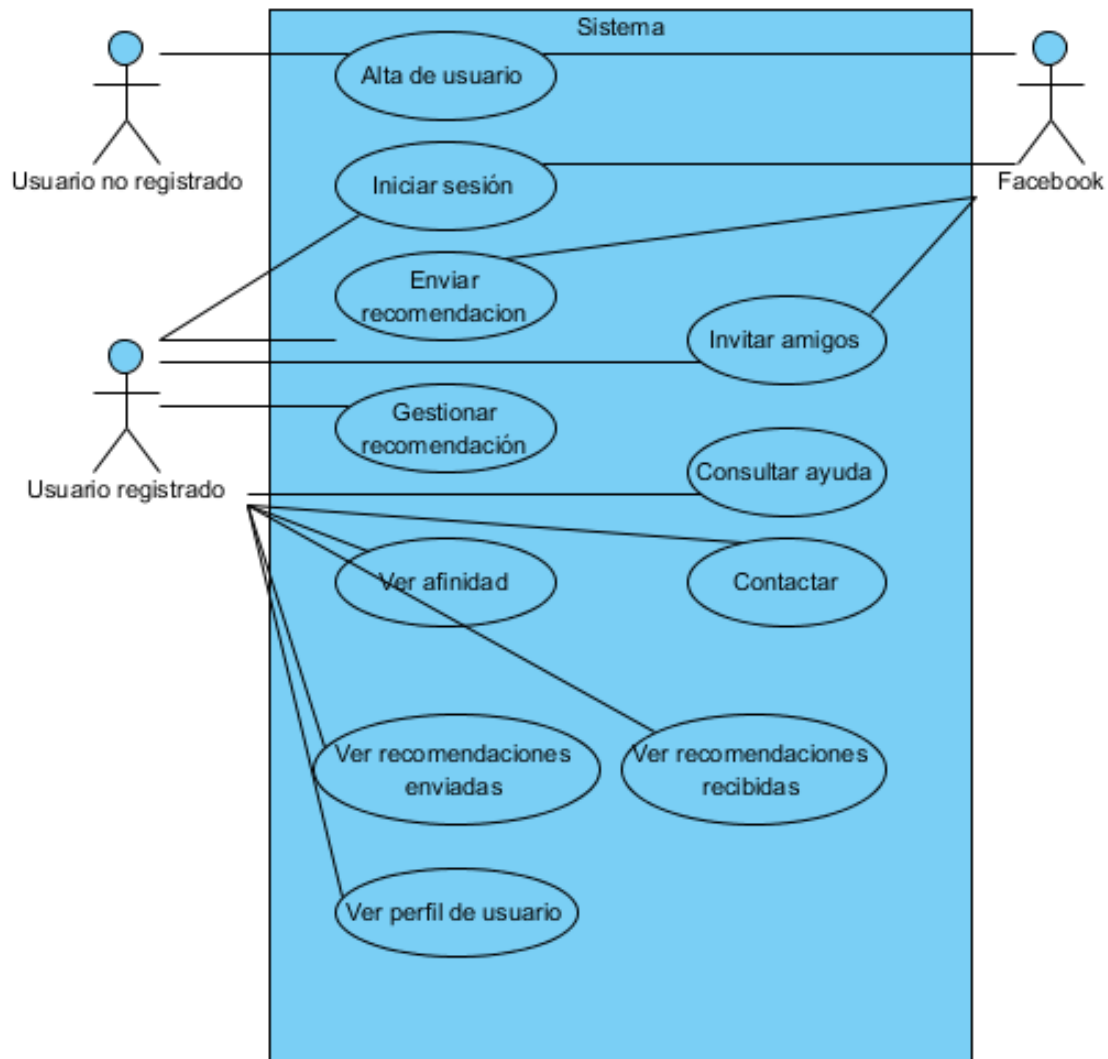


Figura 19 Diagrama frontera del sistema

Los casos de uso mostrados en un diagrama frontera pueden ser lo suficientemente exactos o, por el contrario, pueden ser concretados con un mayor detalle. A la hora de detallar un caso de uso se pueden emplear dos tipos de relaciones:

- **<<extend>>**: es una relación cuya dirección es hacia el caso de uso a detallar que representa comportamientos excepcionales del caso de uso. Se suele emplear para detallar excepciones
- **<<include>>**: es una relación cuya dirección es contraria a la de la relación <<extend>> que representa un comportamiento común del caso de uso.

Se van a detalla los casos de uso en los que es útil obtener un mayor nivel de detalle. El caso de uso Gestionar recomendación incluye otros subcasos, como se puede ver en el siguiente diagrama:

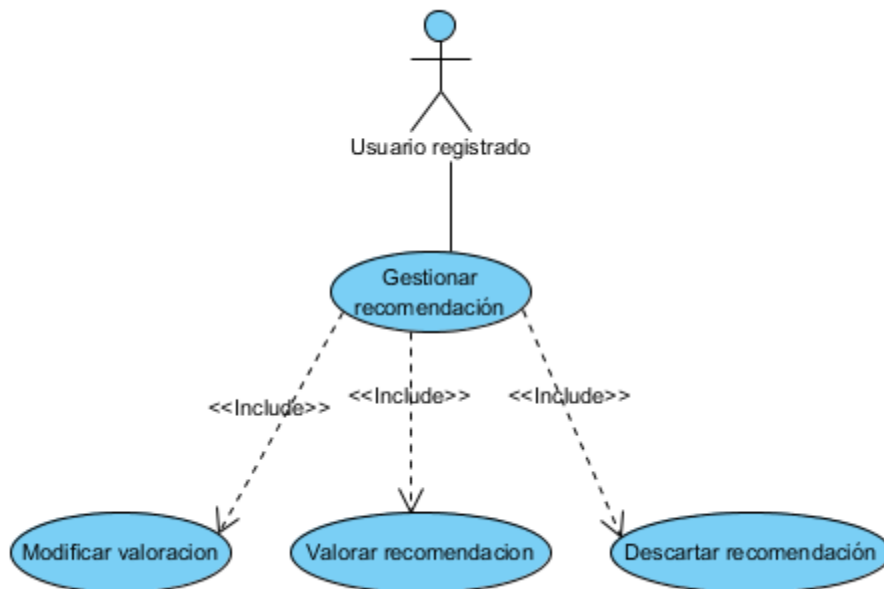


Figura 20 Caso de uso “Gestionar recomendación”

Lo mismo ocurre con el caso de uso de Ver afinidad, pudiendo tratarse de la afinidad global o de la específica:



Figura 21 Caso de uso "Ver Afinidad"

Además, otros casos de uso tienen excepciones que no han sido detalladas en el diagrama frontera. Por ejemplo, caso de uso *Enviar recomendación*:

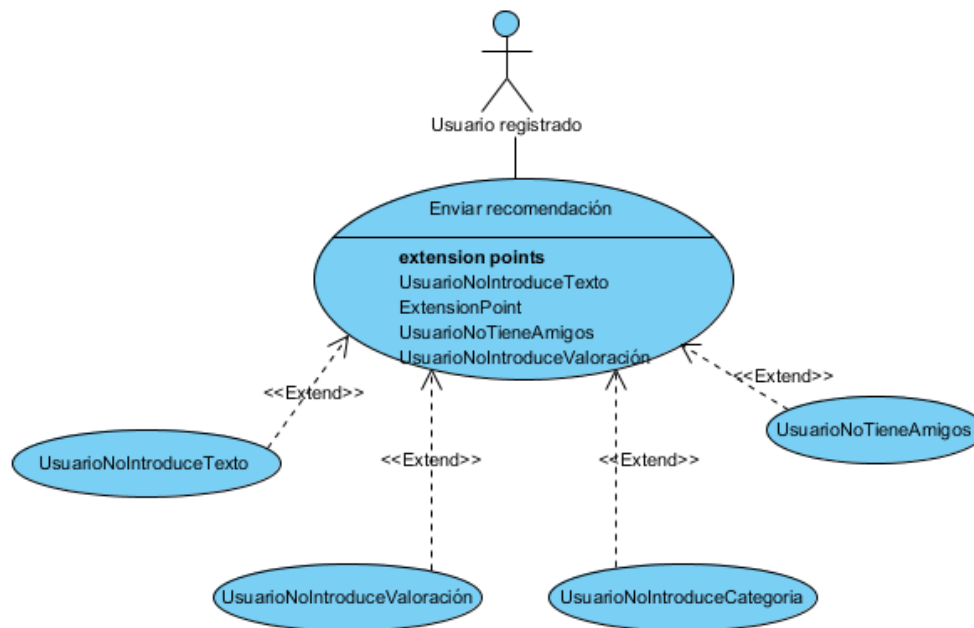


Figura 22 Caso de uso “Enviar recomendación”

Caso de uso *Ver afinidad*:



Figura 23 Caso de uso ver afinidad

Caso de uso *Alta de usuario*:

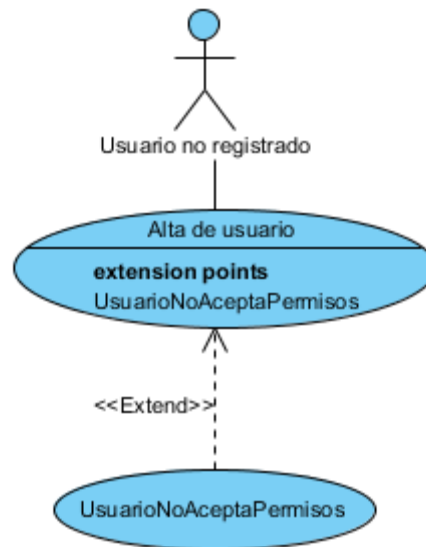


Figura 24 Caso de uso Alta de usuario

La narrativa de casos de uso (Martin Fowler, 1999) es la parte donde realmente quedan especificadas las tareas, funciones e interacciones del sistema. Los elementos mínimos la narrativa de casos de uso son:

- **Actores participantes:** Define que actores participan en ese caso de uso.
- **Condiciones de entrada o precondiciones:** Condiciones que deben darse antes de dispararse, aplicarse, o utilizarse este caso de uso.
- **Condiciones de salida o postcondiciones:** Esta sección indica el resultado tras el desarrollo de este caso de uso.
- **Flujo de eventos:** Describe la secuencia normal de pasos, o la más esperada.
- **Excepciones:** Durante el desarrollo de una tarea es muy probable que no siempre se siga el camino normal y, por tanto, debe especificarse que hará el sistema ante otro tipo de situaciones.

### **Caso de uso 1: Alta de usuario**

---

**Precondición:** El usuario proviene de Facebook y no está registrado en la aplicación.

**Postcondición:** El usuario queda registrado en la aplicación y puede acceder a toda la funcionalidad.

**Actores participantes:** Usuario no registrado.

**Flujo de ejecución:**

1. El usuario quiere acceder a la aplicación.
2. El sistema presenta el formulario con los permisos requeridos para acceder.
3. El usuario acepta otorgar a la aplicación todos los permisos solicitados (E-1)
4. El sistema registra al usuario en el sistema solicitando sus datos básicos a Facebook y almacenándolos en su propia base de datos. Busca los amigos del usuario que ya utilizan la aplicación y los añade como amigos.

**Excepciones:**

E-1 UsuarioNoAceptaPermisos: El usuario no otorga acceso a la aplicación. El sistema no registra al usuario y solicitará permisos de nuevo en caso de que vuelva a intentar acceder a ella.

### **Caso de uso 2: Iniciar sesión en la aplicación**

---

**Precondición:** El usuario debe estar previamente registrado en el sistema.

**Postcondición:** El usuario inicia sesión en el sistema y queda actualizada su información básica.

**Actores participantes:** Usuario registrado.

**Flujo de ejecución:**

1. El usuario accede a la aplicación desde Facebook.
2. El sistema obtiene la información básica y la actualiza en caso de que haya cambios.
3. El usuario inicia sesión en el sistema
4. El sistema presenta la página principal de la aplicación

### **Caso de uso 3: Enviar una recomendación**

---

**Precondición:** El usuario ha iniciado previamente sesión en el sistema

**Postcondición:** Se crea una nueva recomendación desde el usuario hacia el amigo seleccionado.

**Actores participantes:** Usuario registrado, Facebook.

**Flujo de ejecución:**

1. El usuario elige la opción de crear nueva recomendación



2. El sistema le presenta el formulario para que introduzca el texto de la recomendación, la valoración que el usuario le da y su categoría.
3. El usuario introduce el texto de la recomendación (E-1), su valoración (E-2) y selecciona la categoría (E-3).
4. El sistema le muestra la lista de amigos a los que enviar la recomendación.
5. El usuario elige un amigo de su lista de amigos (E-4) y escoge enviar recomendación.
6. El sistema genera una recomendación y una notificación desde ese usuario hacia el amigo seleccionado.
7. Facebook publica en la biografía del usuario una historia para informarle de que ha recibido una nueva recomendación.
8. El usuario obtiene confirmación visual de que la recomendación ha sido enviada.

**Excepciones:**

E-1 UsuarioNoIntroduceTexto: El usuario no introduce el texto de la recomendación. El sistema no le deja enviar la recomendación.

E-2 UsuarioNoIntroduceCategoría: El usuario no selecciona la categoría. El sistema no le deja enviar la recomendación.

E-3 UsuarioNoIntroduceValoración: El usuario no introduce valoración al ítem que va a recomendar. El sistema no le deja enviar la recomendación.

E-4 UsuarioNoTieneAmigos: El usuario no tiene amigos registrados en la aplicación. El sistema le informa de la situación y ofrece la opción de invitar amigos.

#### **Caso de uso 4: Ver recomendaciones recibidas**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación

**Postcondición:** El sistema muestra las recomendaciones recibidas que aún no ha valorado

**Actores participantes:** Usuario registrado.

**Flujo de ejecución:**

1. El usuario escoge la opción de ver las recomendaciones recibidas
2. El sistema selecciona las recomendaciones cuyo destino sea el usuario y que no hayan sido valoradas. Para cada recomendación, realiza el cálculo de la afinidad.

#### **Caso de uso 5: Ver recomendaciones enviadas**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación

**Postcondición:** El sistema muestra las recomendaciones que ha enviado y aún no han sido valoradas por sus amigos.

**Actores participantes:** Usuario registrado.

**Flujo de ejecución:**

1. El usuario escoge la opción de ver las recomendaciones enviadas.

2. El sistema selecciona las recomendaciones cuyo origen sea el usuario y que no hayan sido valoradas. Para cada una, calcula la relevancia que tiene esa recomendación para el usuario destinatario.

### **Caso de uso 6: Descartar recomendación**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación y disponer al menos de una recomendación recibida sin valorar.

**Postcondición:** La recomendación es eliminada del usuario destino.

**Actores participantes:** Usuario registrado.

**Flujo de ejecución:**

1. El usuario escoge la opción de ver las recomendaciones recibidas.
2. El sistema selecciona las recomendaciones cuyo destino sea el usuario y que no hayan sido valoradas.
3. El usuario escoge la opción de descartar recomendación.
4. El sistema elimina al usuario como destinatario de la recomendación y genera una notificación al usuario origen de la recomendación para informarle.
5. El usuario recibe confirmación visual de la eliminación de la recomendación recibida.
6. El sistema decrementa la afinidad entre ambos usuarios.

### **Caso de uso 7: Valorar recomendación**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación y disponer al menos de una recomendación recibida sin valorar.

**Postcondición:** La recomendación queda valorada por el usuario y se actualiza la afinidad entre ambos usuarios.

**Actores participantes:** Usuario registrado.

**Flujo de ejecución:**

1. El usuario escoge la opción de ver las recomendaciones recibidas.
2. El sistema selecciona las recomendaciones cuyo destino sea el usuario y que no hayan sido valoradas.
3. El usuario escoge la valoración que desea asignar a la recomendación.
4. El sistema recoge la valoración. Actualiza el nivel de afinidad entre el usuario origen de la recomendación y el usuario destino tanto global como en la categoría del ítem valorado. Notifica al usuario origen de que el usuario destino ha valorado su recomendación.
5. El usuario recibe confirmación visual de que se ha efectuado la valoración.

### **Caso de uso 8: Modificar una valoración**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación y disponer al menos de una recomendación previamente valorada.

**Postcondición:** La valoración de la recomendación queda actualizada y se actualiza la afinidad entre ambos usuarios.

**Flujo de ejecución:**

1. El usuario escoge la opción de “Historial”
2. El sistema devuelve la lista de películas cuyo origen o destino sea el usuario y que hayan sido valoradas.
3. El usuario escoge la recomendación que desea modificar su valoración del historial de recomendaciones recibidas y selecciona la nueva valoración.
4. El sistema recoge la valoración. Acto seguido realiza el incremento o decremento de afinidad entre esos usuarios y para esa categoría. Añade una nueva notificación al usuario que envió la recomendación informando de la modificación de la valoración.
5. El usuario recibe confirmación visual de la modificación de la valoración.

**Caso de uso 9: Ver afinidad global con amigos**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación.

**Postcondición:** El sistema muestra la afinidad que tiene con cada amigo

**Actores participantes:** Usuario registrado.

**Flujo de ejecución:**

1. El usuario escoge la opción ver afinidad con amigos (E-1).
2. El sistema devuelve la lista de amigos ordenada por mayor grado de afinidad global.

**Excepciones:**

E-1 UsuarioNoTieneAmigos: El usuario no tiene amigos en la aplicación, mostrar opción de invitar amigos.

**Caso de uso 10: Ver afinidad con amigos en una categoría**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación.

**Postcondición:** El sistema muestra la afinidad que tiene con cada amigo

**Actores participantes:** Usuario registrado.

**Flujo de ejecución:**

1. El usuario escoge la opción ver afinidad con amigos (E-1).
2. El sistema devuelve la lista de amigos ordenada por mayor grado de afinidad global.
3. El usuario selecciona la categoría en la que desea ver la afinidad
4. El sistema devuelve la lista de amigos ordenada por mayor grado de afinidad en esa categoría concreta.

**Excepciones:**

E-1 UsuarioNoTieneAmigos: El usuario no tiene amigos en la aplicación, mostrar opción de invitar amigos

**Caso de uso 11: Consultar historial**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación.

**Postcondición:** El sistema muestra el historial de recomendaciones

**Actores participantes:** Usuario registrado.

**Flujo de ejecución:**

1. El usuario escoge la opción de “Historial”
2. El sistema devuelve la lista de películas cuyo origen o destino sea el usuario y que hayan sido valoradas.

### **Caso de uso 12: Invitar amigos**

---

**Precondición:** El usuario debe haber iniciado sesión en la aplicación.

**Postcondición:** Se envían solicitudes de invitación a Sugerest a los amigos del usuario.

**Actores participantes:** Usuario registrado, Facebook.

**Flujo de ejecución:**

1. El usuario escoge la opción de Invitar amigos
2. El sistema realiza la solicitud a Facebook para mostrar el diálogo con los amigos del usuario.
4. Facebook presenta el diálogo con los amigos del usuario que aún no están registrados en la aplicación.
5. El usuario selecciona los amigos a los que quiere enviar la invitación.
6. Facebook envía una solicitud de Sugerest a todos los amigos seleccionados por el usuario.

### **Caso de uso 13: Ver perfil de amigo**

**Precondición:** El usuario debe haber iniciado sesión en la aplicación.

**Postcondición:** Se muestra el perfil del amigo y se regresa al estado original de la aplicación

**Actores participantes:** Usuario registrado, Facebook.

#### **Flujo de ejecución:**

1. El usuario selecciona un amigo en las pantallas de actividad, recomendaciones enviadas, recomendaciones recibidas, historial o afinidad.
2. El sistema muestra una nueva ventana con el perfil del amigo y la información en común con el usuario, la afinidad global y la particular de cada categoría.
3. El usuario cierra la ventana del perfil.
4. El sistema regresa al estado anterior a la pulsación sobre el amigo.

#### **4.3.2 Escenarios**

La representación concreta de un caso de uso se realiza mediante la creación de uno o más escenarios asociados a ese caso de uso que nos permitan ver todas las interacciones entre el usuario y el sistema.

Los escenarios son historias no reales que describen un ejemplo posible de interacción entre el usuario y la aplicación. Aunque no sean reales, los personajes deben tener nombre y motivaciones de uso en la aplicación. Esto



ayuda a los diseñadores ya que es más fácil visualizar una situación concreta que una situación abstracta.

Un escenario está formado por los siguientes elementos:

- Un nombre único y unívoco
- Una descripción
- Los actores participantes
- El flujo de eventos

No se han diseñado un escenario para cada caso de uso, sino que se han escogido los más representativos de la funcionalidad del sistema.

**Nombre:** Enviar\_recomendación

**Descripción:** El usuario José envía una recomendación a su amigo Manuel

**Actor:** Usuario registrado

**Flujo de eventos:**

1. El usuario José accede a la aplicación
2. Facebook comprueba si el usuario ha aceptado los permisos de la aplicación
3. El sistema le muestra la pantalla de bienvenida
4. José elige la opción *Nueva recomendación*
5. El sistema muestra el formulario para enviar una recomendación

6. José introduce *Juego de tronos* en el campo de contenido
7. José selecciona la categoría Cine y TV
8. José valora con cuatro estrellas *Juego de tronos*
9. José selecciona a su amigo Manuel
10. El sistema habilita el botón de *Enviar recomendación*.
11. José pulsa en *Enviar recomendación*
12. El sistema recoge los datos del formulario
13. El sistema crea un nuevo ítem con el nombre y categoría introducido por José.
14. El sistema crea una nueva recomendación con origen José, destino Manuel y con el ítem recién creado.
15. El sistema genera una notificación hacia Manuel informando de que tiene una nueva recomendación
16. Facebook publica en la biografía de Manuel un mensaje para informarle de que José le ha recomendado algo en Sugerest.
17. El sistema muestra a José la lista de recomendaciones que ha enviado

**Nombre:** Enviar\_recomendación\_item\_existente

**Descripción:** El usuario José envía una recomendación a su amigo Manuel de un ítem previamente almacenado en base de datos

**Actor:** Usuario registrado

**Flujo de eventos:**

1. El usuario José accede a la aplicación
2. Facebook comprueba si el usuario ha aceptado los permisos de la aplicación
3. El sistema le muestra la pantalla de bienvenida
4. José elige la opción *Nueva recomendación*
5. El sistema muestra el formulario para enviar una recomendación
6. José comienza a introducir contenido y el sistema le sugiere *Juego de Tronos*.
7. José pulsa en el elemento sugerido.
8. El sistema selecciona la categoría Cine y TV asociada a ese ítem.
9. José valora con cuatro estrellas *Juego de tronos*
10. José selecciona a su amigo Manuel

11. El sistema habilita el botón de *Enviar recomendación*.
12. José pulsa en *Enviar recomendación*
13. El sistema recoge los datos del formulario una nueva recomendación con origen José, destino Manuel e ítem Juego de tronos.
14. El sistema genera una notificación hacia Manuel informando de que tiene una nueva recomendación
15. Facebook publica en la biografía de Manuel un mensaje para informarle de que José le ha recomendado algo en Sugerest.
16. El sistema muestra a José la lista de recomendaciones que ha enviado

**Nombre:** Descartar\_recomendacion

**Descripción:** El usuario José recibe una recomendación de su amigo Manuel pero decide descartarla.

**Actor:** Usuario registrado

**Flujo de eventos:**

1. El usuario José accede a la aplicación
2. Facebook comprueba si el usuario ha aceptado los permisos de la aplicación

3. El sistema le muestra la pantalla de bienvenida
4. José elige la opción *Recomendaciones recibidas*
5. El sistema le muestra la lista de recomendaciones que han enviado a José y que éste aún no ha valorado
6. José elige la recomendación que desea descartar y pulsa en descartar
7. El sistema elimina la recomendación de la lista del usuario y la pone en estado descartada
8. El sistema informa a Manuel de que José ha descartado su recomendación.
9. El sistema decrementa la afinidad entre José y Manuel.

**Nombre:** Valorar\_recomendacion

**Descripción:** El usuario José recibe una recomendación de su amigo Manuel y decide darle una valoración ya que le ha gustado el contenido.

**Actor:** Usuario registrado

**Flujo de eventos:**

1. El usuario José accede a la aplicación
2. Facebook comprueba si el usuario ha aceptado los permisos de la aplicación

3. El sistema le muestra la pantalla de bienvenida
4. José elige la opción *Recomendaciones recibidas*
5. El sistema le muestra la lista de recomendaciones que han enviado a José y que éste aún no ha valorado. Entre ellas se encuentra “Titanic”.
6. José elige valorar “Titanic” y escoge darle cuatro estrellas
7. El sistema almacena la valoración del usuario
8. El sistema actualiza la afinidad global entre José y Manuel y la afinidad entre ellos en la categoría “Cine y televisión”.
9. El sistema mueve la recomendación de la lista de Recomendaciones recibidas de José a su historial
10. El sistema informa a Manuel de que José ha valorado su recomendación.

**Nombre:** Alta\_de\_usuario

**Descripción:** El usuario José recibe una invitación a utilizar Sugerest de su amigo Manuel y decido entrar.

**Actor:** Usuario no registrado

**Flujo de eventos:**

1. El usuario José accede a la aplicación por primera vez
2. Facebook le muestra el diálogo de Ir a la aplicación en el que se aceptan los permisos básicos.
3. José acepta los permisos básicos.
4. Facebook muestra el dialogo de permisos extendidos
5. José acepta los permisos extendidos
6. El sistema pide a Facebook los datos básicos José (nombre, apellidos, imagen) y los almacena.
7. El sistema solicita los amigos de Facebook José que ya utilizan la aplicación.
8. Facebook devuelve Manuel y Pedro.
9. El sistema hace a José amigo de Manuel y de Pedro y establece la afinidad inicial en cero.
10. El sistema le muestra la pantalla de bienvenida a José informándole sobre los amigos que ya están utilizando Sugerest.

### 4.3.3 Modelo del dominio

Un modelo del dominio es una representación de las clases conceptuales del mundo real. Su objetivo es representar la realidad y no las clases software. Se trata de conocer bien el dominio del problema antes de modelar el dominio de la solución.

El modelo del dominio se utiliza como base para el posterior diseño de clases. Gráficamente el modelo del dominio es un diagrama de clases en el que se definen los objetos del dominio o clases conceptuales, los atributos de las clases y las relaciones entre las clases.

