



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica

Universitat Politècnica de València

Pesedia

Red social para concienciar en privacidad

Trabajo Fin de Máster

Máster Universitario en Ingeniería Informática

Autor: Jose Alemany Bordera

Tutor: Ana María García Fornes

Cotutor: Agustín Rafael Espinosa Minguet

Julio de 2016

El presente trabajo fin de master lo dedico...

*a mi pareja, por su paciencia y comprensión,
a mi familia, por su apoyo y ánimo,
a mis amigos y compañeros,
y a mis tutores.*

Gracias a todos.

Resumen

Actualmente el uso de redes sociales se ha convertido en una práctica habitual para informarnos, compartir información, opinar y socializarnos. Entre las que más se utilizan, destacamos Facebook, Twitter, Instagram y Snapchat. El alcance de la información que compartimos se subestima en la mayoría de las ocasiones conformando nuestra identidad virtual, que puede afectarnos sin ser conscientes de forma positiva o negativa en el futuro. Por ello, comprender, conocer y controlar el alcance de nuestra información es tan importante.

En el presente proyecto final de máster se pretende desarrollar una red social completa (con las funciones básicas de compartir información, imágenes, opiniones, comunicación directa entre usuarios, posibilidad de creación de grupos) mediante la utilización de herramientas open-source. La red social dispondrá de interfaz web y móvil, que se desarrollará en Android para facilitar el uso de la propia red social y su explotación. Con el fin de concienciar el posible alcance de la información que compartimos en la red social se estudiarán posibles medidas de privacidad implementando aquellas que ayuden a los propios usuarios a controlar su información (como por ejemplo indicando el riesgo de compartir con determinados usuarios).

Este proyecto está enmarcado dentro del proyecto nacional “Privacidad en Entornos Sociales Educativos durante la Infancia y la Adolescencia” de I+D+I propuesto por el Grupo de Tecnología Informática e Inteligencia Artificial (GTI-IA) del DSIC, en el cual se pretenden aplicar las medidas de privacidad desarrolladas durante el mismo para el uso y aprendizaje centrado en las etapas de la infancia y la adolescencia. Realizando talleres formativos y dejándola disponible para su uso por alumnos de los primeros cursos de la ESO.

Palabras clave: redes sociales, medidas de privacidad, control de la información, Elgg, Android, plugins

Abstract

Currently the use of social networks has become a common practice to inform, share information, express opinions and socialize us. Among the most commonly used, we include Facebook, Twitter, Instagram and Snapchat. The scope of the information we share is underestimated in most cases making up our virtual identity that can affect unaware of positive or negative way in the future. Therefore, understand, know and control the scope of our information is so important.

In this final master project aims to develop a comprehensive social network (with the basic functions to share information, images, opinions, direct communication between users, possibility of creating groups) by using open-source tools. The social network will have web interface and mobile, which will run on Android to facilitate the use of the social network itself and its exploitation. In order to sensitize the possible scope of the information we share on social network possible privacy measures will be studied implementing those that help users manage their information (such as indicating the risk of sharing with specific users).

This project is framed within the national project “Privacidad en Entornos Sociales Educativos durante la Infancia y la Adolescencia” of I+D+I proposed by the Grupo de Tecnología Informática e Inteligencia Artificial (GTI-IA) of DSIC, which are intended apply the privacy measures developed during of it for the use and learning focused on the stages of childhood and adolescence. Conducting training workshops and making it available for use by students of the first courses of the ESO.

Keywords: social networking, privacy measures, information control, Elgg, Android, plugins

Índice general

Resumen	II
1. Introducción	1
1.1. Motivación	1
1.2. Objetivos	2
1.3. Estructura	2
2. Estado del arte	5
2.1. Trabajos relacionados	5
2.1.1. Social Lab	5
2.1.2. FiZZiKiD	6
2.1.3. Facebook	7
2.1.4. Minds	8
2.2. Medidas de privacidad	9
2.3. Conclusiones	10
3. Herramientas	13
3.1. Comparativa de herramientas	13
3.1.1. Herramientas para desarrollar la red social	13
3.1.2. Plataforma móvil de desarrollo	17
3.2. Conclusiones	19
4. La red social Pesedia	21
4.1. Descripción del sistema	21
4.2. Arquitectura	21
4.3. Tecnologías empleadas	22
4.4. Implementación de Pesedia	24
4.4.1. Despliegue de la red social	24
4.4.2. Configuración de las características	27
4.4.3. Implementación de plugins	34
4.4.4. Medidas de privacidad implementadas	41
4.4.5. Creación del API	41
4.5. Implementación de la aplicación móvil	43
4.5.1. Comunicación con el API	44
4.5.2. Gestión de la información	45
4.5.3. Interfaz de usuario	47

5. Evaluación de la plataforma	51
5.1. Validación de la funcionalidad	51
6. Conclusiones	54
6.1. Conclusiones	54
6.2. Trabajo futuro	55
Bibliografía	57

CAPÍTULO 1

Introducción

1.1. Motivación

Las redes sociales se han convertido en una herramienta de gran importancia para la sociedad actual, reflejando las necesidades de expresión y reconocimiento inherentes al ser humano. Con ellas nos mantenemos informados de sucesos en todo el mundo y en nuestros círculos más próximos (familia, amigos, trabajo, etc.); compartimos experiencias, pensamientos y opiniones con el fin de que se nos reconozca.

Existen multitud de redes sociales, las cuales se han especializado dependiendo de la orientación (al contenido o a los usuarios), los ámbitos de uso (personal o profesional), la temática central, el tipo de comunicación y el público al que van dirigidas. Según un estudio realizado por el IAB (*Interactive Advertising Bureau*)[7], las redes sociales más conocidas y utilizadas son Facebook, Whatsapp, Twitter y Youtube; destacando además Instagram y Snapchat como las más utilizadas por los jóvenes.

La gran mayoría de nosotros manejamos varias redes sociales diariamente, compartiendo las actividades que realizamos, nuestros gustos musicales, fotografías y opiniones sin ser conscientes del alcance y valor de nuestra información. El término identidad virtual surge como el conjunto de información pública que está relacionada con nosotros, por ello un mal uso de las redes sociales podría afectar negativamente nuestras vidas. Las políticas de privacidad establecidas en cada una de las redes sociales existentes nos permiten conocer parte del alcance de nuestras acciones, pero es difícil para los usuarios más comunes (inexpertos o principiantes) controlar este aspecto para efectuar un correcto uso de las redes sociales.

Ser capaces de ofrecer una alternativa de red social que les permita comprender, conocer y controlar, mediante medidas de privacidad, el alcance de la información es la solución que se plantea en este trabajo fin de máster, que está además enmarcado dentro del proyecto nacional “Privacidad en Entornos Sociales Educativos durante la Infancia y la Adolescencia”[6]. En dicho proyecto se extenderá el trabajo realizado en el presente trabajo fin de máster, investigando nuevas medidas y alternativas para ofrecer mejores herramientas dentro de nuestra red social; educando a los más jóvenes en el uso de las mismas; y realizándose experimentos con usuarios reales que valorarán la propia red social y su capacidad para controlar el alcance de su información.

1.2. Objetivos

El propósito del siguiente proyecto consiste en el desarrollo de una red social capaz de ofrecer a los usuarios medidas de privacidad que ayuden a comprender el alcance de su información, además de las funciones básicas de buscar información, publicar, compartir y opinar.

Uno de los objetivos principales es realizar un estudio de las herramientas disponibles y seleccionar la más apropiada para el desarrollo y despliegue de la red social. Otro será dotar la red social con todos los componentes arquitectónicos: base de datos, donde almacenaremos toda la información de la red social; lógica del sistema, en las cuales se implementarán las funcionalidades de comunicación, creación de contenido, relaciones y medidas de privacidad; API de comunicación, que permitirá extraer/enviar información de/a la red social bajo autenticación para la aplicación móvil; e interfaz, que en nuestro caso se realizará para web y móvil. Además de las funcionalidades propias de las redes sociales (visualización de contenidos, relaciones, herramientas de comunicación, etc.).

Por otro lado analizaremos medidas o funciones que ayuden a los usuarios a controlar mejor el alcance de su información con el propósito de aplicarlas a nuestra red social. Estas medidas serán extraídas y analizadas de redes sociales existentes o trabajos de investigación.

Finalmente, se pretende desarrollar una aplicación móvil que se comunicará mediante autenticación segura con la API de Pesedia, previamente diseñada, para facilitar el acceso y explotar el uso de la propia red social.

En resumen, los objetivos de este trabajo son:

- Implementar una red social capaz de ofrecer medidas de privacidad a sus usuarios que les permita comprender, conocer y controlar el alcance de su información. Para ello...
 - Se creará una red social con interfaz web mediante la utilización de herramientas de código abierto.
 - Se equipará la red social con las funciones básicas de buscar información, compartir contenidos, opinar, etiquetar, creación de grupos y comunicación directa.
 - Se aplicarán medidas que permitan a los usuarios conocer el alcance de su información.
- Desarrollar una aplicación móvil para dispositivos Android que facilite el uso de la propia red social y su explotación.

1.3. Estructura

Este trabajo se ha organizado de la siguiente forma. En el Capítulo 2 se presentan aquellos trabajos que hemos considerado que tienen mayor relación con el trabajo final de máster; y las técnicas o funcionalidades que utilizan para proteger, educar, concienciar y dotar de mayor capacidad a los usuarios para que controlen su información en las redes sociales con el fin de aplicarlas en nuestra red. Para lograr implementar la red social y cumplir los requisitos

del proyecto, en el Capítulo 3, realizamos un estudio de mercado de las herramientas y plataformas existentes para desarrollar Pesedia y la aplicación móvil que proporcionará un mejor acceso a la misma. A continuación, el Capítulo 4 contendrá todos los aspectos relacionados con el desarrollo de la red social y la aplicación móvil. Y para finalizar, el Capítulo 6 cerrará el trabajo y se discutirán las posibles ampliaciones o trabajos futuros.

CAPÍTULO 2

Estado del arte

2.1. Trabajos relacionados

En el mercado existe una gran cantidad de redes sociales con diferentes formas de comunicación e interacción entre usuarios. Cada una de ellas aplica las políticas de privacidad que estima oportunas para que la información pueda ser alcanzable por el resto de usuarios. En este apartado detallamos las redes sociales y/o trabajos relacionados que ofrecen mejores alternativas para que sus usuarios puedan controlar la información que en ellas se publica. Además, algunas de ellas han sido creadas con propósitos similares a los nuestros debido a la necesidad de control y educación en medidas de privacidad.

Entre los trabajos relacionados, ya sean redes sociales que ofrecen controles en el acceso a nuestra información o herramientas para comprender cómo funcionan las mismas, destacamos los siguientes.

2.1.1. Social Lab

Social Lab[5] es un software de redes sociales orientado a la investigación. Es libre, de código abierto bajo la licencia GPLv3, flexible y no propietario. Esta herramienta permite desplegar tu propia red social con las funciones básicas de perfiles de usuario, peticiones de amistad, mensajería pública y privada, fotos y etiquetado, páginas de interés, etc. Además de integrar la funcionalidad de *social bots* para estimular y simular interacciones sociales dentro de la red; los cuales pueden personalizarse, son trazables y con estados para simular comportamientos más complejos.



Figura 2.1: Aplicación Social Lab.

La propia herramienta proporciona una demo planteada como un *wargame* en privacidad, en el cual el reto está relacionado con la seguridad informática en el que es necesario explotar una vulnerabilidad en el sistema para lograr acceder a la información. Sin embargo, Social Lab es un *wargame* un tanto especial puesto que está orientado hacia la ingeniería social, el arte de manipular a la gente para realizar determinadas acciones o divulgar información confidencial. El objetivo del mismo es conocer las tácticas que utilizan los atacantes para conseguir información privada, confidencial o íntima, para poder prevenirlas en una red social real.

Esta herramienta no nos es útil debido a que la personalización se centra en la elaboración de *social bots* y no en aspectos de funcionalidad de la propia red social o en herramientas para ayudar a sus usuarios a controlar la información. Además, esta explota solamente el aprendizaje de las habilidades sociales para evitar intrusiones. Pero sí resulta interesante este enfoque, el cual podría incluirse en un futuro en nuestra red social.

2.1.2. FiZZiKiD

FiZZiKiD es una red social privada enfocada al entretenimiento y al aprendizaje del correcto uso de las redes sociales, en este caso, de los niños de entre 8 a 14 años (edad en la cual no pueden acceder a las redes sociales libres). Esta red social cuenta con las medidas necesarias de seguridad y privacidad para mantener a los menores a salvo de los riesgos propios de otras redes sociales.

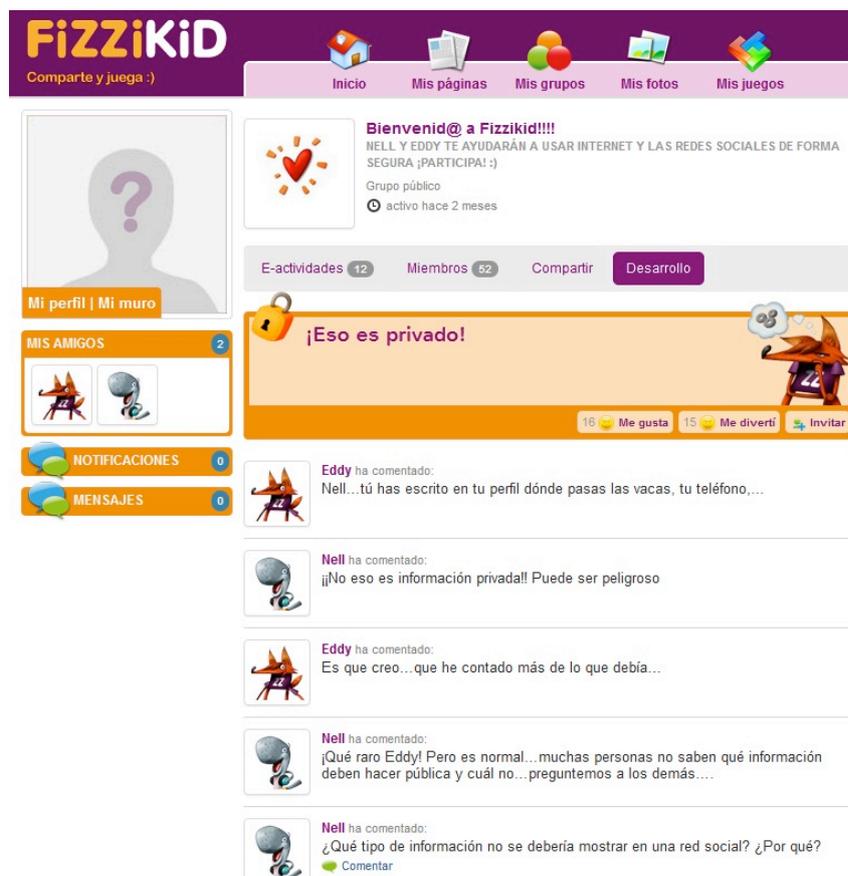


Figura 2.2: Aplicación FiZZiKiD.

El acceso a la red social es efectuado mediante el registro por los propios padres. Y durante el uso de la misma, los niños usuarios de FiZZiKiD se encuentran acompañados por dos mascotas virtuales (Nell y Eddy) que ayudan a los niños a comprender el funcionamiento de cada una de las funciones de la red, las actividades propuestas y la sensibilidad de su información. La finalidad de la misma es que los niños aprendan las habilidades y competencias necesarias para enfrentarse en el futuro a las redes sociales libres.

Esta red social creada por una empresa particular es quizás la más aproximada al proyecto nacional que enmarca este trabajo fin de máster. Aún así, FiZZiKiD al ser de carácter privado (según sus propietarios) no está disponible su acceso por lo que desconocemos si sigue en marcha. Además, esta no cuenta con medidas avanzadas que utilicen la información de la red para recomendar acciones de privacidad a sus usuarios sino que se centra en explicar los conceptos básicos.

2.1.3. Facebook

Facebook[13] es una de las redes sociales más completas y utilizadas en la actualidad. Su evolución con el tiempo les ha permitido explorar y mejorar su interfaz, todos los aspectos de las relaciones y amistades, el alcance de la información y los servicios que ofrecen. En esta red social, a diferencia de muchas otras, los usuarios tienen un mayor control del alcance de su información. Esta preocupación por mejorar la privacidad y su control ha surgido o ha sido provocada por la propia necesidad de los usuarios (a los cuales, probablemente, les haya repercutido negativamente el alcance de la misma). En los últimos años, Facebook ha mejorado sus capacidades para proporcionar un mejor control del alcance de la información, aunque cuando el número de usuarios y relaciones es tan elevado (una media de 335 amigos) es requerido que el propio usuario muestre interés en este tema para saber organizarse en la misma, o sino estas herramientas pierden su funcionalidad al no ser utilizadas.

