Universitat Autònoma de Barcelona

Dipòsit digital de documents de la UAB

This is the **published version** of the bachelor thesis:

Martinez Fernandez, David; Pons, Jordi, dir. Implementación de funcionalidades y mejoras en la red social de la "Xarxa Alumni" del GEI. 2023. (Enginyeria Informàtica)

This version is available at https://ddd.uab.cat/record/280695

under the terms of the license

Implementación de funcionalidades y mejoras en la red social de la "Xarxa Alumni" del GEI

David Martínez Fernández

18/06/2023

Resum– Tras la creación de un prototipo inicial funcional de una red social dedicada al contacto entre los alumni del Grado de Ingeniería Informática de la UAB, los profesores del grado y otras figuras institucionales relacionadas, se encuentra la necesidad de mejorar y optimizar diversas partes del sistema e introducir nuevas funcionalidades para mejorar la experiencia del usuario. Este trabajo se centra en el desarrollo de una serie de mejoras para cubrir esta necesidad. Para ello, se ha utilizado una metodología en espiral durante todo el desarrollo del trabajo, ya que permite un feedback directo del cliente, en este caso la coordinación del grado. Se han utilizado tecnologías convencionales de desarrollo web para construir el proyecto. Se espera que las mejoras introducidas sean útiles para los usuarios y añadan valor a la red social.

Paraules clau– Alumni, Experiencia de usuario, Grado en Ingeniería Informática, HTML, JS, PHP, Red social, SQL, Tailwind.

Abstract– After the development of an initial functional prototype of a social network dedicated to the contact between the GEI alumni from de UAB, the teachers of the grade and others institutional figures that are related, the necessity of improve and optimize various parts of the system and introduce new functionalities to have a better user experience it's present. This project it's focused in the development of a number of improvements to cover this necessity. To achieve this, has been used an spiral methodology during all the development of the project because this methodology allows a direct feedback with the client, in this case the grade coordination. Has been used standard web development technologies to build the project. The introduced improvements are expected to be useful for the users and add value to the social network.

Keywords– Alumni, Computer Engineering Degree, HTML, JS, PHP, Social network, SQL, Tailwind, User Experience.

1 Introducción - Contexto del trabajo

Es una realidad que la mayoría de alumnos y alumnas de las universidades mantienen una relación con la institución ya sea más o menos directa una vez se graduan y empiezan sus carreras laborales. Algunos de ellos simplemente mantienen un contacto esporádico, en Linkedin, Instagram o Twitter, para estar al tanto de las novedades y seguir recordando aquella época en la que ofrecieron esfuerzo y dedicación al estudio. Otros, en cambio, forjan una relación fuerte con docentes y personalidades allegadas con la entidad educativa, llegando a colaborar en eventos, trabajando como profesores asociados o realizando estudios especializados y doctorados bajo su nombre. En cualquiera de los casos, los ex-alumnos, también denominados alumni, de una universidad como la UAB necesitan vías de comunicación para poder mantener un buen contacto. Y es por ello que nacen las redes alumni en el ámbito universitario. Estos sistemas aparecen como solución a la necesidad de crear una vía de comunicación directa entre los docentes de la universidad y los alumni que gustan de seguir teniendo noticias de lo que puede ofrecer una vez la abandonan en calidad de estudiantes.

• E-mail de contacto: david.martinez@autonoma.cat
• Mención realizada: Tecnologies de la Informació
• Trabajo tutoritzado por: Jordi Pons Aróztegui (DEIC)
• Curs 2022/2023

Las redes alumni destacan por su amplia red de contactos en cada uno de los estudios. En nuestro caso, nos centramos en los estudios universitarios relacionados con la ingeniería informática, y es que este proyecto pretende ser la mejora de un primer proyecto de red alumni ya desarrollado por mi compañero Pere Pérez [1] en el semestre anterior. En ese proyecto se creó una herramienta de red social para la Xarxa Alumni para el Grado de Ingeniería Informática (GEI) de la UAB con funcionalidades básicas, y por ello se ha realizado este proyecto, que añade una buena cantidad de mejoras y nuevas funcionalidades que pretenden potenciar el uso del sistema ente los graduados.

Este documento pretende ser una descripción detallada de cuales han sido los objetivos que se han tenido presentes a la hora de realizar el proyecto. Además, se pretende repasar como ha sido el desarrollo de todo el conjunto de novedades añadidas al sistema, describiendo las utilidades de cada nueva funcionalidad, como han sido implementados todos los cambios, qué problemas han surgido durante el desarrollo y cuales han sido las soluciones encontradas a dichos problemas. También se habla sobre la metodología utilizada para realizar el trabajo y los resultados obtenidos una vez finalizado el desarrollo para compararlos con los objetivos propuestos y detallados al inicio de este mismo documento. Finalmente concluiremos el documento con un breve análisis de como se ha desarrollado todo el proyecto en general y de cuales son los posibles siguientes pasos a seguir para mejorar todavía más el sistema.

2 OBJETIVOS

El objetivo principal de este proyecto y por el cual comenzó el desarrollo mi compañero es el de mantener al alumnado que ya se ha graduado conectado entre si y con la institución educativa a través de una plataforma directa y fácil de utilizar y mantener. Además de ese objetivo global que envuelve todo el proyecto en conjunto, el objetivo principal y propio del trabajo desarrollado por el alumno es el de mejorar la plataforma para favorecer la experiencia del usuario y proporcionar una experiencia mas atractiva y satisfactoria, haciendo que de esta manera crezca la popularidad y el uso de la plataforma.

Para poder conseguir esto se requiere del desarrollo de nuevas funcionalidades para cubrir ciertas necesidades, así como de mejoras en el diseño de algunos componentes de la plataforma, ya sean aspectos visuales o de funcionamiento a causa de bugs y problemas de rendimiento. Algunas de estas mejoras fueron detectadas en el desarrollo inicial, pero por limitaciones de tiempo no fueron introducidas en el sistema. A su vez, durante el desarrollo de estas nuevas funcionalidades, se han detectado nuevas mejoras que pueden ser implementadas en un futuro para mejorar todavía mas la plataforma. Como ya se ha mencionado, las mejoras se implementan para suplir ciertas necesidades y carencias de las que sufre la plataforma en el estado inicial. Por este motivo, se ha decidido dividir las mejoras desarrolladas en tres grupos, los cuales abordan una necesidad específica que se pretende cubrir.

- **Feed de la red social**: el feed de la Xarxa Alumni es el corazón del sistema, ya que es el lugar en el que los ususarios pueden interactuar entre ellos de forma directa, conociendo nuevas personas y creando contactos, publicando inquietudes y contestando a publicaciones de otros usuarios, participando en eventos y apuntandose a ofertas y oportunidades tanto laborales como educativas que aparezcán en su feed. Es por ello que una parte importante de las mejoras irán dirigidas al feed principal y los feeds secundarios (feed de perfil y feed de grupos) para mejorar la experiencia de dicha herramienta, ya que es probablemente donde los usuarios inviertan más tiempo. Para ello se pretende desarrollar un sistema de 'Me gusta' (Likes) a los distintos tipos de publicaciones que encontramos en el feed, y un sistema secundario y derivado de los likes que nos permitirá registrar los posts a los que un usuario a dado 'Me gusta' y mostrarlos en su perfil, teniendo así una pestaña de 'Favoritos' personalizada.

  Otra de las mejoras previstas para implementar en el feed es un sistema de archivos adjunto a los posts, el cual permitirá añadir, eliminar y modifcar archivos adjuntos a las publicaciones de los usuarios. Este sistema nos permite adjuntar tanto archivos multimedia (.jpg, .png, .mp4) como otro tipo de archivos como pueden ser archivos comprimidos (.zip, .tar, .rar) o documentos de texto (.pdf, .docx).