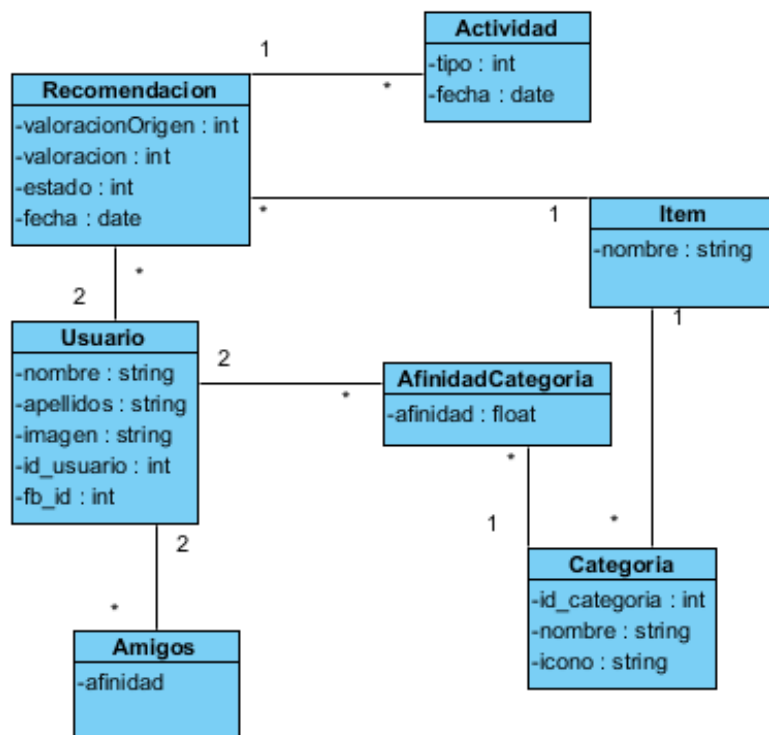


Figura 25 Modelo del dominio



## 4.4 Diseño del sistema

Una vez dejados especificados los requisitos que debe cumplir la aplicación y realizado el análisis, el siguiente paso en el proceso de Ingeniería del Software es el diseño de la misma.

Las dos fases de diseño que veremos a continuación son:

- **Diseño de clases:** se definen las clases que modelan la aplicación.
- **Diseño de los datos:** se define la estructura de los datos.
- **Diseño de la interfaz:** se define la apariencia visual de la aplicación.

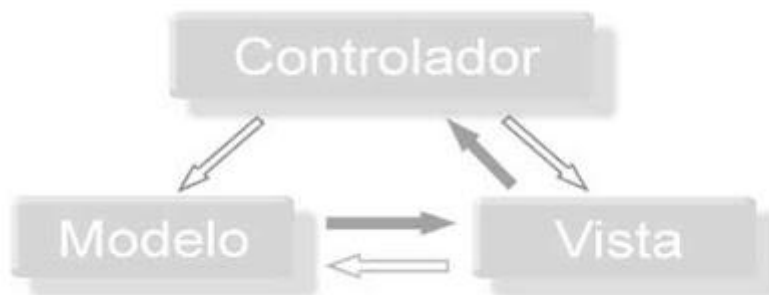
### 4.4.1 Diagrama de clases de diseño

El diagrama de clases de diseño (Ivar Jacobson, 2001) muestra gráficamente las especificaciones de las clases software en una aplicación. Mediante análisis del lenguaje natural obtendremos los distintos elementos del diagrama de clases:

- Clases
- Atributos de las clases
- Asociaciones entre clases
- Métodos de las clases
- Los tipos de datos de los atributos y los métodos.

A diferencia del modelo conceptual, este diagrama contiene la definición de las entidades software en lugar de conceptos del mundo real.

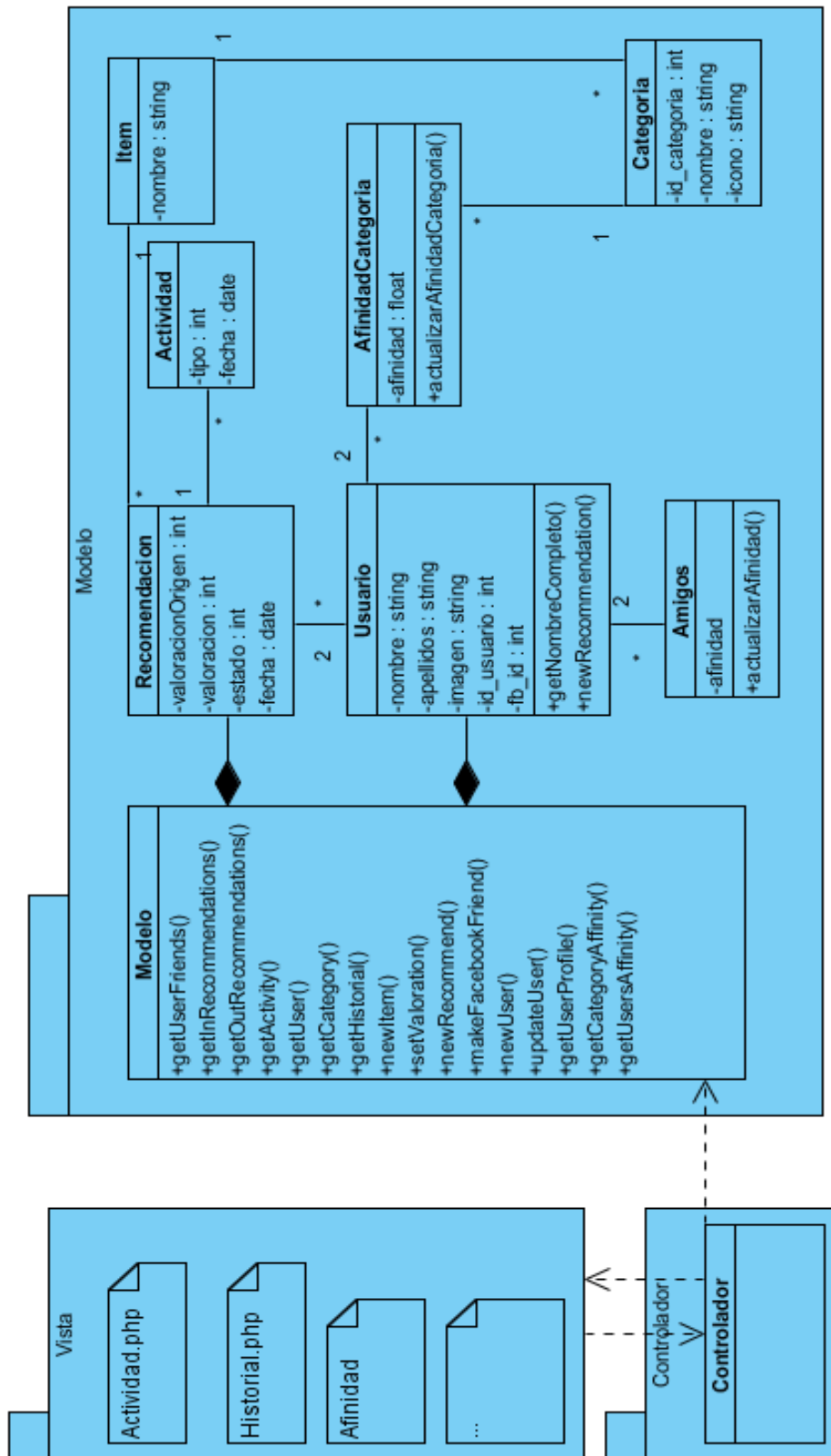
Se va a utilizar el patrón de diseño Modelo-Vista-Controlador. Este patrón separa el software en tres componentes: interfaz de usuario (Vista), datos de la aplicación (Modelo) y lógica de negocio (Controlador). Hay diferentes implementaciones de este patrón. En este proyecto se va a utilizar la de la Figura 26 Patrón Modelo-Vista-ControladorFigura 26.



**Figura 26 Patrón Modelo-Vista-Controlador**

La vista estará compuesta por el código HTML que compone la interfaz del usuario, que realizará llamadas al controlador. El modelo será el encargado de manejar las clases de la aplicación y su persistencia si es necesaria.

El diagrama de clases completo es el siguiente:



#### ***4.4.3 Diagramas de interacción***

Un diagrama de interacción (Martin Fowler, 1999) muestra en detalle el comportamiento del sistema en un caso de uso. Mientras que un diagrama de casos de uso muestra una visión externa del sistema, el diagrama de interacción modela como deben interactuar los objetos dentro de la aplicación.

Al contrario de lo que ocurre en el diagrama de clases de diseño, estos diagramas se utilizan para modelar los aspectos dinámicos de un sistema.

Existen dos tipos de diagramas de interacción y ambos describen información similar:

- **Diagramas de secuencia:** describen el comportamiento de la aplicación haciendo énfasis en los mensajes intercambiados por los objetos a través del tiempo.
- **Diagramas de comunicación:** modela las interacciones entre objetos en términos de mensajes en secuencia. Los diagramas de Comunicación representan una combinación de información tomada desde el diagrama de clases, secuencia, y diagrama de casos de uso describiendo tanto la estructura estática como el comportamiento dinámico de un sistema.

En este proyecto vamos a utilizar diagramas de secuencia, ya que muestran mejor el intercambio de mensajes a lo largo del tiempo. A continuación se presentan diagramas de secuencia de los casos de uso más complejos.

**Diagrama de secuencia de Nueva recomendación**

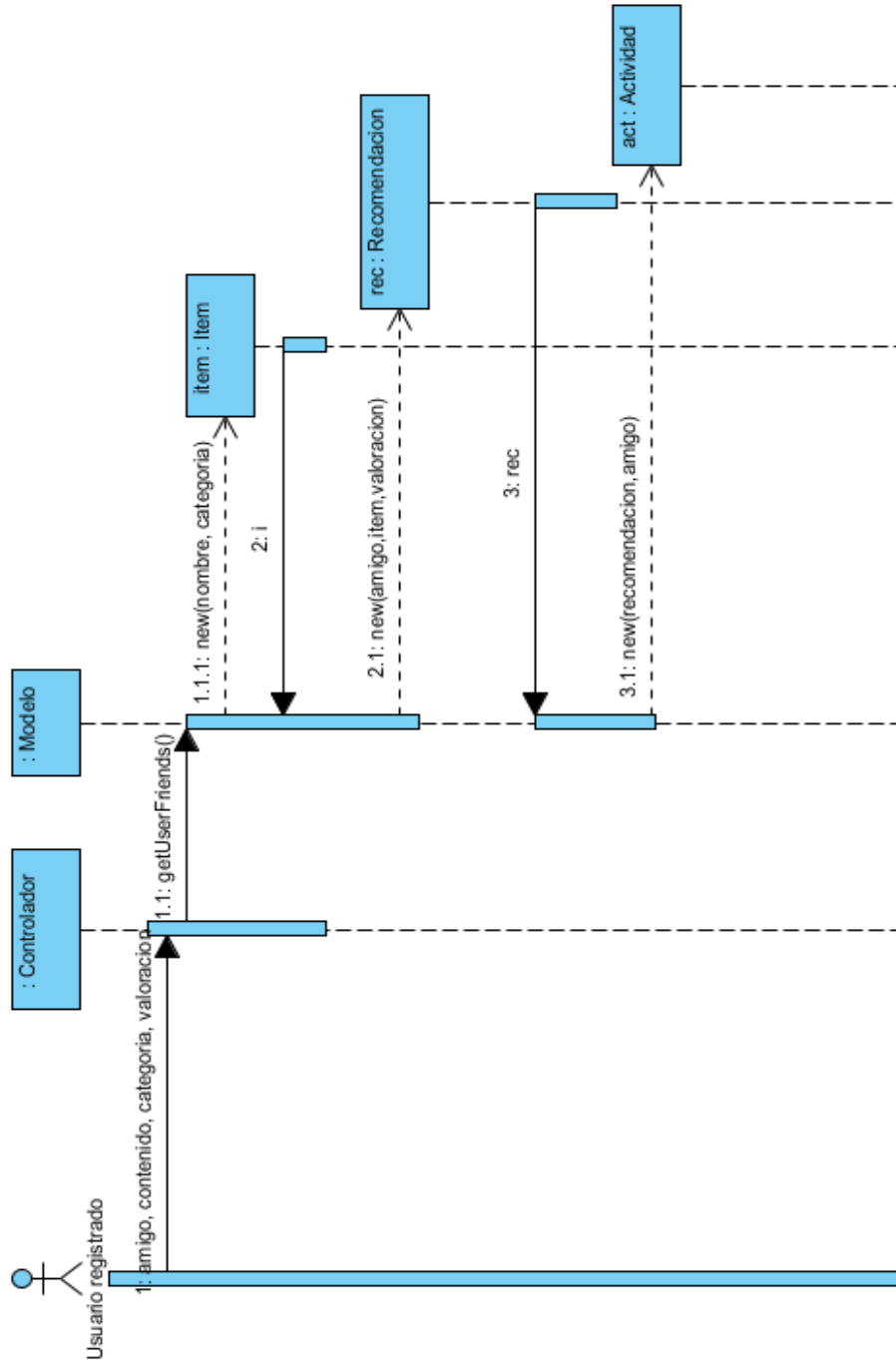


Figura 27 Diagrama de secuencia de Nueva recomendación

**Diagrama de secuencia de Alta de usuario**

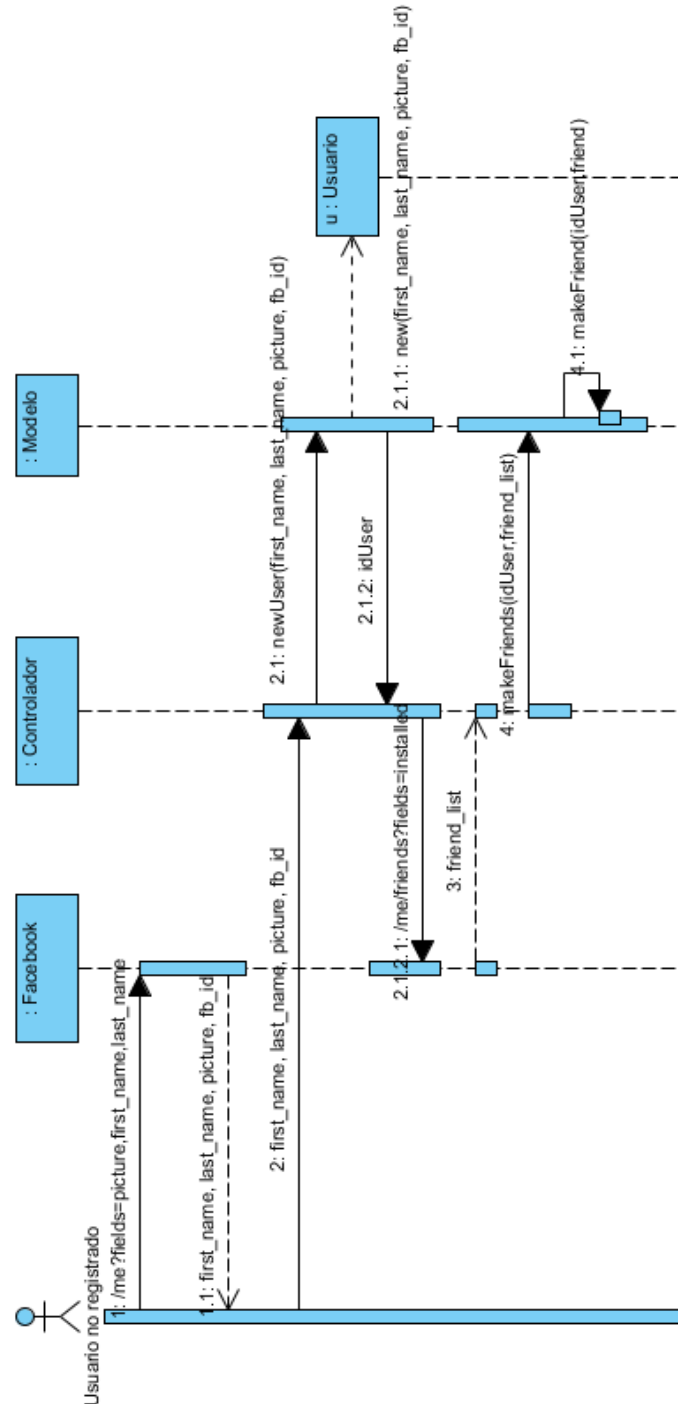


Figura 28 Diagrama de secuencia de Nueva recomendación

**Diagrama de secuencia de *Ver recomendaciones recibidas***

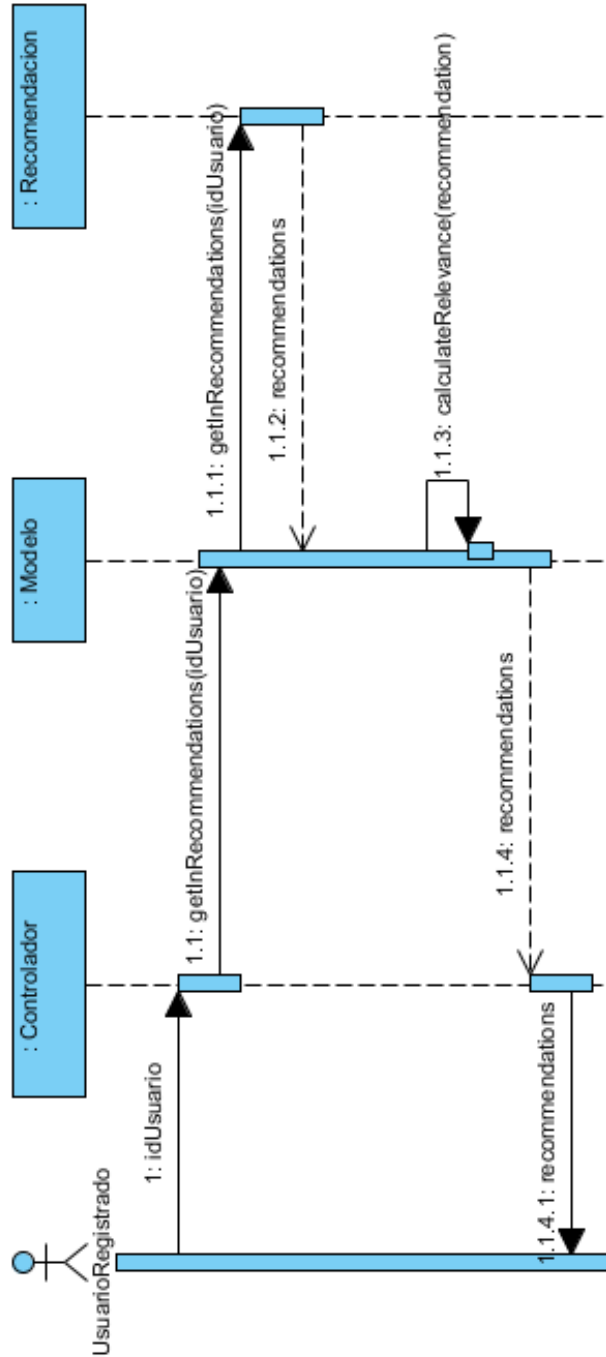


Figura 29 Diagrama de secuencia de Ver recomendaciones recibidas

**Diagrama de secuencia de *Valorar recomendación***

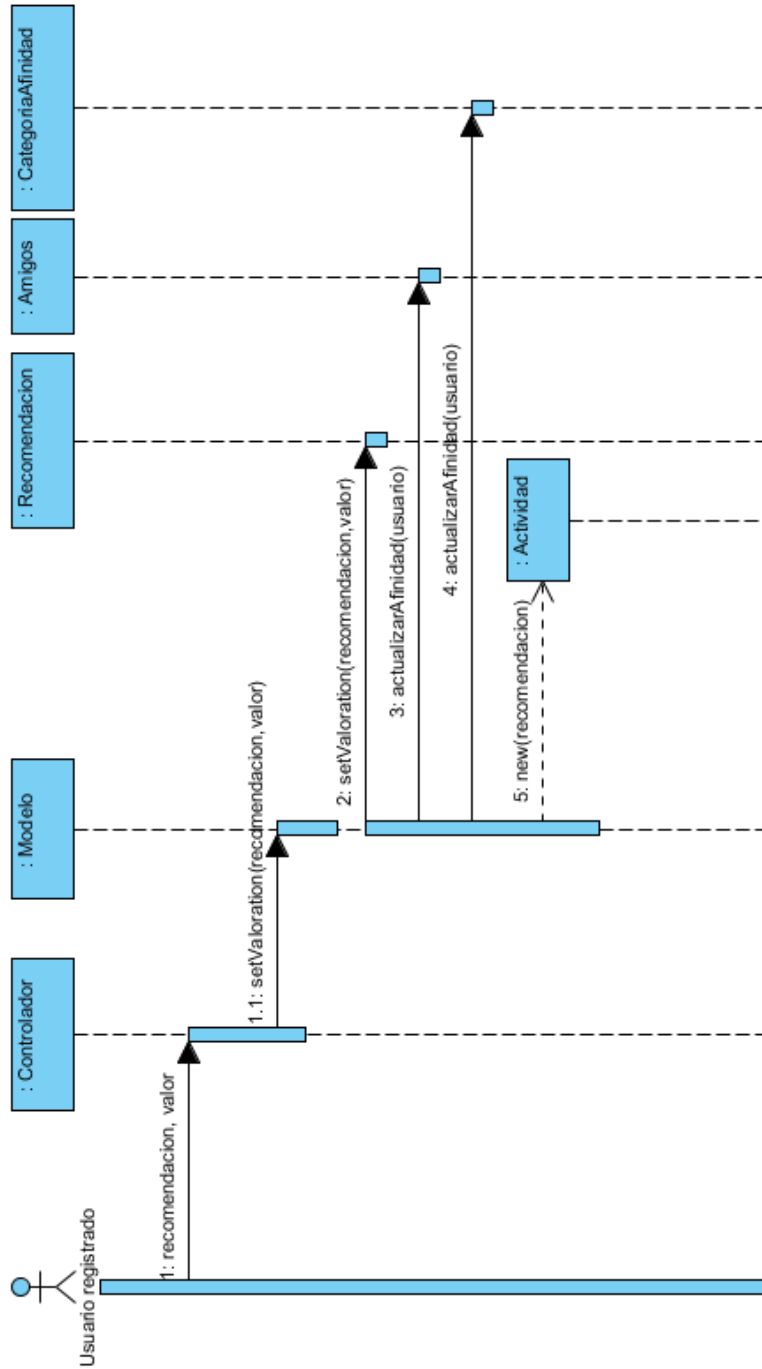


Figura 30 Diagrama de secuencia Valorar recomendación



#### **4.4.3 Diseño de los datos**

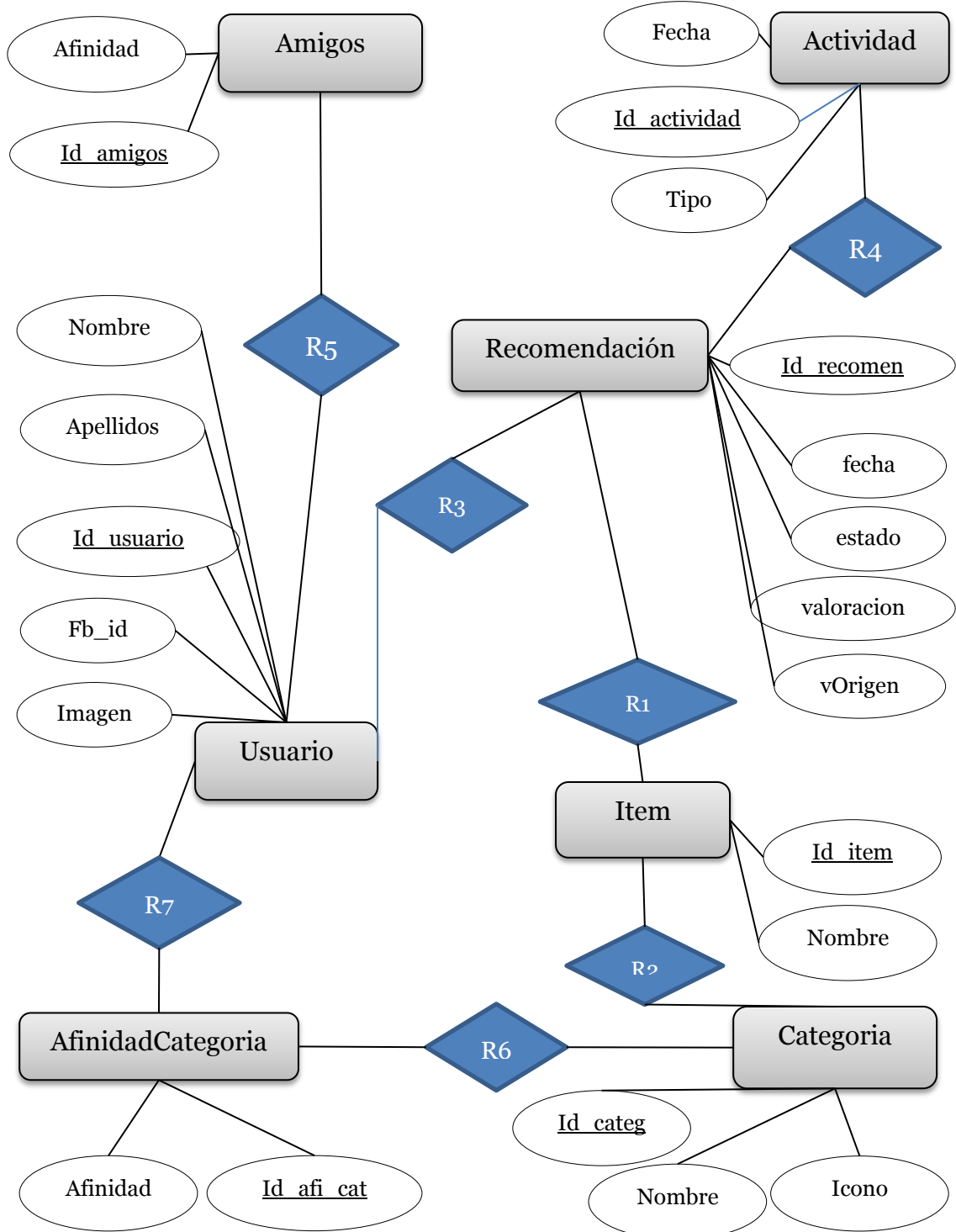
En este apartado vamos a definir la estructura de que van a tener cada uno de los elementos de información con los que se va a trabajar en la aplicación.

El diagrama entidad-relación se propuso inicialmente para el diseño de sistemas de bases de datos relacionales. Se basa en la representación de un mundo compuesto por entidades, atributos y relaciones (Elmasri & Navathe, 2000):

- **Entidades:** cosas u objetos del mundo real con existencia propia y distinguibles del resto. Una instancia es un objeto concreto de una entidad.
- **Atributos:** propiedad o característica de una entidad.
- **Relaciones:** asociación o vínculo existente entre instancias de entidades.