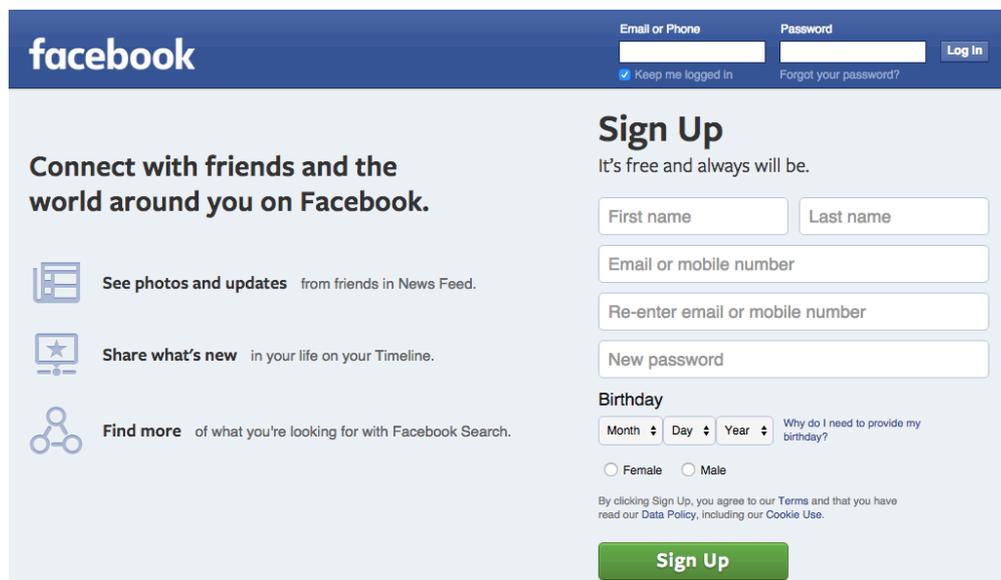


Figura 2.3: Aplicación Facebook.

En Facebook, al igual que en otras redes sociales, la mayoría de acciones están destinadas a compartir información. Las acciones y conceptos que en ellas se manejan no llegan a ser

entendidas por sus usuarios (en la mayoría de ocasiones) y tampoco se definen al inicio de ser utilizados, por lo que el usuario debe indagar para informarse. En Facebook Help[14] podemos encontrar mucha información sobre conceptos, acciones y alcance de la información de mano de Facebook, pero seguramente usted (el lector de este trabajo) utiliza Facebook y nunca ha entrado en esta página para informarse. Ayudar a los usuarios a comprender estas herramientas al inicio (tras crear sus cuentas) y durante su utilización sería una buena alternativa.

El mayor inconveniente de esta red social es que no nos permite el acceso a toda la información (como es lógico) para explorar todos los factores que pudiesen afectar a la privacidad. Es por ello que desarrollar nuestra propia red social, con herramientas de código abierto, nos permite el acceso a todos sus elementos para poder proporcionar nuevas medidas de privacidad resultantes del estudio de la misma.

2.1.4. Minds

Minds[10] es una red social recientemente nueva que se encuentra en su versión beta, la cual apuesta por la privacidad de sus usuarios e incentiva la participación a cambio de promover tus contenidos. Esta red social ha sido creada para, según afirman sus creadores, “Recuperar el control de tu universo social” al no comercializar con la información de los usuarios. Además, se ha diseñado en la misma un sistema de puntos donde el usuario es recompensado por su participación en la red social. Los puntos que los usuarios obtienen al loguearse, visualizar contenido, actualizar su estado, comentar u otras acciones pueden ser consumidos aumentando el alcance de sus publicaciones. Con lo que los usuarios deciden para cada una de sus publicaciones el alcance que tendrá en el conjunto completo de la red.

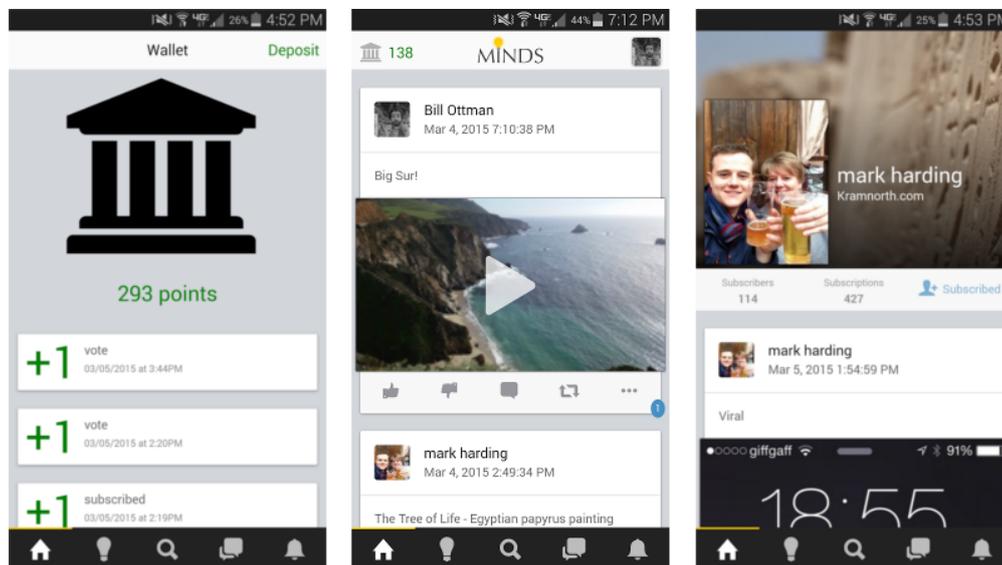


Figura 2.4: Aplicación Minds.

Aunque en dicha red parece que tenemos un mayor control del alcance de nuestra información, como sucede en muchas otras, desconocemos quien conforma la audiencia de nuestras publicaciones. Poder ofrecer esta información beneficiaría a los usuarios y a su privacidad, ya que la audiencia y la sensibilidad de la información que publicamos afectan en gran medida a nuestra privacidad en Internet.

2.2. Medidas de privacidad

La privacidad puede ser definida como el ámbito más personal, íntimo o profundo de la vida de una persona que debe desarrollarse en espacios reservados y que comprende sus sentimientos, vida familiar o relaciones de amistad. En las redes sociales encontramos el concepto de privacidad en dos ámbitos: la privacidad del medio (la tecnología), que no depende de los usuarios y en la cual las comunicaciones deben ser seguras y la información de la base de datos debe estar protegida; y la privacidad en el ámbito social, que depende directamente de los usuarios y de las acciones que realicen. En nuestro caso, el ámbito de la privacidad en el que nos centramos es la relacionada con el ámbito social.

En este apartado vamos a hacer una recopilación de las medidas de privacidad que permiten al usuario conocer, comprender y controlar el alcance de la información que en las redes sociales se publica tomando como ejemplo funciones o herramientas existentes en redes sociales actuales o trabajos de investigación[3].

- **Configuraciones de privacidad.** Tener un apartado donde se recogen todas las opciones relacionadas con elecciones que el usuario debe establecer para personalizar su privacidad en la red es muy importante. La mayoría de redes sociales disponen de elementos en los cuales la privacidad está presente, por ello disponen de un apartado de configuración de la misma. Como la privacidad puede ser compleja de explicar en grandes escenarios, Facebook ha diseñado además un acceso directo y sencillo para que sus usuarios puedan personalizar los aspectos más importantes de la misma en cualquier momento.
- **Selector de audiencia.** Esta característica permite al usuario establecer qué usuarios serán capaces de acceder a cualquier información de la red social, ya sean atributos del propio perfil del usuario o provenientes de acciones de subida, actualización o difusión. Por simplicidad se definen grupos genéricos como **público**, **amigos** (o **amigos de amigos**) y **privado** pero también pueden personalizarse por el usuario o incluso nombrar directamente qué usuarios tendrán acceso. Facebook tiene disponible estas características mientras que Instagram solo diferencia entre seguidores y la selección manual de los usuarios que tendrán acceso.
- **Listas de acceso.** Se consideran listas de acceso aquellos grupos de usuarios con los que se tiene una relación específica (colecciones de amigos, familiares, grupos de trabajo, etc.). Y se definen, en la mayoría de redes sociales, por los usuarios, con el fin de clasificar mejor la audiencia o alcance de la información que se comparte en la red. Estas listas son utilizadas con la funcionalidad de selección de audiencias para personalizar la privacidad de la información. Nuevamente, Facebook hace uso de dicha característica proporcionando incluso listas creadas automáticamente por el sistema. Con este último tema existe una gran cantidad de trabajos de investigación centrados en el estudio de similitudes entre usuarios para determinar diferentes grupos teniendo en cuenta el contexto que los caracteriza.
- **Registro de actividad.** Es un espacio en la red social que permite a los usuarios gestionar rápidamente las acciones realizadas por ellos, visualizando y permitiendo

editar propiedades como el alcance de las mismas con el fin de controlar la privacidad. Esta herramienta está disponible también en Facebook.

- **Audiencia de una publicación.** Conocer la cantidad de usuarios que han interactuado o visualizado un contenido, o incluso conocer la cantidad de los que tienen acceso al mismo permite a los usuarios comprender el alcance de sus publicaciones. Redes sociales como Ello o Minds permiten conocer esta información.
- **Alcance de la publicación.** Otro aspecto no tan detallado como es conocer la audiencia total de una publicación es el de conocer el alcance de la información, es decir, si una publicación es pública o tiene carácter privado (ya sea propio o de círculo cerrado). El alcance de la información es seleccionado por el propio autor de la publicación. En todas las publicaciones realizadas en Facebook podemos apreciar esta característica.
- **Edición de una publicación.** Tener la capacidad de editar la privacidad de una publicación ya publicada por ti, o incluso de borrarla, es una propiedad que afecta directamente a la privacidad y propiedad de la información del usuario.
- **Bloqueo de usuarios.** La capacidad de bloquear usuarios se encuentra disponible en muchas redes sociales, y como el título indica incapacita a dichos usuarios a visualizar contenidos o ser afectados por otras acciones provenientes del usuario que los bloqueó.
- **Potenciar el alcance.** Una herramienta de privacidad utilizada en la red social Minds permite a los usuarios potenciar o no el contenido creado por ellos mismos asignándoles cantidades incrementables de puntos a su información, lo cual incrementa el alcance de la misma. Esta funcionalidad otorga a sus propietarios el control del alcance de su información. Los puntos son conseguidos a cambio de realizar actividades en la red social como loguearse, publicar contenido, comentar u opinar.
- **Recomendaciones/aclaraciones en privacidad.** Informar de las repercusiones de las acciones realizadas en la red social e incentivar sobre las herramientas disponibles para configurar adecuadamente su privacidad, durante los primeros usos de la misma hasta que los propios usuarios sean capaces de comprender completamente su funcionamiento, es una buena forma de ayudar al usuario a conocer las formas de controlar el alcance de su información. Este tipo de funcionalidad no es muy común en las redes sociales, pocas redes sociales como las centradas en el aprendizaje para menores (ej. FiZZiKiD) cuentan con algunas de estas características.

Con las medidas sociales de la privacidad descritas los usuarios disponen de herramientas para lograr un mejor control de la información en las redes sociales.

2.3. Conclusiones

Los trabajos relacionados, aunque aportan características que ayudan a los usuarios a mejorar su privacidad mediante medidas como las descritas en el apartado anterior, no cumplen completamente con los requisitos de nuestro trabajo fin de máster debido a que creemos que estos pueden ser mejorados mediante medidas más efectivas. Por ello pretendemos desarrollar nuestra propia red social que cuente con el conjunto de características de los trabajos analizados, implementar las medidas de privacidad y desarrollar, como aportación futura de

la tesis, nuevas medidas de privacidad con el estudio de la información de los usuarios que se almacena en la red social con su uso.

CAPÍTULO 3

Herramientas

3.1. Comparativa de herramientas

En este apartado se pretende analizar todas aquellas herramientas disponibles en el mercado que nos faciliten la labor de diseñar y desarrollar tanto la red social como la aplicación móvil. El objetivo del análisis será el de escoger las mejores herramientas para nuestro proyecto justificando su elección.

3.1.1. Herramientas para desarrollar la red social

Desarrollar una red social completamente desde cero es una tarea muy costosa y compleja, por ello se ha hecho uso de herramientas que nos han facilitado esta labor. Con el fin de escoger la herramienta de código abierto que más se adecue a las necesidades y requisitos de nuestro proyecto, se ha llevado a cabo un estudio de mercado de las alternativas posibles.

De todas las herramientas que se han encontrado en Internet para el desarrollo de nuestra propia red social, destacamos las siguientes atendiendo que fuesen gratuitas y de código abierto como requisitos indispensables.



Figura 3.1: Logotipo de BuddyPress.

BuddyPress es una extensión de Wordpress, el popular sistema de gestión de contenidos enfocado a la creación de sitios web. Esta extensión es un conjunto de plugins que permite añadir características propias de una red social a nuestro sitio web creado con Wordpress. Fue desarrollada en 2008 por Andy Peatling bajo la licencia GPLv2.0 y en la actualidad se encuentra soportada por una gran comunidad.

BuddyPress tiene fácil instalación, una amplia documentación y su gran comunidad facilita la resolución de problemas durante el diseño, desarrollo y comportamiento de la red social. El código base sobre el que se mantiene y mediante el cual se puede desarrollar nueva funcionalidad es PHP. Además, soporta los modelos de base de datos relacionales como

MySQL o PostgreSQL.

El conjunto de plugin que conforma BuddyPress, y que puede ser extendido por plugins de terceros, proporciona una serie de características propias de las redes sociales. Entre estas destacamos las siguientes:

- Perfil de usuario con posibilidad de añadir campos y secciones para los perfiles, y subida de avatares.
- Publicación de contenidos de diversos tipos (texto, audio, vídeo, ficheros, etc.) y espacios de foro con creación de entradas de blog.
- Sección para visualizar la actividad de los usuarios de la red social.
- Creación de grupos con personalización del logotipo o imagen pública del grupo, foros de discusión.
- Sistema de amistades y relaciones.
- Mensajería privada.
- Sistema de notificaciones.
- Mediante extensión con plugins: etiquetado de usuarios/grupos, servicios web, etc.



Figura 3.2: Logotipo de Oxwall.

Oxwall es un motor para la generación de redes sociales y comunidades bastante reciente (finales del 2010), desarrollado bajo la licencia CPALv1.0 por la empresa Skalfa LLC. El núcleo de Oxwall ha sido desarrollado en el lenguaje PHP el cual ofrece gran cantidad de funcionalidades propias de las redes sociales.

Esta herramienta requiere de un servidor web capaz de soportar PHP 5.3, Apache 2 y MySQL en el cual se instala y configura Oxwall de forma rápida y sencilla. La parte del modelo de la red social y las funcionalidades son proporcionadas por la propia herramienta. Además, la misma puede ser personalizada y extendida mediante plugins, los cuales pueden ser descargados/comprados en la tienda Oxwall (ya sean creados por el equipo Oxwall o por terceros) o desarrollados por nosotros mismos. Esta propiedad de la herramienta, que incluso proporciona un panel de administración de los plugins para activar o desactivar los mismos, permite un gran abanico de posibilidades dentro de la red social. Entre las características proporcionadas por la misma destacamos las siguientes:

- Mensajería privada tanto en forma de chat como en buzón de entrada (tradicional).

- Perfil de usuario con posibilidad de añadir campos y secciones para los perfiles, y subida de avatares.
- Espacios de Blogs, Actividad, Fotos y otros.
- Sistema estándar de amistad y relaciones con peticiones de amistad.
- Capacidad para la creación y administración de grupos.
- Búsqueda de usuarios tanto en la red social creada con Oxwall como en otras redes sociales como Facebook o Twitter.
- Gestión de la privacidad para controlar qué contenidos serán o no visibles por los usuarios.
- Sistema de notificaciones.
- Capacidad para interactuar con el contenido de otros (likes, comentar, opinar, compartir, etc.).
- Publicación de contenidos de diversos tipos (texto, audio, vídeo, ficheros, etc.).
- Mediante extensión con plugins: videojuegos, etc.

Además dispone de varios temas, en cuanto a apariencia de la red social, con aspectos sencillos y atractivos para el usuario que facilita la comprensión de los elementos que la forman. Quizás sea la herramienta que más avanzada este en cuanto a apariencia, pero su comunidad no es tan grande en comparación al resto de herramientas debido a que es de las más recientes.



Figura 3.3: Logotipo de Elgg.

Elgg es una plataforma de servicios de red social creada en 2004 por el grupo Curverider LTD bajo dos licencias: MIT para el núcleo base de la plataforma y GPLv2.0 con los plugins base que trae la propia herramienta. Es quizás una de las herramientas más populares y conocidas para la creación de redes sociales después de que Ning (herramienta para creación de redes sociales) pasase a ser de pago. Por lo que cuenta con una gran comunidad que facilita la resolución de problemas y documentación muy elaborada.