- **Interacción entre los usuarios**: otro de los aspectos más importantes de la plataforma es la interacción de los usuarios fuera del feed principal que, aunque es el lugar más concurrido en cuanto a publicaciones, no es el único sitio por el que los usuarios pueden compartir información y darse a conocer. Es por ello que también se implementarán una serie de funcionalidades dedicadas a mejorar y potenciar esta interacción entre los usuarios y también entre la propia plataforma, para lograr una comunicación más eficiente y fluída por ambas partes.

  Las mejoras en relación a este apartado son la creación de un nuevo tipo de publicación orientada a la organización de eventos en el mundo real, que tiene como objetivo facilitar a los asistentes y organizadores del evento la comunicación entre ambas partes, el desarrollo de un sistema de notificaciones que permite a la plataforma informar en tiempo real al usuario a través de un correo electrónico de diversas situaciones que estan ocurriendo dentro de la plataforma que pueden ser del interés del usuario, la posibilidad de realizar una búsqueda avanzada de usuarios para poder filtrar mejor los contactos que pueden interesar a cada individuo y, por último, la incorporación de un nuevo rol dedicado a los profesores de cada materia y grado.

- **Usabilidad y diseño de la aplicación**: en este último apartado incluimos todos aquellos cambios que se harán una vez se realice una revisión de la herramienta

ya desarrollada. Estos cambios consisitrán en mejoras, tanto pequeñas como grandes, en algunas de las funcionalidades y componentes existentes. Estas mejoras podrán ser estéticas para favorecer la experiencia sensorial de los usuarios, o bien soluciones a funcionalidades que no funcionaban correctamente al cien por cien para poder mejorar la fluidez de la plataforma.

Además de estos objetivos propios del desarrollo de estas nuevas funcionalidades, tenemos una serie de requisitos que vienen derivados del inicio del proyecto y que siguen estando presentes:

- Escalabilidad de la aplicación para poder implementarla en otros grados en un futuro.
- Mantener un diseño responsive para permitir a los usuarios acceder a la aplicación desde cualquier dispositivo.
- Desarrollar una interfaz intuitiva en las nuevas funcionalidades para facilitar lo máximo posible el uso de la aplicación a los usuarios.
- Asegurar la confidencialidad de los datos de los usuarios, así como la seguridad de que dichos datos se almacenan de forma segura y no se utilizan para fines ajenos a los acordados con los usuarios.

3 Estado del arte

En esta sección no indagaremos en otras herramientas de redes alumni de otras instituciones porque el punto de partida de este proyecto es la herramienta ya existente desarrollada por Pere Pérez. En lugar de eso, haremos un breve repaso a los 2 TFGs relacionados con el trabajo. El primero de ellos es la parte de red social ya desarrollada, la cual tiene varias funcionalidades básicas como la posibilidad de crear posts y comentarios que se publican en un feed principal, la creación de grupos, públicos o privados, para que los usuarios puedan también publicar posts relacionados con la temática del grupo (asignaturas, eventos, etc.). Encontramos también un perfil propio, el cual podemos personalizar con una imagen de perfil, una descripción y algunos datos personales como la población, empresa en la que trabajas, redes sociales y etiquetas que estén relacionadas contigo (Backend, Frontend, Xarxes, etc.), y una sección en la que podemos definir la privacidad, que determina quienes pueden ver nuestros posts y nuestro perfil. La herramienta de red social actual, en la cual se trabajará en este proyecto, es una herramienta básica pero que cumple con los objetivos principales de una red social convencional.

El segundo TFG relacionado con la Xarxa Alumni consiste en una herramienta de administración que ofrece un panel para poder consultar estadísticas, administrar la base de datos y también la red social. Esta herramienta aún está en desarrollo y se espera que para principios de 2024 ya este operativa y lista para su uso.

4 Desarrollo

El objetivo de este apartado es el de documentar en profundidad y detalladamente el proceso de desarrollo del trabajo realizado durante los meses de producción. La sección se divide en las distintas funcionalidades implementadas, empezando por la fase de analisis de cada una de ellas, pasando por el desarrollo en sí y acabando con los problemas y soluciones encontradas a dichos problemas.

Antes de empezar a detallar todas las funcionalidades, es preciso realizar un repaso de todas las tecnólogias utilizadas para el desarrollo del proyecto para poder proporcionar algo de contexto cuando se entre en detalle a hablar del desarrollo de las mejoras. Primero hablaremos sobre la base de datos, la cual está construida en base a MariaDB [2], un tipo de base de datos relacional de tipo SQL que se deriva de las bases de datos MySQL convencionales. La parte a destacar de esta base de datos sobre MySQL es que es de tipo GPL (General Public License)[3] al contrario que MySQL. Dicha base de datos está desarrollada para satisfacer las necesidades de la plataforma que compone la red social de la aplicación, pero también está pensada para proporcionar información a una segunda parte dedicada a labores de administración y la cual está siendo desarrollada por otro compañero al mismo tiempo. Dicha base de datos esta compuesta de tablas dedicadas a guardar información sobre los usuarios, como pueden ser datos básicos (NIU, nombre, apellidos, email, ...), datos enfocados a las relaciones académicas (nota media, grado, mención escogida, ...) y datos relevantes para cuestiones estadísticas (año de matriculación, empresa en la que trabaja el usuario, población, ...). También se guarda toda la información necesaria para la privacidad de los perfiles, los roles existentes dentro de la plataforma, publicaciones y comentarios, y nuevas tablas creadas por necesidad al desarrollar las nuevas funciones como las tablas de likes, la tabla de eventos o las relacionadas con el sistema de notificaciones.

En relación a las tecnologías utilizadas, se han seguido usando las mismas herramientas y lenguajes que en el anterior trabajo. Para poder trabajar desde el ordenador del alumno se ha utilizado XAMPP[4], que nos proporciona un servidor de Apache, un gestor de base de datos basado en MariaDB y soporte para PHP. De esta forma, el desarrollo en las fases primerizas de las funcionalidades se han realizado desde el ordenador personal del alumno respaldándose en XAMPP y una vez las funcionalidades han ido llegando a un estado de madurez y solidez funcional correcto, se han subido al servidor de producción localizado en los servidores propios de la universidad. Gracias a XAMPP el desarrollo inicial de las funcionalidades ha sido muy sencillo.

En lo que respecta a lenguajes de programación, se han utilizado HTML nativo para el frontend y PHP nativo para el backend, creando también muchas de las vistas del frontend del lado del servidor (ya que PHP permite renderizar HTML desde el backend). La arquitectura se basa en gran medida en MVC[5], teniendo un controlador central encargado de redirigir las peticiones de los usuarios hacia

las funciones del backend donde se encuentran los modelos correspondientes, y devolviendo respuestas, muchas veces en formato HTML, hacia el cliente modificando así las vistas. Se ha utilizado también JS nativo junto con JQuery para proporcionar asincronía en gran parte de los eventos de la plataforma utilizando AJAX, y por último se ha utilizado un framework de CSS denominado Tailwind[6], que transporta el estilo de los componentes HTML hacia el propio HTML gracias a su sistema de estilos con clases predefinidas, ofreciendo así una gran facilidad y menor carga de archivos CSS a la hora de aplicar estilos.

Ahora que ya hemos repasado las tecnologías generales utilizadas en el conjunto del desarrollo, es hora de presentar las distintas funcionalidades implementadas.

4.1 Likes

Los likes nacen de la necesidad de la plataforma de saber qué contenidos son más acordes con los gustos de los usuarios. De esta manera, si las publicaciones de colaboraciones y ofertas son mejor recibidas que las publicaciones de formación, se crearán más posts en relación a lo que más triunfa. Es por ello que un sistema de likes en las publicaciones y comentarios de los feeds (tanto en el feed principal como en los feeds secundarios de grupos y perfiles de usuario) es una buena herramienta para medir la popularidad de los distintos tipos de posts.