Cada entidad debe tener uno y más atributos que permitan identificar de forma única a cada instancia concreta. A este atributo o conjunto de estos se le denomina clave.

Existen entidades que necesitan estar relacionadas con otras entidades para identificarse, ya que no lo pueden hacer usando solo sus atributos. A estas entidades se las conoce como entidades débiles.



**Figura 31 Modelo entidad relación**

R1: Contiene; R2: Pertenece, R3: Envía; R4: Genera; R5: tiene; R6: contiene R7: tiene.

El diagrama entidad relación ya está normalizado por lo que no hay que realizar ninguna modificación.

Tenemos los siguientes elementos de información, correspondientes a las entidades:

- Los **usuarios** de la aplicación. De ellos conocemos su nombre, apellidos, identificador de Facebook, fotografía, identificador propio del sistema.
- Los **ítems** que se recomiendan en la aplicación. Se trata de un contenido de ocio que tiene un nombre y una categoría temática asociada, como puede ser Cine o Música.
- Las **categorías** en las que se agrupan los ítems a recomendar: Cine, Música, Libros, etc. Tendrá un nombre y una imagen asociada a ese tipo de recomendación.
- La **afinidad en cada categoría**: un usuario tiene una afinidad en cada categoría con cada uno de sus amigos.
- Las **recomendaciones**, de las que se conoce el usuario que las origina, el usuario al que van dirigidas, la fecha, el ítem a recomendar, el estado de la recomendación (si ha sido leída o no), la valoración obtenida
- La **actividad** que generan los usuarios en la aplicación. Se utiliza para notificar en la pantalla principal de la aplicación al usuario sobre recomendaciones recibidas, valoraciones dadas a sus recomendaciones enviadas, etc.

Por tanto, vamos a definir la siguiente estructura de tablas para poder almacenar esa información obtenida del diagrama entidad-relación:

- **Tabla usuarios.** En esta tabla se almacenará toda la información referente a un usuario de la aplicación:

Atributo	Tipo	Información	Requerido	Clave primaria
Id_usuario	Entero	Identificador único de usuario	Si	Si
Nombre	Cadena	Nombre del usuario	Si	No
Apellidos	Cadena	Apellidos del usuario	Si	No
Fb_id	Entero	Identificador único del usuario en la red social Facebook	Si	No
Foto	Cadena	Nombre del archivo con la foto del usuario	Si	No

- **Tabla ítems.** En esta tabla se almacenará toda la información referente a los ítems recomendados por los usuarios:

Atributo	Tipo	Información	Requerido	Clave primaria
Id_item	Entero	Identificador único del ítem	Si	Si
Nombre	Cadena	Texto descriptivo de la recomendación	Si	No
Id_categoria	Entero	Identificador de la categoría a la que pertenece el ítem.	Si	No

- **Tabla categorías.** En esta tabla se almacenará las distintas categorías (Cine, Música, etc.) a las que puede pertenecer un ítem a recomendar.

Atributo	Tipo	Información	Requerido	Clave primaria
Id_categoria	Entero	Identificador único de la categoría	Si	Si
Nombre	Cadena	Nombre descriptivo de la categoría	Si	No
Icono	Cadena	Nombre de la imagen asociada a esa categoría	Si	No

- **Tabla amigos.** En esta tabla se almacenan los usuarios que son amigos dentro de Facebook, así como alguna información adicional:

Atributo	Tipo	Información	Requerido	Clave primaria
Id_usuario1	Entero	Identificador del primer usuario	Si	Si
Id_usuario2	Entero	Identificador del segundo usuario	Si	Si
Afinidad	Real	Grado de afinidad entre ambos usuarios	No	No

- **Tabla actividad.** Se trata de una tabla para recoger y presentar información sobre las acciones del usuario en la aplicación, como enviar una nueva recomendación, iniciar sesión, recibir una recomendación, etc.

Atributo	Tipo	Información	Requerido	Clave primaria
Id_actividad	Entero	Identificador único de la actividad del usuario	Si	Si
Id_recomendación	Entero	Identificador de la recomendación asociada a la actividad	Si	No
Fecha	Fecha	Fecha de la realización de la acción	No	No
Tipo_accion	Entero	Código con el tipo de acción que ha generado la actividad	Si	No
Estado	Entero	Indica si la actividad ha sido mostrada al usuario o no	No	No

- **Tabla recomendación.** Se trata de la tabla que almacena las recomendaciones enviadas por unos usuarios a otros. Consta de la siguiente estructura:

Atributo	Tipo	Información	Requerido	Clave primaria
Id_recomendacion	Entero	Identificador único de la recomendación	Si	Si
Id_origen	Entero	Identificador del usuario que ha	Si	No

		enviado la recomendación		
Id_destino	Entero	Identificador del usuario que recibe la recomendación	Si	No
Fecha	Fecha	Día y hora en la que se envía la recomendación	No	No
Id_item	Entero	Identificador del ítem recomendado	Si	No
Valoración	Entero	Valoración otorgada por el usuario de destino a la recomendación	No	No

- **Tabla categorías\_afinidad.** Se trata de la tabla que almacena la afinidad entre usuarios en cada categoría:

Atributo	Tipo	Información	Requerido	Clave primaria
Id_categorías_afinidad	Entero	Identificador único de la tupla que recoge la afinidad	Si	Si
Id_usuario1	Entero	Identificador	Si	No

		del primer usuario		
Id_usuario2	Entero	Identificador del segundo usuario	Si	No
Id_categoria	Entero	Identificador de la categoría	Si	No
Afinidad	Real	Grado de afinidad entre ambos usuarios en esa categoría	Si	No

Una vez hemos definido las tablas que darán persistencia a los datos de la aplicación, el diagrama entidad-relación resultante es el que se puede ver en la Figura 32 Diagrama E/R final



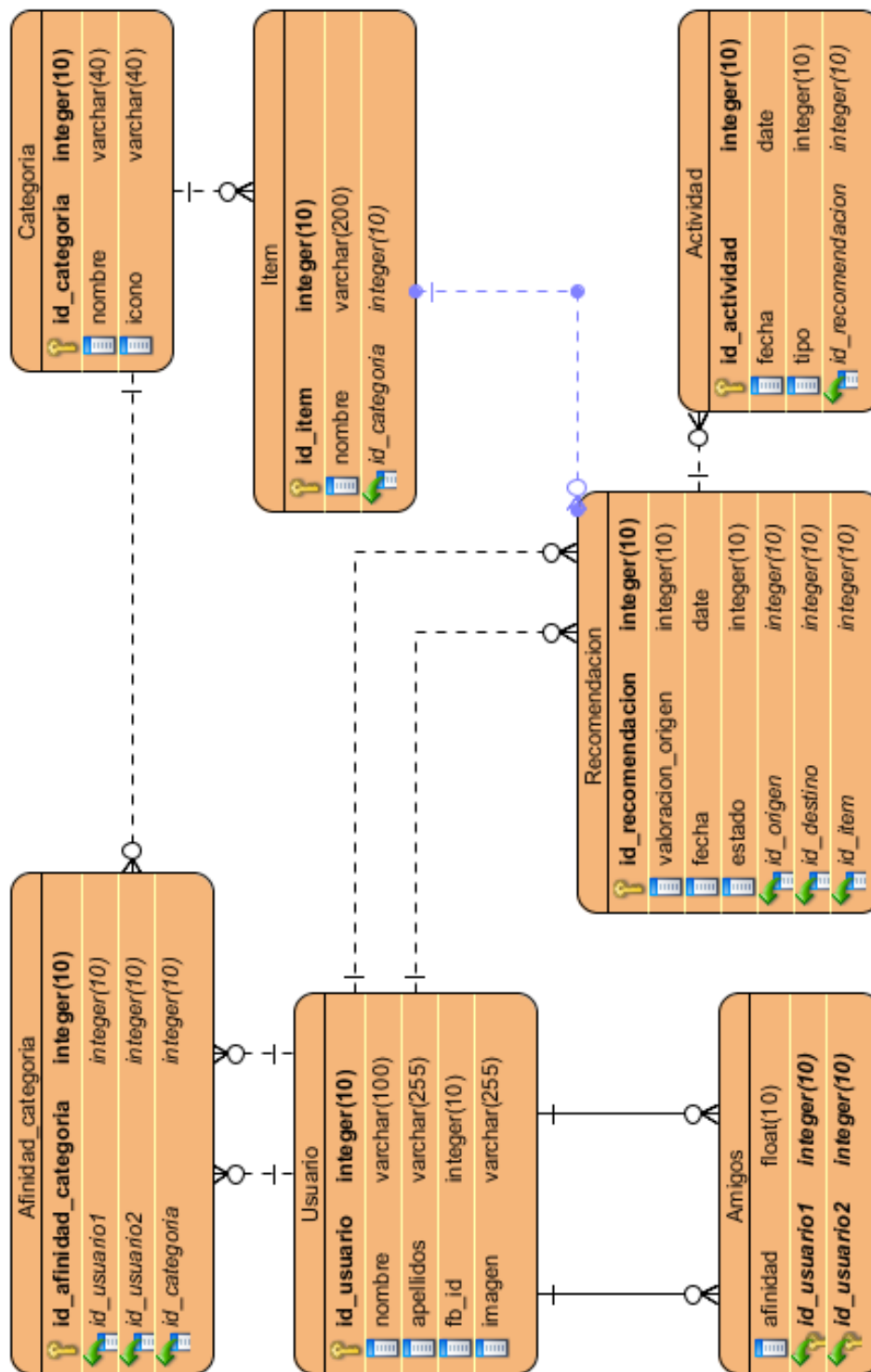


Figura 32 Diagrama E/R final

#### **4.4.4 Cálculo de la relevancia**

Anteriormente se ha hablado de que los usuarios tienen una afinidad global y una afinidad en cada categoría. También se ha comentado que el requisito funcional *Ver recomendaciones recibidas* devolvía al usuario una lista de recomendaciones ordenadas por relevancia o probabilidad de que al usuario le gusten. Los siguientes factores se tienen en cuenta para el cálculo de la relevancia:

- **Afinidad global entre ambos usuarios:** Se trata de la afinidad general que ambos usuarios tienen al haber realizado recomendaciones. Será la valoración media del total de recomendaciones enviadas y recibidas entre ambos usuarios.
- **Afinidad específica de esa categoría:** Es la afinidad concreta en una categoría como cine o música. Será la valoración media del total de recomendaciones enviadas y recibidas entre ambos usuarios en esa categoría concreta.
- **Valoración** dada por el usuario que envía la recomendación, indicando su grado de satisfacción. Alguien puede recomendar un contenido diciendo que está bien, o que es buenísimo. Ese matiz es importante.
- **Tiempo** que ha pasado desde que se envía la recomendación. Se premian las nuevas recomendaciones y se penalizan las que no son valoradas por el usuario. Inicialmente tiene el valor máximo (100%) y se decrementa un 2% por cada día que transcurre desde el envío de la recomendación.

Además, no todos los factores son igualmente relevantes, la afinidad en la categoría será más influyente que la afinidad global o el tiempo. Por tanto, la fórmula del cálculo de la relevancia es la siguiente:

$$r = p_a * A_g + p_b * A_c + V * p_c + T * p_d$$

Donde:

$p_a$ : peso asignado a la afinidad global

$A_g$ : afinidad global entre usuarios

$p_b$ : peso asignado a la afinidad específica de la categoría

$A_c$ : afinidad en la categoría de la recomendación

V: valoración dada por el emisor de la recomendación

$p_c$ : peso asignado a la valoración inicial del emisor

T: tiempo que ha transcurrido desde que la recomendación fue enviada

$p_d$ : peso asignado al factor tiempo

Los pesos que se han asignado a priori son los siguientes:

- Afinidad global: 10%
- Afinidad en la categoría: 60%
- Valoración dada por el emisor: 20%
- Factor tiempo: 10%

Si los usuarios no tienen afinidad en esa categoría por ser la primera recomendación que se envía, se incrementa el porcentaje del resto de factores:

- Afinidad global: 30%
- Valoración dada por el emisor: 60%
- Factor tiempo: 10%

Los porcentajes pueden ser revisados si, tras ver el funcionamiento real de la aplicación, no dan la suficiente importancia a algunos factores frente a otros.

#### **4.4.5 Diseño de la interfaz**

En esta fase se va a definir el aspecto visual que tendrá la aplicación. Es una tarea muy importante porque la interfaz es la parte de la aplicación que percibe el usuario. Una interfaz poco atractiva puede transmitir una imagen bastante mala al usuario.

##### **4.4.5.1 Tipo de aplicación**

El tipo de aplicación es el de una aplicación web integrada en Facebook. El proceso de diseño de la interfaz es similar al de una aplicación o sitio web tradicional, pero con algunos matices al estar integrada en Facebook.

Por un lado, hay que tener en cuenta que la aplicación va a mostrarse en un marco integrado en Facebook, por lo que en la parte superior aparecerá la barra de navegación horizontal de Facebook, y en la parte derecha aparecerá publicidad y otros contenidos de la red social.

Esto hace que la ventana del usuario vaya a estar algo más sobrecargada que en una aplicación web externa, por lo que el principal objetivo en el diseño de la interfaz es la sencillez, por lo que el estilo será algo minimalista omitiendo todo lo que no aporte información o facilidad de uso al usuario y haga la interfaz más compleja.

Además de tener un buen aspecto visual, la interfaz debe resultar fácil de entender y manejar por el usuario, de manera que el tiempo de aprendizaje sea mínimo, y es por ello que debe hacerse un diseño centrado en el usuario.

##### **4.4.5.2 Modelado del usuario**

En una buena interfaz, el usuario podrá realizar sus intenciones de una manera fácil, encontrando la respuesta que esperaba a su acción. Una mala interfaz puede hacer que esta tarea sea frustrante, consiguiendo que el usuario se auto inculpe por no saber utilizarla.

Por tanto, hay que conocer el tipo de usuario que va a utilizar una aplicación, sus características y necesidades, para poder tomar decisiones, especialmente de usabilidad, en el diseño de la interfaz.

Para este proyecto, a diferencia de una aplicación empresarial, el público al que va dirigido es muy amplio. Cualquier usuario con una cuenta de Facebook podría utilizar la aplicación. Si el usuario tiene cuenta en Facebook, implica que tiene un mínimo de conocimientos informáticos, suficientes para crear y manejar un perfil en una red social.

Debido a que el abanico de posibles usuarios es muy amplio, y que la aplicación no realizará una labor profesional sino de ocio, se va a diseñar la interfaz de la manera más usable posible, de manera que una persona con escasos conocimientos informáticos no se sienta agobiada, aunque se corra el riesgo de que parezca una interfaz demasiado simple para usuarios con conocimientos informáticos más avanzados.

#### **4.4.5.3 Análisis de tareas**

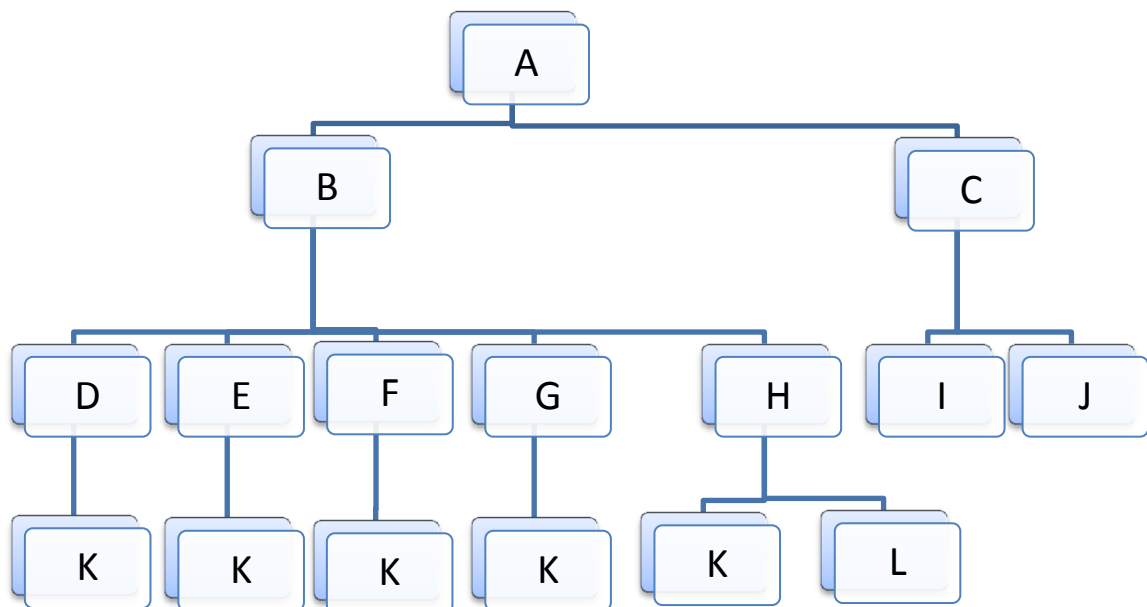
Además de estudiar los posibles usuarios que utilizarán la aplicación, es muy importante saber que tareas podrán realizar estos usuarios en la aplicación.

Se trata de especificar las funcionalidades de la aplicación, las cuales fueron analizadas y quedaron definidas en el apartado *4.1.1 Requisitos funcionales*

#### **4.4.5.4 Modelo conceptual**

Una vez que sabemos al tipo de usuario al que irá dirigida la aplicación y las tareas que deben estar presentes en la interfaz, vamos a organizar la información en niveles de navegación.

Para ello, se presenta el siguiente diagrama de organización utilizando el vocabulario visual de **Garret**. En el, se va a estructurar la funcionalidad de la interfaz, teniendo en cuenta de que se trata de una aplicación web.



**Nivel 0:**

**A:** Ventana principal de la aplicación.

**Nivel 1:**

**B:** Menú principal, que engloba las funciones principales de la aplicación

**C:** Pie de página, en el cual irán funcionalidades secundarias

**Nivel 2:**

**D:** Actividad

**E:** Recomendaciones recibidas

**F:** Recomendaciones enviadas

**G:** Historial

**H:** Afinidad

**I:** Contacto

**J:** Ayuda

**Nivel 3:**

**K:** Ver perfil de usuario

**L:** Filtrar afinidad por categoría

#### **4.4.2.5 Guía de estilo**

A diferencia de las aplicaciones de escritorio, no hay una guía de estilo definida por el sistema operativo que nos indique o aconseje sobre el estilo que debe tener la aplicación: color, tipografía, posición de ciertos botones, etc. El diseñador debe definir su propia guía de estilo, siempre con el objetivo de conseguir un diseño atractivo pero a la vez fácil de usar y familiar para el usuario

El estilo general de la aplicación debe ser claro y estar poco sobrecargado. Hay que pensar que tener en cuenta que la aplicación se utilizará desde la red social Facebook, por lo que hay que tener en cuenta que en el navegador del usuario aparecerán las barras de navegación superior y la derecha de Facebook.

Vamos a ver alguno de los aspectos que definirán el estilo de la aplicación.

### **Logotipo**

El logotipo se ha diseñado de manera que refleje la intención de la aplicación.



Figura 33 Logotipo de la aplicación

Tiene dos partes diferenciadas, imagen y texto:

- La imagen es la seña de identidad de la aplicación y se utiliza como icono del navegador y como icono de la aplicación en Facebook.
- El texto representa, por un lado el título de la aplicación en un tamaño mayor, y por otro lado una breve explicación o eslogan de la finalidad de la misma.

### **Fuentes**

La fuente estándar de la aplicación utilizará la tipografía *Helvética*, de un tamaño de 12puntos. El color será un gris oscuro (#474747). Para encabezados



se utilizarán medidas relativas con un tamaño mayor respecto a la fuente estándar (1.4em).

### **Enlaces**

Los enlaces tendrán el tamaño del texto estándar, pero serán de un color diferente para resaltarlos del texto. El color será un gris algo más claro (#666) y cuando se pase el ratón por encima se mostrará aún más claro (#999). No se subrayará el texto del enlace ni se marcarán con diferente color los enlaces ya visitados.

El destino de los enlaces del menú izquierdo se cargará en el cuadro central de la aplicación. Por el contrario, los enlaces en el pie de página se abrirán en una ventana superpuesta.

### **Colores**

El color predominante será el blanco para mostrar claridad y no sobrecargar la página. No habrá colores fuertes de fondo. Se utilizará el color gris para el fondo y colores fuertes para resaltar algunos elementos como botones o imágenes. El fondo de la aplicación será una imagen granulada con un tono gris.

Los botones de invitar amigos irán en un tono azul en toda la aplicación, mientras que los de enviar recomendación irán en un tono verde.





#### **4.4.5.6 Metáforas**

Las metáforas suponen una parte muy importante del éxito o fracaso de una interfaz, por tanto hay que diseñarlas con mucho cuidado basándose en los datos adquiridos durante el análisis. Lo primero que hay que hacer es identificar aquellas tareas que requieren metáforas y a continuación construir la metáfora.






Para este último paso deberemos identificar los objetos y acciones que intervienen en las tareas, buscar una representación visual, simplificar esa representación y asegurar la coherencia visual.

Tras un análisis exhaustivo de los casos de uso obtenidos durante la fase de análisis hemos seleccionado unas tareas como candidatas para ser identificadas mediante metáforas y se han elaborado estas eligiendo algunos iconos existentes en la red, y combinando algunos de ellos.



Para los tipos de actividad diferentes que se pueden generar en Sugerest, se han elegido los siguientes iconos que identifiquen que tipo de actividad se ha recibido:

- Recomendación recibida: 
- Valoración recibida: 
- Valoración modificada: 
- Valoración rechazada: 

Para la valoración de las recomendaciones, se ha empleado la metáfora de las estrellas de valoración, usada ampliamente en muchos sitios de internet:

- Valoración malísima: 
- Valoración mala: 
- Valoración regular: 
- Valoración buena: 
- Valoración buenísima: 

Para distinguir entre el tipo de recomendación, si es enviada o recibida, se ha utilizado una analogía a las llamadas recibidas o enviadas en los teléfonos móviles, con una flecha entrante o saliente:

- Recomendación enviada: 
- Recomendación recibida: 

Para descartar una recomendación, se empleará el icono de aspa que suelen utilizar los gestores de ventanas para cerrar o cancelar:

- Descartar recomendación: 

#### 4.4.5.7 Pantallas de la aplicación

La lista de las pantallas de la aplicación serán las siguientes:

- Página principal de la aplicación.
- Actividad
- Bienvenida
- Recomendaciones recibidas
- Recomendaciones enviadas
- Historial de recomendaciones
- Nueva recomendación
- Invitar amigos
- Perfil de usuario
- Perfil de ítem
- Ayuda
- Acerca de

- Contacto

#### **4.4.5.8 Prototipos**

Los prototipos nos ofrecen una visión del aspecto visual que tendrá un producto antes de que llegue a crearse. Se han construido unos prototipos de alto nivel en papel, ya que durante la fase de diseño de los prototipos se realizarán muchas modificaciones que son más lentas de realizar con un programa de diseño.

La guía de estilo definida no se aplicará hasta que se implementen los prototipos con el motivo de agilizar el diseño de estos.

Se realizarán todas las modificaciones que sean necesarias hasta que obtengamos un prototipo que sea usable e intuitivo para el usuario.

De las principales pantallas de la sección anterior. Para cada pantalla, se mostrará el boceto a mano de la misma y las decisiones de diseño adoptadas.

#### **Pantalla principal**

La pantalla principal muestra la apariencia visual que tendrá la aplicación.



**Figura 34** Pantalla principal

La pantalla principal de la aplicación tiene cuatro secciones diferenciadas:

- **Cabecera:** En su parte izquierda contendrá el logo de la aplicación. En la parte derecha, dará un saludo al usuario y mostrará el contador de recomendaciones recibidas y enviadas pendientes de valorar o ser valoradas.
- **Menú de navegación izquierdo:** Desde él se accederán a las funciones principales de la aplicación: actividad, recomendaciones recibidas y enviadas, historial y afinidad con amigos.

- **Contenido principal:** Se trata de un contenedor donde se irán cargando las diferentes pantallas del menú de navegación izquierdo. En el inicio de la aplicación mostrará la última actividad en la que esté involucrada el usuario.
- **Pie de página:** contendrá otras secciones de menor relevancia en la aplicación, como contacto, acerca de y manual de usuario. A diferencia de las secciones del menú de navegación izquierdo, las pantallas de los enlaces del pie de página se cargarán en una nueva ventana superpuesta, ya que son independientes del resto de contenido de la aplicación.

### **Actividad**

La pantalla de actividad será la que se cargue inicialmente al iniciar la aplicación, siempre y cuando el usuario haya generado alguna actividad en la aplicación que se pueda mostrar.

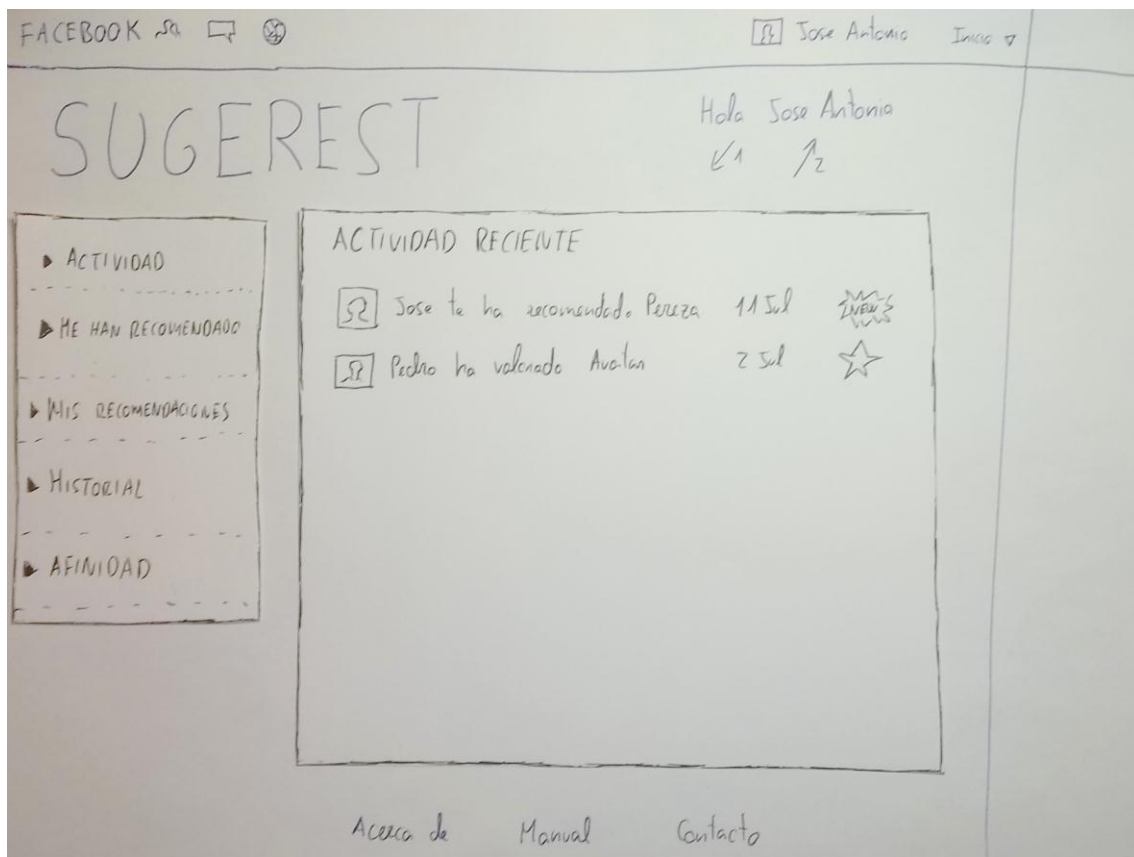


Figura 35 Pantalla de actividad

En ella, se mostrarán las notificaciones que se originan con el uso de la aplicación. Pueden ser del tipo:

- **Nueva recomendación recibida:** Se informa al usuario de que ha recibido una nueva recomendación. Se informa de qué le han recomendado y que amigo la ha enviado.
- **Un amigo ha valorado tu recomendación:** Se informa al usuario de que una recomendación enviada ha sido valorada por el usuario destinatario de la misma.
- **Un amigo ha modificado una valoración:** Se informa al usuario de que un amigo ha modificado al valoración inicial que éste le dio a la recomendación.