Como Oxwall, Elgg corre sobre una plataforma LAMP (Linux, Apache, MySQL y PHP) y su instalación es muy similar ya que necesita la misma tecnología la cual conecta para importar su modelo de base de datos sobre el que se apoya la red social. Además, siguiendo la misma ideología, la propia red social puede ser extendida mediante la instalación de plugins

que pueden ser activados y desactivados desde el panel de administración de la red social. Estos plugins están disponibles a través de la tienda de la comunidad Elgg o en Github, y desarrollar tu propio plugin es posible gracias a los ejemplos y tutoriales proporcionados en la documentación. De las características que dispone esta herramienta destacamos las siguientes:

- Mensajería privada con buzón de entrada (tradicional).
- Perfil de usuario con capacidad para controlar el alcance de la información propia del usuario (número de teléfono, edad, estudios, etc.) y subida de avatares.
- Espacios de Blogs, Actividad, Fotos y otros.
- Sistema amistad y relaciones unidireccional (seguir), con sistema de peticiones de amistad disponible mediante plugin.
- Capacidad para la creación y administración de grupos, y listas de acceso.
- Gestión de la privacidad para controlar qué contenidos serán o no visibles por los usuarios.
- Sistema de notificaciones.
- Sistema de búsqueda de usuarios, grupos y contenido.
- Publicación de contenidos de diversos tipos (texto, audio, vídeo, ficheros, etc.) y visualización de los mismos en la red.
- Mediante extensión con plugins: chat entre usuarios, publicación de posts, etiquetado de fotos, menciones a otros usuarios, servicios web, etc.

Existe una gran cantidad de plugins debido a la extensa comunidad de la herramienta y dispone de un gran soporte para la creación de un API para la comunicación con la red social mediante peticiones RESTful.



Figura 3.4: Logotipo de Humhub.

Humhub es una herramienta/solución sencilla para el desarrollo de redes sociales privadas con un aspecto similar al de Facebook. Esta herramienta fue desarrollada por la compañía Zeros+Ones bajo la licencia AGPL a mediados del 2014, por lo que es también una herramienta bastante reciente.

Su comunidad y documentación no es tan avanzada en comparación con el resto de herramientas debido a su juventud, pero ofrece una programación en PHP basada en el framework Yii que proporciona un fácil uso de la herramienta para crear y lanzar una red

social propia. Como el resto de herramientas, esta también comparte su extensión en la funcionalidad mediante módulos y facilidad para modificar las vistas de la red social con diferentes temas. Entre las características sociales que ofrece la herramienta encontramos las siguientes:

- Sistema de notificaciones.
- Tráfico de la actividad realizada por el propio usuario o del resto de usuarios.
- Perfil de usuario para personalizar su visibilidad y subida de avatares, a modo de tarjeta de presentación.
- Espacios de Blogs, Actividad y otros (ampliables mediante plugins).
- Sistema de relaciones y amistad bidireccional estándar.
- Capacidad para la creación y administración de grupos.
- Mensajería privada con buzón de entrada (tradicional).
- Sistema de intercambio de archivos.
- Sistema de búsqueda de usuarios, grupos y contenido.
- Mediante extensión con plugins: chat entre usuarios, encuestas, gestión de tareas, etc.

Esta herramienta parece estar más destinada a la creación de redes sociales privadas para empresas, por lo que sus módulos conforman. Además, la tienda de módulos es escasa y la documentación para la creación de nuevos módulos es bastante pobre.

Resumen

Tras haber presentado y analizado las herramientas más prometedoras, gratuitas y de código abierto que existen en el mercado para la creación de una red social, sintetizamos dicha información en la Tabla 3.1 que se muestra a continuación.

3.1.2. Plataforma móvil de desarrollo

El acceso a las redes sociales a través de dispositivos móviles como smartphones, smart-watches o smart tvs facilita y explota el uso de las mismas. Es por ello que, aunque muchas de las herramientas para la creación de redes sociales soportan el acceso amigable desde los navegadores móviles, se ha decidido implementar una aplicación móvil que promueva el uso de la misma.

En el caso de este proyecto, nos vamos a centrar en el desarrollo de una aplicación móvil para smartphones debido a que existe mucha más proporción de usuarios que acceden a las redes sociales desde los mismos. De las dos grandes plataformas móviles existentes, Android e IOS, ambas son una buena alternativa para desarrollar nuestra aplicación móvil. Aunque atendiendo a la cuota de mercado de usuarios españoles, Android es el sistema operativo móvil ganador en España (superior a un 80%).

	BuddyPress	Oxwall	Elgg	Humhub
Inicio	2008	2010	2004	2014
Licencia	GPL 2.0	CPAL 1.0	MIT/GPL 2.0	AGPL
Versión	2.6.0	1.8.3	2.2.0	1.0.1
Intalación	Fácil	Fácil	Fácil	Fácil
Código base	PHP, MySQL/PostgreSQL	PHP, MySQL	PHP, MySQL (+)	PHP, MySQL
Comunidad	Grande	Normal	Grande	Reducida
Documentación	Buena	Buena	Excelente	Normal
Escalabilidad	✓	✓	✓	✓
Aspecto de la red social	Normal	Muy atractivo	Normal	Atractivo
Sistema de notificaciones	✓	✓	✓	✓
Mensajería privada	✓	✓+chat	✓+chat (via plugin)	✓+chat (via módulo)
Relaciones de amistad	✓	✓	✓* (standar via plugin)	✓
Perfil de usuario	✓	✓	✓	✓
Grupos	✓	✓	✓	✓
Listas de acceso			✓	
Creación de Posts	✓	✓	✓	✓
Albumes de fotos		✓	(via plugin)	
Etiquetado	(via plugin)	✓	(via plugin)	
Interacción	(via plugin)	✓	(via plugin)	✓
Gestión de la privacidad		✓	✓	
Soporta servicios web	(via plugin)		✓	
Panel de gestión de plugins		✓	✓	

Tabla 3.1: Tabla comparativa de propiedades soportadas por las herramientas.

3.2. Conclusiones

El estudio de mercado realizado sobre las posibles herramientas para el desarrollo del proyecto nos ha permitido conocer el estado actual de las mismas y ser críticos frente a las necesidades inherentes que conlleva la realización del mismo. Por lo que atendiendo a los siguientes requisitos se ha seleccionado la más adecuada.

- Funciones sociales de amistad e interacción.
- Capacidad para soportar grandes cargas.
- Licencia de uso.
- Soporte a la gestión de la privacidad.
- Posibilidad de comunicación (API) desde la aplicación móvil con la red social.
- Calidad de la documentación.
- Estado de la comunidad que respalda la herramienta.
- Interfaz web atractiva.

Todas las herramientas presentadas para la creación de la red social disponían de las propiedades necesarias y propias de las redes sociales; y aunque alguna característica no estuviese incluida inicialmente, esta es fácilmente extensible mediante plugins. Además, todas ellas aseguraban soportar grandes cargas de usuarios en la red social.

En cuanto a la capacidad de soporte para la gestión de la privacidad de los usuarios, la calidad de la documentación y el estado de la comunidad que respalda la herramienta se ha observado que las más adecuadas corresponden con las herramientas Elgg y Oxwall, siendo levemente superior la primera de ellas.

Finalmente se ha optado por **Elgg** debido a que ofrece soporte para la creación de un API de comunicación mientras que en Oxwall, aunque presenta una interfaz claramente superior al resto por su atractivo diseño, debería desarrollarse completamente por nosotros el soporte para la comunicación con la aplicación móvil **Android**.

CAPÍTULO 4

La red social Pesedia

4.1. Descripción del sistema

La red social que se ha desarrollado a lo largo de este trabajo de fin de máster tiene como objetivo ayudar a los usuarios a mejorar el alcance de la información, para perfeccionar los círculos de usuarios o comunidades que tienen acceso a la misma. Además de servir como herramienta de estudio y aprendizaje para el proyecto nacional de “Privacidad en Entornos Sociales Educativos durante la Infancia y la Adolescencia”, en el cual se pretenden explorar medidas más avanzadas para gestionar la privacidad.

Con la herramienta Elgg para la creación de redes sociales, se ha logrado desarrollar y desplegar nuestra propia red social, cuyo nombre es Pesedia. La capacidad ofrecida por la herramienta para la extensión de las propiedades de la red social mediante plugins, nos ha facilitado nuestra tarea, personalizando la misma mediante la instalación de plugins de terceros que nos han proporcionado funcionalidades sociales adecuadas a los objetivos descritos en el Capítulo 1 (Sección 1.2) y a las medidas de privacidad explicadas en el Capítulo 2 (Sección 2.2). Además, con la creación de la interfaz móvil para la plataforma Android rompemos con la limitación del acceso facilitando su uso y explotación.

La red social es actualmente de acceso público mediante una dirección web segura (HTTPS) y pública, pero de registro cerrado¹ para controlar los accesos a la misma. Por lo que tan solo pueden acceder a la misma aquellos usuarios que dispongan de clave y contraseña, por el momento. En cuanto a la aplicación móvil, se espera publicar en el Google Play Store en breve (cuando esté finalizada la versión *release*).

Con la realización del trabajo se espera que los objetivos queden satisfechos y que la utilización de la misma facilite a los usuarios el control de su información.

4.2. Arquitectura

Los componentes que conforman nuestro proyecto son los inherentes a la red social y a la aplicación móvil. Cómo se conectan entre sí para su funcionamiento es el aspecto que definirá la estructura del conjunto de nuestro sistema. La Figura 4.1 muestra la representación visual de la misma.

¹El registro estará habilitado excepcionalmente durante la impartición de los talleres (ámbito de la “Escola de Estiu”) en Julio.

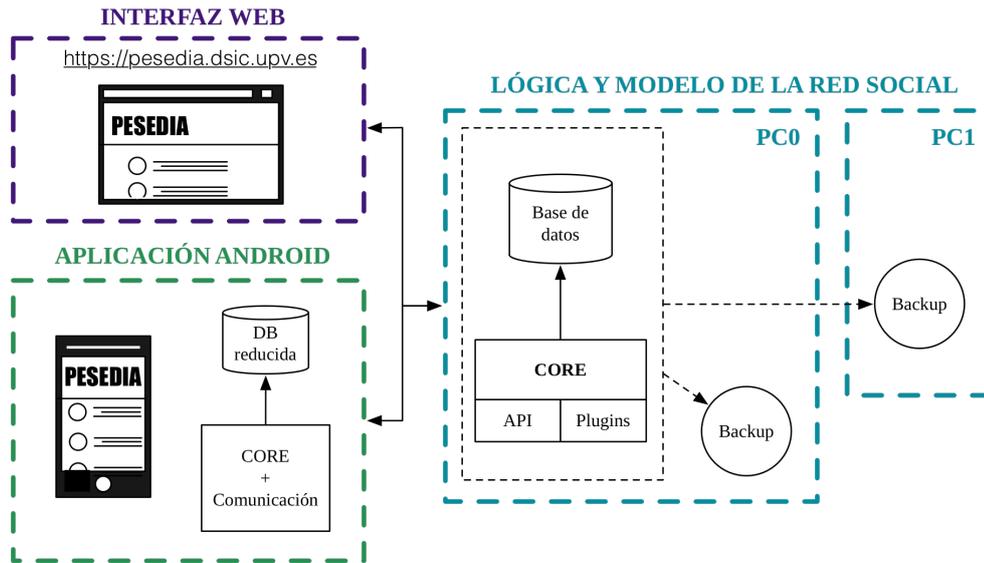


Figura 4.1: Arquitectura completa de nuestro proyecto de red social.

La instalación y despliegue de la red social mediante la herramienta Elgg será llevada a cabo en el servidor (con sistema operativo Ubuntu), el cual atenderá tanto las llamadas desde los navegadores de los clientes (interfaz web) como las efectuadas desde los dispositivos móviles. Nuestra red social estará configurada de acuerdo a una serie de plugins de terceros y propios instalados en el servidor. Durante la instalación se crearán las conexiones apropiadas con la base de datos que será donde resida el modelo de la propia red social. Por seguridad y como recomendación de la propia herramienta, se ha configurado la generación de copias de seguridad (backups), tanto en la propia máquina como en otra de confianza, mediante trabajos Cron para reforzar la seguridad de la información y poder hacer frente a posibles fallos en el sistema.

Se ha creado en el propio servidor un plugin donde se encuentran todas las funciones del API que han sido diseñadas, incluyendo diferentes restricciones de acceso a las mismas mediante identificación. Dichas funciones serán utilizadas desde la aplicación Android mediante consultas HTTP RESTful. La aplicación móvil estará formada por la propia lógica de la aplicación, y contará con dos tipos de almacenamiento persistente: *SQLite Databases*, para almacenar datos estructurados en memoria (obtenidos del API) con carácter privado; y *Shared Preferences*, que almacena primitivas encriptadas (`clave-valor`) para almacenar claves de sesión o contraseñas de los propios usuarios.

4.3. Tecnologías empleadas

A continuación, se explica qué tecnologías o herramientas se han utilizado para implementar el conjunto de la red social y la aplicación Android. Destacaremos las razones por las cuales se han elegido estas herramientas y haremos referencia a las partes de la arquitectura a las cuales pertenecen (ver Figura 4.1).

- **Elgg.** La herramienta para la creación de redes sociales Elgg nos ha permitido desarrollar gran parte de la “Lógica y modelo de la red social”, además de proporcionarnos

la “Interfaz web”. Las razones del uso de la misma han sido expuestas en el Apartado 3.1.1 del Capítulo 3.

- **PHP-Javascript-jQuery-HTML5-CSS.** Han sido utilizados todos estos lenguajes para el desarrollo o modificación de plugins para la red social (lenguajes web). PHP ha sido el principal lenguaje de desarrollo de los plugins, mientras que los otros lenguajes han sido utilizado como complementos para mejorar las vistas de la red social. El uso del lenguaje PHP ha sido una imposición de la propia herramienta Elgg; Javascript y jQuery han sido utilizados para dinamizar el contenido de las vistas debido a su facilidad de uso; y HTML5 y CSS por imposición de las tecnologías web para organizar el contenido y dotarlo de estilo.
- **Bash-Cron.** Se ha utilizado el lenguaje Bash desde la terminal para instalar los requisitos de la herramienta Elgg, realizar la propia instalación y configurar Cron en el servidor. Cron es un administrador regular de procesos en segundo plano que se ha utilizado para realizar copias de seguridad de la propia red social. En la propia documentación de Elgg se encuentra la información necesaria para configurar los procesos Cron.
- **MySQL.** Aunque la herramienta Elgg tiene soporte para diferentes tipos de bases de datos, se ha utilizado finalmente MySQL porque es escalable, por su facilidad de uso y mantenimiento, y por su gran comunidad de usuarios. El lenguaje para realizar consultas sobre MySQL es el lenguaje estándar SQL. La tecnología MySQL ha sido utilizada para la base de datos de la parte de la “Lógica y modelo de la red social”.
- **Apache.** Al igual que con la base de datos, la herramienta Elgg ofrece la posibilidad de gestionar el servidor web HTTP con cualquiera de las opciones existentes en el mercado. En nuestro caso hemos elegido Apache por comodidad ya que en la propia documentación de Elgg viene recogida la configuración del servidor Apache.
- **Android-Java.** La plataforma Android ha sido la elegida para desarrollar la aplicación móvil para smartphones, de acuerdo a las razones expuestas en el Apartado 3.1.2 del Capítulo 3. Con la tecnología Android, la cual está basada en el lenguaje Java, hemos sido capaces de desarrollar una interfaz de la red social más accesible para los usuarios. Esta nos ha proporcionado librerías para personalizar las vistas, adaptadores para contenidos y la tecnología para el almacenamiento de la información de forma segura (*SQLite Databases* y **Shared Preferences**).
- **Retrofit2.** Para realizar las consultas desde la “Aplicación Android” con el API de Pesian se ha utilizado Retrofit2. Al igual que esta librería, existen otras con la misma funcionalidad pero su facilidad de aprendizaje, uso y rendimiento nos ha convencido para utilizarla. Esta librería proporciona una capa de alto nivel que, además de facilitar las consultas, permite serializar las respuestas del API para recogerlas como objetos Java. Además cuenta con comunicación síncrona y asíncrona entre sus características.
- **Git.** Aunque este proyecto ha sido realizado por una sola persona, la utilización de sistemas de control de versiones es muy beneficioso porque permite organizar tu código frente a diferentes versiones del mismo (“releases” o “feature branches”). Además, puedes detectar errores con mayor facilidad y recuperar versiones antiguas. Por ello se ha utilizado el controlador de versiones Git.

Además de las que se han descrito, se han utilizado otras librerías de menor importancia para el proyecto que afectan sobretodo a aspectos de las vistas en la aplicación Android como *CircularImageView* o *CircularFloatingActionButton*, entre otras.

4.4. Implementación de Pesedia

En este apartado se describen todos los pasos seguidos para la implementación y el despliegue de la red social. Incluyendo un razonamiento de los plugins instalados y aquellos desarrollados por nosotros mismos.

4.4.1. Despliegue de la red social

La red social Pesedia estará ubicada en una máquina Ubuntu (versión 14.04 LTS), la cual actuará como servidor. Para llevar a cabo el despliegue e instalación de la red social mediante la herramienta Elgg se han seguido los siguientes pasos.

