

Proyecto PhotoTDS 2022/2023

Junio 2023

PhotoTDS



Francisco Jesús Camacho Molina - 49441861A Grupo 2
Jose Manuel Lorca Aguilar- 48755446T Grupo 2

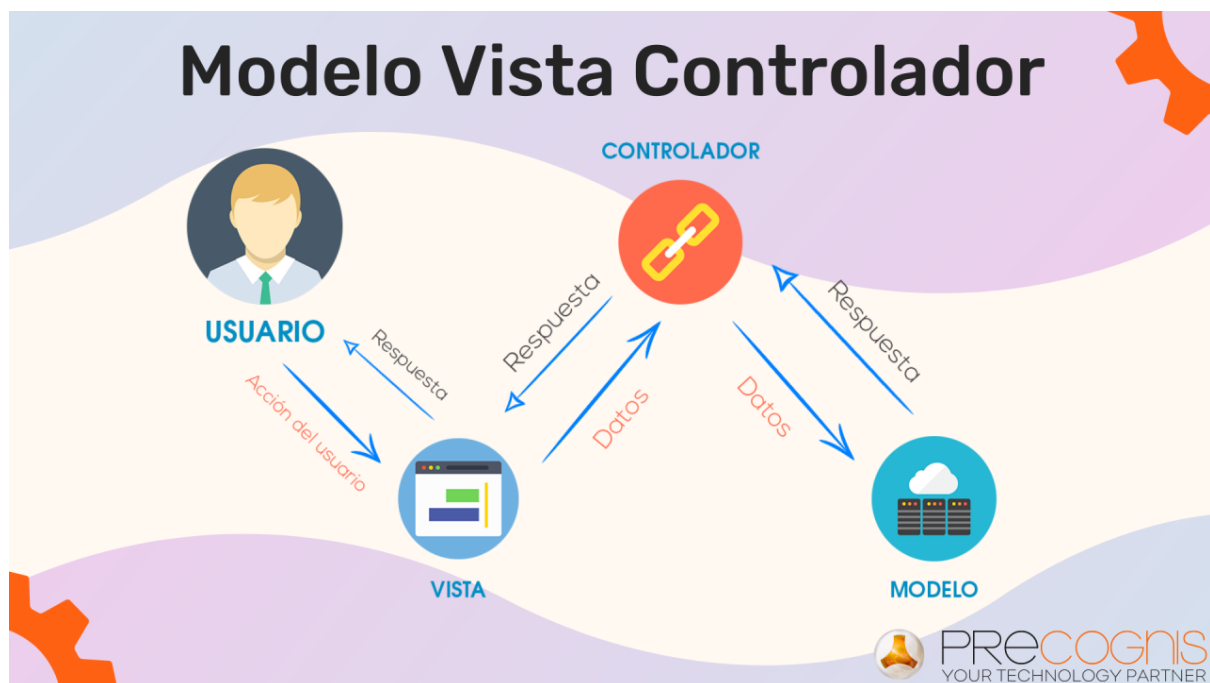
1. Introducción.....	3
2. Arquitectura de la aplicación.....	3
3. Diagrama de clases.....	4
4. Diagrama de secuencia para crear una foto.....	4
5. Patrones de diseño.....	5
5.1 Singleton:.....	5
5.2 Bridge:.....	5
5.3 Observer:.....	5
5.4 Composite:.....	6
5.5 Estrategia + Composite.....	6
5.6 DAO: Adapter + Abstract Factory.....	6
6. Decisiones de diseño.....	7
7. Componentes JavaBean.....	8
7.1 Cargador de fotos.....	8
7.2 Luz.....	8
7.3 Jcalendar.....	8
8. Tests Unitarios.....	8
9. Manual de Usuario.....	8
9.1 Ventana login.....	8
9.2 Ventana registro.....	9
La ventana de registro se puede acceder desde la ventana de inicio de sesión y tiene como propósito permitir el registro de un nuevo usuario en la aplicación. A continuación, se muestran los campos correspondientes en la siguiente figura.....	9
9.3 Ventana inicio.....	9
9.4 Subir una foto.....	9
9.5 Ver foto.....	9
9.6 Perfil.....	9
9.7 Álbumes.....	10
9.8 Crear Álbumes.....	10
9.9 Subir fotos a álbumes.....	10
10. Uso de GitHub.....	10
11. Observaciones finales.....	11