- **Tu recomendación ha sido descartada:** Un amigo ha descartado una recomendación enviada por el usuario.

Cada actividad tendrá un icono asociado a la misma, de manera que el usuario pueda ver rápidamente el tipo de actividad sin necesidad de leer previamente el texto de la misma.

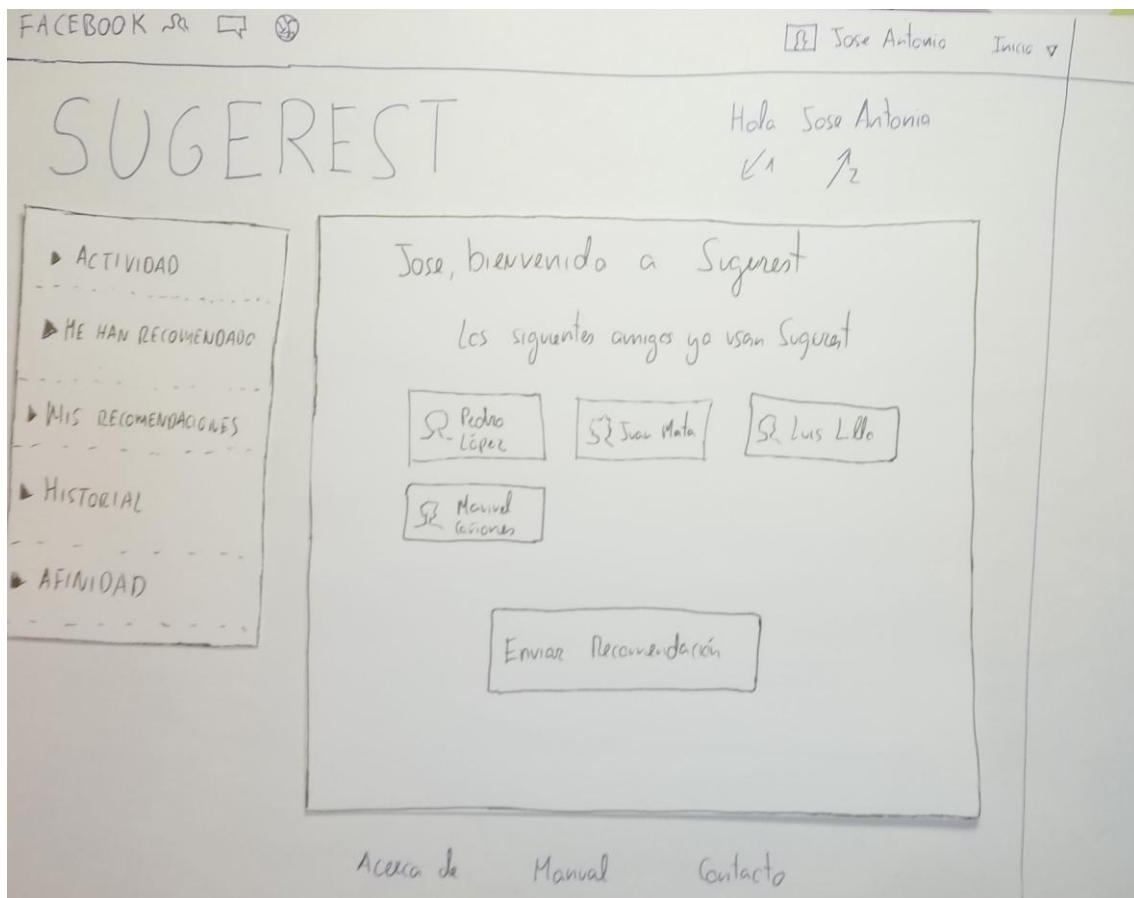
### **Bienvenida**

La pantalla de bienvenida será mostrada al usuario nuevo en la aplicación. Si un usuario ha generado actividad, la pantalla de actividad será la pantalla por defecto. Pero en caso de ser nuevo y aún no haberla generado, se le mostrará esta pantalla de bienvenida.

Cuando un usuario llega nuevo a la aplicación pueden ocurrir dos situaciones: que tenga ya amigos registrados en la aplicación y pueda comenzar a enviar recomendaciones, o que no tenga aún ningún amigo.

Si ya tiene amigos, la acción natural del usuario será la de enviar una recomendación y debemos incitar a ello. La siguiente pantalla cubre este caso:





**Figura 36 Pantalla de bienvenida con amigos**

Por el contrario, si no tiene amigos la aplicación debe sugerirle que invite a sus amigos a utilizar la aplicación. Además, en este último caso, también sería recomendable ofrecerle al usuario leer el tutorial o manual de ayuda. La pantalla de bienvenida en este caso es:



Figura 37 Pantalla de bienvenida sin amigos

### **Recomendaciones recibidas**

La pantalla de recomendaciones recibidas presenta todas las recomendaciones que le han enviado al usuario sus amigos y éste aún no ha valorado. Las recomendaciones vienen ordenadas según el grado de afinidad con el usuario que las envía. La lista de recomendaciones tendrá las siguientes columnas:

- **Recomendación:** Texto de la recomendación recibida
- **Tipo:** Icono con el tipo de recomendación

- **Amigo:** Fotografía del amigo que ha enviado la recomendación al usuario.
- **Fecha:** Fecha en la que se recibió la recomendación.
- **Valoración:** Valoración que el usuario puede dar a la recomendación. Se utilizará la metáfora de las estrellas, para valorar desde una a cinco estrellas la recomendación recibida. El usuario puede seleccionar el número de estrellas y la valoración quedará asignada
- **Afinidad:** Afinidad del usuario con el amigo que envió la recomendación. Las recomendaciones se presentarán ordenadas por este campo, ya que es más probable que sean del gusto del usuario al tener un mayor grado de afinidad con el usuario que las envía.

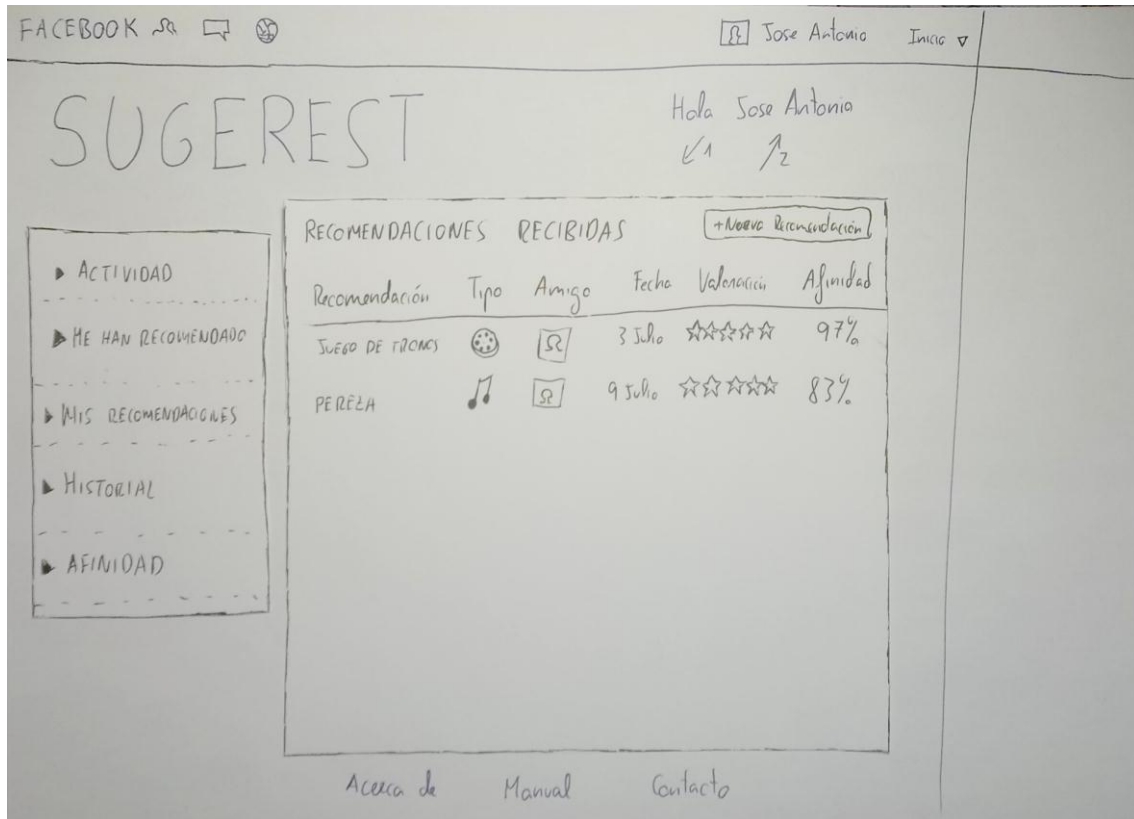


Figura 38 Pantalla de recomendaciones recibidas

### Recomendaciones enviadas

La pantalla de recomendaciones enviadas mostrará todas las recomendaciones que ha enviado el usuario a sus amigos y que aún no han sido valoradas por estos.

Mostrará el listado en forma de tabla con las siguientes columnas:

- **Recomendación:** Texto de la recomendación enviada
- **Tipo:** Icono con el tipo de recomendación
- **Amigo:** Fotografía del amigo al que se ha enviado la recomendación

- **Fecha:** Fecha en la que se envió.
- **Estado de la recomendación:** Indica si la recomendación ha sido visualizada por el usuario destinatario de la misma. Una recomendación puede estar en los estados: no leída, no valorada y valorada. Los dos primeros aparecerán en esta pantalla. En el momento en el que pase a estar valorada, aparecerá en el historial.

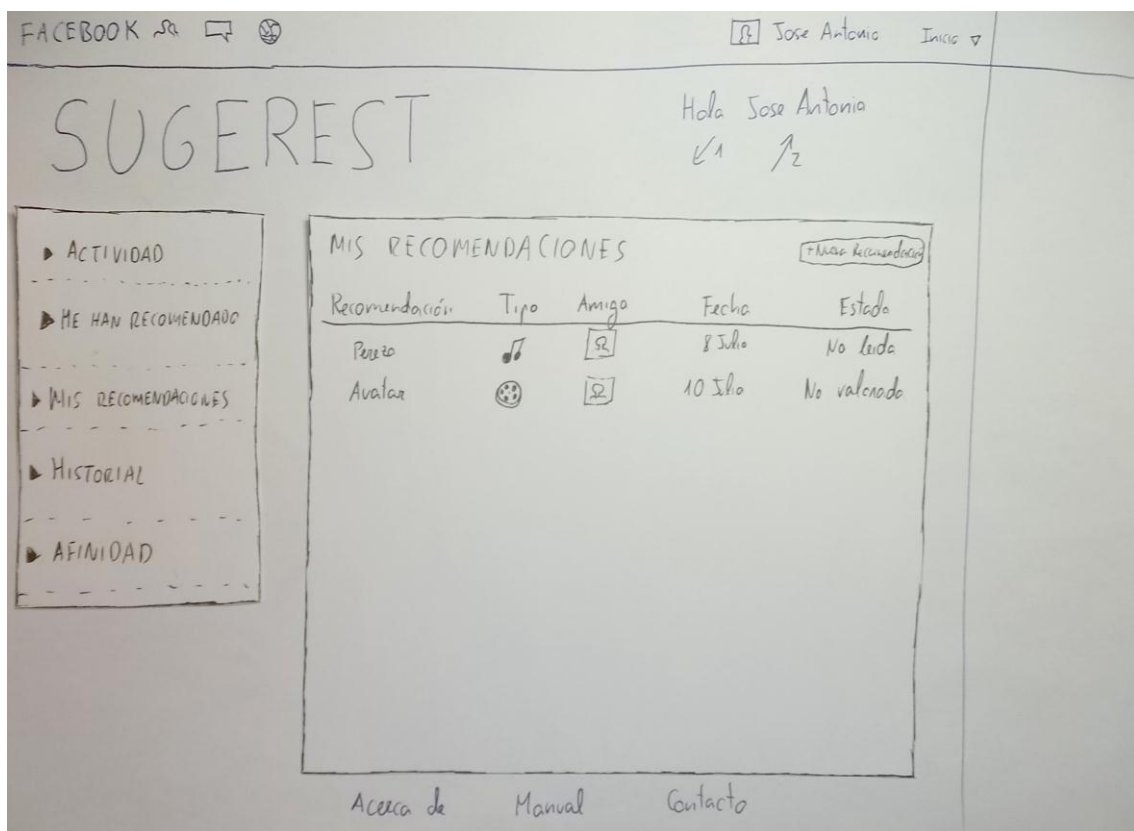


Figura 39 Pantalla de recomendaciones enviadas

### Historial de recomendaciones

La pantalla de historial de recomendaciones recoge todas las recomendaciones enviadas que han sido valoradas por el usuario destino y las

recomendaciones recibidas que el usuario ya ha valorado. Es decir, todas las recomendaciones que han obtenido respuesta.

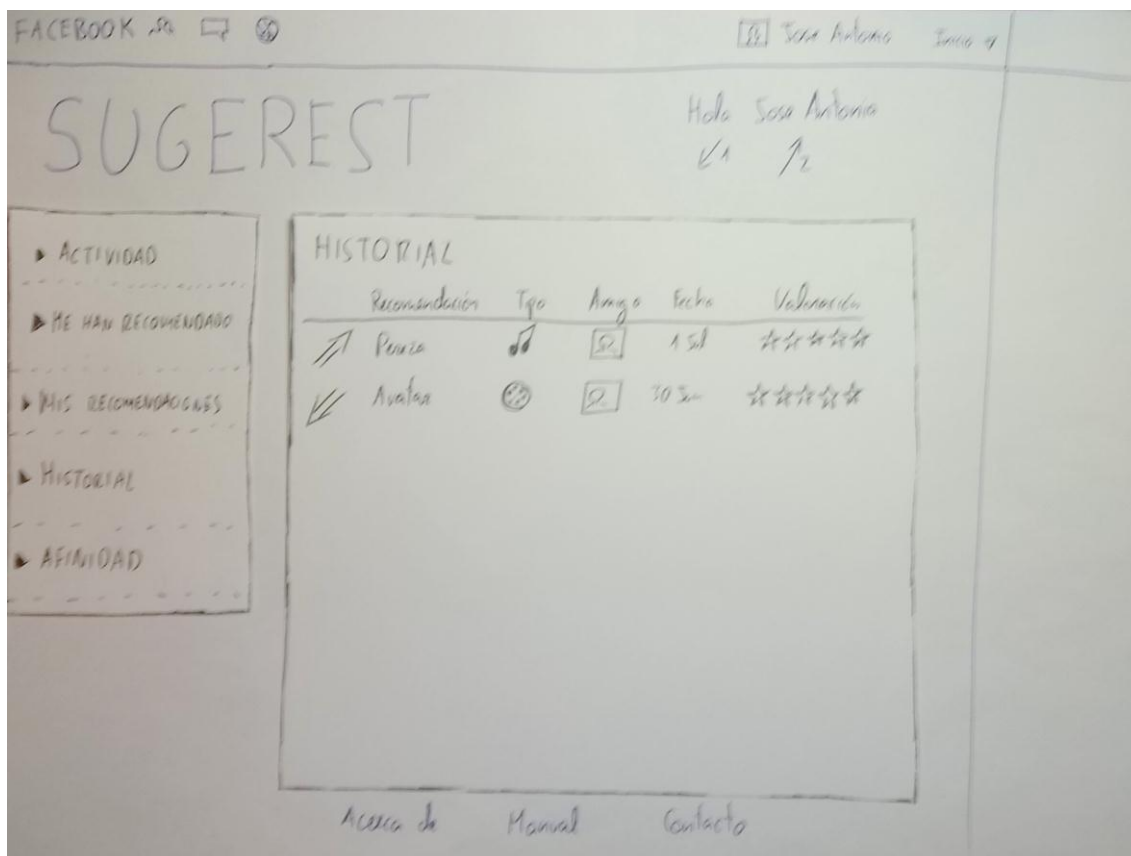


Figura 40 Pantalla de historial

Cuando el usuario valora una recomendación en la pantalla de “Recomendaciones recibidas” automáticamente aparece en el historial. Del mismo modo, cuando una recomendación enviada por el usuario que aparece en “Mis recomendaciones” es valorada por el usuario destino, también pasa a ocupar el historial.

Se mostrará un listado con la siguiente información:

- Tipo de recomendación: Se trata de una imagen que ayuda a reconocer rápidamente si la recomendación es enviada o recibida. Se utiliza una

metáfora similar a las llamadas de un teléfono móvil, representadas con flechas de entrada o salida si son entrantes o salientes respectivamente.

### **Nueva recomendación**

La pantalla de nueva recomendación es posiblemente la pantalla más importante de Sugerest. Debe de ser una pantalla atractiva y fácil de usar para que el usuario novel sea se sienta cómodo enviando su primera recomendación.

Es por ello, que se ha diseñado como un formulario muy intuitivo, en el que solo hay que escribir el título del ítem a recomendar y la selección de categorías y amigos se realiza con un clic. Para seleccionar una categoría o amigo, solo basta con pulsar sobre recuadro que los contiene y visualmente quedarán marcados como seleccionados.

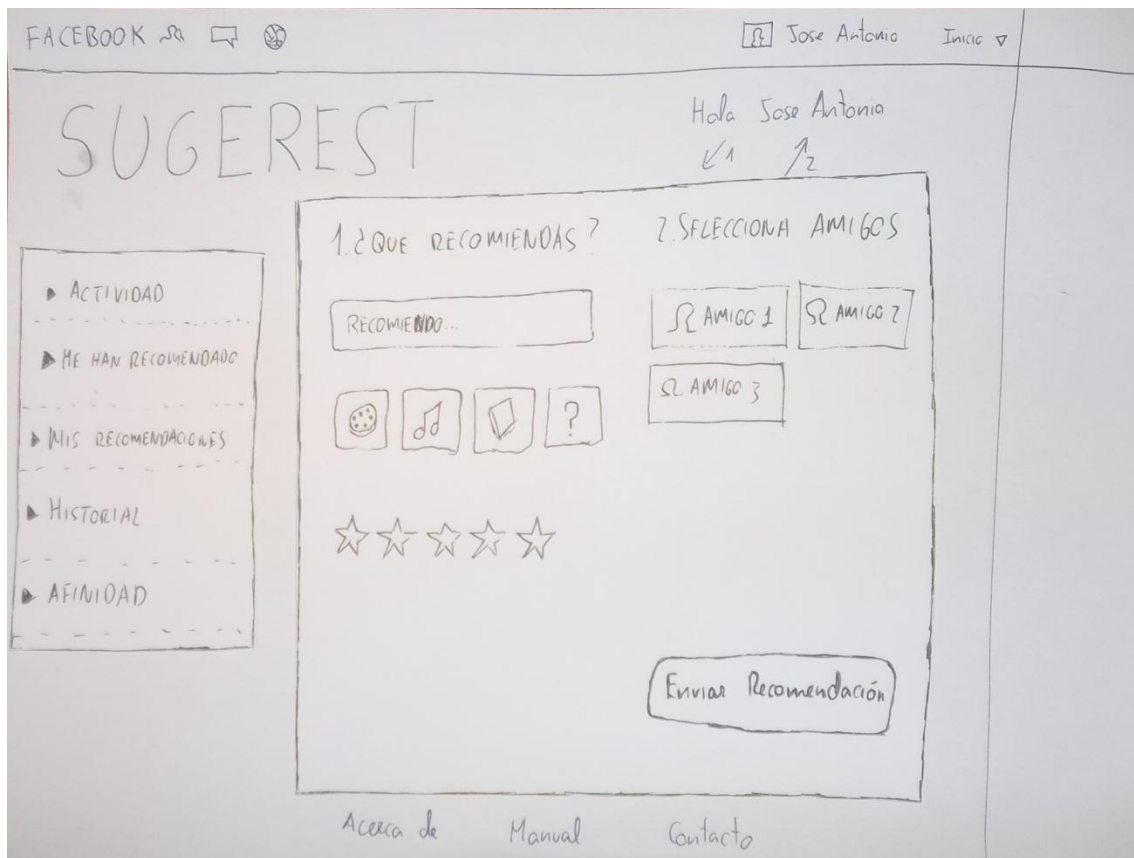


Figura 41 Pantalla de nueva recomendación

El botón de “Enviar recomendación” quedará deshabilitado siempre que no esté toda la información seleccionada: un ítem a recomendar, una categoría y un amigo. Cada vez que el usuario modifique esa información, se comprobará si hay que habilitar o deshabilitar el botón de “Enviar recomendación”. Del mismo modo, cuando el usuario pulse sobre él, se deshabilitará para evitar dobles pulsaciones.

### Afinidad

La pantalla de afinidad tiene la finalidad de mostrar el listado de amigos con los que hemos realizado algún intercambio de recomendaciones y que al



menos una de ellas ha sido valorada para poder tener información sobre si ha gustado o no.



**Figura 42** Pantalla de afinidad

El listado quedará ordenado por la última columna, Afinidad, mostrando primero los amigos con mayor afinidad y en último lugar los amigos con los que el usuario tenga menor afinidad.

Por defecto, aparece activada la afinidad global entre usuarios. Para ver la afinidad por cada categoría, en la parte superior derecha se ubican unos botones que permitirán el filtrado por categorías. Cuando se pulse el botón de una categoría, se mostrara el listado de amigos ordenados igualmente por afinidad pero para esa categoría concreta, con lo que puede variar el orden de la lista.

## **Ayuda**

La pantalla de ayuda contendrá una lista de temas de ayuda que expliquen al usuario como interactuar con la aplicación. Cuando el usuario pulse en el tema de ayuda concreto, se mostrará en la parte inferior la información referente a ese tema.

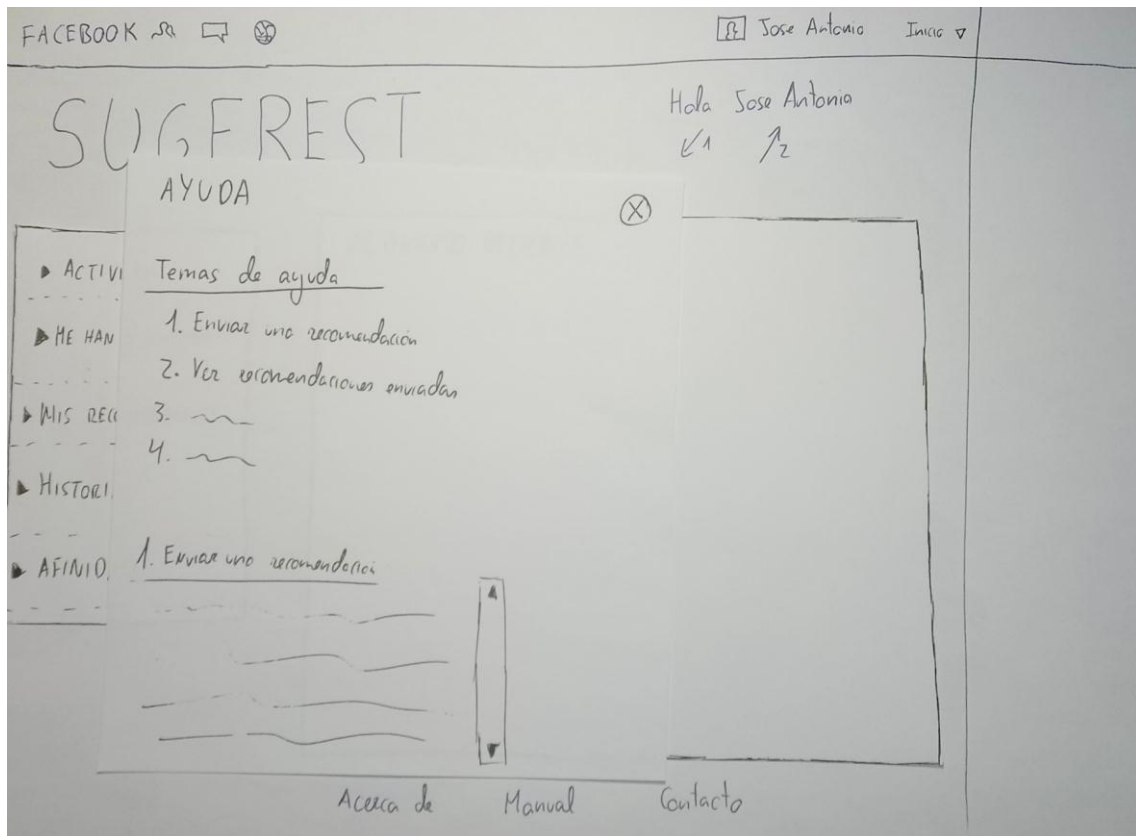


Figura 43 Pantalla de ayuda

## Acerca de

La pantalla de Acerca de simplemente se limitará a mostrar información acerca de Sugerest. Responderá preguntas del tipo ¿Qué es Sugerest? o ¿Por qué surge Sugerest?