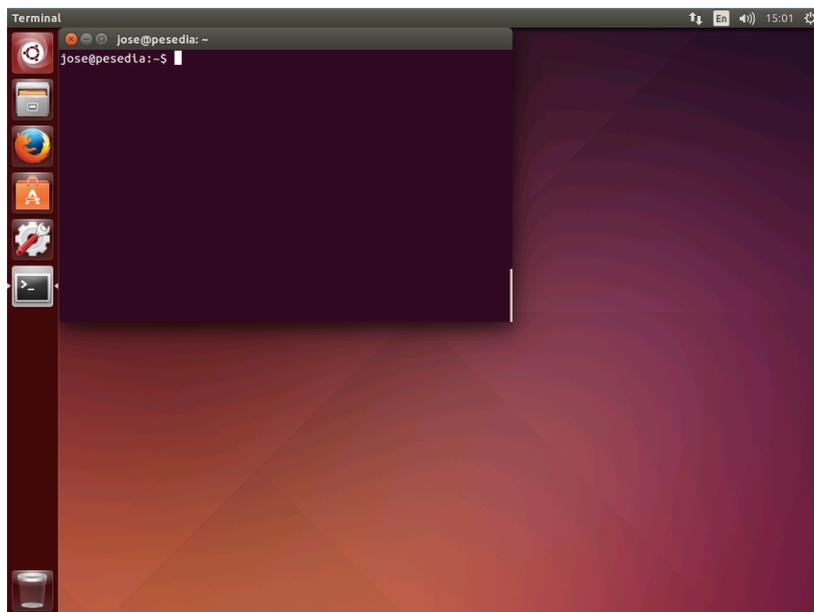


Figura 4.2: Máquina Servidor con Ubuntu 14.04 LTS que alojará a Pesedia.

1. Desde la terminal de Ubuntu, se ha actualizado el sistema de gestión de paquetes que viene en la instalación por defecto (**apt**). Esta actualización se ha realizado porque se va a utilizar **apt-get** para instalar los requerimientos necesarios de la herramienta Elgg.

```
$ sudo apt-get update
$ sudo apt-get upgrade
```

2. A continuación hemos instalado todos los requerimientos para tener configurada nuestra máquina de acuerdo al modelo LAMP (Linux, Apache, MySQL y PHP).

```
$ sudo apt-get install mysql-client mysql-server apache2 php5 php5-mysql
php5-gd php5-imap php5-ldap php5-odbc php-pear php-xml-parser php5-
xmlrpc php5-curl sysv-rc-conf wget unzip
```

Los paquetes que se han instalado corresponden con...

- Componentes de MySQL 5+ (`mysql-client` y `mysql-server`)
- Servidores web con soporte para reescritura de URL (`apache2`)
- PHP 5.5+ (`php5`) con extensión para bases de datos (`php5-mysql` y `php5-odbc`), para procesado de gráficos (`php5-gb`), para correo electrónico (`php5-imap`), para acceso simplificado a directorios (`php5-ldap`) y para reutilización de código (`php5-pear`).

3. Arrancamos los servicios LAMP.

```
$ sudo /etc/init.d/apache2 start
$ sudo /etc/init.d/mysql start
```

Y configuramos las propiedades de seguridad de MySQL de acuerdo a las necesidades de nuestro proyecto.

```
$ sudo mysql_secure_installation
```

4. Configuramos MySQL para Elgg creando la base de datos Pesedia, el usuario administrador y los permisos del mismo.

```
$ mysql -u root -p
mysql> create database pesedia;
mysql> grant all on pesedia.* to padmin@localhost identified by 'padmin';
mysql> flush privileges;
```

Y configuramos también el servidor web Apache con conexión segura (HTTPS) para Elgg que atenderá las peticiones de los usuarios.

```
$ sudo vim /etc/apache2/sites-available/pesedia.conf
<VirtualHost *:80>
    ServerName pesedia.dsic.upv.es
    ServerAdmin webmaster@localhost
    DocumentRoot /pesedia/public_html/
    Redirect permanent / https://pesedia.dsic.upv.es/
    <Directory /pesedia/public_html/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride All
        Order allow,deny
        allow from all
        Require all granted
    </Directory>
</VirtualHost>
$ sudo vim /etc/apache2/sites-available/pesedia-ssl.conf
<VirtualHost _default_:443>
    ServerName pesedia.dsic.upv.es
    ServerAdmin webmaster@localhost
    DocumentRoot /pesedia/public_html/
    SSLEngine on
    SSLCertificateFile /etc/apache2/ssl/certificado.pem
    <Directory /pesedia/public_html/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride All
        Order allow,deny
```

```

        allow from all
        Require all granted
    </Directory>
</VirtualHost>
$ sudo a2enmod rewrite
$ sudo /etc/init.d/apache2 restart

```

5. Creamos el espacio de directorios como recomienda Elgg.

- `/pesedia/public_html/`, directorio donde estará almacenada la instalación de Elgg, toda la lógica y vistas de la red social.
- `/pesedia/data/`, directorio donde la herramienta Elgg almacenará datos de la red social que no pueden ser almacenados en la base de datos (iconos de perfil, imágenes, documentos, etc.).
- `/pesedia/sitebackups/`, directorio donde se almacena las copias de seguridad de Pesedia. Además, otra copia será enviada a otra máquina por mayor seguridad.

Será necesario otorgar a la aplicación permisos para modificar las carpetas que contiene el directorio `/pesedia/` con la siguiente orden de comando.

```

$ chown -R www-data:www-data /pesedia/*
$ chmod -R 770 /pesedia/*

```

6. Descargamos Elgg desde su web oficial o bien desde terminal mediante las siguientes órdenes de comando y volcamos su contenido en el directorio `/pesedia/public_html/`.

```

$ wget https://elgg.org/getelgg.php?forward=elgg-2.1.2.zip
$ unzip getelgg.php\?forward=elgg-2.1.2.zip
$ sudo mv elgg-2.1.2 /pesedia/public_html/

```

7. Finalmente, para verificar que la instalación se realizó correctamente y configurar las últimas características de Elgg accedemos a la dirección web: `https://pesedia.dsic.upv.es/`

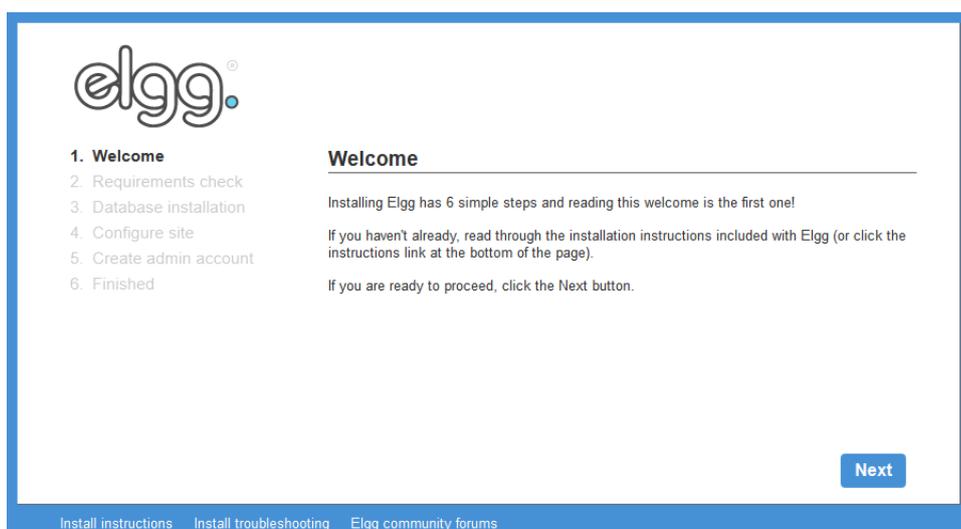


Figura 4.3: Últimos pasos para desplegar nuestra red social.

Tras la instalación se habrán creado las tablas en la base de datos siguiendo el modelo proporcionado por Elgg (ver Figura 4.4), el cual está formado principalmente por entidades (*ElggEntities*), relaciones entre entidades (*Relationships*), colecciones de acceso (*AccessCollections*), metadatos (*Metadata*) y anotaciones (*Annotations*).

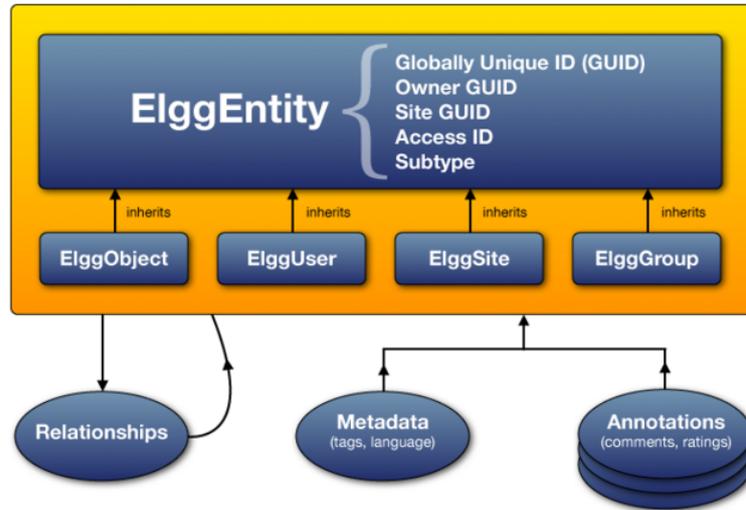


Figura 4.4: Diagrama del modelo de datos de Elgg.

4.4.2. Configuración de las características

El resultado de la instalación genera como producto la red social que puede apreciarse en la Figura 4.5. El núcleo base de la red social generada durante la instalación de Elgg cuenta con muy pocas características entre las que se incluyen las siguientes.

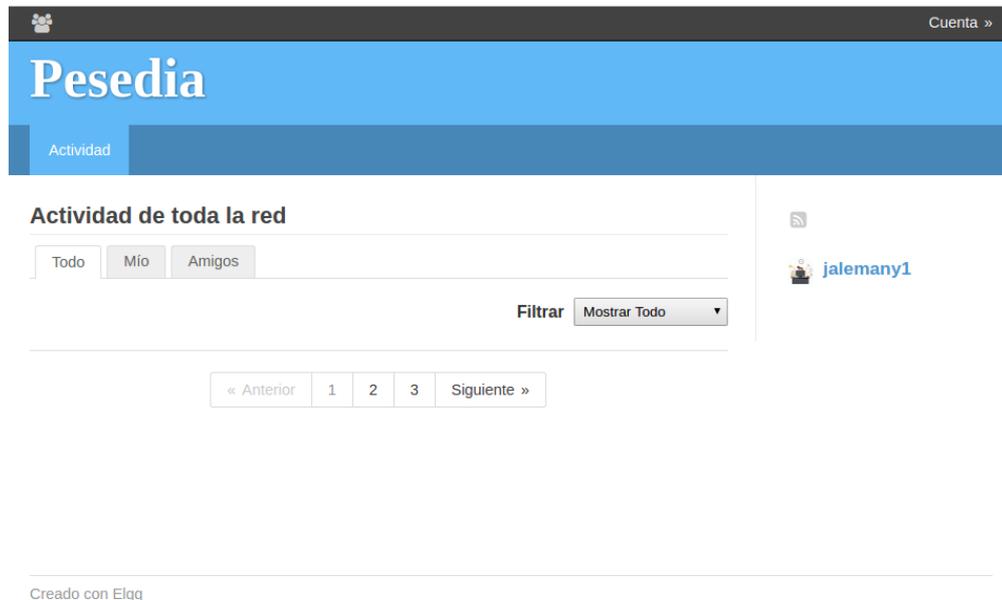


Figura 4.5: Red social base generada por Elgg.

- **Detalles de configuración** como icono de usuario, información personal con gestión de la privacidad para cada campo, nombre público, gestión de contraseñas, etc.
- **Gestión de la amistad** de carácter unidireccional (similar a la acción “seguir” en Twitter) y creación de colecciones de amigos (listas de acceso).
- **Página principal de actividad**, la cual incluye un selector de propiedad y de tipo de contenido.

Por ello se ha realizado una búsqueda entre los plugins ofrecidos por Elgg y los de terceros con el fin de dotar a Pesedia con mejores funcionalidades y cubrir con algunas de las medidas de seguridad y privacidad (vistas en el Apartado 2.2 del Capítulo 2). A continuación vamos a describir la selección de plugins instalados en Pesedia justificando su aplicación en la misma.

Profile + Profile Manager + Rounded Avatars	Añade un espacio personal del usuario donde se encuentra su información y puede gestionar qué espacios o widgets (muro, amigos, álbumes, etc.) relacionado con él estarán disponibles para el resto de usuarios.
--	--

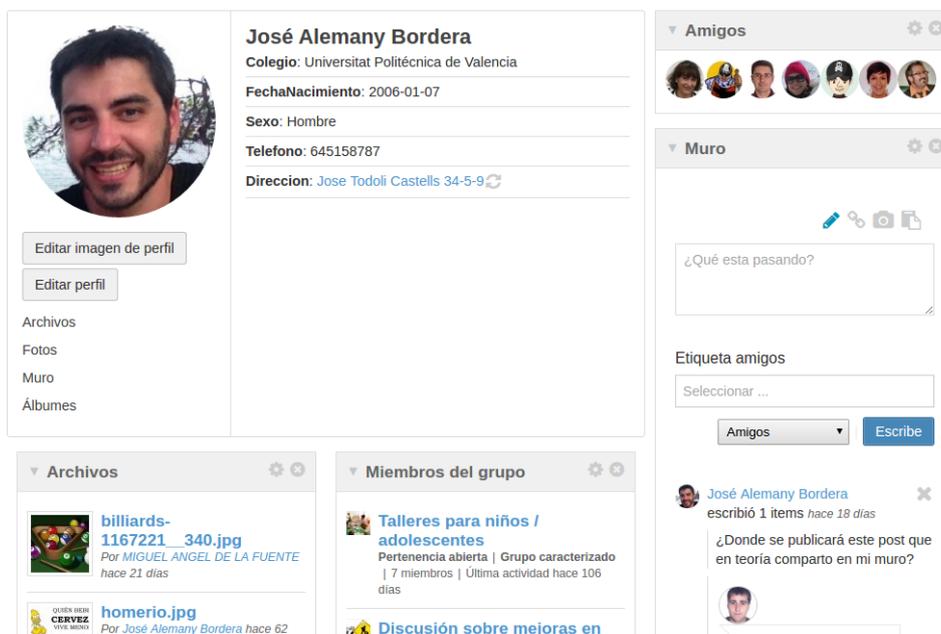


Figura 4.6: Plugins que afectan al perfil de usuario en Pesedia.

Friend Request + User Friends	Convierte la relación de amistad por defecto (unidireccional, similar a la acción “seguir” en Twitter) a una relación del tipo bidireccional como en Facebook con peticiones de amistad. Y mejora el espacio de Amistad con una interfaz más amigable.
---	--

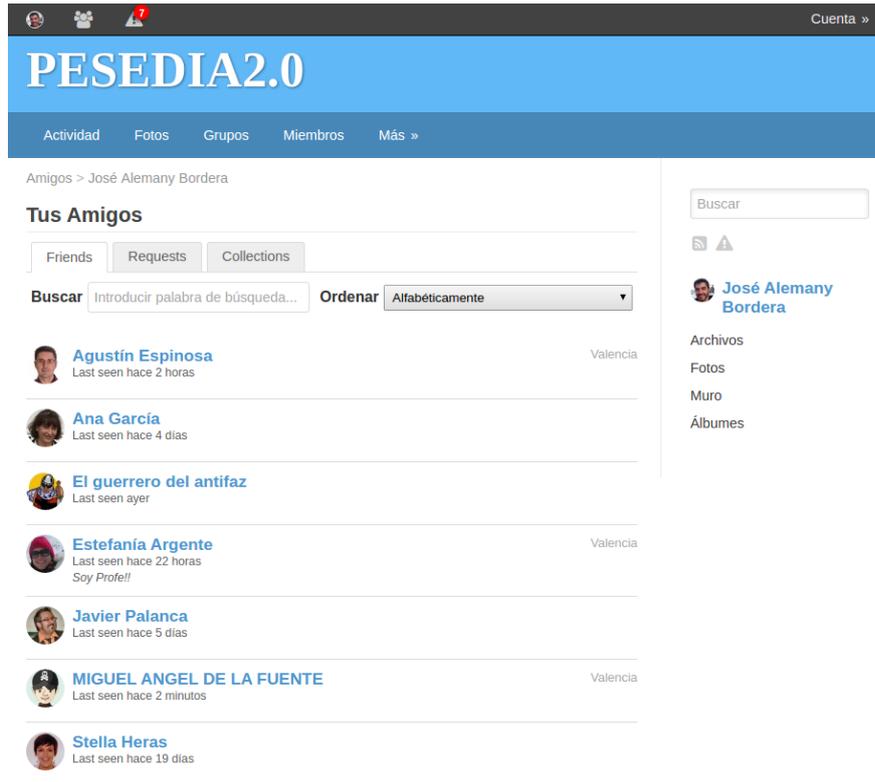


Figura 4.7: Plugins que afectan a la gestión de la amistad en Pesedia.

<p>Notifications + Notifier + Notification Templates</p>	<p>Añade un espacio en la barra superior donde se informa al usuario de las acciones de otros usuarios que estén relacionadas directamente contigo. Además dispone de su propia configuración para que el usuario establezca que tipo de acciones quiere que se le notifiquen y cuáles no.</p>
--	--



Figura 4.8: Plugins para notificaciones en Pesedia.