Para implementar este sistema se ha añadido un nuevo botón en los posts en forma de corazón, junto a los botones de comentarios, edición y eliminación (estos dos últimos solo se muestran si somos autores del post o administradores). Este bóton actua de forma que, una vez pulsado por primera vez, se rellena con un color rojizo indicando que dicho post está marcado como 'Me gusta' por el usuario. A su vez, si volvemos a pulsar el bóton, el color rojizo del interior desaparecerá y dejará de estar marcado. Este comportamiento lo podemos ver en las figuras 1 y 2. Además, en dichas figuras también podemos ver un indicador númerico al lado derecho del botón. Este indicador nos muestra cuantos likes generales ha obtenido cada publicación, sabiendo de esta manera si es un post más o menos popular entre los usuarios. Como podemos observar al pulsar el like en la figura 2, el número aumenta en 1.

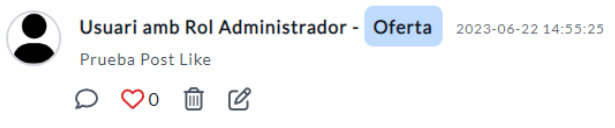

Fig. 1: Post sin like

Fig. 2: Post con like

Otra de las utilidades del sistema de likes es la posibilidad de que los usuarios accedan de forma sencilla a los posts que han marcado como 'Me gusta'. Esto se ha implementado realizando una remodelación del perfil del usuario, de la cual hablaremos en detalle en el último subapartado de esta misma sección. En el perfil, como podemos observar en la figura C.2, vemos un botón con la etiqueta de 'M'agrada', el cual si pulsamos, nos muestra los posts y comentarios a los cuáles hemos dado like, creando de esta forma una pestaña de 'Favoritos' personalizada para cada usuario.

Entrando en profundiad en el desarrollo de la funcionalidad, se han creado 3 tablas que gestionan los likes. Existe una tabla que almacena los likes de las publicaciones, guardando simplemente el NIU del usuario y el 'id' del post al que se le ha dado a like, y 2 tablas para los likes de los comentarios. Una de las tablas guarda los likes de los comentarios a posts convencionales de igual forma (NIU, id del comentario) y la otra guarda los likes de comentarios a eventos, los cuales son un tipo de post específico con una tabla en la base de datos propia. De esta forma logramos llevar un registro de los likes muy simple pero completamente funcional para lo que se busca. En cuanto al funcionamiento interno, se han creado funciones asíncronas con JQuery para poder cambiar el botón y el número de likes sin necesidad de recargar la página, y se han creado funciones de PHP para poder seleccionar los posts y comentarios con likes de cada usuario y poderlos mostrar en el perfil. Dichos posts se cogen de la base de datos gracias al 'id' almacenado en las tablas de likes y realizando un *inner join* con el 'id' en las tablas que almacenan toda la información de los posts.

4.2 Sistema de ficheros

Otra de las grandes carencias de las que sufrían los posts del feed en la red social era la imposibilidad de adjuntar ficheros a los mismos. Esto ocasionaba limitaciones a la hora de publicar la información, ya que no era posible añadir un poster informativo, una imagen de un horario, un documento que hable de algun tema de interés general, etc. Es por ello que la necesidad de implementar un sistema de ficheros adjunto a los posts era predominante y una de las funcionalidades mas prioritarias en el desarrollo del proyecto. A causa de esto y de la complejidad de la implementación ha sido la funcionlidad más costosa, tanto en tiempo como en dificultad, de todo el trabajo.

En la fase de análisis de la funcionalidad, se concluyó que la mejor forma de tratar el sistema de ficheros era con una organización de carpetas jerárquica, que se guardara en la parte del servidor y que dividiera los archivos en relación al tipo de publicación al que están adjuntos. En un inicio, solamente existían dos carpetas, una para los archivos adjuntos a publicaciones convencionales y otra para los comentarios en dichas publicaciones. Al implementar los eventos, estas carpetas han aumentado en 4, añadiendo las carpetas que almacenan los archivos adjuntos a eventos y

una que almacena los archivos adjuntos a comentarios de eventos (los cuales son tratados de forma distinta al resto de comentarios). De esta forma la jerarquía de carpetas quedaría tal y como vemos en la figura 3. Como podemos

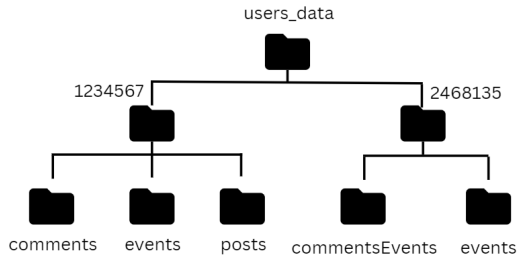

Fig. 3: Estructura de carpetas

observar, la jerarquía se estructura con una carpeta general que almacena todos los archivos existentes, y luego dentro de esta carpeta genérica se dividen por el NIU de cada usuario. Además, no existen las carpetas de forma estática, sino que si un usuario no ha adjuntado ningún archivo a, por ejemplo, un comentario de un post, dicha carpeta no existe hasta que el usuario lo haga. Es por ello que en la figura 1 vemos como el usuario con NIU 2468135 no tiene una carpeta para guardar archivos adjuntos a posts convencionales, pero el usuario con NIU 1234567 si que la tiene. Esto genera un sistema de ficheros que responde a las necesidades del usuario, ya que si no tiene archivos adjuntos a un post, dicha carpeta no se crea y no ocupa espacio en el servidor.

Pasando al funcionamiento e implementación del sistema, como podemos ver en la figura A.1, en el modal de creación y modificación de posts, eventos y comentarios, ahora hay dos botones nuevos que sirven para adjuntar los archivos. El primer bóton sirve para adjuntar archivos multimedia, tales como fotos o vídeos, y el segundo botón sirve para adjuntar archivos de otros tipos, como PDFs o archivos comprimidos. Se realiza esta división entre archivos con previsualización y archivos sin previsualización. Esta diferencia también es visible en el momento de crear la publicación con archivo adjunto, como vemos en las figuras A.2 y A.3. En la primera de las figuras comentadas vemos como se ha adjuntado un archivo .png, el cual podemos ver en la propia publicación. En cambio, en la otra imagen observamos que ha aparecido un nuevo botón en la parte inferior con el símbolo de una carpeta, el cual si pulsamos nos descargará el archivo (.pdf, .zip, .docx,...) adjunto a la publicación. Existe una limitación a la hora de adjuntar archivos, y es que no pueden superar los 4MB de tamaño por razones de seguridad, así evitaremos almacenar archivos muy pesados en el servidor.

A la hora de adjuntar un archivo, este se guarda en la carpeta correspondiente en función de en qué tipo de publicación se adjunte. Si la carpeta no existe previamente, se crea una nueva. El guardado del archivo se realiza antes de crear el post, para poder obtener una ruta que almacenar en la base de datos en el registro referente a la nueva publicación. En este momento, el archivo se guarda con un nombre genérico que puede provocar sobreescrituras en el sistema. Por este motivo, una vez creada la publicación, se le cambia el nombre al archivo para obtener un formato único y evitar que se sobreescriban. Dicho formato es *nombreOriginal_tipoPublicación_idPublicación.extensión*. Como ejemplo, si adjuntamos un archivo a un comentario de un evento con el nombre de 'foto.png', y este nuevo comentario tiene id 3, el nombre del archivo será **'foto_commentEvent_3.png'**. Una vez almacenado el archivo en el sistema, en la base de datos hemos añadido un nuevo campo a las tablas de 'publicacions' y 'comentaris' llamado 'file' en el que se guarda la ruta relativa al archivo, para que de esta forma el sistema sea capaz de encontrar el archivo y mostrarlo o adjuntarlo en la publicación. Todo este funcionamiento se realiza mediante AJAX, lo cual ha sido lo más complicado de hacer a causa de que los formularios que guardan archivos se comportan de forma especial a la hora de enviarse, y su funcionamiento por defecto es recargar la página para almacenar el archivo en el sistema correctamente. Gracias al uso de AJAX y a la modifiación de cómo se envían los datos hacia el backend, hemos conseguido almacenar dichos archivos sin necesidad de recargar la página, lo que proporciona una sensación de fluidez y evita la pesadez de recargar cada vez que se crea, modifica o elmina un post con un archivo adjunto.