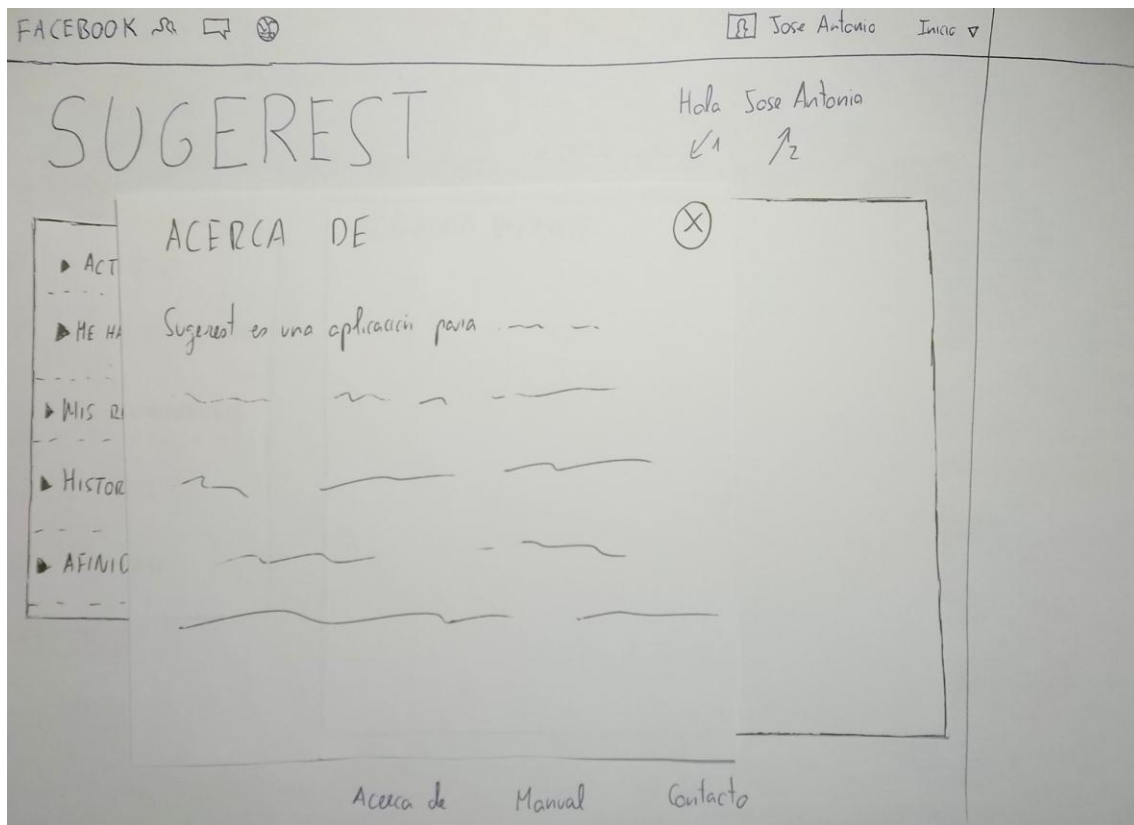


Figura 44 Pantalla de acerca de

## Contacto

Al igual que las pantallas anteriores, la pantalla de contacto también se mostrará superpuesta al contenido principal de la aplicación. Consta de un formulario a través del cual el usuario puede contactar con el administrador de la aplicación.

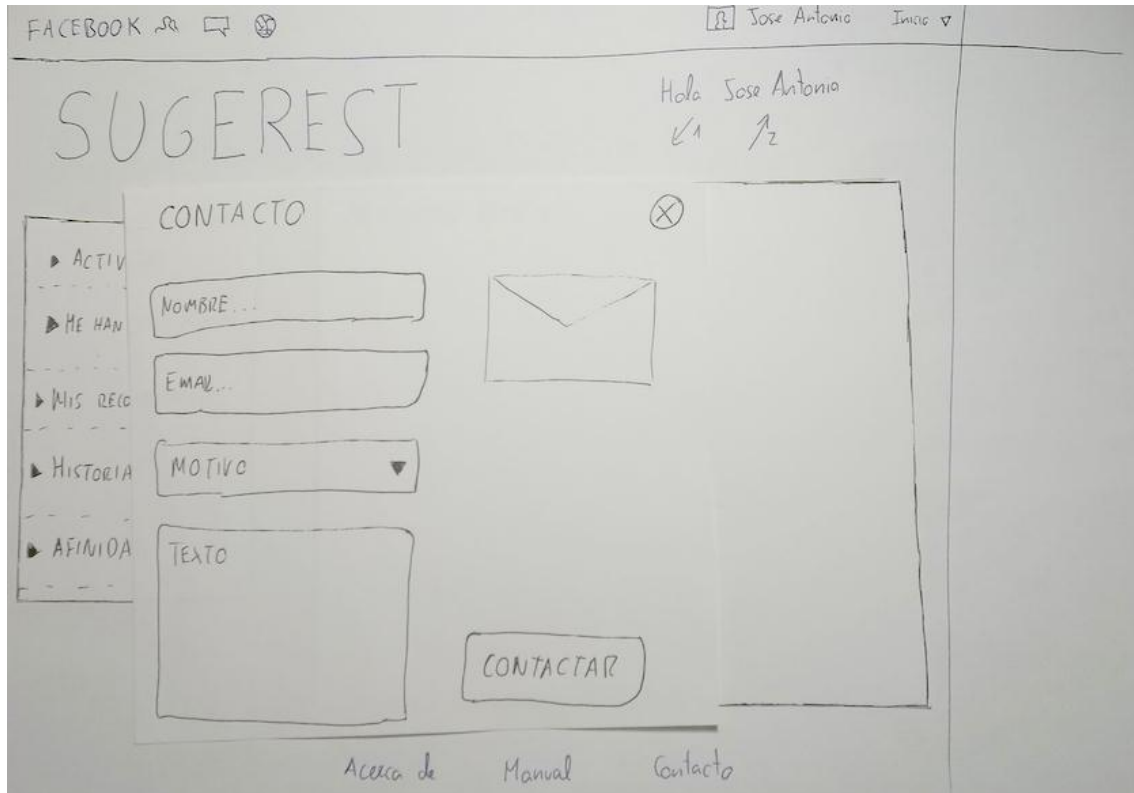


Figura 45 Pantalla de contacto

Los campos del formulario son:

- **Nombre:** Nombre del usuario que desea formular la pregunta. Se completará automáticamente con los datos del usuario y no se permitirá modificar.
- **Email:** Correo electrónico del usuario. Se completará también con los datos del usuario, pero se podrá modificar por si el usuario desea recibir la respuesta en otra dirección de correo electrónico
- **Motivo:** motivo por el que se contacta con los administradores de la aplicación. Un ejemplo de motivo puede ser reportar un bug o sugerir una nueva funcionalidad.

- **Texto:** Texto explicativo que desarrollará el motivo por el que se ha contactado.

El botón de “contactar” estará inicialmente deshabilitado, y se habilitará cuando todos los campos estén correctamente rellenos. Una vez se pulse en contactar, el formulario se enviará y se cerrará la ventana de contacto para volver a la aplicación.

### **Ver perfil de usuario**

La pantalla del perfil de usuario mostrará la información existente de un amigo del usuario en la aplicación. Por un lado, se mostrará la información general del usuario amigo: número de recomendaciones recibidas y enviadas. Por otro lado, se mostrará las recomendaciones que tiene el usuario y su amigo: recomendaciones que el usuario ha enviado a su amigo y recomendaciones que el amigo ha enviado al usuario. Además, se mostrará el grado de afinidad entre ambos amigos.

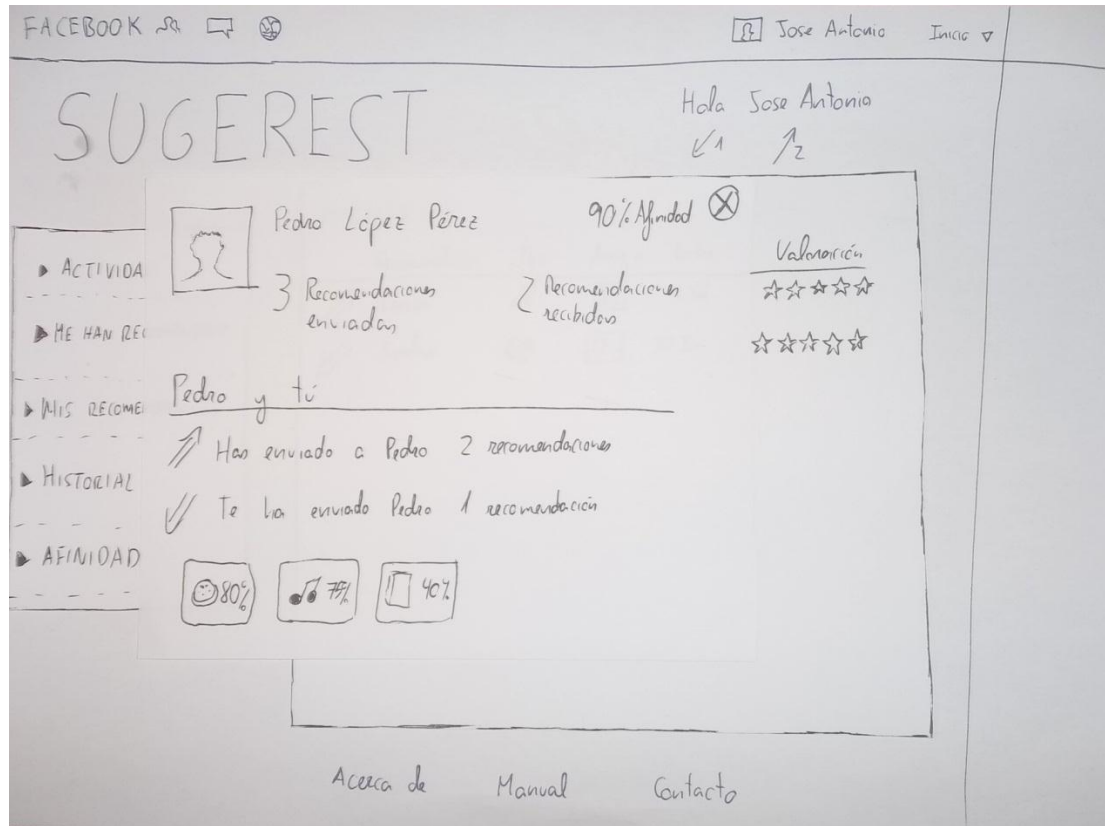


Figura 46 Pantalla de perfil de usuario

#### 4.4.2.9 Caminos de navegación

Las pantallas y los prototipos especificados en las secciones anteriores tienen un carácter aislado y estático. De ellas no se deduce la interacción del usuario con la aplicación. Es por ello que se va a recurrir a diseñar la secuencia de acciones que tiene que realizar el usuario para conseguir un objetivo deseado en la aplicación. Esto nos va a permitir comprobar como de usable es nuestra aplicación.

Se va a utilizar una herramienta llamada *Storyboard*, la cual consiste en mostrar la secuencia de pantallas por las que el usuario se va encontrando en el transcurso de la interacción con la aplicación. Será muy útil para describir los escenarios de situaciones concretas que pueden existir en la aplicación. Nos

permitirá comprobar que la interacción es adecuada y ver si es necesario añadir alguna pantalla intermedia que facilite la interacción, o si por el contrario hay que eliminar alguna de las existentes para simplificar.

Se van a utilizar las pantallas diseñadas anteriormente para indicar la secuencia de interacción y se va a considerar el punto de partida para realizar cada tarea la página principal de la aplicación.

Por tanto, se definen los siguientes storyboards para los caminos más relevantes, ya que todos los elementos del menú izquierdo siguen la misma estructura de navegación, al igual que los del pie de página. Hay que destacar, que los caminos de navegación no son únicos, y que por ejemplo, para enviar una nueva recomendación hay un enlace desde varias secciones del sitio, igual que para ver el perfil de un usuario.

**Ver recomendaciones recibidas:**

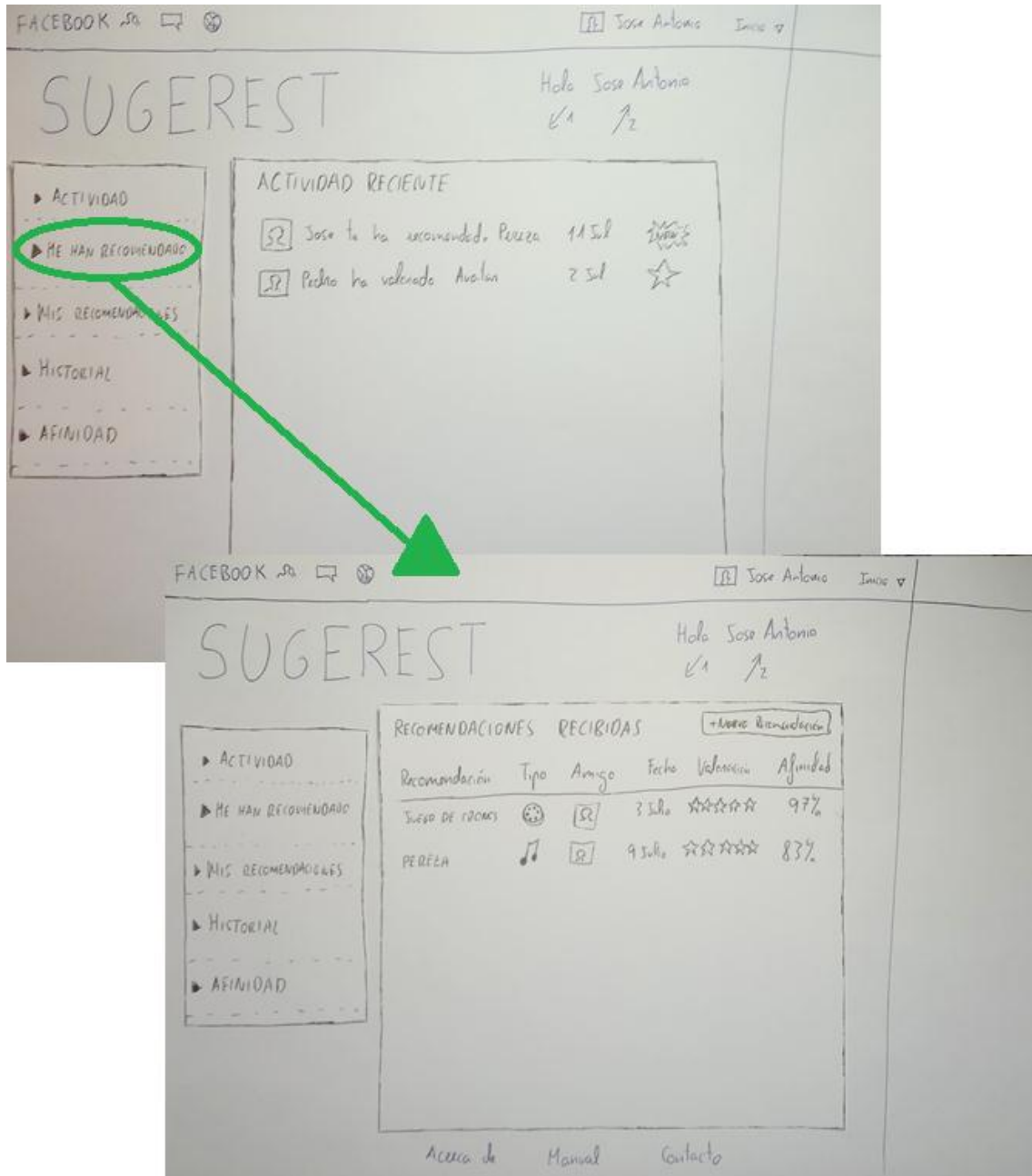


Figura 47 Story board de Ver recomendaciones recibidas



**Enviar recomendación**

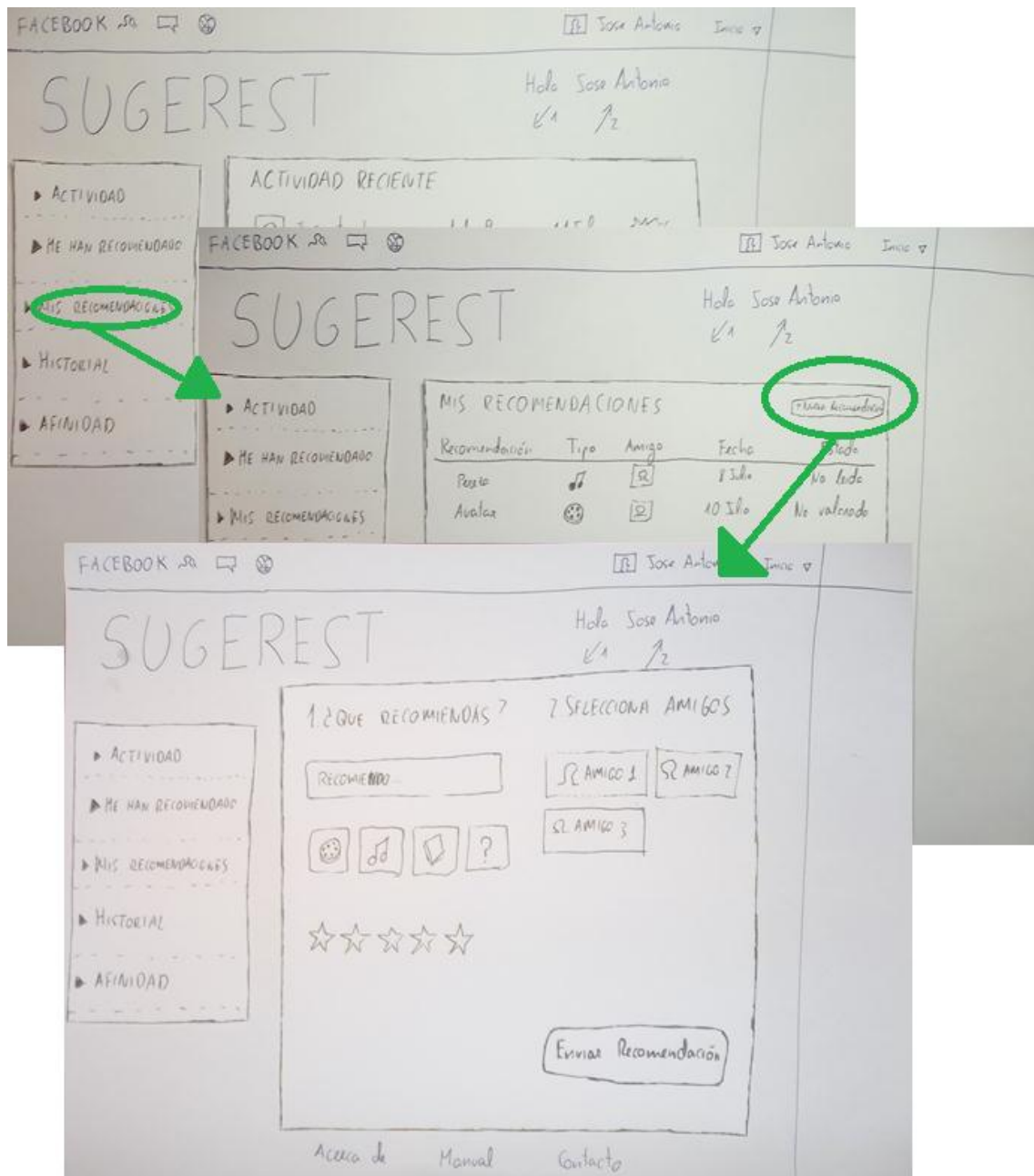


Figura 48 Storyboard de Enviar recomendación

**Ver perfil de un usuario**

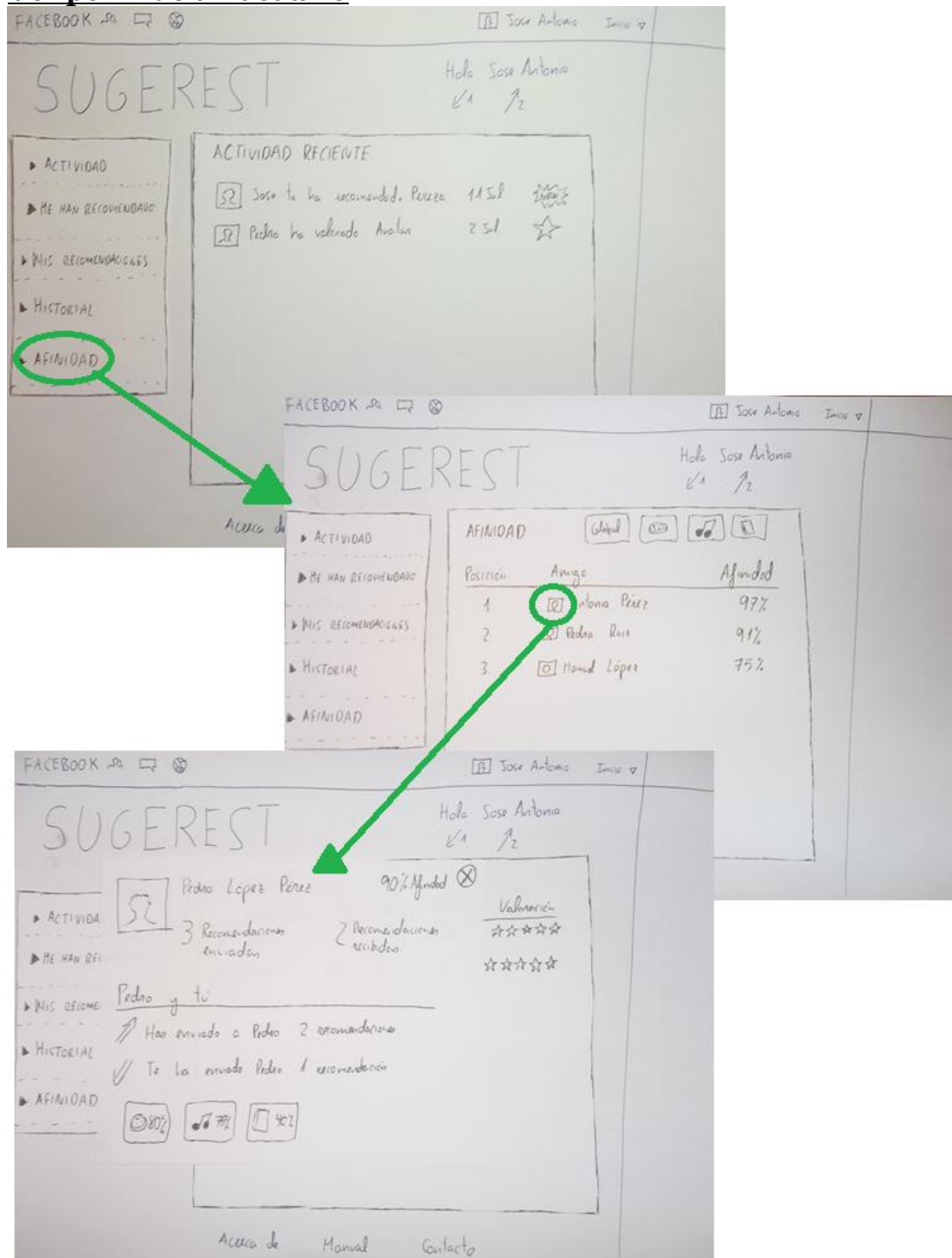


Figura 49 Storyboard de Ver perfil de usuario

**Valorar recomendación**

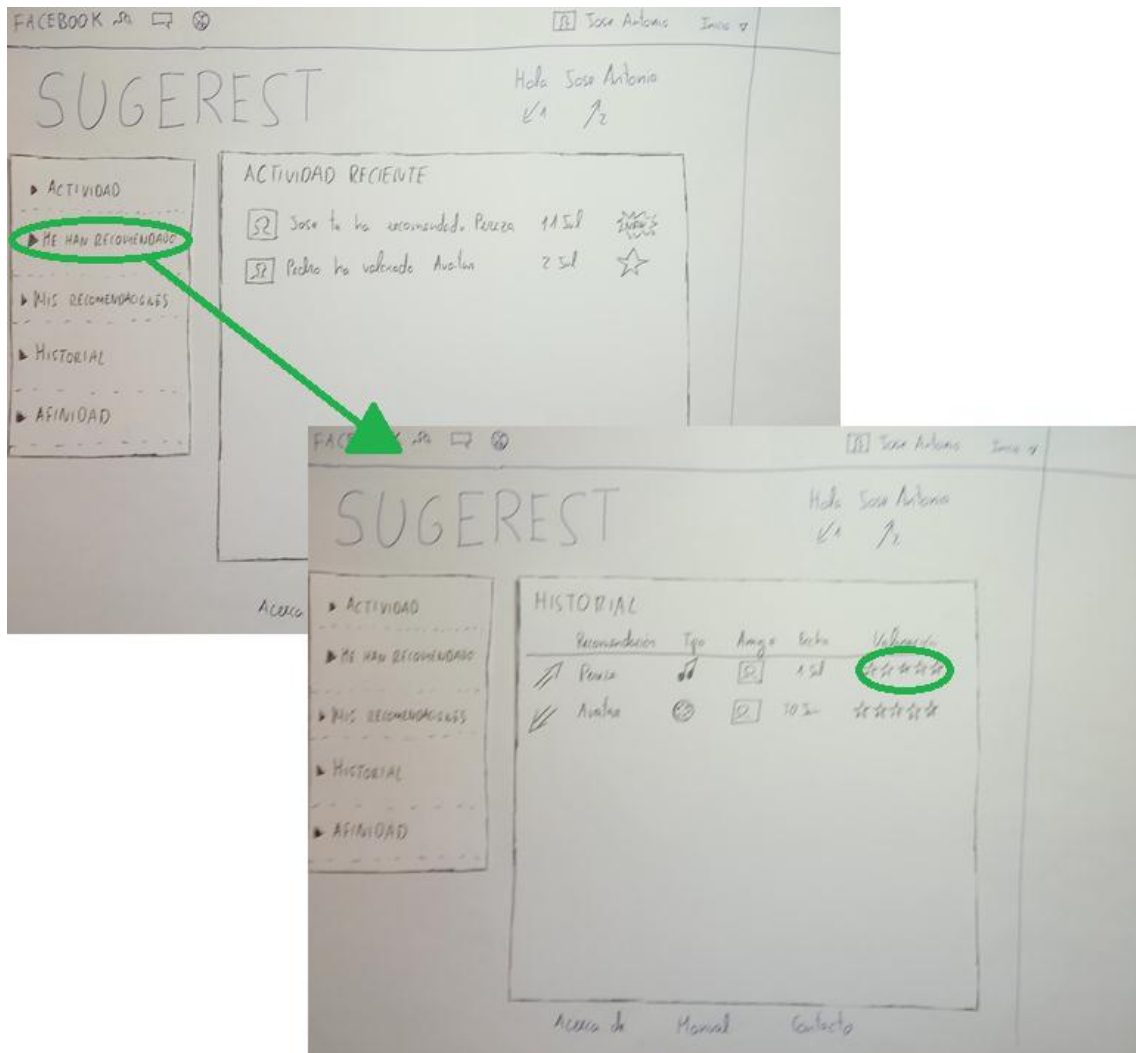


Figura 50 Storyboard de Valorar recomendación

## Ver ayuda

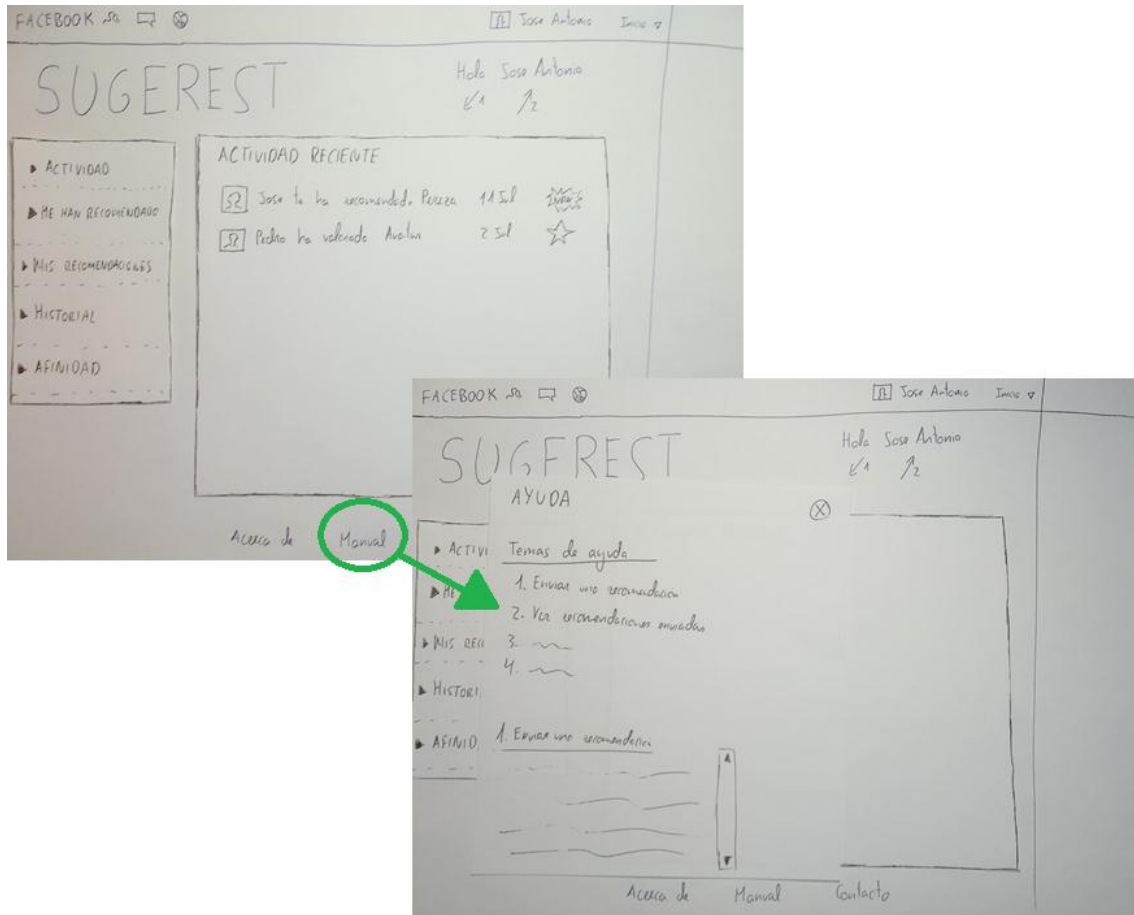


Figura 51 Storyboard de Ver ayuda

### 4.4.5.10 Mensajes de error

No se van a mostrar mensajes de error al usuario. La interfaz controlará que el usuario no llegue a un estado de error.