Search + Members + Tokenizing Autocomplete	Permite a los usuarios realizar todo tipo de búsquedas en la red social, y entre ellas listar a todos los usuarios de la red. Con el autocompletado se facilita esta tarea a sus usuarios proporcionándoles alternativas.
---	---

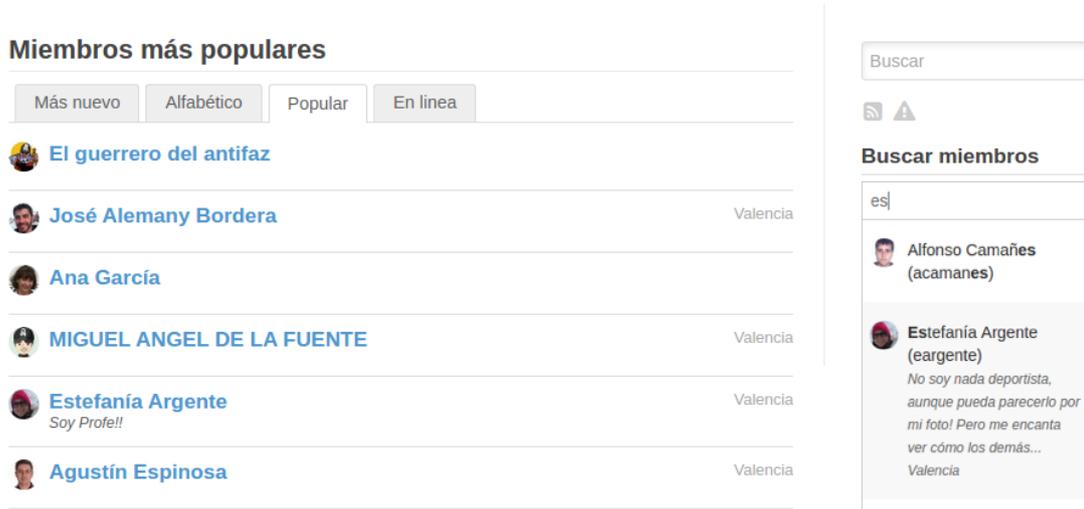


Figura 4.9: Plugins para búsqueda y autocompletado en Pesedia.

Reported Content	Facilita la denuncia de contenido inadecuado en la red social, e incluso de usuarios con mala conducta. Esta característica es importante para detectar malos usos de la red social.
------------------	--

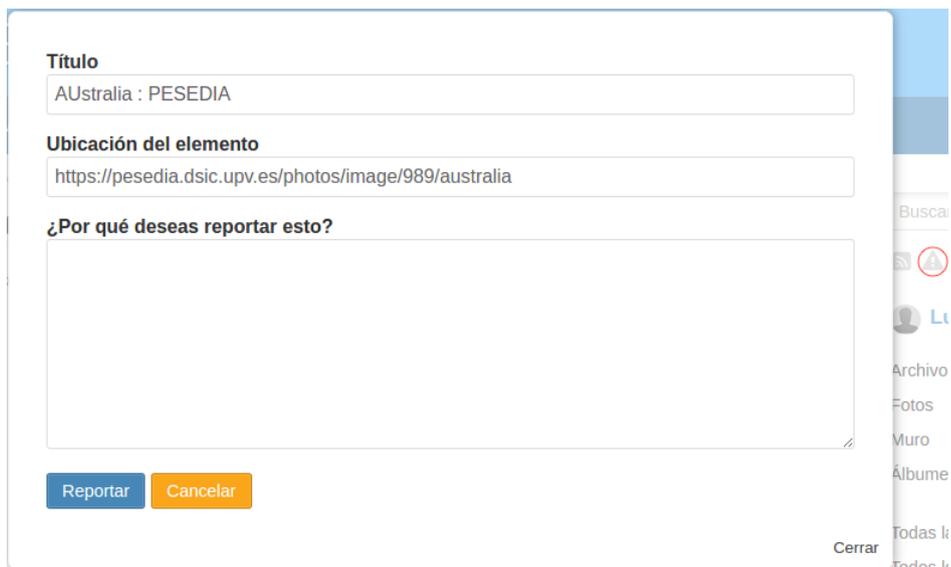


Figura 4.10: Plugin para reportar contenido en Pesedia.

<p>File + Tidypics Photo Gallery</p>	<p>Añade los espacios y la capacidad para gestionar, publicar y listar contenidos visuales, videos y otros. Además permite la creación de álbumes, etiquetado de fotos y, como todo en la red social, gestión de la privacidad de los mismos.</p>
--	---

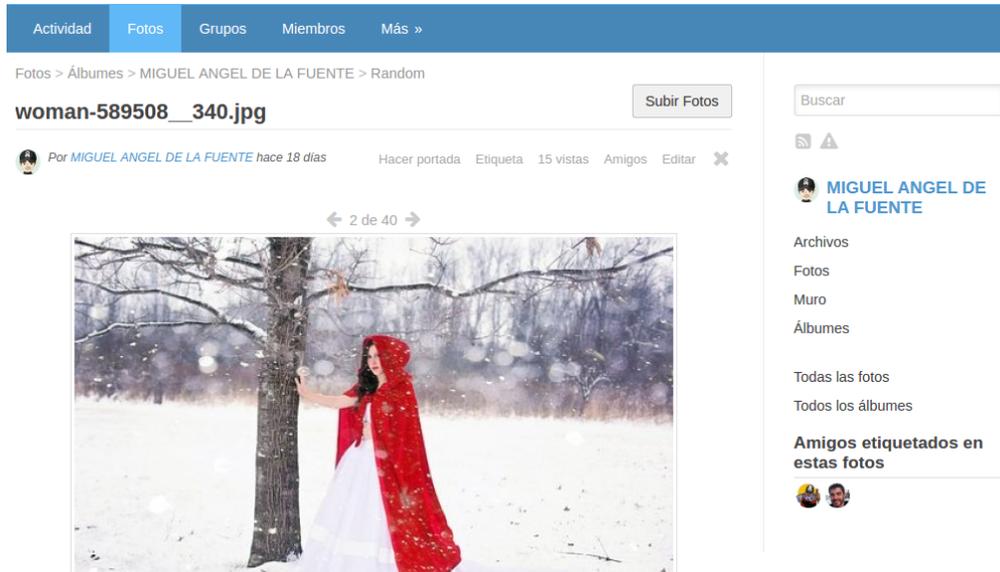
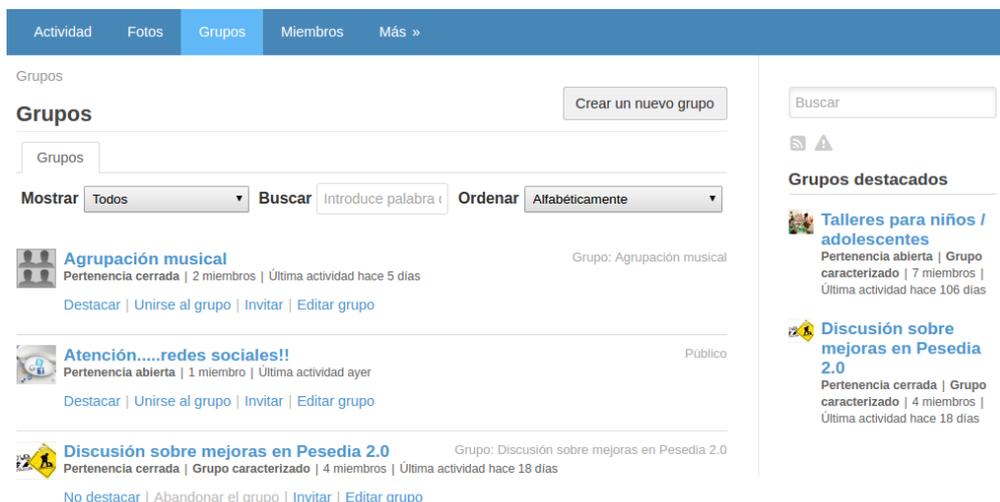


Figura 4.11: Plugins para soportar contenido visual y etiquetado en Pesedia.

<p>Groups + Group Lists + Group Membership Lists + Group List Sorting + Group invitations</p>	<p>Este conjunto de plugins ofrece la capacidad en la red social para crear y administrar grupos, roles dentro del grupo y configuración de privacidad del mismo (acceso, visibilidad, alcance, etc.). Además, gestiona tanto peticiones como invitaciones a los mismos.</p>
---	--



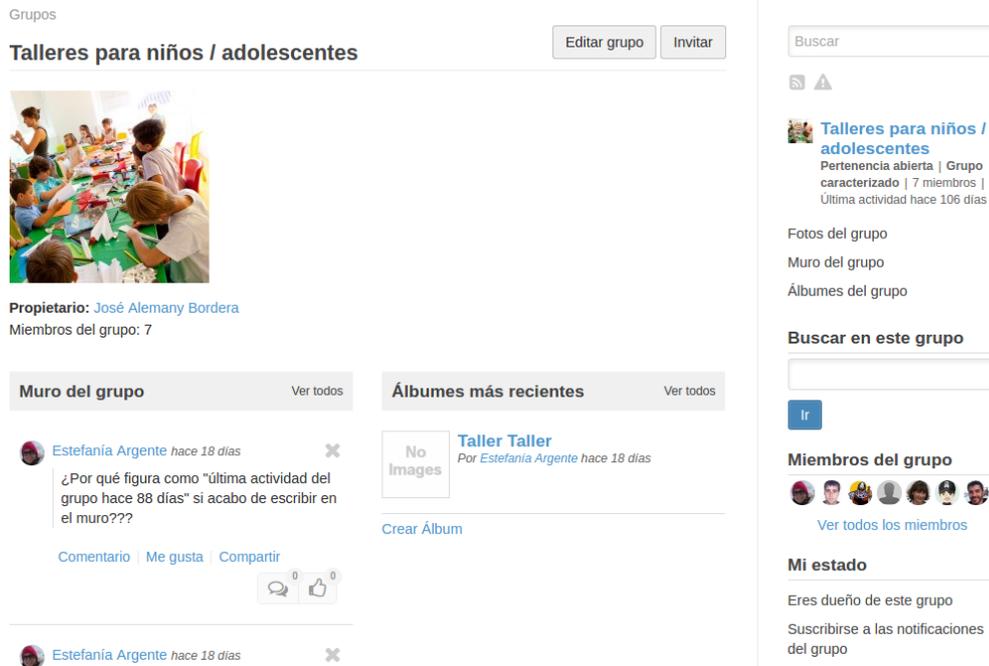


Figura 4.12: Plugins para soportar grupos en Pesedia.

<p> hypeWall + hypeApps + hypeDropzones + hypeLists + hypeScraper </p>	<p>Este conjunto de plugin permite la creación de contenido en la red social tanto en la página principal de la red social como en el muro de los usuarios. Las propiedades de privacidad relacionadas con estas funcionalidades se encuentran personalizables en el espacio de configuración de cada usuario. La herramienta para crear publicaciones en la red social está dotada con capacidad para publicar todo tipo de archivos, compartir enlaces con previsualización de los mismos, etiquetar a los usuarios con los que estas, el lugar desde el que publicas y seleccionar la audiencia con la que vas a compartir la publicación.</p>
<p> Comments + Likes + Tagg Tools + hypeInteractions </p>	<p>Estos plugins corresponden con funcionalidades para la interacción con el contenido que otros usuarios han creado o compartido. Con lo que las publicaciones podrán ser comentadas (incluso los comentarios podrán ser respondidos), votar por su popularidad mediante “likes” o etiquetar usuarios.</p>

Muro > Muro de El guerrero del antifaz

Muro de El guerrero del antifaz

¿Qué esta pasando?

Etiqueta amigos

Seleccionar ...

Amigos ▾ Escribe

Estefanía Argente *hace 5 días* ✕

Hola estamos reunidos - con Ana Garcia, El guerrero del antifaz

Comentario | Me gusta | Compartir

El guerrero del antifaz compartió un enlace *hace 12 días* ✕

¿Qué pasará si enlazan vídeos "no-apropiados" ? [DETRÁS DE CÁMARAS: Game of Thrones Episodios 3 & 4 | Season6 \(Subtitulado\)](#)

DETRÁS DE CÁMARAS: Game of Thrones Episodios 3 & 4 | Season6 (Subtitulado)
[youtu.be](#)
 VÍdeo donde se muestra el detrás de todo el trabajo del equipo de producción. Efectos Visuales, prostéticos, etc. Aquí sabremos como se hicieron esas escenas...

Comentario | Me gusta | Compartir

Figura 4.13: Plugins para soportar el “Muro” en Pesedia, y la creación de contenido en él.

El guerrero del antifaz escribió en el muro de Estefanía Argente *hace 5 días* ✕

Muro de Fany, etiquetando a Ana - con Ana Garcia

Comentario | Me gusta | Compartir

Estefanía Argente: Probando probando *hace 5 días*

Responder | Me gusta | Compartir | Editar | Eliminar

Comentar

Figura 4.14: Plugins para interactuar con el contenido en Pesedia.

Web Services	Este plugin es necesario porque proporciona el marco de trabajo y las funciones necesarias para definir nuestro propio API.
Elgg Developer Tools + Diagnostics + Garbage Collector + HTMLawed + Legacy URL Support + Login As + Log Browser	Conjunto de plugins que proporcionan características de administrador y desarrollador de la red social, entre las que se encuentran: logueo como otro usuario de la red, visualización de acciones y eventos, propiedades de seguridad de la red social, etc.

Tabla 4.1: Resto de plugins referentes a la parte de desarrollo y administración en Pesedia.

Con todas estas funcionalidades instaladas la red social Pesedia dispone de la propiedades y algunas medidas de privacidad requeridas para nuestro proyecto. Aún así las opciones de “compartir contenido” (factor altamente afectante a la privacidad por su capacidad de propagación de la información) y “comunicación con el API” desde la aplicación móvil tendrán que ser implementadas mediante la creación de nuestros propios plugins.

4.4.3. Implementación de plugins

La herramienta Elgg proporciona una serie de guías de diseño para el desarrollo de plugins, además de funciones de librería que facilitan la implementación de los mismos. Como gran advertencia nos recomienda no modificar directamente el “core” de la red social ya que podríamos dañar parte de la misma, e insiste en que desarrollemos nuestras ampliaciones a través de los plugins. Entre las recomendaciones que ofrece está la estructuración de los plugins siguiendo las directrices que detallamos a continuación.

- Se deben definir obligatoriamente en la raíz de nuestros plugins los siguientes ficheros:
 - `start.php`, este fichero contiene todas las llamadas de inicio del plugin, registro de acciones y vistas, y captura de eventos. Este código habilitará toda la funcionalidad del plugin implementado.
 - `manifest.xml`, este fichero debe contener una descripción del plugin con los siguientes elementos requeridos: *id*, *name*, *author*, *version*, *description* y *requires*. Este fichero será consultado desde el panel de administración de plugins, si todo está correcto podremos activar nuestro plugin.

Como ficheros opcionales que pueden ser definidos también en la raíz, se encuentran `activate.php` y `deactivate.php` que contendrán código que se ejecutará automáticamente en la activación y desactivación, respectivamente, de nuestro plugin.

- Los directorios que lo formarán deberán ser los siguientes:
 - `./actions/`, donde se definirán acciones que afecten a la base de datos principalmente. Estas acciones serán registradas en el fichero de inicio `start.php` y estará asociadas con formularios de la red social lanzados por un botón de “submit” (ej. “Publicar”). Para que las acciones se ejecuten la ruta dentro de `./actions/` debe coincidir con la de la vista situada en `./views/default/forms`.

- `./classes/`, en este directorio serán el lugar donde deberemos definir nuestras clases, las cuales serán reconocidas automáticamente por Elgg al activar el plugin.
- `./languages/`, este directorio contiene diccionarios (`clave-valor`) de palabras para cada uno de los idiomas contemplados por los desarrolladores para soportar el cambio de idioma desde las preferencias de los usuarios.
- `./vendor/`, aquí es donde se incluyen las librerías de terceros. Las dependencias que estas presentan pueden ser gestionadas mediante *Composer*.
- `./views/default/`, todo el código referente a la generación de las vistas de la red social será almacenado en este directorio. Las rutas de los ficheros definidos dentro de este directorio tendrán una especial relevancia para la red social. La coincidencia de rutas dentro de este directorio provocará la sobrescritura de las vistas. Además, las funciones proporcionadas por Elgg para referenciar vistas estarán relacionadas dentro de los siguientes subdirectorios:
 - `forms/`, en este directorio se almacenarán los formularios que tendrán asociada una acción en la red social. Elgg proporciona la función `elgg_view_form()` para encapsular las vistas tipo formulario, a la cual pasaremos la ruta del fichero partiendo de este directorio.
 - `resources/`, para la renderización de vistas completas se utilizará este directorio que, al igual que el anterior, tiene una función Elgg que facilitará su llamada `elgg_view_resource()`.
 - `widgets/`, aquí es donde se diseñan paneles que conformarán el perfil del usuario. Para definir un *widget* es necesario registrarlo en el fichero de inicio (`start.php`) y crear los ficheros: `edit.php`, que contendrá los parámetros de configuración del mismo; y `content.php`, que tendrá en cuenta los anteriores parámetros de configuración para mostrar el contenido del *widget*.