1. Introducción.

En este documento vamos a explicar las partes más importantes de nuestra implementación del proyecto **PhotoTDS**. Intentando siempre hacer código fuente limpio, modular y fácilmente mantenible, es decir: seguir los principios SOLID.

2. Arquitectura de la aplicación.

Para el desarrollo de la aplicación hemos hecho uso de la arquitectura MVC, que promueve una separación clara de responsabilidades y facilita el desarrollo y mantenimiento del software.



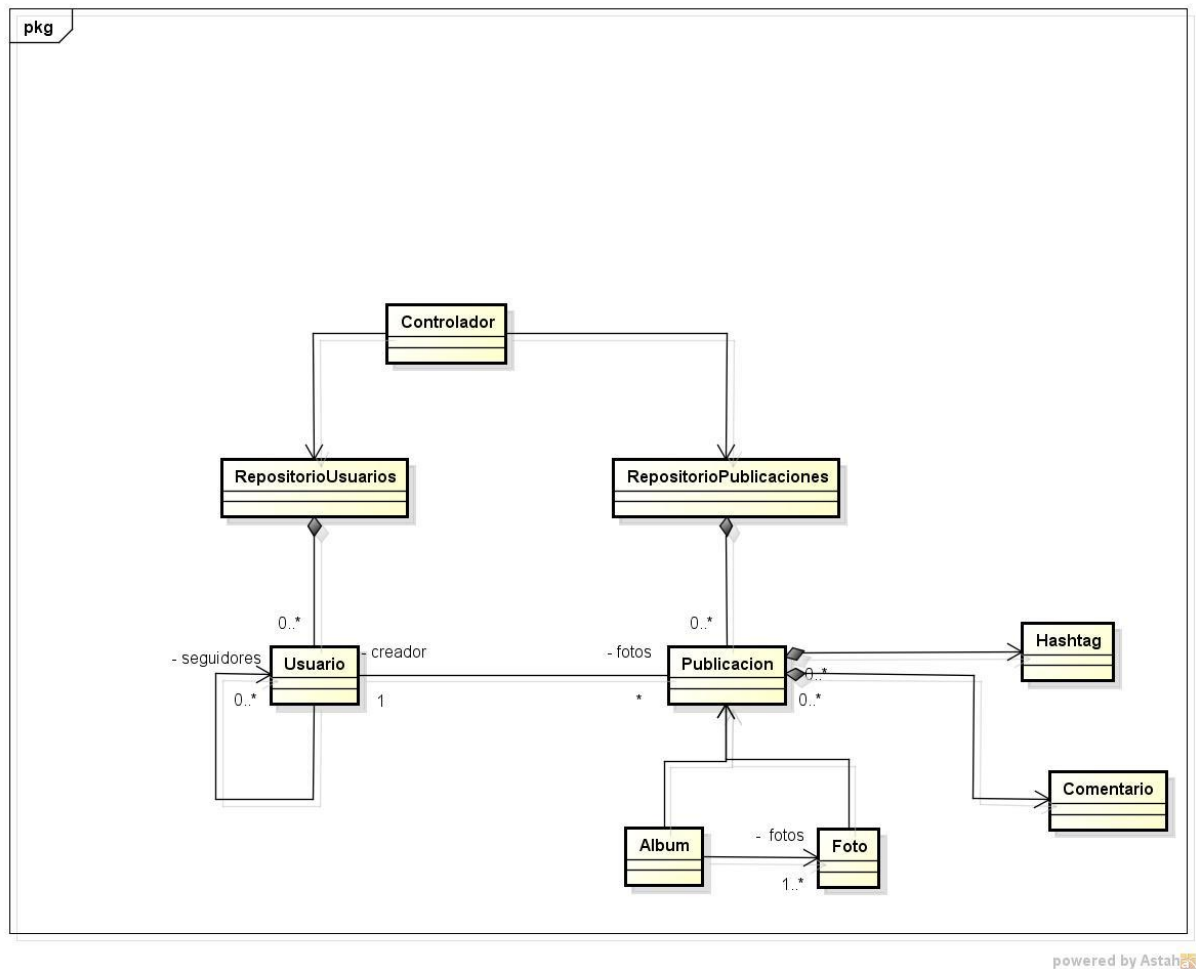
Este patrón de diseño consta de tres componentes principales:

1. Modelo (Model): El modelo representa la capa de datos de la aplicación y en esta capa se definen las estructuras de datos y las reglas de negocio.

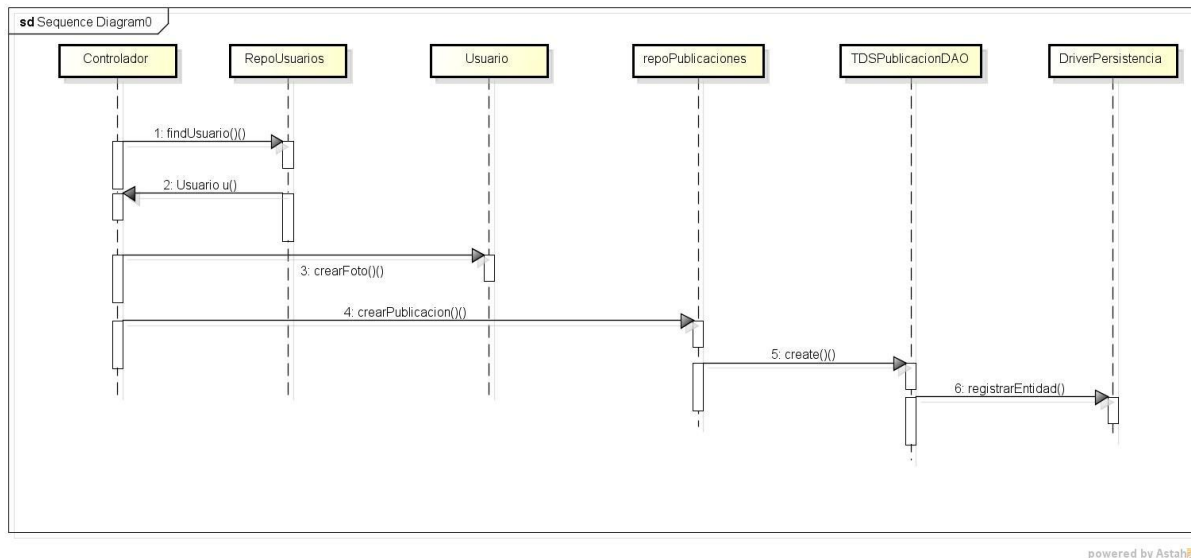
2. Controlador (Controller): El controlador actúa como intermediario entre el modelo y la vista. Responde a las interacciones del usuario y gestiona las solicitudes entrantes. Se encarga de actualizar el modelo según las acciones del usuario y de actualizar la vista con los datos actualizados del modelo. El controlador es responsable de orquestar el flujo de la aplicación y manejar la lógica de coordinación entre el modelo y la vista.

3. Vista (View): La vista es la capa de presentación de la aplicación. Es responsable de mostrar los datos al usuario y de proporcionar una interfaz de usuario interactiva. La vista no realiza ninguna lógica de negocio y se limita a mostrar los datos proporcionados por el modelo.