Esto adquiere especial relevancia en el formulario de Enviar recomendación, en la que se habían definido una serie de excepciones en los casos de uso que hay que controlar:

- El usuario no introduce nada en el campo de texto de la recomendación

- El usuario no introduce una valoración al ítem que va a recomendar
- El usuario no selecciona categoría
- El usuario no tiene amigos o no selecciona ninguno

La solución adoptada es deshabilitar por defecto el botón de Enviar recomendación. Cada vez que el usuario introduzca la información en alguno de los campos, se comprobará el estado de cada elemento del formulario. Si todo es correcto, se habilitará el botón para poder enviar la recomendación. En caso contrario, el botón permanecerá deshabilitado.

## **4.5 Implementación**

La actividad que finaliza el proceso de Ingeniería del Software es la implementación del modelo obtenido en las fases anteriores para convertirlo en código fuente. Para ello, hay que tener en cuenta aspectos como el tipo de arquitectura de la aplicación, el lenguaje de programación que más se adecue a los propósitos de la aplicación y la herramienta utilizada para generarla.

### ***4.5.1 Arquitectura de la aplicación***

Vamos a desarrollar una aplicación con una arquitectura cliente/servidor, con una interfaz en un entorno web para la comunicación con el usuario.

### ***4.5.2 Lenguajes y tecnologías utilizadas***

La aplicación será en un entorno web, por lo tanto uno de los lenguajes a utilizar será HTML. Aunque resulta obvio que para el funcionamiento requerido no es suficiente, sino que se necesita un lenguaje que genere contenido

dinámico del lado del servidor y que presente al usuario el contenido generado en formato HTML.

Para ello se va a utilizar el lenguaje PHP. La decisión de utilizar PHP se ha meditado entre las tres alternativas posibles más utilizadas hoy en día: PHP, Java y Ruby.

PHP es el lenguaje dominante en Internet. La mayoría de los sitios y aplicaciones web lo utilizan para la parte del lado del servidor. Es el que se ha escogido frente a los rivales por varios motivos:

- Es un lenguaje con el que ya se ha trabajado, por lo que no requiere demasiado esfuerzo adicional su aprendizaje para el desarrollo de éste proyecto.
- Es un lenguaje orientado a sitios y aplicaciones web, por lo que es bastante eficiente y su integración en el código HTML es muy directa.
- Facebook ofrece dos SDK para el desarrollo de aplicaciones para su red social: JavaScript SDK y PHP SDK. Por ello, el hecho de utilizar un lenguaje como PHP con el que Facebook nos provee el SDK será mas conveniente. Hay que indicar que también hay SDK para Java o Ruby, pero no son oficiales por lo que pueden estar más desactualizados o ser propensos a errores.

Para la parte de la interfaz y el diseño del aspecto visual de la aplicación se ha utilizado hojas de estilo CSS y JavaScript. Por un lado, hoy no se concibe un sitio web sin utilizar CSS. La tendencia de la web es la de separar claramente el contenido de la presentación del mismo. La página HTML debería contener solamente el contenido de la información que se mostrará, dejando a la hoja de estilo decidir que aspecto visual y presentación tendrá cada elemento en la página.

Por otro lado, JavaScript nos permite programar en el lado del cliente, siendo código que se ejecuta en el propio navegador. Esto nos permite, entre otras cosas, modificar aspectos de la interfaz sin recargar toda la página.

Se va a hacer uso de la librería JQuery, programada en JavaScript, con una gran cantidad de funciones de mucha utilidad tanto para mejorar la interfaz como para realizar peticiones dinámicas de contenidos al servidor mediante AJAX.

Vamos a describir un poco los lenguajes utilizados y su función en la aplicación:

### **HTML**

Es el lenguaje de marcado predominante para la elaboración de páginas web, describiendo la estructura y la información, así como añadiendo elementos multimedia como imágenes. Aunque tiene etiquetas que pueden describir la apariencia estética del documento, estas han caído en desuso dejando paso a las hojas de estilo CSS y, de hecho, van a ser removidas del nuevo estándar de HTML5. El texto se escribe en forma de etiquetas, englobadas en corchetes angulares: <body>, <html>, etc. En nuestra aplicación, el código HTML es lo que se muestra al usuario, aunque en gran parte es generado dinámicamente tras interpretar el código PHP en el servidor.

### **CSS**

Sus siglas vienen de Cascading Style Sheets, hojas de estilo en cascada. Se utiliza para definir la presentación de un documento HTML. Surge fundamentalmente para separar la información de la presentación. HTML estructura la información y CSS define que aspecto visual tendrá la misma.

El estándar actual es el 2.1, pero muchas de las funcionalidades del futuro estándar CSS3 ya han sido implementadas por las últimas versiones de los principales navegadores. Algunas de estas propiedades se utilizan en la aplicación.

Una hoja de estilos tiene un listado de reglas que se aplican a un elemento a través de un selector. El selector selecciona los elementos a los que se le aplicará un estilo concreto, y puede ser un elemento estándar html, un identificador, una clase, un tipo, etc.

Hay tres formas de dar formato con CSS a un documento HTML

- **Hoja de estilo externa:** Las reglas CSS están almacenadas en un archivo diferente al que contiene el código HTML, separando totalmente ambas partes. El fichero con las reglas CSS se incluye en la sección <head> del fichero HTML. Es la opción más elegante y apropiada de definir el estilo de la web.
- **Hoja de estilo interna:** Las reglas están incrustadas dentro del propio documento HTML, en una sección englobada por las etiquetas <style> dentro de la sección <head> del fichero HTML.
- **Un estilo en línea:** Se inserta el código dentro de una etiqueta HTML, a través de un atributo *style*. Es la menos recomendada, ya que añade mayor complejidad al código HTML y complica su mantenimiento.

En la aplicación se utilizan varias hojas de estilo. La principal contiene todo el estilo base de la aplicación, pero además se añaden otras hojas de estilo que se utilizan en algunos plugins de JQuery. Todas son hojas de estilo externas.

## PHP



Es un acrónimo “PHP: Hypertext Preprocessor”. Se trata de un lenguaje de alto nivel, con sintaxis similar a Java o C, orientado a especialmente al desarrollo web, aunque también siendo posible el desarrollo de aplicaciones de escritorio. Se integra en el código de una página HTML permitiendo la creación de sitios webs dinámicos, a través de las etiquetas de inicio `<?php` y de fin `?>`. Básicamente se pueden realizar tres tipos de desarrollos con PHP:

- **Scripts del lado del servidor:** Es la principal utilidad de PHP. Requiere de tres componentes: un servidor web, un navegador y un intérprete de PHP.
- **Aplicaciones en línea de comandos:** Se puede ejecutar un script desde la línea de comandos sin necesidad de un servidor web o un navegador. Un uso muy común es la ejecución de tareas planificadas con Cron en Linux, pero pueden realizar cualquier tarea de procesamiento.
- **Aplicaciones gráficas:** Aunque no es el uso principal y más extendido, es posible realizar aplicaciones gráficas de escritorio con PHP, gracias en gran parte al uso de la extensión PHP-GTK, no incluida en la distribución principal de PHP.

## JavaScript

JavaScript es un lenguaje de programación interpretado, utilizado principalmente en el lado cliente de las arquitecturas cliente-servidor. En un sitio web, permite mejorar la interfaz de usuario y hacer sitios dinámicos mediante la implementación del DOM (Document Object Model), que proporciona el conjunto de objetos que representan HTML y presenta una interfaz para acceder, añadir o modificar dichos elementos. Actualmente todos los navegadores interpretan el código JavaScript de las páginas web.

## **JQuery**

Es una biblioteca de JavaScript, que simplifica enormemente la manera interactuar con documentos HTML, manipulando sus elementos, manejando eventos, utilizando peticiones AJAX, etc. Permite simplificar tareas que de otra manera necesitarían bastante código JavaScript. La biblioteca es muy completa, y además, hay una gran comunidad desarrollando plugins basados en la misma. Para este proyecto, la lista de plugins de JQuery que se han utilizado es:

- **Fancybox:** Se trata de un plugin para mostrar contenido como imágenes o código HTML en una ventana emergente sobre el sitio o aplicación web. Es empleado en las secciones: Contacto, Acerca de, Manual de usuario y Perfil de usuario.
- **Rating:** Se trata de un plugin para votar o valorar un contenido. Sustituye un elemento de tipo radio de un formulario por la metáfora de las estrellas de valoración. Es usado por el usuario para valorar las recomendaciones recibidas.
- **Tipsy:** Es un plugin para mostrar tooltips con un estilo similar a los utilizados en Facebook al pasar el ratón por un elemento del sitio web. Es empleado para mostrar el nombre de un amigo al pasar el ratón por su imagen, o el nombre de la categoría al pasar el ratón por encima del icono.
- **TableSorter:** Es un plugin para ordenar tablas creadas en HTML. Permite tanto ordenar a priori una tabla, como hacerlo cuando el usuario pulse en el encabezado de una de las columnas de la tabla.

### ***4.5.3 Herramientas de desarrollo utilizadas***

Realmente, para escribir código HTML, CSS y PHP no es necesaria ninguna herramienta más allá de un editor de texto. Esto quizás será suficiente

para una pequeña página, o para un código muy simple. Pero para gestionar un proyecto algo más complejo, es de gran ayuda contar con una herramienta que nos ayude a la gestión y elaboración del código fuente. En este proyecto se ha utilizado Netbeans como herramienta de desarrollo.

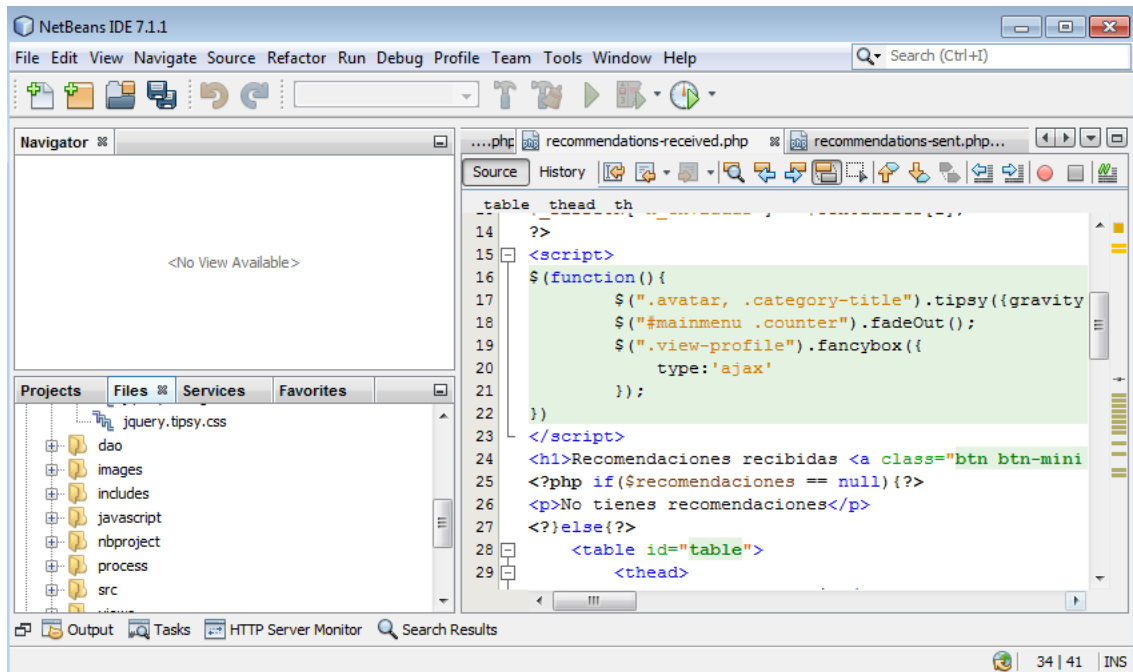


Figura 52 Pantalla principal de Netbeans

Netbeans es un entorno de desarrollo integrado multiplataforma, gratuito y de código abierto, escrito en Java, y con soporte para multitud de lenguajes de programación, como el propio Java, C, C++, PHP, etc.

Posee algunas características que ayudan en la codificación de la aplicación, como el resaltado de código y el autocompletado de funciones.

Por otro lado, para el manejo de la base de datos se ha utilizado el gestor con interfaz web phpMyAdmin

## 4.6 Pruebas

El objetivo que se persigue en esta fase es la realización de un conjunto de pruebas sobre el sistema completo para comprobar que no existen errores o situaciones no esperadas que disminuyan la calidad del software realizado. Por ello se realizarán pruebas que abarquen la mayor parte posible de la funcionalidad del sistema, provocando excepciones e intentando llegar a situaciones no deseadas.

### 4.6.1 Test de prueba

Cada test de prueba indicará el requisito (o requisitos) funcionales que va a comprobar. Los requisitos funcionales quedaron definidos en la sección “4.1.2 Requisitos funcionales”, pero por comodidad se añaden a continuación el código del requisito funcional y su nombre:

Código del requisito	Título
RF1	Alta de usuarios
RF2	Iniciar sesión en la aplicación
RF3	Enviar recomendaciones
RF4	Recibir recomendaciones
RF5	Descartar recomendaciones
RF6	Valorar recomendación
RF7	Modificar valoración
RF8	Consultar ayuda
RF9	Ver afinidad global con usuarios
RF10	Ver afinidad específica con usuarios
RF11	Calcular relevancia
RF12	Ver listado de recomendaciones recibidas

RF13	Ver listado de recomendaciones enviadas
RF14	Consultar historial de recomendaciones
RF15	Invitar amigos
RF16	Publicar en la biografía
RF17	Ver perfil de un amigo
RF18	Contactar con el administrador

**Tabla 4** Resumen de requisitos funcionales

**Test 1:** Acceso de usuario no registrado aceptando permisos

<b>Requisitos testeados</b>	RF1
<b>Acción del usuario</b>	Un usuario no registrado en la aplicación accede a la aplicación desde Facebook. Se le solicitan los permisos para acceder a la aplicación y los acepta
<b>Comprobación</b>	El usuario queda registrado en nuestra aplicación e inicia sesión en la aplicación.

**Test 2.** Acceso de usuario no registrado no aceptando permisos

<b>Requisitos testeados</b>	RF1
<b>Acción del usuario</b>	Un usuario no registrado en la aplicación accede a la aplicación desde Facebook. Se le solicitan los permisos pero los rechaza

<b>Comprobación</b>	El usuario no queda registrado en la aplicación. Se le pide de nuevo que acepte los permisos en caso de intentar volver a acceder a la aplicación.
---------------------	--

**Test 3:** Acceso de usuario registrado

<b>Requisitos testeados</b>	RF2
<b>Acción del usuario</b>	Un usuario registrado accede a la aplicación desde Facebook.
<b>Comprobación</b>	El usuario inicia sesión en la aplicación. Toda la información generada en anteriores accesos a la aplicación debe ser correcta

**Test 4:** Enviar una recomendación

<b>Requisitos testeados</b>	RF3
<b>Acción a realizar</b>	El usuario escoge la opción Nueva recomendación
<b>Comprobación 1</b>	El sistema debe mostrar el formulario para enviar una nueva recomendación, donde aparecerá un campo de texto para el nombre, una lista de categorías y una lista de amigos.
<b>Acción a realizar</b>	El usuario introduce el título,

selecciona categoría y amigo y pulsa en Enviar recomendación

**Comprobación 2**

El sistema almacena la nueva recomendación correctamente e informa visualmente al usuario de ello.

**Test 5:** Enviar una recomendación sin introducir ningún ítem a recomendar.

**Requisitos testeados**

RF3

**Acción a realizar**

El usuario escoge la opción Nueva recomendación

**Comprobación 1**

El sistema debe mostrar el formulario para enviar una nueva recomendación, donde aparecerá un campo de texto para el nombre, su valoración, una lista de categorías y una lista de amigos.

**Acción a realizar**

El usuario selecciona la categoría y el amigo y pulsa en Enviar recomendación

**Comprobación 2**

El sistema no debe permitir el envío hasta que no estén todos los campos completos

**Test 6:** Enviar una recomendación sin seleccionar ningún amigo

<b>Requisitos testeados</b>	RF3
<b>Acción a realizar</b>	El usuario escoge la opción Nueva recomendación
<b>Comprobación 1</b>	El sistema debe mostrar el formulario para enviar una nueva recomendación, donde aparecerá un campo de texto para el nombre, una lista de categorías y una lista de amigos.
<b>Acción a realizar</b>	El usuario introduce el nombre, selecciona la categoría y pulsa en Enviar recomendación
<b>Comprobación 2</b>	El sistema no debe permitir el envío hasta que no estén todos los campos completos

**Test 7:** Enviar una recomendación sin seleccionar ninguna categoría

<b>Requisitos testeados</b>	RF3
<b>Acción a realizar</b>	El usuario escoge la opción Nueva recomendación
<b>Comprobación 1</b>	El sistema debe mostrar el formulario para enviar una nueva recomendación, donde aparecerá un campo de texto para el nombre, una lista de categorías y una lista de amigos.
<b>Acción a realizar</b>	El usuario introduce el nombre y



selecciona el amigo y pulsa en Enviar recomendación

**Comprobación 2**

El sistema no debe permitir el envío hasta que no estén todos los campos completos

**Test 8:** Enviar una recomendación con un ítem igual o similar al enviado previamente por otro usuario

**Requisitos testeados**

RF3

**Acción a realizar**

El usuario escoge la opción Nueva recomendación

**Comprobación**

El sistema debe mostrar el formulario para enviar una nueva recomendación, donde aparecerá un campo de texto para el nombre, una lista de categorías y una lista de amigos.

**Acción a realizar**

El usuario introduce el nombre del ítem a recomendar, selecciona categoría y un amigo; y pulsa en Enviar recomendación

**Comprobación 2**

El sistema comprueba existe ese ítem y lo utiliza para la recomendación del usuario sin generar uno nuevo en la tabla de ítems.

**Test 9:** Valorar una recomendación recibida.

<b>Requisitos testeados</b>	RF6
<b>Acción a realizar</b>	El usuario escoge la opción de ver Recomendaciones recibidas
<b>Comprobación 1</b>	El sistema muestra la lista de recomendaciones recibidas con un campo para que el usuario escoja la valoración
<b>Acción a realizar</b>	El usuario escoge la valoración deseada
<b>Comprobación 2</b>	El sistema almacena la valoración, actualiza la afinidad con su amigo, notifica al amigo de que el ítem ha sido valorado y muestra confirmación visual al usuario.

**Test 10:** Descartar una recomendación recibida.

<b>Requisitos testeados</b>	RF5
<b>Acción a realizar</b>	El usuario escoge la opción de ver Recomendaciones recibidas
<b>Comprobación 1</b>	El sistema muestra la lista de recomendaciones recibidas con una opción de descartar la recomendación
<b>Acción a realizar</b>	El usuario escoge la opción de descartar recomendación
<b>Comprobación 2</b>	El sistema elimina la recomendación

	de la lista de recomendaciones recibidas del usuario y notifica al usuario origen de esa recomendación.
--	---

**Test 11:** Modificar una valoración

<b>Requisitos testeados</b>	RF7
<b>Acción a realizar</b>	El usuario escoge la opción de ver el Historial
<b>Comprobación 1</b>	El sistema presenta el historial de recomendaciones enviadas y recibidas
<b>Acción a realizar</b>	El usuario escoge una recomendación recibida y escoge una nueva valoración
<b>Comprobación 2</b>	El sistema almacena la nueva valoración, actualiza la afinidad entre usuarios y notifica al usuario que envió la recomendación.

**Test 12:** Comprobar incremento de afinidad con amigo

<b>Requisitos testeados</b>	RF7, RF9
<b>Acción a realizar</b>	El usuario escoge la opción de ver el Historial
<b>Comprobación 1</b>	El sistema presenta el historial de recomendaciones enviadas y recibidas
<b>Acción a realizar</b>	El usuario modifica la valoración sobre una recomendación recibida

---

	incrementando el valor de ésta.
<b>Comprobación 2</b>	El sistema incrementa la afinidad entre ambos usuarios.

---

**Test 13:** Invitar amigos a la aplicación.

---

<b>Requisitos testeados</b>	RF13
<b>Acción a realizar</b>	El usuario escoge la opción de invitar amigos
<b>Comprobación 1</b>	El sistema solicita a Facebook que genere el diálogo de Invitar amigos.
<b>Acción a realizar</b>	El usuario selecciona los amigos que desea invitar
<b>Comprobación 2</b>	Los usuarios solicitados reciben una solicitud de ingreso a Sugerest.

---

**Test 14:** Comprobar contadores de recomendaciones

---

<b>Requisitos testeados</b>	RF3, RF4
<b>Acción a realizar</b>	El usuario envía una recomendación
<b>Comprobación 1</b>	El sistema incrementa en uno el número de recomendaciones enviadas
<b>Acción a realizar</b>	El usuario recibe una recomendación
<b>Comprobación 2</b>	El sistema incrementa en uno el número de recomendaciones recibidas

---

**Test 15:** Ver perfil de un usuario amigo

<b>Requisitos testeados</b>	RF15
<b>Acción a realizar</b>	El usuario escoge la opción Afinidad
<b>Comprobación 1</b>	El sistema le muestra la lista de sus amigos ordenada por afinidad con el usuario
<b>Acción a realizar</b>	El usuario pulsa sobre la imagen de un amigo tanto global como en esa categoría
<b>Comprobación 2</b>	El sistema le muestra el perfil del usuario en una ventana superpuesta con la información del mismo, afinidad entre ellos y recomendaciones mutas que se han enviado.
<b>Acción a realizar</b>	El usuario cierra la ventana
<b>Comprobación 3</b>	El sistema sigue en la pantalla previa al pulsar sobre la imagen del amigo.

**Test 16:** Calcular relevancia de recomendación

<b>Requisitos testeados</b>	RF11
<b>Acción a realizar</b>	El selecciona Recomendaciones recibidas
<b>Comprobación 1</b>	El sistema incrementa muestra la lista de recomendaciones ordenadas por la relevancia calculada en función de varios factores entre el usuario origen

y destino.

#### 4.6.2 Resultados obtenidos

Se han obtenido los siguientes resultados en los test de prueba:

Test 1	
Comprobación	Correcta
Test 2	
Comprobación	Correcta
Test 3	
Comprobación	Correcta
Test 4	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 5	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 6	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 7	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 8	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 9	

Comprobación 1	Correcta
Comprobación 2	Correcta
Test 10	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 11	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 12	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 13	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 14	
Comprobación 1	Correcta
Comprobación 2	Correcta
Test 15	
Comprobación 1	Correcta
Comprobación 2	Correcta
Comprobación 3	Correcta
Test 16	
Comprobación 1	Correcta

**Tabla 5 Resultados test de pruebas**

Todos los resultados de las pruebas han sido satisfactorios y el sistema se comporta como se esperaba en cada uno de los casos que han sido probados, abarcando la mayoría de las funcionalidades de la aplicación.