El resto de vistas pueden ser definidas en los directorios que estimemos oportunos, teniendo en cuenta que podemos sobrescribir otras vistas por lo que se recomienda crear directorios con el nombre de nuestro plugin para evitar esta situación. Las vistas personalizadas serán llamadas con la función Elgg `elgg_view()`.

Tomando en consideración estas recomendaciones y analizando plugins con características similares a la funcionalidad que se pretende alcanzar se han diseñado los siguientes plugins que facilitarán el estudio de nuevas técnicas más avanzadas para mejorar la privacidad en las redes sociales.

Share Content

Debido a la necesidad de disponer de los elementos naturales de una red social para poder estudiar la privacidad en la misma se ha diseñado e implementado un plugin que ofrece dicha funcionalidad. La funcionalidad completa del plugin implementado es la que describimos a continuación.

- Añade la acción de “Compartir” a las publicaciones que pueden ser compartidas.
- Permite difundir contenidos entre los usuarios de la red de forma sencilla.

- Puede seleccionarse el destino sobre el que se le aplicará la publicación compartida:
 - Nuestro propio “Muro”, sobre el cual tendremos capacidad para determinar su alcance o privacidad mediante el selector de audiencia.
 - El “Muro” de un amigo, permitiéndonos el autocompletado durante la búsqueda.
 - En un grupo, permitiéndonos el autocompletado durante la búsqueda.
- Incluir un comentario para la difusión.
- Nuevas configuraciones para la privacidad como: quien puede publicar en tu “Muro” y que grado de privacidad otorgas a las publicaciones que en él se hacen.

En cuanto al desarrollo del mismo, la estructuración del árbol de directorios seguida en el plugin desarrollado es la que se aprecia en la Figura 4.15. El plugin incluye los dos ficheros requeridos `manifest.xml` y `start.php` en el que se definen y establecen las propiedades y funcionalidades del mismo. El código que contienen puede apreciarse en las Figuras 4.16 y 4.17.

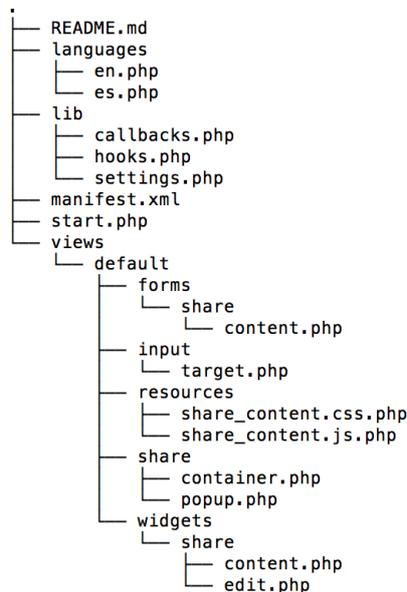


Figura 4.15: Árbol de directorios del plugin *Share Content*.

Por motivos de organización del documento `start.php`, se han repartido las funciones en diferentes archivos dentro del directorio `./lib/` atendiendo a sus utilidades:

- `callbacks.php`, se encuentran las funciones donde se determina el espacio de entidades que se ofrecen a los usuarios para el autocompletado. Este plugin proporciona autocompletado para la selección de los usuarios que nos permitan publicar en su muro y para los grupos a los cuales tengamos acceso.
- `settings.php`, aquí se determinan las configuraciones por defecto del plugin tanto globales como individuales de cada usuario. Ya que en este plugin se han incluido, además, nuevas propiedades de privacidad para el objeto “Muro” relacionadas con compartir.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <plugin_manifest xmlns="http://www.elgg.org/plugin_manifest/1.8">
3     <name>Share Content</name>
4     <author>Jose Alemany</author>
5     <version>1.0</version>
6     <description>Ability to share content social network with whom you
7         choose.</description>
8     <copyright>(C) Jose Alemany 2016</copyright>
9     <license>GNU Public License version 2</license>
10
11     <requires>
12         <type>elgg_release</type>
13         <version>2.0.2</version>
14     </requires>
15 </plugin_manifest>

```

Figura 4.16: Archivo *manifest* del plugin *Share Content*.

- `hooks.php`, en este fichero se encuentran aquellas funciones que se ejecutan al capturar eventos en la red social. Algunos de los eventos capturados son: la creación de las vistas de publicaciones para la inclusión de la acción “Compartir”, la acción de creación de una publicación para establecer el destino de la misma y la comprobación de permisos para publicar en el “Muro” de otro usuario atendiendo a las nuevas propiedades de privacidad incluidas.

Como puede apreciarse en el árbol de directorios (Figura 4.15), el plugin desarrollado tiene soporte para la lengua española e inglesa al crearse los diccionarios de palabras de estos dos idiomas. Por otra parte, las vistas que han sido implementadas han sido las correspondientes a una ventana de tipo *popup* cuyo contenido utiliza elementos de los plugins de creación de contenido para dotar a la red de mayor coherencia. Dicha vista extenderá el contenido de la vista utilizada (*wall_publicator*) con un nuevo campo *target* que determinará donde compartiremos una publicación: en nuestro muro, en el muro de un amigo o en un grupo. Elementos de estilo y dinamismo de la propia vista se han definido en el directorio `resources/`.

Con el desarrollo del plugin *Share Content* y su instalación en la red social conseguimos añadir un apartado de funcionalidad importante que será controlado mediante las configuraciones de privacidad, el cual se pretende investigar en el proyecto nacional para determinar su relevancia en cuanto a privacidad. El aspecto final del desarrollo de este plugin puede apreciarse en las Figuras 4.18a y 4.18b, que reflejan los dos espacios ampliados.

Experiments

Para el estudio y validación de los factores que son capaces de influir en la privacidad de la información en las redes sociales, se ha desarrollado un plugin que realiza pruebas sobre datos sintéticos. Se espera que con los resultados obtenidos de este marco de trabajo se pueda profundizar en medidas de privacidad más efectivas para los usuarios. Este plugin se ha desarrollado íntegramente para la investigación más avanzada que se efectuará en el

```

1  <?php
2  /**
3   * share_content
4   *
5   * @package pesedia
6   * @subpackage share_content
7   *
8   * @author Jose Alemany <jalemany1@dsic.upv.es>
9   */
10
11 require_once __DIR__ . '/lib/settings.php';
12 require_once __DIR__ . '/lib/hooks.php';
13 require_once __DIR__ . '/lib/callbacks.php';
14
15 elgg_register_event_handler('init', 'system', 'share_init');
16
17 /**
18  * Initialize the plugin
19  * @return void
20  */
21 function share_init($segments) {
22
23     elgg_extend_view('elgg.css', 'resources/share_content.css');
24     elgg_extend_view('elgg.js', 'resources/share_content.js');
25
26     // handler to include the option of sharing
27     if (elgg_is_active_plugin('hypeInteractions')) {
28         elgg_register_plugin_hook_handler('register', 'menu:interactions',
29             'share_river_menu_setup');
30     } else {
31         elgg_register_plugin_hook_handler('register', 'menu:river', '
32             share_river_menu_setup', 400);
33     }
34
35     // view and form to share a post
36     elgg_register_ajax_view('share/popup');
37     elgg_extend_view('forms/wall/content', 'forms/share/content', 499);
38     elgg_register_plugin_hook_handler('action', 'wall/content', '
39         set_share_target_destination');
40
41     // audience of publishing that other users post on the owner's Wall
42     elgg_extend_view('widgets/wall/edit', 'widgets/share/edit', 501);
43     elgg_extend_view('widgets/wall/content', 'widgets/share/content', 499)
44         ;
45
46     return true;
47 }

```

Figura 4.17: Archivo *start* del plugin *Share Content*.

proyecto nacional que enmarca este trabajo fin de máster y en el que está enmarcada la realización de mi tesis doctoral.

Como puede observarse en la Figura 4.19, este plugin es más complejo. Y su acceso

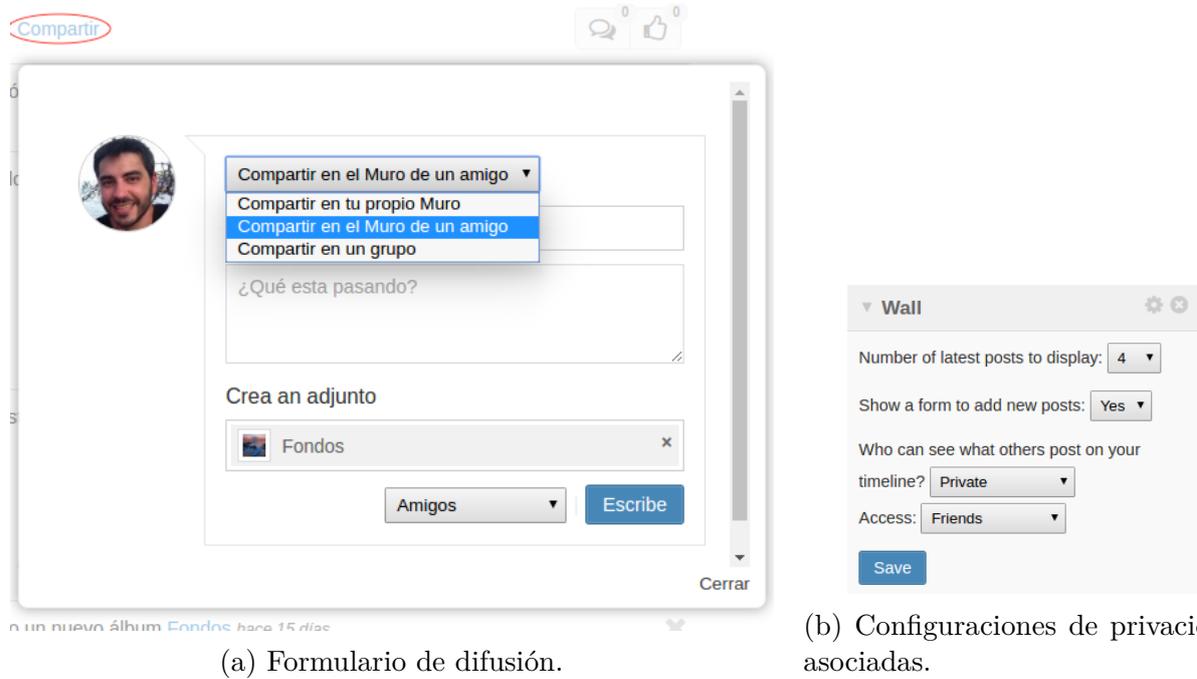


Figura 4.18: Aspecto final del plugin (*Share Content*).

se restringe exclusivamente para los administradores de la red social. La funcionalidad del plugin *Experiments* implementado es la que describimos a continuación.

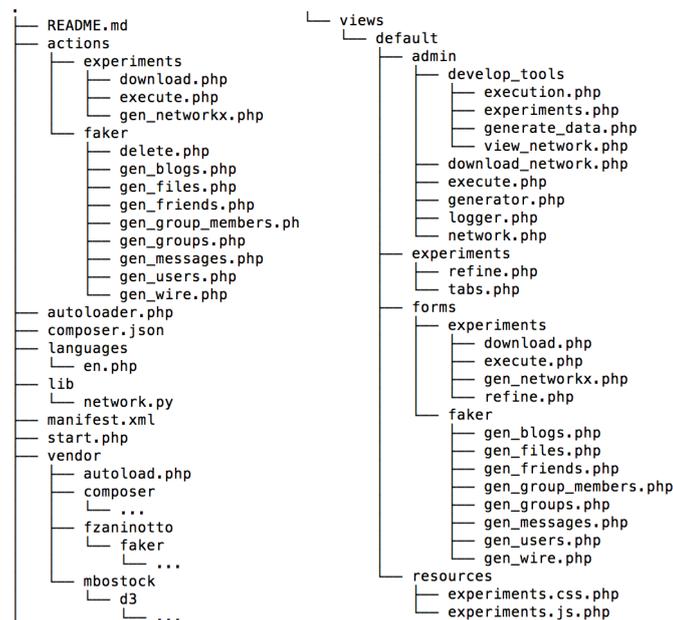
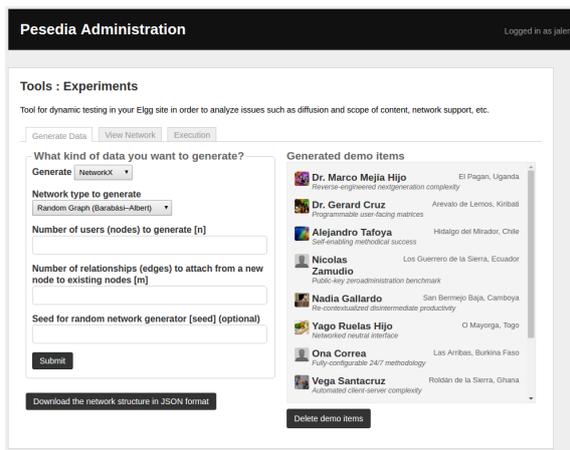


Figura 4.19: Árbol de directorios del plugin *Experiments*.

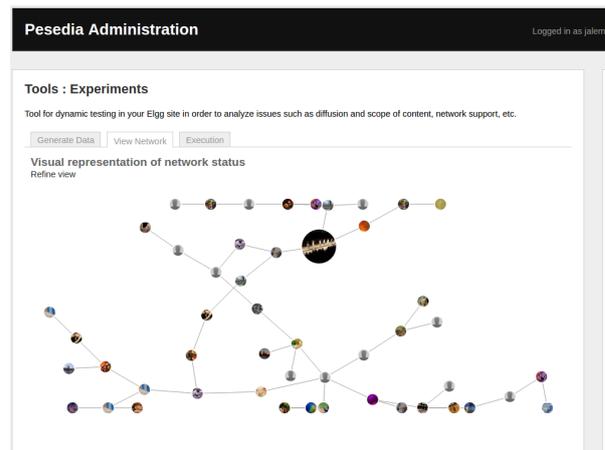
- Permite la creación de datos sintéticos (usuarios, relaciones, contenido, grupos, etc.) de dos formas:
 - Individual: generando los diferentes tipos de datos de forma aleatoria y por necesidad.

- Directa: mediante la utilización de la librería *NetworkX* la cual genera tipos de redes sociales específicas de acuerdo a unos parámetros de entrada.
- Capacidad para descargar en formato *JSON* una descripción y la estructura de la red para ser tratada externamente con la librería *NetworkX*.
- Proporciona una ventana de log, para gestionar los datos creados.
- Representación visual de la red social y capacidad de manipulación de la misma mediante la librería gráfica *D3js*.
- Ejecución controlada de actividades evolutivas en el tiempo con posibilidad de monitorización y visualización de resultados. El estudio que se realiza con las ejecuciones es el del cálculo de la probabilidad del alcance de la información, mediante la difusión, en los diferentes niveles de profundidad de la red social, detectando así los usuarios más influyentes. Para determinar qué tipo de acción realiza cada usuario, se han utilizado datos estadísticos, pero se pretende incluir aspectos más complejo como la personalidad y la similitud entre usuarios.

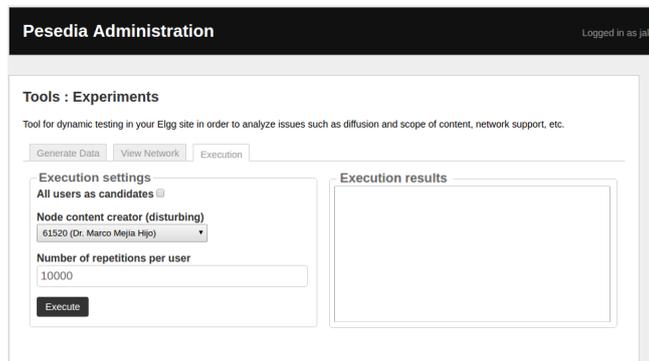
Una pequeña muestra del aspecto final de este plugin puede apreciarse en la Figura 4.20, que refleja los espacios disponibles para la realización adecuada de las pruebas en la red social.



(a) Generación de datos sintéticos.



(b) Visualización de la estructura de la red social.



(c) Ejecución y visualización de resultados.

Figura 4.20: Aspecto final del plugin (*Experiments*).

4.4.4. Medidas de privacidad implementadas

Una vez desplegada la red social, con la instalación de plugins de terceros y la implementación de nuestras propias características, esta es capaz de soportar la mayoría de las medidas de privacidad que mencionamos en el Apartado 2.2 del Capítulo 2. En este apartado vamos a resaltar aquellas que se han llevado a cabo y el nivel de adaptación con la red social conseguido.

La red social Pesedia permite a sus usuarios gestionar la mayoría de aspectos de la privacidad desde el apartado de configuración del usuario. Tan solo, en el caso de las propiedades del “Muro”, dichas propiedades de privacidad no se encuentran recogidas en el apartado de configuración global sino en la configuración del *widget* porque resultan más intuitivas. En cuanto a la capacidad de seleccionar la audiencia o nivel de acceso, la mayoría de características de los usuarios en la red social permiten establecer dicha privacidad (PRIVATE, FRIENDS, PUBLIC o configurados por el usuario) cumpliéndose ampliamente este aspecto de la privacidad.

En cuanto a la posibilidad de creación de listas de acceso, la red social soporta dicha característica que denomina con el nombre de “colecciones de amigos”, las cuales son utilizadas para personalizar el nivel de acceso de su información. La edición de las publicaciones y la capacidad de borrarlas es propio de la funcionalidad base de la red social al igual que el alcance de la publicación, con lo que estos aspectos están cubiertos. Aun así, se estudia mejorar la visibilidad de los identificadores del alcance para que los usuarios sean capaces de identificarlos con un simple vistazo. El registro de actividad también se incluye, pero se está estudiando mejorarlo para poder modificar la privacidad de publicaciones ya publicadas.

Además, se ha dejado preparado el entorno para la realización pruebas sobre la red social con el fin de diseñar una medida capaz de detectar, obteniendo información del estado de la red social, los usuarios más influyentes, las personalidades de los mismos y la afinidad entre ellos para proporcionar recomendaciones de privacidad a sus usuarios (como recomendar colecciones de amigos o futuras audiencias de una publicación atendiendo a la sensibilidad de la información que se comparte). Incluso, indicarles la cantidad de usuarios que potencialmente accederán a dicha publicación si alguien la comparte o etiqueta en la misma, atendiendo a un valor de riesgo en privacidad asociado a cada usuario, el cual se ha calculado de acuerdo a su actividad y a datos estadísticos. Actualmente se está recogiendo esta información en la red social para proporcionar funcionalidad futura, la cual depende de los datos reales de uso.

4.4.5. Creación del API

La creación de un API para la comunicación entre la plataforma móvil y la red social ha sido desarrollada como una extensión de la red social (mediante plugins). Se ha dedicado un apartado específico para la descripción de este plugin por su relevancia.

Entre las extensiones que proporciona la herramienta Elgg se encuentra un plugin (*Web Services*) que es el encargado de definir los elementos y establecer los requisitos necesarios para la comunicación. Entre las funciones utilizadas que proporciona dicho plugin se encuentran:

- `elgg_ws_expose_function`, esta es la más importante ya que nos permitirá expo-

ner o publicar las funciones disponibles para nuestra API. Esta función requiere que definamos:

- `$api_function`, el nombre de la función que estará disponible en el API.
 - `$local_function`, la función donde se encuentra la funcionalidad capaz de atender la consulta y devolver un resultado de la misma.
 - `$params`, los parámetros que se esperan recibir como entrada de la función con sus propiedades (tipos, nombres, valores por defecto, etc.).
 - `$descr`, una descripción de la funcionalidad del mismo.
 - `$type`, el tipo de consulta RESTful (GET / POST).
 - `$api_auth`, sí será necesaria una autenticación del API para acceder a la función.
 - `$user_auth`, sí será necesaria una autenticación de usuario para acceder a la función.
- Y dos ejemplos de funciones expuestas que son básicas y necesarias en cualquier API.
 - `auth.gettoken`, función para conseguir la autenticación de usuario en la red social mediante usuario y contraseña. Esta nos proporciona un *token* el cual deberemos utilizar para autenticarnos como usuarios en posteriores consultas al API. Dicho *token* tiene caducidad por lo que tendrá que ser renovado periódicamente.
 - `system.api.list`, esta función nos permite listar todas las funciones disponibles en el API y sus propiedades.

Tomando como ejemplo estas funciones se ha implementado nuestro propio plugin *PesediaAPI* que contiene todas las funciones necesarias para la aplicación móvil. En la tabla que se muestra a continuación (ver Tabla 4.2), se describen todas las funciones disponibles para la comunicación con el API, una explicación de su comportamiento y los parámetros de entrada que se esperan.

Para comunicarse con el API RESTful es necesario realizar las peticiones a la URL formada por la dirección web de Pesedia (<https://pesedia.dsic.upv.es>), más `/services/api/rest/json/`, el método y los parámetros que requiere. Un ejemplo de dirección URL de una petición sería el siguiente:

```
https://pesedia.dsic.upv.es/services/api/rest/json/&method=list.river?limit=20&auth_token=56f23c6a8d9a24846b67ee0353e8a031
```

Como puede observarse, esta consulta tiene el parámetro `$auth_token` incluido. Este parámetro debe utilizarse, incluyéndose en las consultas, debido a que todas nuestras funciones del API requieren una previa autenticación como usuario en el sistema. El resultado obtenido de las consultas se obtendrá siempre en formato *JSON*.

<code>reg.user</code> (<code>\$email</code> , <code>\$password</code>)	Esta función permite a los usuarios registrarse en la red social. Para ello deben proporcionar un correo electrónico y una contraseña válida (al menos 6 caracteres).
<code>list.river</code> (<code>\$interval_ts</code> , <code>\$limit</code>)	Con esta función seremos capaces de listar el contenido de la página principal para visualizarlo desde la aplicación móvil. Los parámetros que utilizamos como entrada de la función corresponden con el intervalo de tiempo, del cual queremos obtener publicaciones, y el límite numérico del mismo, con el fin de no obtener resultados duplicados.
<code>list.entities</code> (<code>\$type</code> , <code>\$subtype</code> , <code>\$limit</code>)	Esta función tiene un comportamiento similar a la anterior pero esta está más relacionada con entidades de tipo usuario, grupos o álbumes; en las cuales no tiene sentido utilizar intervalos de tiempo para obtener resultados. Sus parámetros hacen referencia al tipo de contenido que deseamos listar.
<code>get.entity</code> (<code>\$id</code>)	Permite obtener las propiedades de cualquier tipo de entidad (usuarios, grupos, álbumes). Sólo recibe como parámetro de entrada el identificador de entidad que lo define para obtener sus propiedades.
<code>post.entity</code> (<code>\$type</code> , <code>\$title</code> , <code>\$tags</code> , <code>\$target</code> , <code>\$access</code>)	Esta función nos permite publicar contenido en la red social (texto, imágenes, etiquetas, etc.). Esta función es utilizada tanto para crear una nueva publicación, como para compartir o comentar una ya existente. Esto será posible variando los parámetros de “tipo” y “destino”. Las publicaciones que se realicen podrán ser configuradas con un nivel de acceso determinado.
<code>like.entity</code> (<code>\$id</code>)	La función “me gusta” solo tomará como parámetro el identificador de la entidad que queramos popularizar para añadirle un nuevo <i>like</i> .
<code>friend.request</code> (<code>\$id</code>)	Envía una petición de amistad al usuario que hayamos seleccionado, en este caso, al usuario identificado con el valor de entrada pasado a la función.
<code>refresh.token</code> (<code></code>)	Esta función genera un nuevo <i>token</i> con una fecha de caducidad extendida respecto a la del anterior valor del token.

Tabla 4.2: API de comunicación de Pesedia.

4.5. Implementación de la aplicación móvil

La aplicación Android es el medio mediante el cual los usuarios interactúan de forma más sencilla dentro de la red social, ya que pueden subir contenido multimedia (fotos) y geolocalizar sus publicaciones de forma más sencilla. Además, la posibilidad de acceso mediante la aplicación móvil facilitará la explotación de la propia red social. La aplicación tiene como nombre Pesedia, que coincide con el nombre de la propia red social. Esta aún no se encuentra disponible en la tienda oficial de Android (Google Play Store), pero se espera publicar la primera versión en breve. El icono diseñado para la aplicación puede verse en la Figura 4.21.



Figura 4.21: Icono de la aplicación móvil Pesedia.

Entre los aspectos más importante de la funcionalidad de la aplicación móvil, resaltamos los siguientes.

- Como es lógico los usuarios podrán loguearse en la red social.
- La aplicación tendrá capacidad para visualizar publicaciones realizadas en la red social e interactuar con las mismas mediante comentarios, *likes* y la opción de compartirlas.
- Los usuarios serán capaces de crear sus propias publicaciones incluyendo texto, imágenes tomadas desde sus *smartphones*, incluir su localización, etiquetar a otros usuarios y compartir enlaces. Además de poder establecer el alcance de dicha publicación.
- Se podrá buscar usuarios para solicitar peticiones de amistad.
- Y se dispondrá de un espacio de usuario para visualizar las propiedades de los mismos.

A continuación vamos a describir cada uno de los aspectos que componen la aplicación móvil que se ha implementado para este trabajo fin de máster. Describiendo cómo se realiza la comunicación con el API, cómo gestiona la aplicación la información obtenida y la funcionalidad de las vistas (interfaz de usuario) disponibles en la aplicación.

4.5.1. Comunicación con el API

La aplicación móvil Android desarrollada es parte de la interfaz de la red social Pesedia, por lo que todo su contenido es obtenido de la misma. Debido a ello, la comunicación con el API RESTful es tan importante para el correcto funcionamiento de la aplicación. En este apartado se pretende describir cómo se realiza dicha comunicación, la tecnología empleada y el mecanismo de autenticación requerido.

Las consultas se realizan mediante el protocolo HTTP siguiendo el estándar de comunicación de arquitecturas REST. Para realizar estas consultas se estudió brevemente las alternativas que ofrecían las librerías disponibles en Android, ya que el propio sistema no tiene soporte para realizar consultas de alto nivel con el API. Finalmente se concluyó que la mejor alternativa era utilizar la librería Retrofit2 por su facilidad de aprendizaje, uso y rendimiento en comparación con el resto de alternativas. Esta librería proporciona una capa de alto nivel que, además de facilitar las consultas, nos permite serializar las respuestas del API para recogerlas como objetos Java. Además cuenta con comunicación síncrona y asíncrona entre sus características.

Para el correcto uso de la librería Retrofit2 en nuestra aplicación se han definido los siguientes objetos o clases:

- **ServiceGenerator**, es la clase donde se definen los objetos de la librería Retrofit2 responsables de establecer el rol cliente en la comunicación y de establecer la correcta conexión (canal) con el servidor (ver Figura 4.22). Esta clase ha sido definida siguiendo el patrón *Singleton* para que se realice una única instancia de la misma y de ella se soliciten servicios de comunicación con el API .

```

1 public class ServiceGenerator {
2
3     public final String API_BASE_URL = "https://pesedia.dsic.upv.es/";
4
5     private OkHttpClient.Builder httpClient = new OkHttpClient.Builder
6         ();
7     private Retrofit.Builder builder = new Retrofit.Builder().baseUrl(
8         API_BASE_URL);
9     ...
10
11    public <S> S createService(Class<S> serviceClass) {
12
13        Retrofit retrofit = builder
14            .addConverterFactory(GsonConverterFactory.create())
15            .client(httpClient.build()).build();
16
17        return retrofit.create(serviceClass);
18    }
19 }

```

Figura 4.22: Creación del servicio de consulta con el API.

- **PesediApi**, que se define como una interfaz y que contiene las asociaciones entre las llamadas realizadas en Android y las consultas RESTful efectuadas al servidor. Parte de la definición de nuestro fichero puede observarse en la Figura 4.23.
- **XResponse**, donde la X corresponde a los objetos obtenidos de las consultas realizadas (usuarios, contenido, archivos, etc.). Estas clases son definidas con decoradores (@) para serializar la información (JSON) obtenida de las consultas en objetos de Java. Un ejemplo de la definición de un objeto para su serialización con Retrofit2 puede observarse en la Figura 4.24.

Con los objetos definidos, pueden llevarse a cabo las consultas necesarias desde cualquier actividad o punto de la aplicación donde se requiera mostrar o enviar información a la red social. Para la solicitud de información tan solo deberán ejecutarse las siguientes líneas de código (Figura 4.25) para una comunicación asíncrona:

En el caso del envío de contenido a la red social se efectúa mediante comunicación síncrona. Además, como se habrá podido observar en los diferentes fragmentos de código que se ha mostrado el tipo de autenticación que se realiza por el usuario es mediante *token*.

4.5.2. Gestión de la información

Con el fin de mejorar la navegabilidad en la aplicación móvil, se ha decidido almacenar pequeñas cantidades de información provenientes de las consultas a la red social para así

```

1 public interface ElggApi {
2
3     String PROTOCOL = "rest";
4     String RET_FORMAT = "json";
5
6     /* LOGIN */
7     @POST("services/api/" + PROTOCOL + "/" + RET_FORMAT + "&method=
8         auth.gettoken")
9     Call<MyResponse<String>> doLogin(@Query("username") String name,
10         @Query("password") String pass);
11
12     /* LIST RIVER CONTENT */
13     @GET("services/api/" + PROTOCOL + "/" + RET_FORMAT + "&method=
14         list.river")
15     Call<MyResponse<List<RiverItemResponse>>> getRiverItems(@QueryMap
16         Map<String, String> options, @Query("auth_token") String token)
17         ;
18     ...
19 }

```

Figura 4.23: Vinculación de funciones Java con consultas REST.

```

1 public class RiverItemResponse {
2
3     // AUTHOR
4     @SerializedName("author")
5     private UserResponse author;
6
7     // TYPE
8     @SerializedName("object_type")
9     private String objectType;
10
11     // ACTION
12     @SerializedName("action_type")
13     private String actionType;
14
15     // PRIVACY LEVEL
16     @SerializedName("access")
17     private String access;
18     ...
19 }

```

Figura 4.24: Ejemplo de serialización de un objeto (*RiverItemResponse*).

disminuir el número de consultas realizadas al API de Pesedia, eliminando esperas innecesarias y la redundancia de consultas e información.

Android nos proporciona diferentes opciones para el almacenamiento de la información en las aplicaciones. Para decidir qué opciones de almacenamiento se ajustaban mejor a los requerimiento de nuestra aplicación se realizó un estudio de las diferentes alternativas (*Shared Preferences*, *Internal Storage*, *External Storage*, *SQLite Databases* y *Network Connection*). Finalmente, la alternativa o alternativas elegidas para la gestión de la información en la