En el caso de querer modificar el archivo adjunto a una publicación, el sistema comprobará si se quiere eliminar o si se quiere sustituir dicho archivo. En caso de que el archivo quiera ser eliminado, simplemente se eliminará del sistema y el path almacenado en la base de datos cambiará a null. Por el contrario, si se quiere sustituir por otro archivo, se eliminará el archivo antiguo de igual manera, pero la referencia de la base de datos cambiará a la del nuevo archivo, que sigue el mismo proceso que al crear un post, cambiandole el nombre al formato antes comentado. De igual forma, si se elimina el post de forma completa, se realiza la misma acción con respecto al fichero adjunto eliminándolo por completo del sistema.

Dicha funcionalidad funciona correctamente en el servidor de producción, a pesar de un problema inicial que provocaba que no se llamara a la función de guardado de archivos y la referencia guardada en la base de datos fuese errónea. Este error fue solucionado moviendo la función a otro fichero y realizando la llamada desde el controlador de PHP. Otro error en el servidor de producción era que no se podian cambiar los nombres de los archivos al formato correcto ni eliminarlos cuando fuese necesario. Esto se debía a un error por permisos y un error en los paths guardados en la base de datos que no eran correctos en el servidor pero si en local. Este error también ha sido solucionado en las últimas etapas del desarrollo del proyecto.

4.3 Eventos

Pasando ahora a las funcionalidades dedicadas a la interacción entre los usuarios, la más importante de esta sección y problablemente del proyecto en su totalidad, es la incorporación de los eventos. Los eventos nacen de la necesidad de organizar de forma óptima situaciones que suceden en una franja temporal fija en el mundo real, es decir, actividades o eventos que tienen lugar tanto online como en persona y

que pueden ser del interés de los usuarios de la aplicación. Un ejemplo podría ser una jornada de puertas abiertas de l'Escola d'Enginyeria en la que se necesita la colaboración de algunos alumni del grado de ingeniería informática para realizar unas charlas para promocionar los estudios. Dichas charlas se realizan un día y hora determinados dentro de la facultad. Estos eventos de la aplicación permiten que los usuarios interesados se apunten a dicho evento, el cual ha sido organizado por otro usuario (al que podemos denominar como el administrador del evento en cuestión) que ha indicado las fechas y horas en las que el evento se va a celebrar. Además, el propio post de evento permite al resto de usuarios consultar una lista con los usuarios apuntados. De esta forma podemos 'organizar' un evento dentro de la aplicación de forma sencilla y coordinada entre todas las partes. Esta es la premisa que se ha seguido a la hora del desarrollo

En la fase de análisis, se concluyó que un evento necesitaría de la posibilidad de determinar fecha y hora tanto de inicio como de final, un botón para que los usuarios puedan apuntarse o desapuntarse a dicho evento, un indicador para mostrar las horas del evento y si está en proceso o no, y un botón para mostrar la lista de usuarios apuntados. Además, el post de evento debe de determinar el estado en el que se encuentra a partir de la fecha real y las fechas de su configuración para poder determinar el estado del evento. Estos estados pueden ser 'Per començar', 'En procés' o 'Finalitzat'.

Ya en la fase de desarrollo, lo primero que se hizo fue modificar los modales de creación y modificación de posts, así como añadir un nuevo tipo de publicación. Como podemos ver en la figura B.1, si se selecciona el tipo de publicación como 'Esdeveniment', nos aparecerá la opción de añadir las fechas y horas inciales y finales del evento. Por lo demás, un evento se comporta exactamente igual que un post convencional, solamente que en lugar de dar a like al post, nos apuntamos al evento (o nos desapuntamos si ya estabamos apuntados). Como podemos ver en la figura B.2, un evento es visualmente muy parecido a un post, pero vemos que el botón de like ahora es un '+' o un '-' en relación a si estamos o no apuntados. En la parte superior se observa el botón de un reloj, que muestra el horario del evento y, en caso de haber comenzado, se muestra un mensaje 'En procés', si no ha comenzado aún se muestra un mensaje 'Per començar', y si el evento ya ha finalizado se muestra el mensaje 'Finalitzat' como vemos en las figuras 4, 5 y 6. Por último, si se pulsa el botón situado en el lado izquierdo del reloj, se abre la lista de participantes tal y como se ve en la figura B.3. En dicha lista se observan los nombres de los inscritos en el evento y un botón que permite enviar una solicitud de amistad. Además, en el perfil de cada usuario existe un botón que permite visualizar los eventos a los que el propio usuario está inscrito sin necesidad de buscarlos en el feed principal. Esta funcionalidad es muy similar a la de 'Favoritos' con los likes pero con los eventos.

En cuanto a la implementación, todo funciona con funciones asíncronas muy similares a las que se utilizan con el resto de posts y comentarios. Incluso la función que

Fig. 4: Evento por empezar

Fig. 5: Evento en proceso

Fig. 6: Evento finalizado

comprueba si un evento ha acabado lo hace de forma totalmente asíncrona sin necesidad de recargar la página. Dicha función se basa en un intervalo de 10 segundos que comprueba la fecha real con la fecha inicial y final del evento para saber si está en proceso o si ha finalizado. Una vez que finaliza el evento, se detiene el intervalo. En lo que respecta a la base de datos, al ser un tipo de post muy concreto, ha sido necesario crear una nueva tabla que, además de la información de los posts convencionales, también almacena las fechas y horas de los eventos. Es por ello que también ha sido necesario crear una nueva tabla de comentarios para no provocar interferencias. Por último, se ha creado también una tabla que relaciona cada evento con sus participantes.

4.4 Sistema de notificaciones

La idea del sistema de notificaciones de la aplicación es fomentar el uso de la misma entre los usuarios informando en tiempo real de algunas situaciones que pueden ser del interés del individuo. De esta forma el usuario puede ver la notificación en su correo electrónico y entrar e interactuar con la plataforma, aumentando así su utilización.

Para ello se ha utilizado un objeto ya desarrollado llamado PHPMailer que implementa todas las funcionalidades necesarias para poder implementar el sistema de notificaciones. También se ha creado un objeto propio llamado *Mail Manager* que integra el PHPMailer y se encarga de hacer de puente entre el sistema y el mailer. Dicho manager debe de configurar el servidor SMTP, necesario para poder enviar las notificaciones a través del correo electrónico, el email y la contraseña del propio email que es el que utiliza el sistema para enviar las notificaciones. De este modo, cuando suceden cualquiera de los siguientes eventos dentro de la aplicación:

- Recibir solicitud de amistad.
- Recibir solicitud de unión a un grupo.
- Recibir un baneo de tu usuario.
- Recibir un desbaneo de tu usuario.

El sistema envía en el mismo momento una notificación al email del usuario informando del evento sucedido.

El hecho de recibir notificaciones se puede configurar

en el perfil de cada usuario, como podemos ver en la figura C.2. En dicha imagen observamos un botón de una campana acompañado de un texto. Este botón nos permite habilitar el hecho de recibir notificaciones o, por el contrario, desactivarlo si no se desea recibir dichos mensajes. Además, para los usuarios administradores, existe la opción de configurar el sistema de notificaciones de forma global, activando o desactivando la posibilidad de recibir notificaciones para toda la plataforma, modificando la dirección de email que utiliza el sistema para enviar las notificaciones junto con la contraseña de la cuenta de email indicada, y el servidor SMTP que usa la cuenta de correo. Esta configuración se almacena en un archivo de configuración interno que el sistema utiliza para configurar el mailer.