### 4.6.3 Validación

Un aspecto muy importante en un sitio web o aplicación web es que cumpla con los estándares de HTML y CSS. De este modo, cualquier usuario que disponga de un navegador que cumpla con los estándares.

Para la realización de las pruebas se van a utilizar las herramientas proporcionadas por el W3C (World Wide Web Consortium). El World Wide Web es un consorcio internacional que produce recomendaciones para la web. Está dirigido por Tim Berners-Lee, el creador del protocolo HTTP, la URL y el lenguaje de marcado HTML que constituye la base de la web.

Vamos a utilizar dos herramientas:

- **Markup Validation Service:** Se encarga de validar documentos XHTML, HTML, MathML... En nuestro caso, comprobaremos que la aplicación cumpla con los estándares de XHTML. La dirección url es: <http://validator.w3.org/>
- **CSS Validation Service:** Se encarga de validar la hoja de estilos CSS incrustada en un documento HTML o XHTML. Se recomienda primero comprobar que la construcción con el lenguaje de marcado sea correcta y después analizar la hoja de estilos. La dirección url es: <http://jigsaw.w3.org/css-validator/>

Para validar el código HTML, Markup Validation Service nos ofrece tres posibles opciones:

- Ingresar directamente la URL del sitio
- Cargar el archivo HTML a analizar
- Pegar directamente el código HTML a ser analizado.



En nuestro caso, vamos a utilizar la tercera opción. La primera opción no es posible al utilizar el sitio sesiones y generarse el contenido dinámicamente para el usuario. Para poder generar el código HTML de una página es requisito estar logueado en Facebook e iniciar sesión en la aplicación.

La segunda opción también la descartamos debido a que la aplicación está escrita en PHP, código que se ejecuta del lado del servidor. El código HTML es el resultado que envía al navegador, pero el fichero debe ser procesado previamente por el parseador de PHP.

Por tanto, la opción más viable es la de pegar directamente el código, para ello pulsamos en la opción *Validate by direct input*.



The image shows the W3C Markup Validation Service interface. At the top, there is a blue header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below the header, there are three tabs: "Validate by URI", "Validate by File Upload", and "Validate by Direct Input". The "Validate by Direct Input" tab is highlighted with an orange border. Below the tabs, the "Validate by direct input" section is visible, featuring a text input field with the placeholder "Enter the Markup to validate:". Below the input field, there is a "More Options" link and a "Check" button.

Figura 53 Validar código HTML

Una vez pegado el código en el área de texto, se pulsa el botón Check y se procederá a la validación del código HTML.

En nuestro caso, lo ha validado correctamente para el estándar XHTML 1.0 estricto, por lo que cualquier navegador que cumpla con éste estándar deberá mostrar correctamente la aplicación. Solo muestra un *warning* informando que el validador da por supuesto que el código pegado en el área de texto está en formato UTF-8.

The image shows the W3C Markup Validation Service interface. At the top, there is a blue header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below the header, there are navigation links: "Jump To:", "Notes and Potential Issues", and "Congratulations · Icons". A green banner states "This document was successfully checked as XHTML 1.0 Strict!". Below this, a table displays the validation results:

<b>Result:</b>	Passed, 1 warning(s)	
<b>Source :</b>	<pre>&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"&gt; &lt;html xmlns="http://www.w3.org/1999/xhtml"&gt; &lt;head&gt; &lt;title&gt;Sugerest - Envía recomendaciones a tus amigos&lt;/title&gt; &lt;meta http-equiv="Content-Type" content="text/html; charset=utf-8" /&gt; &lt;link rel="stylesheet" href="./css/estilo_base.css" type="text/css" media="screen" /&gt;</pre>	
<b>Encoding :</b>	utf-8	<input type="text" value="utf-8 (Unicode, worldwide)"/>
<b>Doctype :</b>	XHTML 1.0 Strict	<input type="text" value="(detect automatically)"/>

Figura 54 Resultado correcto de la validación HTML

Una vez validado el código HTML, vamos a comprobar que la hoja de estilos sea también válida. En este caso, si podemos indicar la dirección URL directamente, ya que solo necesita obtener la dirección de la hoja de estilos para su análisis. Introducimos la dirección del servidor donde está alojada la aplicación y se pulsa al igual que en el caso anterior en Check.



Figura 55 Validación de código css

El resultado es satisfactorio y nuestro código cumple con los estándares de CSS3.

#### 4.6.4 Visualización en distintos navegadores

Es importante comprobar que la aplicación se visualiza correctamente en los navegadores más extendidos. Aunque cumpla los estándares, es conveniente revisar que el aspecto visual sea similar, ya que algunas propiedades no son tratadas del mismo modo en diferentes navegadores, sobre todo las nuevas propiedades de CSS3.

Se van a realizar pruebas en los siguientes navegadores:

- Chrome

- Internet Explorer
- Firefox

Las pruebas se consideraran exitosas si no hay diferencias visuales graves a la hora de visualizar la aplicación en diferentes navegadores. Si hay diferencias visuales mínimas se consideran aceptables siempre y cuando no afecten a la usabilidad de la aplicación o empeoren la apariencia estética. Las siguientes imágenes muestran la apariencia física en los distintos navegadores:

### Chrome

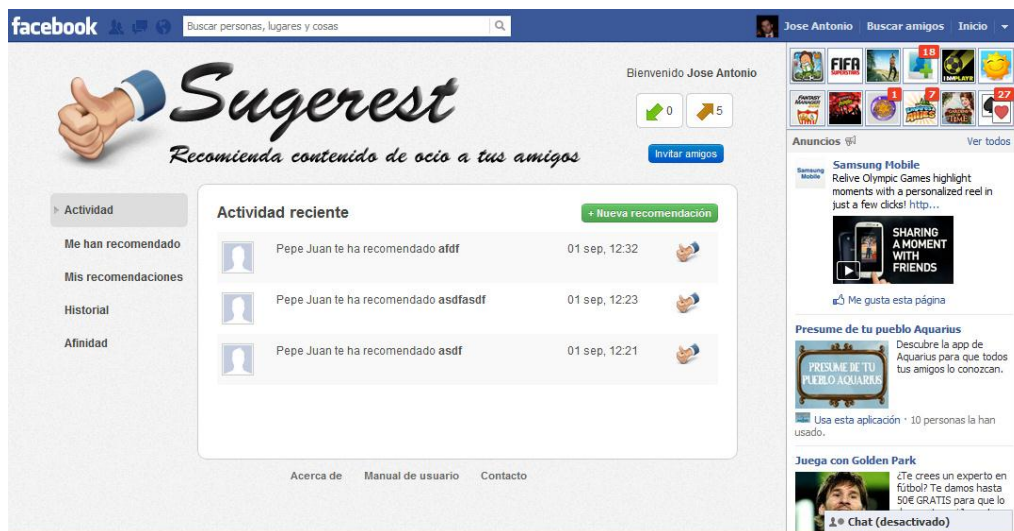


Figura 56 Aplicación en navegador Chrome

### Internet explorer

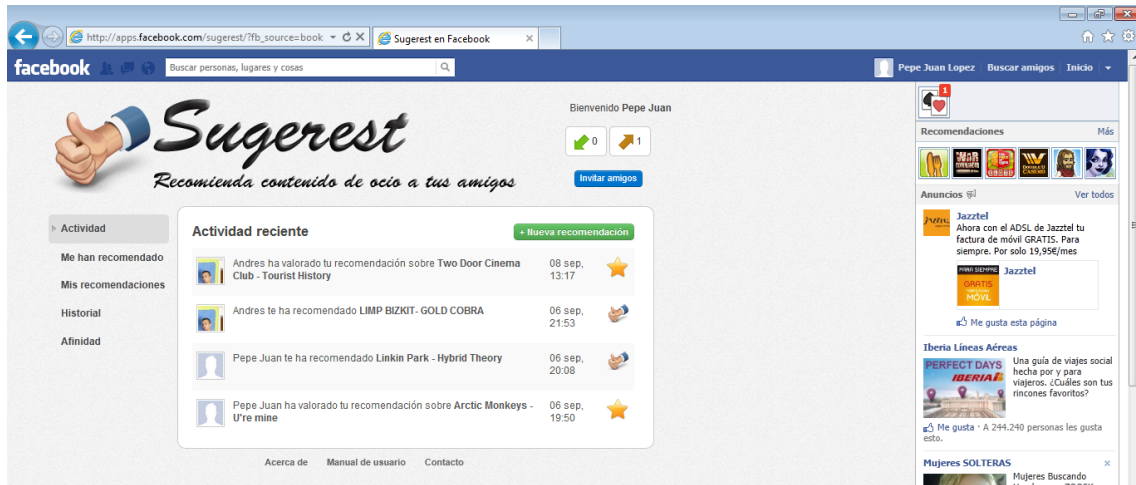


Figura 57 Aplicación en Internet Explorer

## Firefox

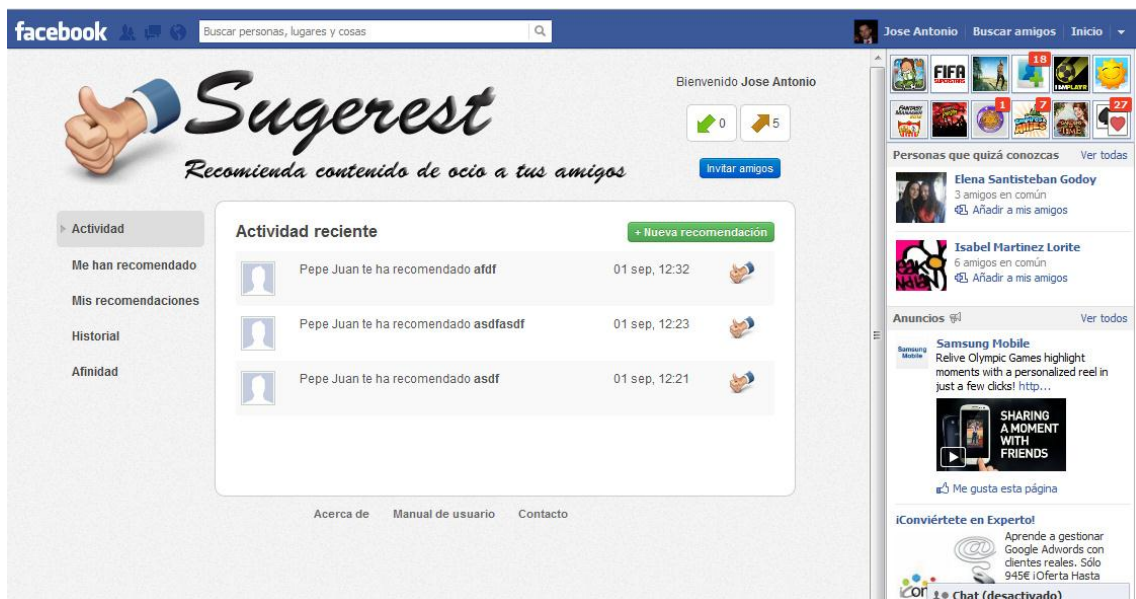


Figura 58 Aplicación en navegador Opera

## **4.7 Instalación de la aplicación**

La instalación de la aplicación tiene dos partes. Por un lado, la parte en torno a Facebook que requiere el registro de la aplicación en su herramienta de Desarrolladores, con el fin de obtener el identificador de la aplicación y el código secreto para poder realizar llamadas a la API de Facebook desde la propia aplicación.

Cuando registramos la aplicación en Facebook, hay que indicar la dirección desde la cual obtendrá el canvas. Por tanto, hay que subir la aplicación a un servidor web y crear la estructura de base de datos.

El proceso de registro de la aplicación en Facebook quedó documentado en el apartado “3.4 Registrando la aplicación de Sugerest”.

La instalación de la aplicación en el del servidor y la creación de la base de datos viene documentada paso a paso en el Anexo I. Por su parte, en el Anexo II se encuentran disponible el manual de usuario de la aplicación.

# 5. Conclusiones

---

## 5.1. Conclusiones

La realización de este proyecto fin de carrera surgió de la necesidad de gestionar las recomendaciones que nuestros amigos y conocidos nos realizan en el día a día. Básicamente, tenía que solucionar dos problemas: la gestión de las recomendaciones por parte de tus amigos y calcular si esa recomendación podría gustarnos en base a recomendaciones pasadas entre esos usuarios.

Se optó por la integración en Facebook, ya que ofrecía varias ventajas como analizó en el apartado 2.2. Y además de las ventajas, era bastante interesante estudiar la API de Facebook y desarrollar una aplicación integrada en ella, ya que es algo bastante novedoso y muy popular a día de hoy.

Esto ha requerido un estudio previo de esta red social y su servicio a desarrolladores. Una pequeña barrera ha sido el constante cambio de la API de Facebook, que hace que libros de hacer un par de años estén obsoletos, así como artículos y foros de discusión.

Además del desarrollo de aplicaciones en Facebook, las tecnologías usadas apenas se han visto en la carrera, solo algunas pinceladas básicas. Por ello, se ha tenido que profundizar en campos como diseño web, programación del lado de servidor con PHP, etc.

Tras el estudio de los conocimientos previos que eran necesarios para el desarrollo del proyecto, se han realizado las fases propias de Ingeniería del Software, para finalizar probando la aplicación con usuarios reales, haciendo que esté disponible en Facebook para cualquier usuario. Gracias a estos, se han ido descubriendo algunos fallos en la aplicación que han sido corregidos.



## 5.2 Desarrollos futuros

La aplicación desarrollada en este proyecto es totalmente funcional y cumple con los objetivos fijados. Pero es posible extender la funcionalidad de la misma.

En la versión actual de la aplicación, un usuario tiene un círculo de amigos con los cuales intercambiar recomendaciones y tiene una cierta afinidad. Por tanto, cada usuario, tendrá un grafo de usuarios y afinidades. Cada grafo es aislado para ese usuario y solo incluye sus amigos.

Hay una serie de ítems recomendados que estarán presentes en más de un círculo de amigos, por ser películas de estreno o best sellers. Estos ítems serán recomendados por gente de círculos distintos sin ninguna conexión entre sí, pero está claro que si recomiendan lo mismo, tendrían una buena afinidad en caso de ser amigos.

Un posible desarrollo futuro sería establecer un mecanismo para agrupar usuarios en función de los ítems similares que hayan recomendado a sus amigos. De esa manera, la aplicación podría ofrecer al usuario información interesante del estilo: los diez usuarios más afines a ti de España, permitiendo incluso que ambos usuarios se conozcan y se añadan como contactos en la red social.

Otra posibilidad de ampliación futura sería permitir al usuario la creación de categorías personalizadas para sus recomendaciones. Un usuario que tenga amigos a los que le guste esquiar, puede tener la necesidad de recomendarle estaciones de esquí. El sistema le permitiría crear la categoría personalizada *Estaciones de esquí* la cual sería visible por sus amigos cuando él envíe una recomendación.



# Referencias y bibliografía

---

## **Libros**

Elmasri, R., & Navathe, S. B. (2000). *Sistemas de Bases de Datos. Conceptos fundamentales. 2ª edición*. Addison-Wesley.

Ivar Jacobson, G. B. (2001). *El proceso Unificado de Desarrollo de Software*. Pearson Educacion, S.A.

Martin Fowler, K. S. (1999). *Uml gota a gota*. Mexico: Addison Wesley Longman de México, S.A.

Pérez, J. E. (2008). *Introducción a CSS*.

Pérez, J. E. (2008). *Introducción a XHTML*.

Pérez, J. E. (2010). *Introducción a AJAX*.

Stay, J. (s.f.). *Facebook Application Development for Dummies*. Wiley Publishing.

## **Enlaces web**

Facebook Developers (<https://developers.facebook.com/>). Documentación oficial de Facebook para desarrolladores.

# **Anexo I: Manual de instalación**

---

## A.1 Instalación en el servidor

La aplicación de Sugerest está desarrollada en PHP y hace uso de una base de datos MySQL, por lo que necesitaremos:

- Un servidor web con soporte para PHP4 o superior.
- Una base de datos MySQL con un mínimo de 50mb para almacenar la información generada en la aplicación.

En el cd adjunto a esta memoria, se podrá encontrar la siguiente estructura de carpetas:

- **doc**: ficheros de documentación del proyecto.
- **source**: código fuente de la aplicación.
- **sql**: fichero de texto con las consultas necesarias para construir la estructura de tablas de la aplicación.

Nombre	Fecha de modifica...	Tipo	Tamaño
doc	02/09/2012 16:34	Carpeta de archivos	
source	02/09/2012 16:34	Carpeta de archivos	
sql	02/09/2012 16:34	Carpeta de archivos	

Figura 59 Ficheros de la aplicación

Un paso previo a subir los archivos, es editar el fichero con los datos de la conexión a la base de datos MySQL. Estos datos se encuentran en el fichero “conexion.php” contenido en la carpeta /includes. Se puede editar con cualquier editor de texto plano. Una vez abierto, hay que modificar los siguientes campos:

```
1  <?
2
3
4  $link = mysql_connect("SERVIDOR", "USUARIO", "CONTRASEÑA")
5      or die(mysql_error());
6  mysql_select_db("NOMBREDB", $link) or die(mysql_error());
7
8
9  ?>
```

Figura 6o Configurando el fichero de conexión a base de datos

- **Servidor:** dirección del servidor donde está alojada la base de datos.
- **Usuario:** nombre de usuario de la base de datos.
- **Contraseña:** contraseña asociada al usuario anterior.
- **NombreDB:** nombre de la base de datos creada para la aplicación

Toda esta información nos es suministrada por el proveedor del servicio de bases de datos, que normalmente será el mismo proveedor con el que se ha registrado previamente el servidor web.

Una vez configurado este fichero, procedemos a guardar los cambios y a subir todo el código fuente de Sugerest al servidor. Para ello, se puede utilizar alguna herramienta software que haga la conexión FTP con el servidor. Un ejemplo de software es Filezilla, proyecto de código abierto de la fundación Mozilla. Se puede descargar desde <http://filezilla-project.org/>.

Una vez abierto, el programa presenta la siguiente apariencia:

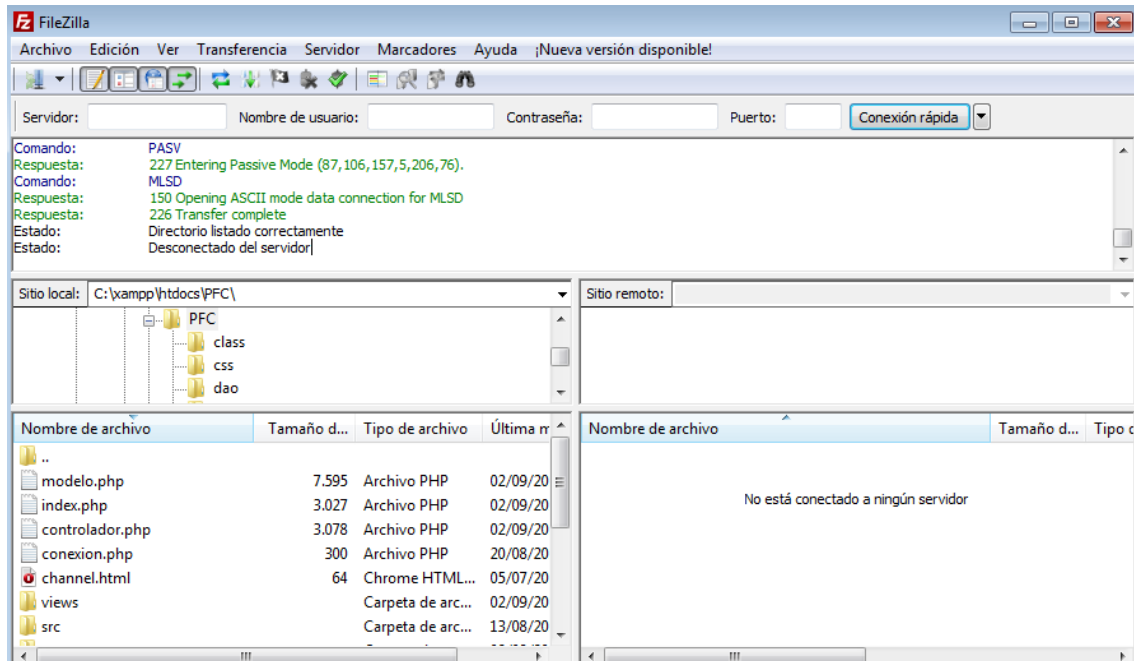


Figura 61 Pantalla inicial de Filezilla

Podemos distinguir tres zonas:

- **El panel central:** Muestra las peticiones que se van realizando al servidor de conexión, ficheros, etc. Si, por ejemplo, la contraseña de acceso al servidor es incorrecta, se notifica en este recuadro.
- **El panel izquierdo:** Muestra los ficheros locales del usuario en su máquina. En nuestro caso, en estructura de carpetas superior habría que ubicar la carpeta con el código fuente de la aplicación. Una vez seleccionada arriba la carpeta, en la parte inferior aparecen todos los ficheros que contiene la misma.
- **El panel derecho:** Muestra los ficheros existentes en el servidor. Inicialmente aparece en blanco debido a que no estamos aún conectados a ningún servidor.



Para realizar la conexión al servidor, debemos añadir la información de acceso del servidor. Se puede realizar pulsando en Gestor de sitios dentro del menú Archivo. Una vez en la ventana de Gestor de sitios, se pulsa en Nuevo sitio.

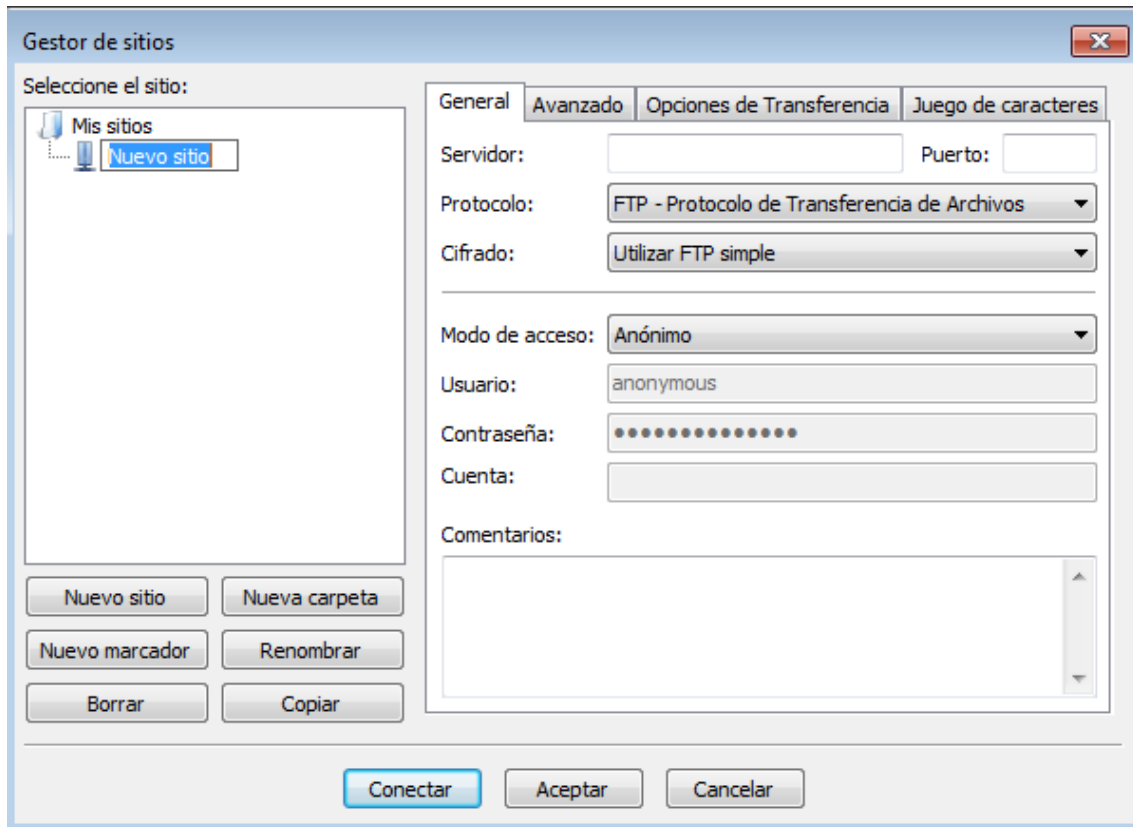


Figura 62 Introducir nuevo sitio en Filezilla

Aparecerá en la parte derecha el formulario para introducir la información del servidor. Los campos básicos que son necesarios son:

- **Servidor:** dirección del servidor web que hemos seleccionado.
- **Puerto:** Puerto FTP del servidor web. Si se deja en blanco, se utiliza el puerto por defecto (21).

- **Modo de acceso:** Por defecto viene seleccionado anónimo, pero lo normal es que el que nos ofrezca el servicio de alojamiento web nos facilite un usuario y contraseña para acceder. Por tanto, se selecciona en Modo de acceso la opción “Normal”.
- **Usuario:** Usuario facilitado por el servicio de alojamiento web.
- **Contraseña:** Contraseña facilitada por el servicio de alojamiento web para ese usuario.

Una vez completados los campos se pulsa en Aceptar y quedará registrado el sitio. Para conectarse rápidamente basta con pulsar en la pestaña del primer icono de la lista de botones de Filezilla, y seleccionar el sitio recién añadido.

Una vez conectados, veremos la estructura de carpetas existente en el servidor en el panel derecho. Procedemos a arrastrar todo el contenido de la carpeta *source* con el código fuente de la aplicación al servidor, ya sea arrastrando y soltando o seleccionando todo y pulsando en Subir en el menú contextual.