```

1 | PesediaApi client = ServiceGenerator.createService(PesediaApi.class);
2 | Call<<List<RiverItemResponse>> call = client.getRiverItems(options, token);
3 |
4 | call.enqueue(new Callback<<List<RiverItemResponse>>>()
5 | {
6 |     @Override
7 |     public void onResponse(Call<List<RiverItemResponse>> call, Response<
8 |         List<RiverItemResponse>> response)
9 |     {
10 |         List<RiverItemResponse> riresponse = response.body().getResult();
11 |         ...
12 |     }
13 |     @Override
14 |     public void onFailure(Call<List<RiverItemResponse>> call, Throwable t)
15 |     {
16 |         System.out.println("Error: " + t.toString());
17 |     }
18 | });

```

Figura 4.25: Ejemplo de consulta asíncrona para solicitar información al API de Pesedia.

aplicación se presentan y justifican a continuación.

- *Shared Preferences*: este tipo de almacenamiento es utilizado generalmente para el almacenamiento de las características o preferencias de las aplicaciones u otros pequeños bits de datos. La información almacenada de esta forma se mantiene segura (encriptada) y es accesible desde la aplicación con consultas clave-valor. Entre la información que almacenamos en la aplicación de Pesedia se encuentra la relativa a la del inicio de sesión, almacenando de esta forma la clave de sesión o *token*.
- *SQLite Databases*: por otro lado, este tipo de almacenamiento permite una mejor gestión de datos estructurados y, aunque no es muy escalable, es muy conveniente para aplicaciones pequeñas por su rapidez. Además, la información almacenada se encuentra en un único fichero `.db` de carácter privado mediante el cual se interactúa con funciones para almacenar y recuperar la información. Este tipo de almacenamiento ha sido utilizado para almacenar la información relativa a la red social (usuarios, publicaciones, imágenes, etc.).

Con la solución implementada, se comprueba previamente a la realización de cualquier consulta si dicha información está disponible o no por la aplicación (localmente). De esta forma se mejoran los tiempos de respuesta y la fluidez de la misma proporcionando una mejor experiencia a los usuarios.

4.5.3. Interfaz de usuario

Por último, después de diseñar todos los componentes lógicos de la aplicación móvil y conectarlos entre sí, estos son visualizados por los usuarios a través de las diferentes vistas con las cuales interactúan. Como se habrá podido apreciar del icono de la aplicación (visto en el Apartado 4.5) y como se verá en las vistas que se presentan a continuación, los colores primarios de estilo escogido son el azul y el verde, que acompañaremos en muchas ocasiones

del color blanco (neutral). Disponer de combinaciones atractivas de color le otorga a la aplicación un mejor aspecto.

Entre las vistas diseñadas para la aplicación móvil se encuentra la vista de acceso a la red social (Figura 4.26), donde el usuario debe loguearse como requisito para acceder a la misma. En esta vista podemos observar la marca corporativa de la aplicación, los campos para el acceso y el enlace para efectuar el registro en caso de no disponer de cuenta. En caso de no introducir los valores correctos, la aplicación proporciona *feedback* a los usuarios indicando el motivo.

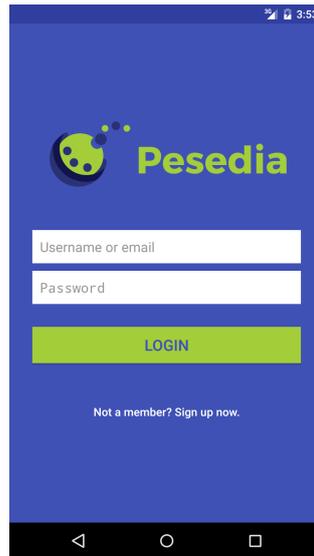
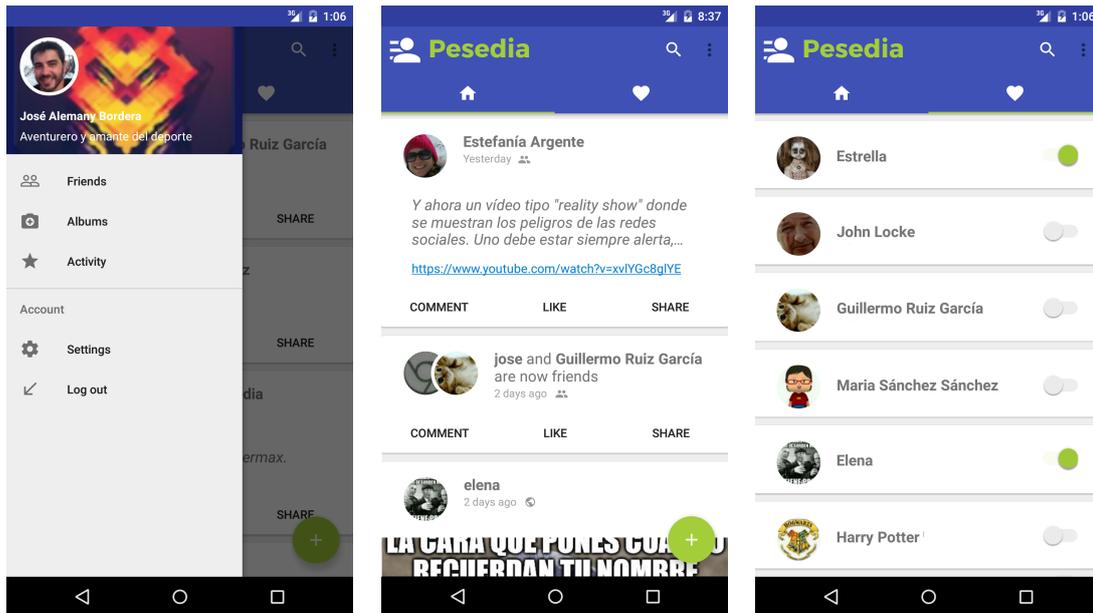


Figura 4.26: Vista de Logueo en Pesedia.

Tras acceder a Pesedia, la aplicación dispone de un conjunto de vistas principales que muestran la red social de acuerdo a los requisitos de funcionalidad que contemplamos para esta primera versión. Las Figuras 4.27a, 4.27b y 4.27c corresponden con: el apartado propio del usuario donde puede acceder a su contenido, personalizar sus configuraciones o salir de su cuenta; la vista principal de la aplicación donde se muestra la actividad accesible para el usuario, y con la que interactúa y crea contenido; y el espacio para listar a los usuarios de la red con el fin de solicitarles peticiones de amistad; respectivamente.

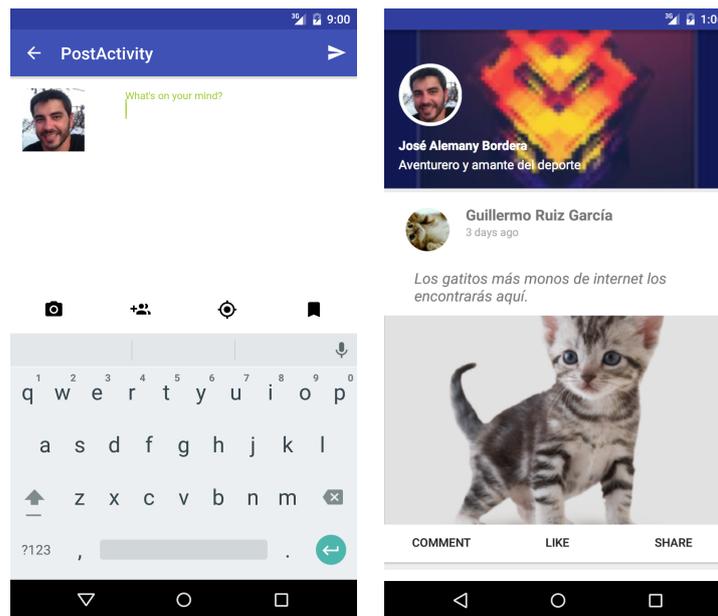
En las vistas principales de la aplicación resaltamos que las publicaciones que se visualizan se han realizado de forma adaptativa a los diferentes tipos de contenidos (imágenes, texto, enlaces, aceptación de amistad, contenidos compartidos, etc.). En cuanto a la vista de amistad, las solicitudes de amistad se envían activando el *switch* del correspondiente usuario.

Por último, presentados las dos vistas restantes correspondientes a la creación de contenido (Figura 4.28a), para la cual se utiliza un botón flotante que aparece solamente en la vista de publicaciones para transitar a dicha vista; y al espacio de usuario (Figura 4.28b), donde puedes visualizar su “Muro”. La vista de creación de publicaciones se ha utilizado también para la creación de comentarios o compartir publicaciones adaptando su funcionalidad a la acción que va a llevarse a cabo.



(a) Apartado de usuario. (b) Vista de publicaciones. (c) Vista de amistad.

Figura 4.27: Vistas principales de los usuarios logueados en Pesedia.



(a) Publicación de noticias. (b) Espacio de usuario.

Figura 4.28: Resto de vistas de Pesedia.

Con el conjunto de vistas que se han expuesto en este apartado queda definida completamente la interfaz de usuario de la aplicación móvil desarrollada para el presente trabajo fin de máster. No se descarta modificar la apariencia de la misma en futuras versiones para adaptarlas al gusto de los usuarios, tras haberlas evaluado por un gran número de usuarios.

CAPÍTULO 5

Evaluación de la plataforma

5.1. Validación de la funcionalidad

Con el fin de evaluar la funcionalidad de la red social y todos sus componentes, los investigadores del Grupo de Tecnología Informática e Inteligencia Artificial (GTI-IA) del DSIC que participan en el proyecto nacional “Privacidad en Entornos Sociales Educativos durante la Infancia y la Adolescencia” (aprox. 20) hemos llevado a cabo usos prolongados de la misma. Utilizando la red social, en primera instancia, para detectar y reparar fallos en la misma, y con la finalidad de diseñar talleres formativos de sensibilización en el uso de las redes sociales.

Con el uso constante de la misma, permitió rediseñar aspectos de la red social que quedaban confusos como la duplicidad de algunas opciones para realizar la misma acción y reparar fallos de plugins implementados por terceros, incluso solucionar conflictos entre diversos plugins.

Con la implementación completada de Pesedia, se han diseñado una serie de talleres para la sensibilización de las redes sociales en cuanto a aspectos de privacidad. La finalidad de los mismos es que los participantes aprendan a valorar la sensibilidad de su información y qué medidas pueden tomar para protegerla de forma que se cree un equilibrio perfecto entre la sociabilidad y la privacidad. Además se pretende extraer información de los usuarios para estudios posteriores de la actividad, personalidad, afinidad e influencia para desarrollar medidas más efectivas de privacidad.

Los talleres se realizarán durante el mes de Julio en el ámbito de la “Escola de Estiu” del politécnico (con el logo de la Figura 5.1), en los cuales participarán 7 grupos con 25 niños aproximadamente por grupo (175 aprox. en total). Las edades de los participantes estarán comprendidas de entre 12 y 14 años. Y la duración de los talleres será de 90 minutos por sesión, habiendo un total de 2 sesiones.

La infraestructura utilizada para la impartición de los talleres es la de los laboratorios del DSIC, en concreto, se dispondrá del laboratorio 1, 2 y 3 para la impartición de los mismos. Cada laboratorio cuenta con los recursos necesarios (escritorios, sillas y ordenadores con acceso a Internet) para un máximo de 32 individuos. Para proporcionar acceso a los ordenadores del laboratorio se han creado las cuentas necesarias, que serán asignadas a los participantes.

En cuanto a la red social, se han destinado los recursos suficientes para la impartición



Figura 5.1: Logo del taller.

de los talleres; se han asignado cuotas de almacenamiento para los usuarios; y se pretende monitorizar la cantidad de recursos utilizados por usuario para asegurar la continuidad del servicio.

Los contenidos de las sesiones de los talleres se han diseñado de acuerdo a que durante la primera sesión, los participantes actuarán libremente por la red motivados por retos referentes a su actividad en la red, en los cuales obtendrán puntos, sin centrarnos en aspectos como la privacidad; y durante la segunda sesión, se propondrán retos más complejos en los cuales los participantes del taller serán penalizados (en puntos) si no son capaces de controlar el alcance de su información. La motivación de los participantes será la de ganar la mayor cantidad de puntos para obtener así un premio al finalizar el taller.

Además del acceso a la red social durante la realización de los talleres, esta estará disponible también durante todo el mes de Julio para que sea utilizada por parte de los participantes, los cuales se registrarán en la realización de la primera sesión del taller. Para incentivar su uso, los monitores publicarán las imágenes de las actividades realizadas en la “Escola de Estiu”. Añadir que se cumple con la ley de protección de datos, ya que se ha dado de alta (previamente) en el Registro General de Protección de Datos el fichero que contiene los datos para el funcionamiento de la red. Además, se cuenta con las autorizaciones necesarias (al ser menores) para la participación en los talleres.

La validación de la funcionalidad de la red social se ha obtenido: a pequeña escala por el propio grupo de investigación GTI-IA mediante informes personales de los mismos integrantes; y a gran escala se pretende obtener de los resultados de la realización de un cuestionario, realizado por los participantes del taller, que mide su grado de satisfacción con la red social.

CAPÍTULO 6

Conclusiones

6.1. Conclusiones

Se espera que este trabajo fin de máster tenga una relevancia social en el ámbito de la confianza digital y la privacidad en redes sociales, ayudando a los usuarios de las mismas a ser más conscientes de los riesgos a los que se exponen y a saber aprovechar mejor las ventajas que nos ofrecen.

Dada la laboriosa y compleja tarea de implementar una red social con todas las propiedades de la misma, y además añadir medidas que proporcionen a sus usuarios las capacidades necesarias para gestionar su privacidad, el presente trabajo es un *hito* importante alcanzado para el estudio y aplicación avanzada de medidas de privacidad. Siendo capaces de desarrollar una plataforma completa de red social con todos los componentes necesarios (servidores, lógica de la red social e interfaces) y una aplicación móvil, que mejora su uso y capacidad de explotación para la recogida de datos reales. Aún siendo una aproximación al desarrollo de medidas de privacidad, se han abordado algunas de las más importantes, como los selectores de audiencia para la información que se almacena en la misma, capacidad para listas de acceso, registros de actividad y visualización del alcance de un post; cumpliendo así con los objetivos marcados en la Sección 1.2 del Capítulo 1.

Al principio de realizar el proyecto se desconocían los componentes y tecnologías necesarias para la implementación de una red social (como Elgg, PHP, Javascript, jQuery, HTML5, CSS, Cron, MySQL, Apache, Android, Retrofit2 y Git) y las herramientas posibles para mejorar el control de la información, pero el estudio y aprendizaje realizado en el presente trabajo fin de máster nos ha permitido adquirir los conocimientos necesarios e implementar nuestra propia red social con los elementos indispensables para la gestión de la privacidad. El estudio de las medidas de privacidad existentes nos han permitido, además, detectar las carencias que ofrecen las mismas en cuanto a que el usuario debe ser un experto, en la mayoría de ocasiones, para poder utilizarlas; y, aún así, no disponen de un control total.

Entre las posible aplicaciones de la red social implementada tendríamos las relacionadas con la enseñanza, en las cuales se podrían estudiar diferentes aspectos del aprendizaje y observar si las conductas se modifican o no como pretende este trabajo [12] pero con la posibilidad de realizar el aprendizaje sobre una red social real y con acceso a la información a largo plazo. Otra posible aplicación sería la relacionada con la investigación, en la cual la red social actuaría como el medio (al cual tenemos acceso completamente) para validar propuestas y algoritmos con actividad real de los usuarios.

Como resultado final del proyecto, se ha logrado crear una red social que ofrece medidas de privacidad a sus usuarios para que sean capaces de mejorar el alcance de su información, en la cual tenemos el control total para estudiar futuras medidas más avanzadas e incluirlas en la misma.

6.2. Trabajo futuro

Como beneficiario de una beca predoctoral (FPI) para la realización del doctorado en “Investigación de medidas de privacidad en redes sociales” y la colaboración con el proyecto nacional “Privacidad en Entornos Sociales Educativos durante la Infancia y la Adolescencia”, que enmarcan la temática de este trabajo fin de máster, se pretenden alcanzar en un futuro próximo los siguientes objetivos en privacidad.

- Desarrollar algoritmos que permitan determinar los rasgos de personalidad de un usuario a partir de la información recuperada de la red social[1, 2].
- Diseñar y desarrollar algoritmos de clasificación de usuarios en función de la afinidad entre ellos[4].
- Hacer uso de técnicas de recomendación que permitan sugerir patrones específicos de privacidad a los usuarios.
- Desarrollar redes de consenso para el establecimiento de un valor de reputación de usuarios[11].
- Hacer uso de medidas estructurales de redes sociales para determinar la importancia, visibilidad e influencia de un usuario en la red, lo cual permitirá establecer el riesgo de difusión de la información al compartirla con un usuario[8, 11, 9].
- Establecer una medida personalizada y multidimensional de la privacidad que no sólo utilice aspectos técnicos de la privacidad sino también su dimensión social.

En cuanto a la interfaces, tanto web como móvil, se pretende mejorar su apariencia para que resulte más atractiva a los usuarios simplificando las vistas y las opciones disponibles. De esta forma facilitaremos el objetivo final del mismo, explotar y comercializar la plataforma de la red social como un nuevo producto tecnológico.

Bibliografía

- [1] ADALI, S., AND GOLBECK, J. Predicting personality with social behavior. In *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)* (2012), IEEE Computer Society, pp. 302–309.
- [2] BACHRACH, Y., KOSINSKI, M., GRAEPEL, T., KOHLI, P., AND STILLWELL, D. Personality and patterns of facebook usage. In *Proceedings of the 4th Annual ACM Web Science Conference* (2012), ACM, pp. 24–32.
- [3] FOGUES, R., SUCH, J. M., ESPINOSA, A., AND GARCIA-FORNES, A. Open challenges in relationship-based privacy mechanisms for social network services. *International Journal of Human-Computer Interaction* 31, 5 (2015), 350–370.
- [4] FOGUÉS, R. L., SUCH, J. M., ESPINOSA, A., AND GARCIA-FORNES, A. Bff: A tool for eliciting tie strength and user communities in social networking services. *Information Systems Frontiers* 16, 2 (2014), 225–237.
- [5] GARAIZAR, P., AND REIPS, U.-D. Social lab: crea tu propio laboratorio de redes sociales. <http://sociallab.es/index.es.html>.
- [6] GRUPO DE TECNOLOGÍA INFORMÁTICA E INTELIGENCIA ARTIFICIAL (GTI-IA). *Privacidad en Entornos Sociales Educativos durante la Infancia y la Adolescencia* (2014).
- [7] IAB-SPAIN. Estudio anual de redes sociales. http://www.iabspain.net/wp-content/uploads/downloads/2016/04/IAB_EstudioRedesSociales_2016_VCorta.pdf, Abril 2016.
- [8] LESKOVEC, J., ADAMIC, L. A., AND HUBERMAN, B. A. The dynamics of viral marketing. *ACM Transactions on the Web (TWEB)* 1, 1 (2007), 5.
- [9] LIU, K., AND TERZI, E. A framework for computing the privacy scores of users in online social networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 5, 1 (2010), 6.
- [10] OTTMAN, B., OTTMAN, J., AND HARDING, M. Minds. <https://www.minds.com/>.
- [11] REBOLLO PEDRUELO, M., DEL VAL NOGUERA, E., CARRASCOSA CASAMAYOR, C., PALOMARES CHUST, A., AND PEDROCHE SÁNCHEZ, F. Consensus over multiplex network to calculate user influence in social networks. In *International Journal of Complex Systems in Science* (2013), vol. 3, Interlude, pp. 71–75.

- [12] VANDERHOVEN, E. Educating teens about the risks on social network sites. an intervention study in secondary education/enseñar a los adolescentes los riesgos de las redes sociales: Una propuesta de intervención en secundaria. *Comunicar* 22, 43 (2014), 123.
- [13] ZUCKERBERG, M., MOSKOVITZ, D., SAVERIN, E., MCCOLLUM, A., AND HUG, C. Facebook. <https://www.facebook.com/>.
- [14] ZUCKERBERG, M., MOSKOVITZ, D., SAVERIN, E., MCCOLLUM, A., AND HUG, C. Facebook help center. <https://www.facebook.com/help/>.