Actualmente, el sistema de notificaciones solamente funciona de forma local y con un email bajo el servidor SMTP de Gmail. Se está intentando modificar la dirección de email y el servidor para conseguir que el sistema funcione desde los servidores de producción pasando por los firewalls configurados en la universidad. En este momento, la coordinación del grado ha conseguido el correo institucional 'alumni.gei@uab.cat' para utilización exclusiva de la aplicación y se están realizando pruebas en la configuración del mailer.

4.5 Búsqueda avanzada de usuarios

A continuación hablaremos de las mejoras implementadas en relación a la búsqueda de usuarios en la plataforma. Primero, se ha modificado el directorio general de usuarios para que muestre las poblaciones y empresas indicadas por cada usuario para permitir que el resto de individuos puedan realizar una búsqueda a partir de esta información. Además, se ha implementado una nueva pestaña que permite realizar una búsqueda limpia aplicando filtros como són las etiquetas asociadas de forma personalizada a cada usuario y también por las empresas que nos interesan. Estos filtros pueden tratarse como un AND o un OR, es decir, puedes buscar por más de una etiqueta o empresa y indicar que los usuarios deben de tener asociadas todas las etiquetas seleccionadas, o por el contrario tienen que tener como mínimo una de ellas.

4.6 Rol de profesor

Por último y acabando con las mejoras relacionadas con la interacción entre usuarios, se ha añadido un nuevo rol para los profesores que les permite identificarse como docentes dentro de la plataforma. De esta manera, los alumni podrán comunicarse con ellos de una forma sencilla y serán conscientes de que son profesores de asignaturas o grados que les interesa. Como se puede ver en la figura B.4, tanto en el directorio de alumni como en el directorio de usuarios administradores, se ve una tabla con los usuarios que tienen el rol de profesor.

4.7 Mejoras y optimizaciones generales

Para acabar con la exposición del desarrollo general del proyecto, se pretenden detallar las mejoras y optimizaciones generales realizadas en el conjunto de la plataforma. La mayoría de estas optimizaciones son cambios en la parte del frontend, para hacer más ligera y amigable la interfaz del usuario. En la figura C.1 podemos observar el rediseño del feed, concretamente de las publicaciones, donde se han cambiado los colores para crear una sensación más limpia, y se han unido todos los posts en una única columna para aportar una sensación de solidez.

En cuanto al perfil, el cual podemos observar en la figura C.2, se han añadido los botones que dividen el feed personal en tres secciones. El primer botón muestra las publicaciones realizadas por el propio usuario, el segundo botón muestra los posts con likes del usuario, y el tercer botón muestra los eventos a los que el usuario se ha apuntado. De esta forma logramos generar un feed menos recargado y más específico a la hora de buscar información concreta. También se ha realizado un remodelado del componente que muestra la información del usuario, unificando todas las opciones de edición en un solo botón situado a la derecha del nombre del usuario y que muestra el modal que se ve en la figura 7. Dicho modal divide la edición en tres apartados. El primero de ellos corresponde a información genérica como el email, la población o la empresa en la que trabaja el usuario. El segundo corresponde a las redes sociales y el tercero a la privacidad.

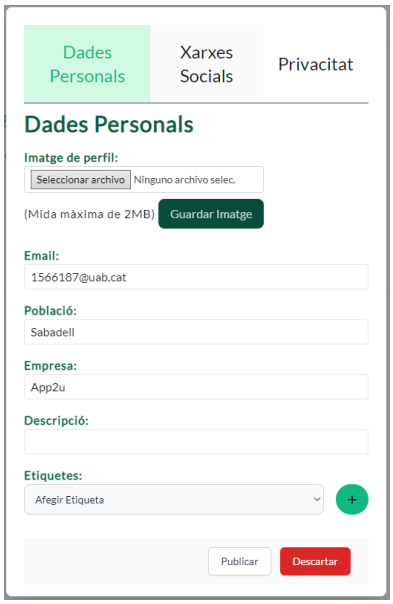

Fig. 7: Modal de edición de perfil

Por último, se han realizado algunas correcciones menores en funcionalidades que no funcionaban del todo bien, como la asincronía en el cambio de roles o algunos botones que no cambiaban al clickarlos.

5 METODOLOGÍA

En este apartado comentaremos la metodología que se ha utilizado a lo largo del proyecto y también analizaremos la planificación confeccionada en un inicio y si se han producido desviaciones a lo largo del desarrollo natural

del trabajo. En este apartado se hará referncia de forma general a las figuras que aparecen en el apéndice D, puesto que són las figuras relacionadas con las planificaciones desarrolladas a lo largo del proyecto.

La metodología de desarrollo utilizada durante todo el proyecto ha sido una metodología en espiral[7] en la cual se pedia feedback del cliente (en este caso el tutor representando a la coordinación del grado) y aplicando el feedback recibido en el producto en etapas donde todavía no era crítico realizar cambios en las funcionalidades implmenetadas. Este feedback se ha recbido generalmente al finalizar cada funcionalidad o en algunos casos al finalizar algunos paquetes de funcionalidades más pequeñas o que formaban un bloque como las mejoras relacionadas con el feed. Una vez recibido el feedback se ha procedido a realizar los cambios pertinentes para adaptar el producto a las demandas del cliente, como puede ser la incorporación del contador de likes o la búsqueda avanzada por tags y empresas. Se ha decantado por utilizar esta metodología a causa de la facilidad con la que se pueden aplicar los cambios que nacen a partir del feedback del cliente, ya que al hacerlo cuando solamente se ha desarrollado un bloque de todas las mejoras permite modificar el código con mucho menos riesgo a romper otras partes del sistema más delicadas.

En cuanto a la planificación inicial que se encuentra en el apéndice D, podemos destacar que se ha seguido de forma correcta hasta la mitad del proyecto, cumpliendo así con el calendario hasta finales de abril. En cuanto al resto de la planificación, se retrasó durante el desarrollo de la funcionalidad de los eventos a causa de una mala estimación del tiempo que se necesitaría a la hora de implementar dicha funcionalidad de forma óptima. Es por esto que se realizó una segunda planificación, que también se encuentra en el apéndice D, que aborda desde finales de mayo hasta el final del proyecto. Esta segunda planificación se ha podido seguir de forma correcta y sin complicaciones.

6 RESULTADOS DEL PROYECTO

Ahora procederemos al análisis de los objetivos y el desarrollo para determinar si se han cumplido dichos objetivos propuestos y detallados al inicio de este mismo documento. De la misma forma también analizaremos si se han cumplido con los requisitos también expuestos en el apartado 2 del documento.

Recordemos que el objetivo principal del proyecto era el de mejorar la plataforma a nivel general para proporcionar una mejor experiencia de usuario y potenciar el uso de la aplicación entre los distintos usuarios objetivos como pueden ser los alumni, docentes, y otrs profesionales de la UAB y más concretamente del grado de ingeniería informática. Para ello, como ya se ha explicado, se ha dividido dicho objetivo en tres partes más concretas enfocadas en mejorar y optimizar distintos aspectos de la aplicación.

6.1 Mejoras en el feed

Las mejoras en el feed desarrolladas, que han sido el sistema de likes en posts convencionales y comentarios, y el sistema de ficheros en las publicaciones cumplen una función vital a la hora de conseguir el objetivo principal de mejorar la experiencia del usuario. Estas implementaciones ayudan a aumentar la calidad del eje central de la aplicación y donde los usuarios van a pasar la mayor parte del tiempo. El sistema de likes proporciona una sensación de calidad en el feed personal de cada usuario, ya que si aparecen publicaciones con más likes quiere decir que lo que se muestra en tu pantalla principal es contenido que en general gusta y puede considerarse de buena calidad, ya sea por su utilidad o por su originalidad. Además, como usuario, también proporciona la sensación de que puedes llegar a mejorar dicha calidad gracias al feedback que proporcionas a los autores de los posts. Si reaccionas con un 'Me gusta' a una publicación de un contenido concreto, el usuario autor sabrá que su contenido tiene cabida en la aplicación y seguirá subiendo posts similares en la medida de lo posible. De la misma forma, si un post de un mismo usuario recibe menos likes de lo normal también es beneficioso, puesto que sabrá que ese contenido no es tan popular en su red de contactos.