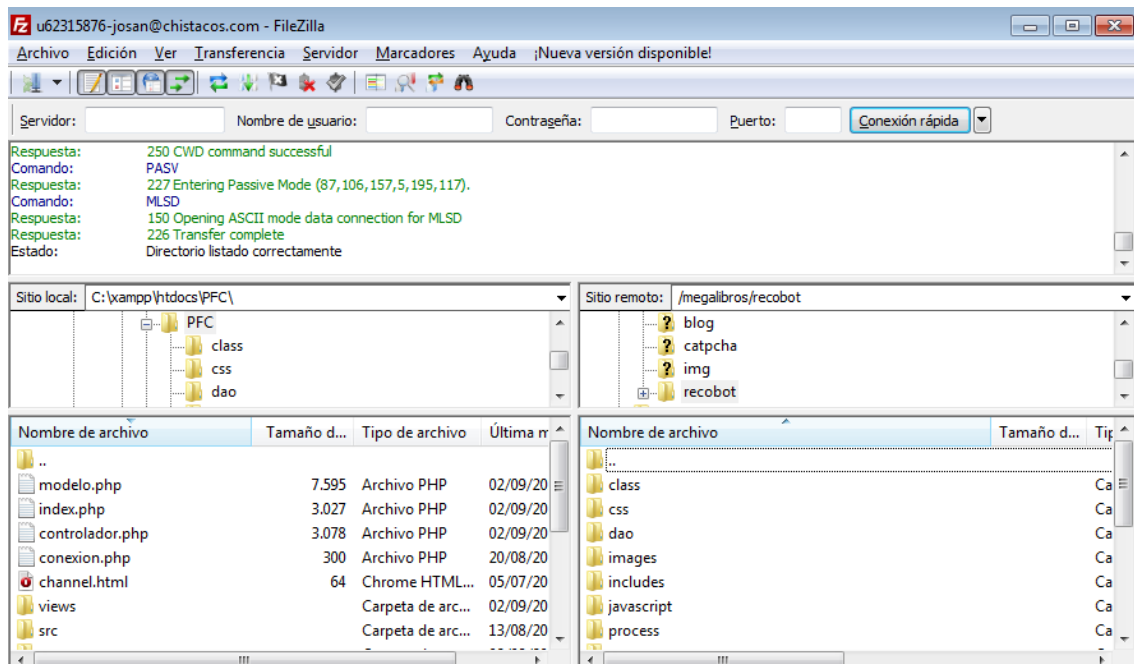


Figura 63 Conexión con el servidor en Filezilla

Una vez terminado, ya tendremos el código fuente de Sugerest subido en el servidor.

El siguiente paso es importar la estructura de la base de datos que utilizará la aplicación. Para ello, desde el panel de administración de la base de datos de tu proveedor de alojamiento, hay que importar el fichero con la estructura que utiliza Sugerest. La mayoría de proveedores ofrecen como interfaz de gestión de la base de datos el software *phpMyAdmin*, por lo que va a ser explicado utilizando este gestor.

Una vez accedemos a la pantalla inicial de *phpMyAdmin*, nos dirigimos al apartado del menú superior titulado SQL.



Figura 64 Consulas SQL en phpMyAdmin

Una vez en este apartado, nos aparecen varias opciones para introducir el código SQL mediante texto o para importar un archivo. Escogemos la segunda opción, dejando la compresión en *Autodetectar*, y pulsamos en Seleccionar archivo. Escogemos el fichero que hay en la carpeta sql de este proyecto con el nombre de *Sugerest\_bbdd.sql*, y pulsamos en Continuar. Este fichero contiene todo el código SQL necesario para la creación de las tablas de la aplicación.

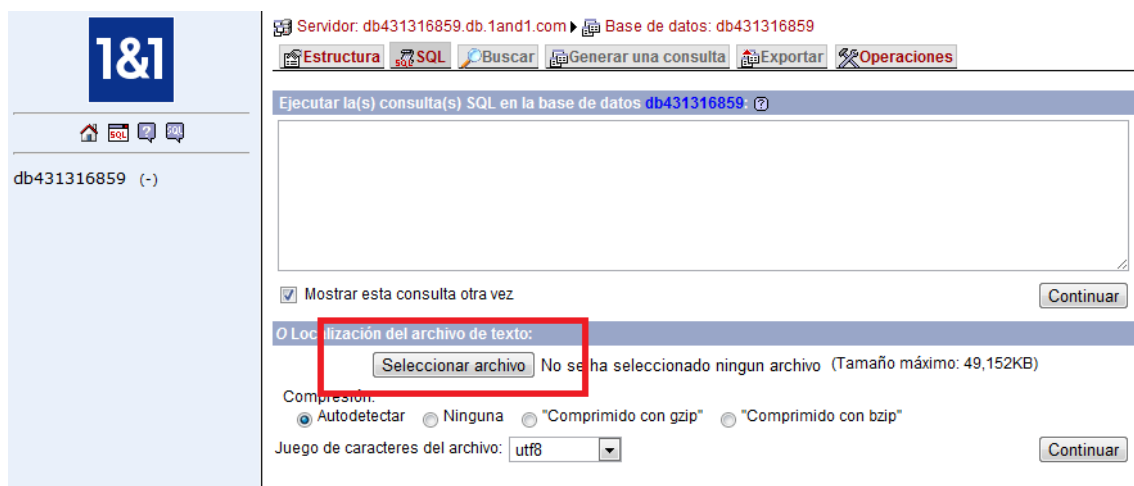


Figura 65 Importar archivo SQL

Si todo ha ido bien, nos informará de que la consulta se ejecutó con éxito y veremos en la parte izquierda las tablas de la aplicación ya creadas.



Servidor: db431316859.db.1and1.com ▶ Base de datos: db431316859

Estructura SQL Buscar Generar una consulta Exportar Operaciones

Su consulta se ejecutó con éxito

consulta SQL:

```
-- phpMyAdmin SQL Dump
-- version 2.6.4-pl3
-- http://www.phpmyadmin.net
--
-- Servidor: db427382039.db.1and1.com
-- Tiempo de generación: 10-09-2012 a las 16:28:21
-- Versión del servidor: 5.0.91
-- Versión de PHP: 5.3.3-7+squeeze14
--
-- Base de datos: `db427382039`
--
-- Estructura de tabla para la tabla `actividad`
--
CREATE TABLE `actividad` (
  `id_actividad` INT(11) NOT NULL AUTO_INCREMENT,
  `id_recomendacion` INT(11) NOT NULL,
  `tipo_accion` INT(11) NOT NULL,
  `id_item` INT(11) NOT NULL,
  `fecha` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_actividad`),
  KEY `id_recomendacion` (`id_recomendacion`),
  KEY `id_item` (`id_item`)
) ENGINE = INNODB AUTO_INCREMENT = 97 DEFAULT CHARSET = utf8 # MySQL ha devuelto un valo
```

Figura 66 Base de datos creada con éxito

Una vez creada la base de datos, la aplicación ya debería estar operativa si se han seguido todos los pasos, desde su registro en Facebook hasta aquí.

## A.2 Instalación en el cliente

El requisito que tiene que cumplir el cliente para comenzar a utilizar la aplicación es disponer de un navegador web que cumpla con los estándares y una cuenta en la red social Facebook.

Para comenzar a utilizar la aplicación, lo normal es buscarla en la barra superior de Facebook, introduciendo el nombre (Sugerest). Hasta que la aplicación no tiene diez usuarios mensuales, Facebook no la incluye en el índice de aplicaciones. Mientras no esté indexada, la manera más sencilla de acceder es a través de la url <https://apps.facebook.com/sugerest>.

La primera vez que se inicie la aplicación, el usuario debe aceptar los permisos que requiere la aplicación. En una pantalla inicial, debe aceptar que la aplicación acceda a su información básica (nombre, sexo, edad, amigos, etc.).



Figura 67 Aceptando permisos básicos para Sugerest

Si la aplicación necesita algunos permisos extendidos, se solicitan al pulsar en “Ir a la aplicación”. En el caso de Sugerest, se necesita un permiso de publicación en la biografía del usuario cuando se le envía una nueva recomendación.

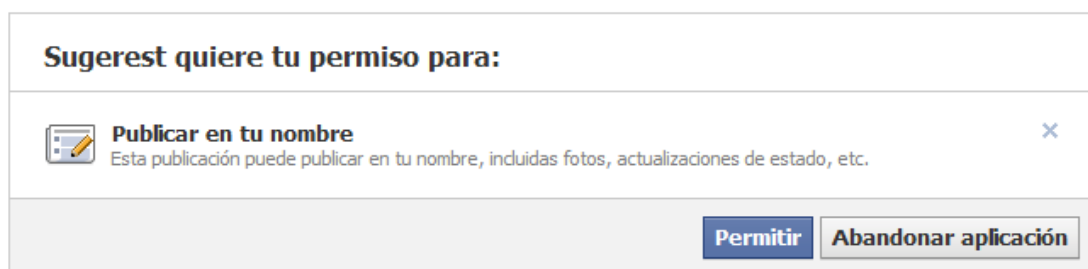


Figura 68 Aceptando permisos extendidos en Sugerest

Una vez que el usuario acepte estos permisos, ya tendrá la aplicación instalada en su cuenta para comenzar a ser utilizada. Se le añadirá un acceso directo en su lista de Marcadores y le aparecerá en las primeras posiciones del formulario de búsqueda en la barra superior, incluso aunque la aplicación aún no esté indexada en el listado de aplicaciones.





# **Anexo II: Manual de usuario**

---

A través de este manual de usuario se va a hacer un recorrido guiado a través de todas las funcionalidades de la aplicación. Ante todo, definir que es Sugerest:

*Sugerest es una aplicación integrada en Facebook que te permite recomendar contenido de ocio a tus amigos, como pueden ser películas o música.*

Seguro que alguna vez le has recomendado a un amigo o un familiar que viera determinada película o leyera algún libro. Sugerest intenta trasladar estas clásicas recomendaciones a una aplicación informática que permita: por un lado, gestionar todas las recomendaciones que te envían a tus amigos y las que tu envías a ellos; y por otro lado, obtener una retroalimentación sobre si la recomendación ha sido del agrado del usuario al que iba destinada, almacenando dicha valoración y estableciendo un grado de afinidad con el tiempo.

## **1. Primeros pasos en Sugerest**

Cuando entramos por primera vez a la aplicación, puede darse dos casos diferenciados: que ya tengamos amigos en la aplicación o que no tengamos. En función de ello, la pantalla de bienvenida será distinta, ya que para poder enviar una recomendación tenemos que tener algún amigo que utilice la aplicación.

Si tenemos amigos, se nos mostrará la pantalla siguiente:



**Figura 69 Bienvenida a Sugerest con amigos**

En ella, se nos informa de los amigos de Facebook que ya están utilizando Sugerest y se nos invita a enviarles una recomendación.

Por el contrario, si no tenemos amigos, la pantalla de bienvenida es la siguiente:



**Figura 70 Bienvenida a Sugerest sin amigos**

Si no tenemos amigos en la aplicación no podremos enviarles recomendaciones ni que nos envíen, por lo tanto lo principal es que podamos invitar a nuestros amigos a utilizar la aplicación. Además, se nos muestra un enlace al manual para poder obtener más información de la aplicación.

Además, no solo podemos invitar amigos desde esta pantalla, sino que en la parte superior derecha hay un botón permanente para invitar amigos a la aplicación. Si pulsamos en el botón en cualquiera de los botones de “Invitar amigos”, se nos muestra el selector de amigos para enviarles las invitaciones.



**Figura 71 Invitar amigos a Sugerest**

Una vez seleccionados todos los amigos que deseamos invitar, pulsamos en Enviar invitaciones y Facebook le enviará la solicitud a cada uno de los amigos que hayamos seleccionado.

## 2. Enviar una recomendación

La actividad principal de Sugerest es la de enviar una recomendación a un amigo. Como hemos visto en la sección “Primeros pasos”, necesitamos tener algún amigo en la aplicación para poder enviar una recomendación.

Si ya disponemos de algún amigo, vamos a proceder a enviar una recomendación.

Para ello, se puede utilizar el botón existente en la pantalla de bienvenida, o utilizar el botón de “+ Nueva recomendación” ubicado en las secciones de Actividad, Recomendaciones enviadas, Mis recomendaciones o Historial.



**Figura 72 Botón de nueva recomendación**

Una vez pulsamos en el botón de nueva recomendación, se nos presenta la siguiente pantalla para introducir los datos de nuestra recomendación:

facebook Buscar personas, lugares y cosas

Bienvenido Jose Antonio

2 2

Invitar amigos

Actividad

Me han recomendado

Mis recomendaciones

Historial

Afinidad

1. ¿Qué quieres recomendar?

2. Elige un amigo

Recomiendo...

Mi valoración ★★★★★

Andres Antonio J.

Francisco Lorenzo

Maria Natalia

Pepe Juan

Enviar recomendación

Acerca de Manual de usuario Contacto

Figura 73 Formulario de nueva recomendación

Se requiere introducir la siguiente información:

- **¿Qué quiero recomendar?:** Tenemos que introducir un texto descriptivo del contenido que se desea recomendar, como por ejemplo, Toy Story 3.
- **¿Qué valoración le doy?:** Tenemos que introducir la valoración que le damos al contenido que vamos a recomendar. Puedes recomendar algo indicando que es excelente, o simplemente diciendo que está bien. El amigo al que se lo envíes verá esta información.

- **¿De que categoría es?:** Hay que seleccionar la categoría en la que se encuadre el ítem que queremos recomendar: Cine y TV, Música, Libros, etc.
- **¿A quién se lo quiero recomendar?:** En la parte derecha se nos muestra la lista de nuestros amigos que ya utilizan Sugerest. Pulsamos sobre el amigo al que vaya destinada la recomendación.

Una vez que hayamos completado ambos pasos, se habilitará el botón de Enviar recomendación. Pulsamos este botón y la recomendación quedará enviada. La aplicación mostrará el listado de recomendaciones enviadas en la que se mostrará la recomendación que acabamos de enviar.

Las posibles categorías son:

- Cine y televisión
- Música
- Libros
- Videojuegos
- Sitios webs
- Aplicaciones móviles
- Varios

### **3. Recomendaciones recibidas**

En la sección de recomendaciones recibidas podemos ver las recomendaciones que nuestros amigos nos han enviado y aún no las hemos valorado. Podemos acceder a través del enlace del menú izquierdo *Me han recomendado*.



The screenshot shows the 'Sugerest' web application interface. At the top, there is a Facebook logo and a search bar. The main header features a thumbs-up icon and the word 'Sugerest' in a large, stylized font, with the tagline 'Recomienda contenido de ocio a tus amigos' below it. A navigation menu on the left includes 'Actividad', 'Me han recomendado', 'Mis recomendaciones', 'Historial', and 'Afinidad'. The main content area is titled 'Recomendaciones recibidas' and contains a table of recommendations. The table has columns for 'Recomendación', 'Tipo', 'Amigo', 'Fecha', 'Nota', 'Relevancia', and 'Valorar'. Two recommendations are listed: 'LIMP BIZKIT - GOLD COBRA' and 'Linkin Park - Hybrid Theory'. At the bottom, there are links for 'Acerca de', 'Manual de usuario', and 'Contacto'.

Recomendación	Tipo	Amigo	Fecha	Nota	Relevancia	Valorar
LIMP BIZKIT - GOLD COBRA	Música	[Avatar]	06 sep, 21:53	5	51,6%	★★★★★
Linkin Park - Hybrid Theory	Música	[Avatar]	06 sep, 21:07	5	50,6%	★★★★★

**Figura 74 Recomendaciones recibidas**

Se nos muestra la lista de recomendaciones que hemos recibido con una recomendación en cada fila. Podemos ver la siguiente información:

- **Recomendación:** Nombre que nuestro amigo a dado a la recomendación.
- **Tipo:** Categoría de la recomendación, como puede ser Cine o Música.
- **Amigo:** Fotografía del amigo que ha enviado la recomendación. Si se pasa el ratón por encima se podrá ver el nombre completo del mismo.
- **Fecha:** Fecha en la que nuestro amigo nos ha enviado la recomendación.
- **Nota:** Valoración que nuestro amigo le ha dado al ítem que ha recomendado.

- **Relevancia:** Porcentaje que indica el grado de relevancia o fiabilidad con la que la recomendación será de nuestro gusto
- **Valorar:** Nos permite dar una valoración al ítem que ha sido recomendado expresando el grado de satisfacción sobre la recomendación recibida. Esto producirá un incremento o decremento de la afinidad con tu amigo en función de la valoración.

Si pasamos el ratón por la columna de valoración, se iluminarán las estrellas. Una estrella es la peor valoración frente a cinco estrellas que sería la mejor. Solo con pulsar sobre la estrella y la valoración quedará almacenada.



Figura 75 Valorar una recomendación recibida

Cuando una recomendación es valorada, desaparece de la lista de recomendaciones recibidas. Si se quiere volver a consultar, se puede hacer desde

la sección de Historial, en la cual se almacenan todas las recomendaciones enviadas y recibidas que ya han sido valoradas.



**Figura 76 Recomendaciones recibidas tras valorar una recomendación**

Por último, tenemos la opción de descartar una recomendación. Si por algún motivo, sabemos que esa recomendación no nos gustará, tenemos la opción de descartarla. Ten en cuenta que esto hará que decrezca la afinidad que tienes con ese amigo, y se le notificará informándole que has descartado la recomendación.

Para descartar una recomendación pasamos el ratón por la parte derecha de la recomendación y aparecerá el icono de descartar. Si se pulsa, la recomendación quedará descartada.



Figura 77 Descartar una recomendación

#### 4. Recomendaciones enviadas

Si pulsamos en el menú izquierdo en *Mis recomendaciones*, veremos la sección con todas las recomendaciones que hemos enviado a nuestros amigos y aún no han sido valoradas por estos.

La recomendación enviada nos muestra la siguiente información:

- **Recomendación:** Nombre que le hemos dado al contenido que hemos recomendado.
- **Tipo:** Categoría en la que hemos asignado el elemento recomendado.
- **Amigo:** Fotografía del amigo al que se ha enviado la recomendación. Si se pasa el ratón por encima se podrá ver el nombre completo del mismo.
- **Fecha:** Fecha en la que realizamos la recomendación a nuestro amigo.

- **Estado:** estado en el que se encuentra la recomendación.



**Figura 78 Recomendaciones enviadas**

Una recomendación enviada en Sugerest pasa por tres estados:

- **No leída:** Indica que la recomendación ha sido enviada a tu amigo, pero esta aún no ha sido vista por él.
- **No valorada:** Indica que tu amigo ha visto la recomendación que le has enviado, pero aún no ha dado ninguna valoración a la misma.
- **Valorada:** La recomendación ha sido valorada, se ha actualizado la afinidad y la recomendación desaparece de esta sección para ir al *Historial*.

## 5. Historial

El historial nos muestra el listado de recomendaciones que has recibido y has valorado, o que has enviado y tu amigo ya la ha valorado. Una vez que valores una recomendación recibida, esta pasará al historial. Del mismo modo, una recomendación enviada que sea valorada por tu amigo pasa a aparecer en esta sección.



Figura 79 Historial de recomendaciones

El listado muestra la siguiente información:

- **Enviada/Recibida:** Muestra si la recomendación es enviada por nosotros (flecha saliente) o nos la ha enviado un amigo (flecha entrante).
- **Recomendación:** Nombre que le hemos dado al contenido que hemos recomendado.
- **Tipo:** Categoría en la que hemos asignado el elemento recomendado.

- **Amigo:** Fotografía del amigo al que se ha enviado la recomendación. Si se pasa el ratón por encima se podrá ver el nombre completo del mismo.
- **Fecha:** Fecha en la que realizamos la recomendación a nuestro amigo.
- **Valoración:** valoración que has dado a la recomendación o que tu amigo ha dado. Se puede modificar la valoración que ya hayas dado previamente, con su correspondiente actualización de afinidad entre tú y tu amigo.

## 6. Afinidad

Puedes acceder a la sección de afinidad a través de la opción inferior del menú izquierdo. La página de afinidad nos mostrará la afinidad que tenemos con cada uno de nuestros amigos. El listado se ordena de mayor a menor afinidad.

Bienvenido Jose Antonio

Invitar amigos

**Afinidad** Global

Posición	Amigo	Afinidad
1.	Andres Santiago	100%
2.	Francisco Jose	90%
3.	Antonio J. Pérez	80%
4.	Maria Martinez Santiago	80%
5.	Pepe Juan Lopez	50%
6.	Natalia Santacreu	40%

Acercas de Manual de usuario Contacto

Figura 80 Afinidad global

En Sugerest hay dos tipos de afinidad:

- **Afinidad global:** Se trata de la afinidad global entre los usuarios en todas las categorías.
- **Afinidad por categoría:** Se trata de la afinidad concreta entre los usuarios para una categoría concreta.

Inicialmente, se mostrará la afinidad global con tus amigos. Si quieres ver la afinidad para una categoría concreta, basta con pulsar sobre el icono de la categoría y se mostrará el listado para esa categoría concreta.





Figura 81 Afinidad por categorías

Es posible que desaparezcan amigos de este listado, ya que con ellos quizás no tengas ninguna afinidad aún en esta categoría al no haber realizado ninguna recomendación en ella.

## 7. Actividad

En el capítulo 1 de este manual de usuario, vimos la pantalla de inicio o bienvenida a la aplicación. Esta pantalla de bienvenida se te muestra mientras no hayas generado ninguna actividad.

En el momento en el que recibas la primera notificación, la pantalla de bienvenida será sustituida por la de actividad. Esta pantalla se mostrará en el inicio de la aplicación, cuando se pulse sobre el logo en la cabecera o sobre el enlace de *Actividad* en el menú izquierdo.



Figura 82 Actividad

- **Recomendación recibida:** Esta notificación te llegará siempre que un amigo te envíe una nueva recomendación. Su icono es 👍.
- **Valoración recibida:** Esta notificación te llegará siempre que un usuario valore una recomendación que tu le hayas enviado. El icono asociado es ★.
- **Valoración modificada:** Esta notificación te llegará siempre que un usuario modifique la valoración previa a una recomendación que tu hayas enviado. El icono es ★.
- **Valoración rechazada:** Esta notificación te llegará siempre que un usuario descarte la recomendación que tú le has enviado. El icono asociado a esta notificación es 🗑️.

## 8. Perfil de un amigo

El perfil de un amigo nos muestra la información sobre tu amigo en aplicación, viendo su información general por un lado, y la actividad y afinidad entre tu amigo y tú.

Para acceder al perfil, basta con pulsar en el la imagen de un amigo en cualquiera de las secciones anteriores. Una vez pulsamos, sobre la imagen, se abrirá la ventana emergente con dicho perfil.



Figura 83 Perfil de un usuario

El perfil podría dividirse en tres partes. Por un lado, en la parte superior podemos ver la fotografía y el nombre completo de nuestro amigo, la afinidad global con el y el número de recomendaciones totales que ha enviado y recibido en la aplicación.

En la parte intermedia, podemos ver el número de recomendaciones que tu has enviado a tu amigo, y el número que él te ha enviado a ti.

Finalmente, en la parte inferior, se muestra la afinidad que tienes con tu amigo en cada una de las categorías en las que habéis intercambiado recomendaciones.

## 9. Acerca de

La pantalla de Acerca de te ofrece una descripción breve sobre en que consiste la aplicación



Figura 84 Acerca de

Puedes acceder a ella a través del enlace del pie de página titulado *Acerca de*.

## 10. Contacto

El formulario de contacto es la herramienta que tienes a tu mano para comunicarte con la administración de Sugerest. Se trata de un formulario en el que hay que introducir tus datos básicos y el motivo y texto del formulario de contacto. El motivo puede ser uno de los siguientes:

- Información general: Para obtener más información sobre cualquier aspecto de Sugerest.
- Reportar error: Para informar sobre errores o fallos que has encontrado con el uso de la aplicación
- Sugerencias: Si tienes alguna idea para mejorar la aplicación o añadir una nueva funcionalidad podrás comunicarlo.

En el campo de texto es donde puedes desarrollar el motivo de la consulta enviando el mensaje que desees con la explicación acerca del motivo del correo de contacto.

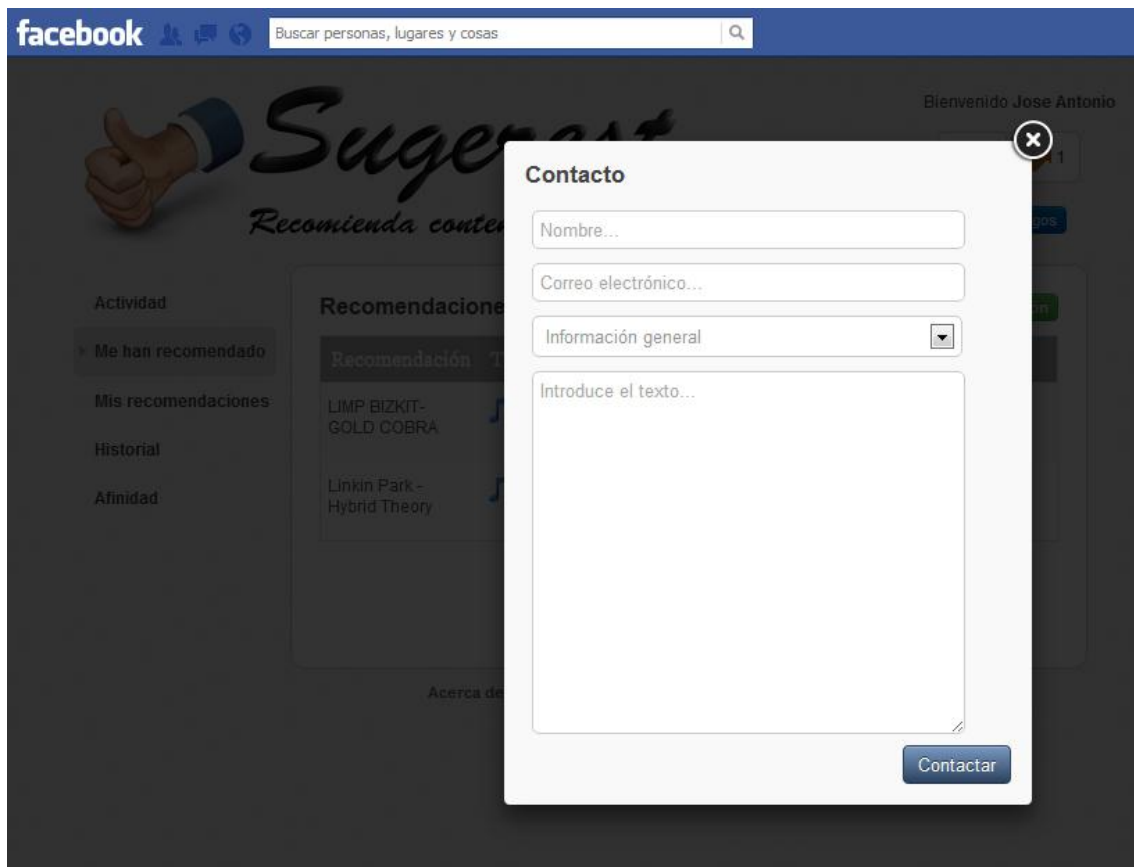


Figura 85 Contacto

## 11. Manual de usuario

El apartado manual de usuario es un resumen de éste manual que estás leyendo, menos amplio, pero integrado en la misma aplicación para una consulta rápida.



**Figura 86 Manual de usuario**

En la parte superior se puedes seleccionar el tema de ayuda que deseas consultar.