La situación con el sistema de ficheros es similar, pero sin ese punto de feedback hacia los autores de los posts. Los ficheros proporcionan un contenido mucho mas dinámico, visual y ligero, que entra por la vista en el caso de fotografías y videos. No es lo mismo publicar una oferta de trabajo en formato texto que adjuntar una imagen que resuma dicha oferta de una forma gráfica que en general suele gustar más a las personas. A su vez proporciona una gran facilidad para compartir documentos sin necesidad de recurrir al email o al contacto privado. En el mismo caso de la oferta de trabajo, un usuario podría responder adjuntando su CV en un comentario justo debajo, lo cual nos ofrece una experiencia dinámica y fluida, ayudando así a mejorar en gran parte la experiencia del usuario y favorenciendo el uso de la plataforma.

6.2 Mejoras en la interacción de usuarios

Las mejoras en relación con la interacción entre los propios usuarios y la interacción de la propia plataforma con el usuario buscan potenciar tanto el uso de la plataforma gracias a la fluidez y la mayor presencia de la aplicación en la vida del usuario, como en forjar contactos en el mundo real que, a fin de cuentas, es uno de los grandes objetivos de una red social de este tipo. Para ello, los eventos han sido la mejor de las implementaciones, puesto que es una funcionalidad muy útil para crear estos contactos. Gracias a su interactividad con el autor del evento y a su condición de ser una situación del mundo real, puede generar que una gran cantidad de usuarios con intereses similares se conozcan a raiz de participar en dichos eventos. Esto favorece mucho a la creación de relaciones tanto dentro como fuera de la red social forjando así vinculos laborales, educativos e incluso personales. Esta funcionalidad por lo tanto se convierte en el eje central de la interacción entre usuarios, y constituye un engranaje principal del núcleo de

la plataforma.

A su vez, el sistema de notificaciones pretende favorecer la interacción entre la plataforma y el usuario, además de buscar cumplir el objetivo de acercar más la aplicación al día a día de los individuos que la utilizan mostrándoles mensajes e información que está sucediendo a tiempo real y que puede ser de su interés. Al fin y al cabo las notificaciones son una parte esencial de las redes sociales ya que nos recuerdan constantemente que existen, que están ahí y que hay cosas de nuestro interés dentro de ellas. Si no hacemos este sistema de notificaciones los usuarios entrarán cada vez menos a la aplicación y finalmente puede quedar en el olvido. Es por esto que esta funcionalidad es muy útil para potenciar el uso de la plataforma, acercándola a las rutinas constantes de los usuarios.

Por otro lado, las funcionalidades de búsqueda avanzada y el rol de profesor cumplen un papel más discreto pero no menos importante, ya que favorecen a la fluidez y a la simpleza a la hora de utilizar el sistema. La búsqueda avanzada puede llegar a ahorrar una buena cantidad de tiempo, ofreciendo a los usuarios unos resultados mucho más concretos de lo que están buscando y de una forma sencilla de utilizar. A su vez, el rol de profesor pemite esepcificar todavia más qué papel cumple cada usuario dentro de la aplicación, acercando a los profesores y a los alumni de forma mucho más directa. Podemos concluir que estas dos últimas funcionalidades favorecen la interacción entre el usuario y la plataforma aumentando la simpleza de la misma y ofreciendo una experiencia de usuario más satisfactoria.

6.3 Mejoras de usabilidad y diseño

Por último, cabe mencionar que las mejoras generales tanto de diseño de interfaces[8] (remodelación de posts y página del perfil de usuario) como las mejoras de usabilidad (correcciones de bugs de funciones de asincronía y conexión con la BD) son una parte importante para ofrecer una experiencia de calidad y no ofrecer al usuario un mal rendimiento por parte de la aplicación, lo cual podría generar rechazo y actuar en nuestra contra al buscar potenciar el uso de la plataforma entre los usuarios.

Finalmente, se ha de comentar que durante el desarrollo de todas las funcionalidades se han tenido en cuenta los 4 requisitos que a fin de cuentas no se pueden dejar de lado en algunas partes y en otras no, sino que se debe de realizar el desarrollo siempre teniendo estas pautas en mente para poder ofrecer un producto que cumpla con las expectativas. Por tanto, se ha hecho un desarrollo pensando en la escalabilidad de la aplicación, manteniendo un diseño responsive, creando interfaces intuitivas en las nuevas funcionalidades y asegurando los datos de los usuarios.

7 Ampliaciones y líneas de mejora

Ahora que ya hemos realizado un repaso de todas las funcionalidades implementadas y de su desarrollo, y hemos analizado los resultados junto con los objetivos propuestos al inicio del documento, es hora de hablar del futuro de la aplicación y de posibles lineas de mejora para conseguir un proyecto mucho más pulido y optimizado del que hay a día de hoy. Si bien es cierto que con el trabajo realizado estos meses se ha logrado un proyecto más maduro y con mayor valor, todavía queda camino por recorrer y margen para desarrollar un sistema óptimo y diferencial del resto de alternativas existentes.

A lo largo del desarrollo de esta segunda versión de la Xarxa Alumni se han detectado nuevas posibles mejoras a implementar en un futuro, al igual que siguen existiendo algunas opciones ya detectadas en el desarrollo de la primera versión (como la aplicación móvil y la implementación de una base de datos no relacional), las cuáles no desarrollaremos en este documento.

7.1 Recomendaciones

Una incorporación interesante a la red social podría ser una sección de recomendaciones personalizadas generada a partir de un algoritmo y que muestre 3 o 4 usuarios que podrían resultarnos interesantes bajo el criterio de la aplicación. Dicho criterio estaría construido en base a los likes realizados por el usuario y a los contactos que ya tiene. El algoritmo debería de ser capaz de encontrar y reconocer patrones en los likes del usuario, recomendando así usuarios que realicen o bien publicaciones parecidas, o bien un número determinado de publicaciones con likes por parte del usuario. Dicha sección podría situarse tanto en uno de los laterales del feed principal como en uno de los laterales del perfil del usuario.

7.2 Chat privado

Otra funcionalidad que puede ser interesante es la posibilidad de añadir un sistema de mensajería privada para poder compartir informacón de tú a tú con otro usuario sin necesidad de recurrir a un feed público o a un grupo en concreto. De esta forma la comunicación entre usuarios se vería reforzada al aumentar así las vías de contacto y además también favorecería a la privacidad de los mensajes.

7.3 Recarga asíncrona del feed

Algo que podría ser interesante y útil al mismo tiempo es la posibilidad de ofrecer una recarga automática del feed principal cada pocos segundos. De esta forma el usuario no debe de estar pendiente de recargar constantemente la aplicación para poder ver nuevo contenido, sino que la propia aplicación se encargaría de notificar al usuario cuando se ha detectado nuevo contenido en su feed principal que ha sido publicado desde la última recarga. Cabe decir que esta funcionalidad sería de gran utilidad en el caso de un tráfico considerable de publicaciones en la plataforma.

7.4 Comentarios en varios niveles

Actualmente el sistema solamente permite un nivel de comentarios, es decir, existe la publicación principal y comentarios a la misma publicación, pero no se pueden realizar comentarios a los comentarios. Esto es una limitación a la hora de realizar una comunicación fluída puesto que no

sabes si un comentario va dirigido a tu comentario o a la publicación original. Por esto creo que una gran mejora sería la implementación de varios niveles de comentarios para crear hilos parecidos a los de Twitter[9], que son una gran herramienta a la hora de debatir o comentar sobre ciertos temas. Esto podría realizarse guardando en la base de datos un campo que indique el nivel del comentario y guardando una referencia al comentario al que se está respondiendo, al igual que con las publicaciones. De esta forma también es mucho más sencillo limitar el número de niveles para no generar conversaciones infinitas que afecten al rendimiento.

7.5 Sistema de ficheros 2.0

El sistema de ficheros implementado actualmente significa una gran mejora para el sistema, pero tiene algunas limitaciones que sería interesante solucionar a partir de un par de cambios. El primero de estos cambios es la implementación de la posibilidad de añadir más de un fichero adjunto a una misma publicación, ya que en el sistema actual solamente se puede adjuntar un archivo y no podemos crear, por ejemplo, un post con una imagen y un documento PDF. El segundo de los cambios consiste en implementar la posibilidad de previsualización de otro tipo de archivos además de fotos y videos, como documentos PDF o archivos de word.

8 Conclusiones

Como conclusión del propio sistema desarrollado, puedo decir que, aunque sigue siendo un producto con un gran margen de mejora, actualmente es un producto más completo y atractivo que al iniciar el proyecto. Se han añadido funcionalidades básicas de cualquier red social, como los likes y los ficheros adjuntos, pero que añaden un gran peso al valor de la aplicación y que ofrecen una experiencia de usuario mucho más completa, y también se han añadido funcionalidades más específicas en relación a las necesidades del sistema como los eventos o el rol de profesor, para favorecer en gran medida el contacto entre los usuarios, que a fin de cuentas es un objetivo principal de una red social.

En cuanto a conclusión personal, no solo he tenido que realizar un desarrollo de código, sino también una lectura y un proceso de comprensión de un código ya escrito, que a veces resulta más complicado que escribir el propio código. Me he visto forzado a entender código escrito por otros, he aprendido lenguajes y frameworks que apenas había utilizado como PHP y Tailwind, y he profundizado en otras tecnologías como JavaScript o HTML. Pero además del aprendizaje práctico, he desarrollado mi capacidad de análisis de los problemas a los que hay que buscarles una solución. He aprendido a analizar mucho mejor los requisitos, he conseguido construir desde cero y a partir de dichos requisitos soluciones propias a los problemas planteados y he desarrollado más habilidad a la hora de enfrentarme a bugs y problemas en el código, aprendiendo a debuggar y analizar el código línea a línea de forma más eficiente.

En definitiva considero que desarrollar este proyecto o uno de características similares requiere de dedicación, constancia y perseverancia, pero al mismo tiempo te proporciona una gran experiencia en los ámbitos en los que trabajas y merece la pena por eso y por la satisfacción de ser capaz de construir algo por ti mismo.

Espero que la Xarxa Alumni en general, tanto la parte de red social como la parte de administración, sea una herramienta útil para los alumni del GEI y, en un futuro, para alumni de otros grados, y que esté disponible para el uso público lo antes posible.

Agradecimientos

Para finalizar, agradecer a mi tutor Jordi Pons por toda la ayuda brindada a lo largo del desarrollo y por los consejos y el feedback proporcionado, además de la constante disponibilidad en cualquier momento del día para resolver mis dudas. Agradecer también a mi compañero Pere Peréz que siempre ha estado dispuesto a ayudarme con cualquier problema en relación al código y a echarme una mano, sobretodo durante el desarrollo del sistema de ficheros.

Por último agradecer a mi familia, mis amigos y mi pareja por despejarme la cabeza cuando las cosas no funcionaban y hacerme ver los problemas desde una perspectiva más calmada y un mejor ángulo en los momentos de frustración.

Referències

[1] Pere Pérez López, "Disseny i implementació d'una plataforma per a la gestió de la Xarxa Alumni del GEI: La xarxa social", Escola Enginyeria (EE) UAB, grado en ingeniería informática.

[2] MariaDB.
Disponible: https://mariadb.org/

[3] GPL.
Disponible: https://es.wikipedia.org/wiki/GNU_General_Public_License

[4] XAMPP.
Disponible: https://www.apachefriends.org/es/index.html

[5] Qué es MVC.
Disponible: https://desarrolloweb.com/articulos/que-es-mvc.html

[6] Tailwind Docs.
Disponible: https://tailwindcss.com/docs/installation

[7] Deloitte. ¿Qué es el desarrollo en Espiral? Disponible: https://www2.deloitte.com/es/es/pages/technology/articles/que-es-el-desarrollo-en-espiral.html

[8] Arsys. 7 buenas prácticas para diseñar interfaces de usuario.
Disponible: https://www.arsys.es/blog/buenaspracticas-interfaces-usuario

[9] Twitter.
Disponible: https://twitter.com/home

APÉNDICE

A FUNCIONALIDADES DEL FEED

Modal de creación de un post

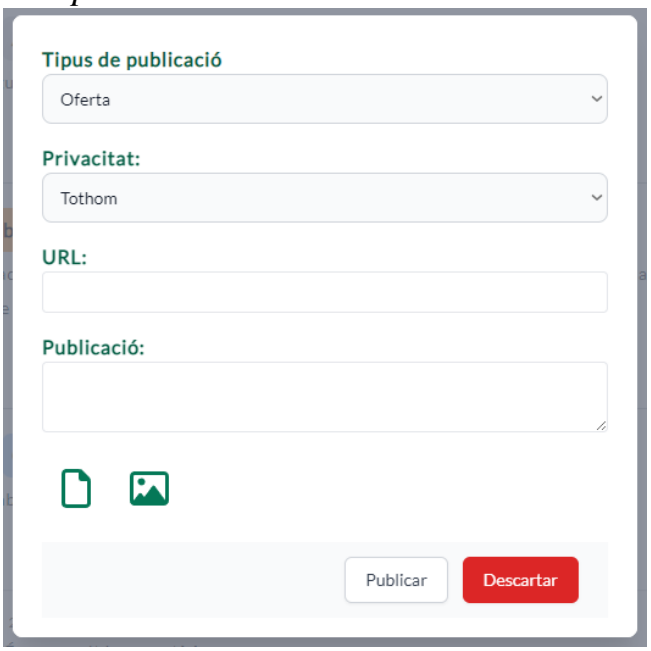

Figura A.1

Post con imagen adjunta

Figura A.2

Post con documento PDF adjunto

Figura A.3

B FUNCIONALIDADES DE INTERACCIÓN ENTRE USUARIOS

Modal para crear un evento

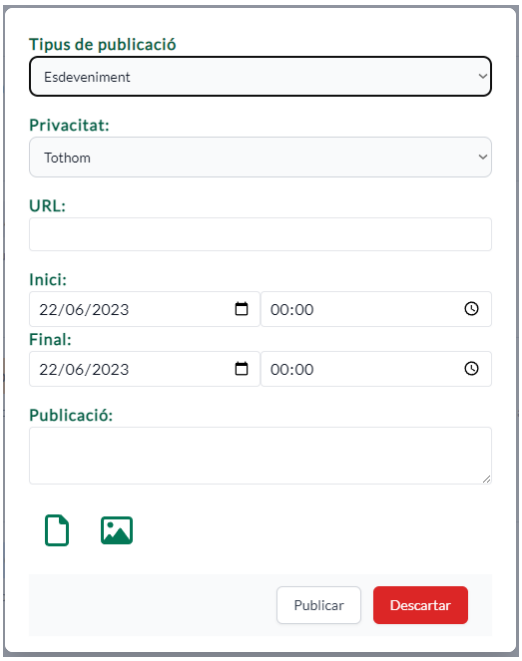

Figura B.1

Publicación de tipo 'Esdeveniment'

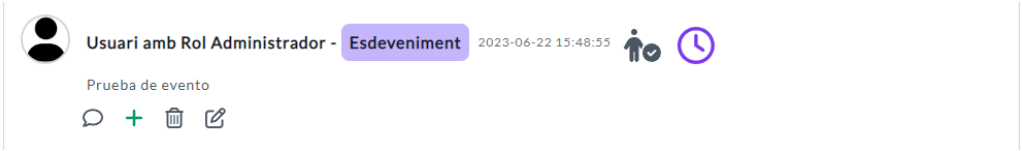

Figura B.2

Lista de participantes del evento

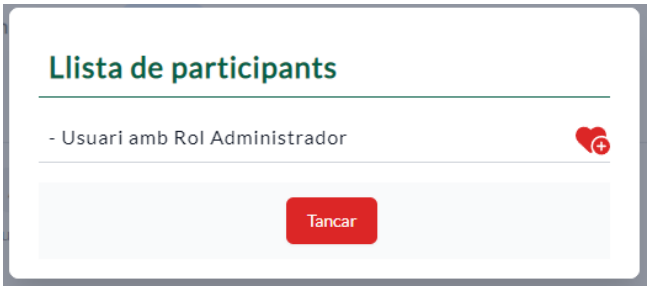

Figura B.3

Tabla de profesores

Figura B.4

C MEJORAS DE USABILIDAD Y DISEÑO

Nuevo diseño de posts

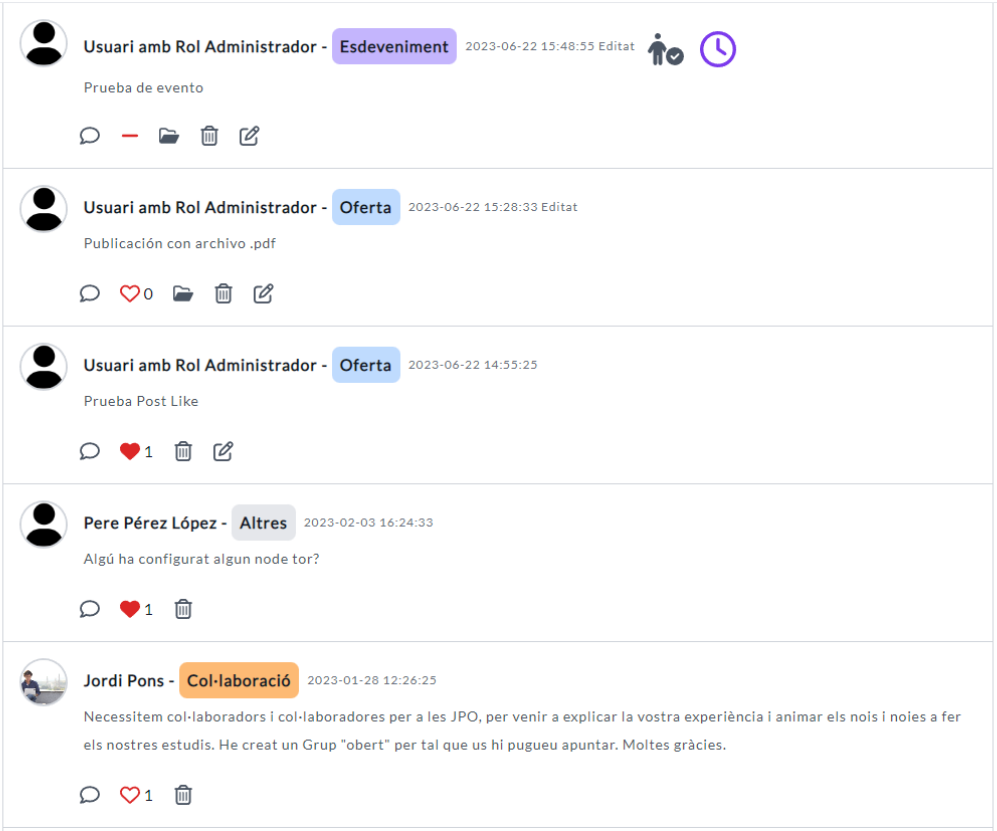

Figura C.1

Nueva pestaña de perfil

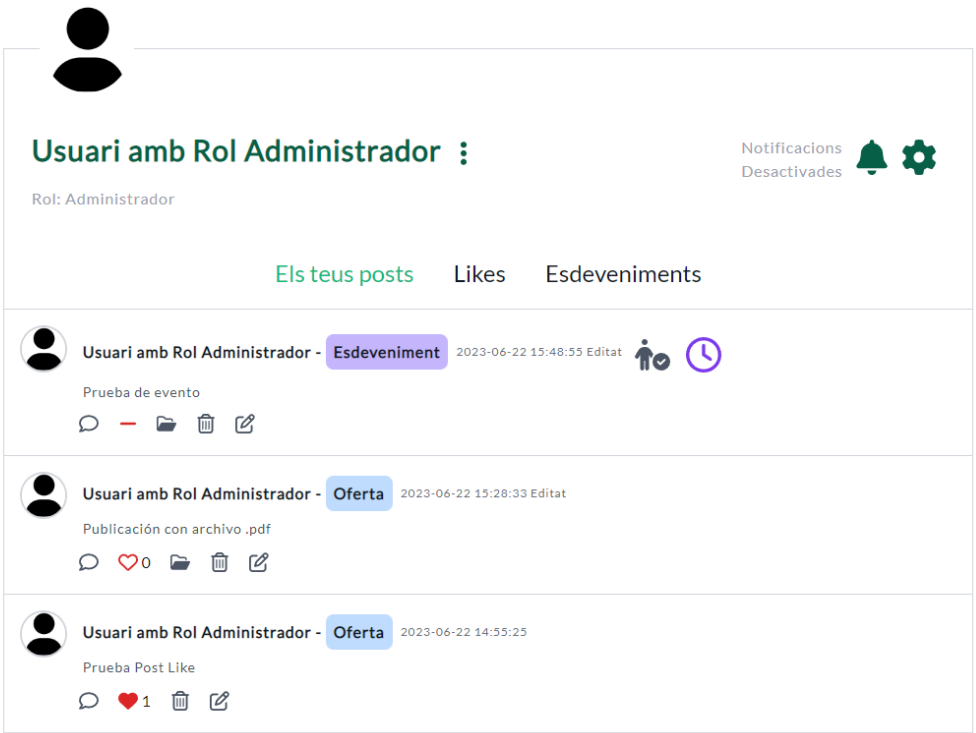

Figura C.2

D PLANIFICACIONES

Planifiación inicial

	Feb		Mar				Abr				May				Jun			
	S2	S3	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4
Definición del contexto y los objetivos	X	X																
Lectura del código y estudio de funcionalidades		X	X	X														
Desarrollo de funcionalidades del feed (Análisis)					X	X												
Desarrollo de funcionalidades del feed (Desarrollo)						X	X	X										
Desarrollo de funcionalidades del feed (Test)								X										
Desarrollo de funcionalidades de usuario (Análisis)									X	X								
Desarrollo de funcionalidades de usuario (Desarrollo)										X	X	X						
Desarrollo de funcionalidades de usuario (Test)												X						
Modificación y mejoras de la interfaz web													X	X				
Revisión y resolución de problemas encontrados															X	X	X	X

Figura D.1

Planificación modificada

	May	Jun			
	S	S1	S2	S3	S4
Desarrollo sistema notificaciones (Desarrollo)	X				
Desarrollo sistema notificaciones (Test)	X				
Desarrollo rol profesor (Análisis)		X			
Desarrollo rol profesor (Desarrollo)		X	X		
Desarrollo rol profesor (Test)			X		
Desarrollo Búsqueda avanzada (Análisis)		X			
Desarrollo Búsqueda avanzada (Desarrollo)			X	X	
Desarrollo Búsqueda avanzada (Test)				X	
Modificación y mejoras de la interfaz web				X	X
Revisión y resolución de problemas encontrados					

Figura D.2