3. Diagrama de clases.



4. Diagrama de secuencia para crear una foto.



5. Patrones de diseño.

Los patrones de diseño son esenciales en el desarrollo de software, ofreciendo soluciones probadas, fomentando la reutilización de código y facilitando la comunicación entre los desarrolladores. Permiten crear software flexible, escalable y fácilmente mantenible.

En el desarrollo de este proyecto, hemos utilizado los siguientes:

5.1 Singleton:

Este es uno de los patrones más importantes de nuestro proyecto, ya que lo usamos en multitud de casos en PhotoTDS: Repositorios, Adaptadores, Factorías... etc.

Consiste en crear una instancia global de estas clases.

5.2 Bridge:

Aunque no utilizamos directamente el patrón Puente en nuestro proyecto, es empleado en el núcleo de Swing. Un ejemplo de su uso se encuentra en la clase JButton, la cual contiene un atributo de implementación que evidencia el uso del patrón Puente.

5.3 Observer:

Hemos utilizado el patrón de forma tanto directa como indirecta en nuestro proyecto. En la implementación directa, aplicamos el patrón con el uso del Cargador de Fotos, el cual será explicado con mayor detalle en la sección de Componentes.

Por otro lado, de forma indirecta, Swing emplea el Modelo de Delegación de Eventos para implementar los Action Listeners de los botones, entre otras funcionalidades.

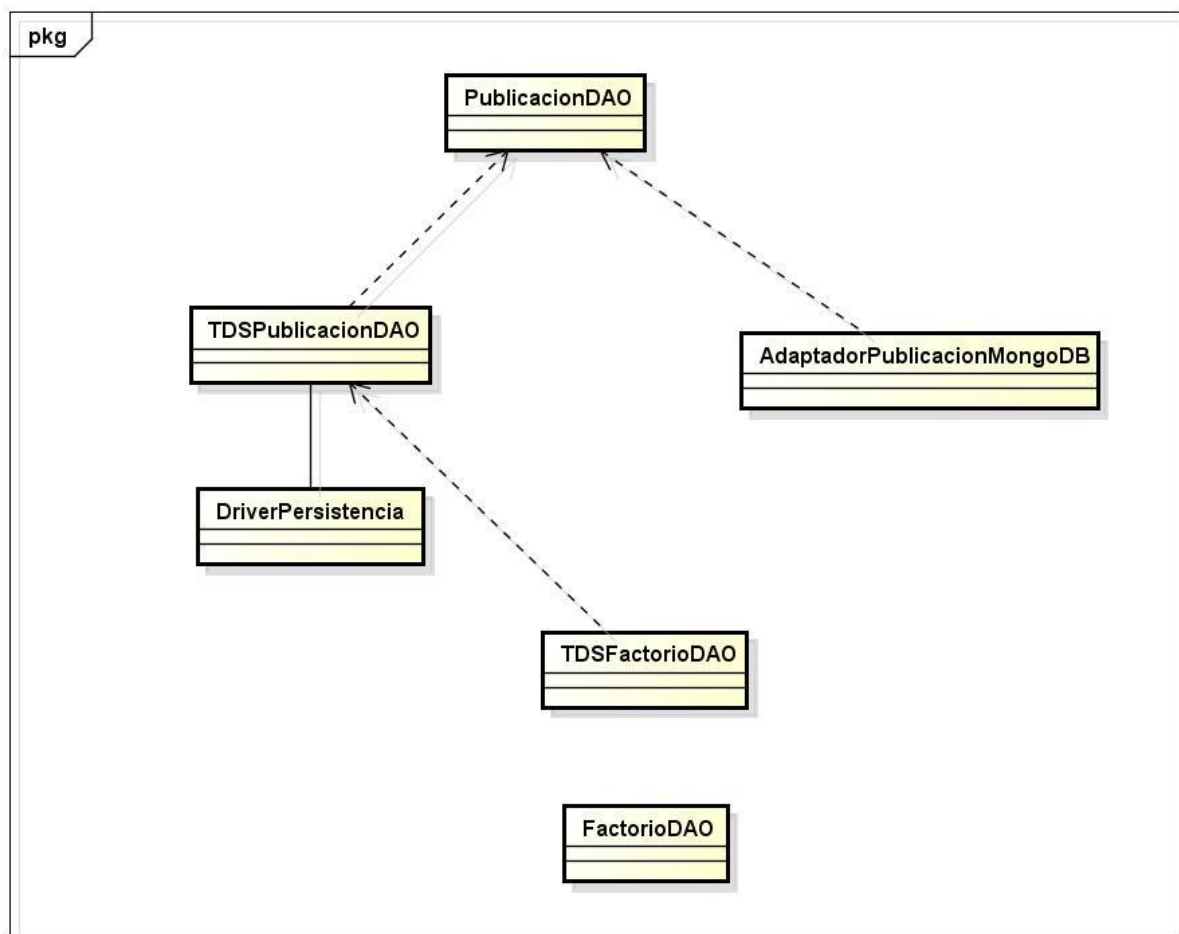
5.4 Composite:

Una vez más, el patrón se utiliza de manera indirecta en Swing. En particular, está relacionado con la clase Component de Swing. Todos los componentes que agregamos a las ventanas pueden estar compuestos por otros componentes a su vez.

5.6 DAO: Adapter + Abstract Factory.

Para lograr que PhotoTDS sea independiente de una tecnología específica de base de datos, hemos incorporado el patrón DAO. En realidad, este patrón consta de dos subpatrones: el Adaptador y la Factoría Abstracta.

El patrón Adaptador nos permite abstraernos de una tecnología concreta de base de datos, mientras que la Factoría Abstracta nos brinda la capacidad de crear una familia de clases asociadas a una tecnología de persistencia específica. En nuestra implementación, el diagrama de clases correspondiente se muestra a continuación:



powered by Astah

Ponemos un ejemplo de un adaptador que podríamos implementar para que la persistencia fuera en MongoDB.

6. Decisiones de diseño.

-Diferenciar publicaciones a nivel de persistencia en vez de repositorio:

Como podemos comprobar en el Diagrama de Clases de la Sección 2, tanto las Fotos, como los Álbumes son tipos de Publicación. Para hacer la persistencia de los mismos y respetar la jerarquía de clases, tenemos un repositorio de publicaciones, y diferenciamos como se guardan en persistencia en los Adaptadores del DAO. pero creemos que es como mejor se respeta el diagrama de clases.

Paquete Helpers : En esta paquete definimos funciones estáticas que usamos en varios puntos de nuestro modelo de la aplicación para que podamos acceder a ellas de manera sencilla y sin repetir código. Ejemplos de estas funciones pueden ser funciones auxiliares para conversores de fechas a string, etc.

Login Obligatorio: Para poder usar cualquier funcionalidad del controlador previamente ha tenido que ser logueado el usuario.

7. Componentes JavaBean.

7.1 Cargador de fotos.

Componente Java Beans creado por nosotros con el cual podemos añadir todas las fotos que indiquemos en un archivo xml con tal solo pulsar un botón (Luz explicado a continuación).

7.2 Luz.

Componente Java Beans proporcionado en los recursos de la asignatura es un botón circular que se puede utilizar en una ventana. Este componente se integra fácilmente con WindowsBuilder y se puede utilizar como cualquier otro componente de Swing. Para utilizar este componente, es necesario implementar la interfaz IEncendidoListener.

7.3 Jcalendar.

Lo usamos dentro de la VentanaRegistro, es un componente que nos permite seleccionar la fecha de nacimiento mediante un calendario interactivo. Para utilizar este componente, simplemente debemos añadir una entrada en el archivo pom.xml buscando la referencia en el repositorio Maven.

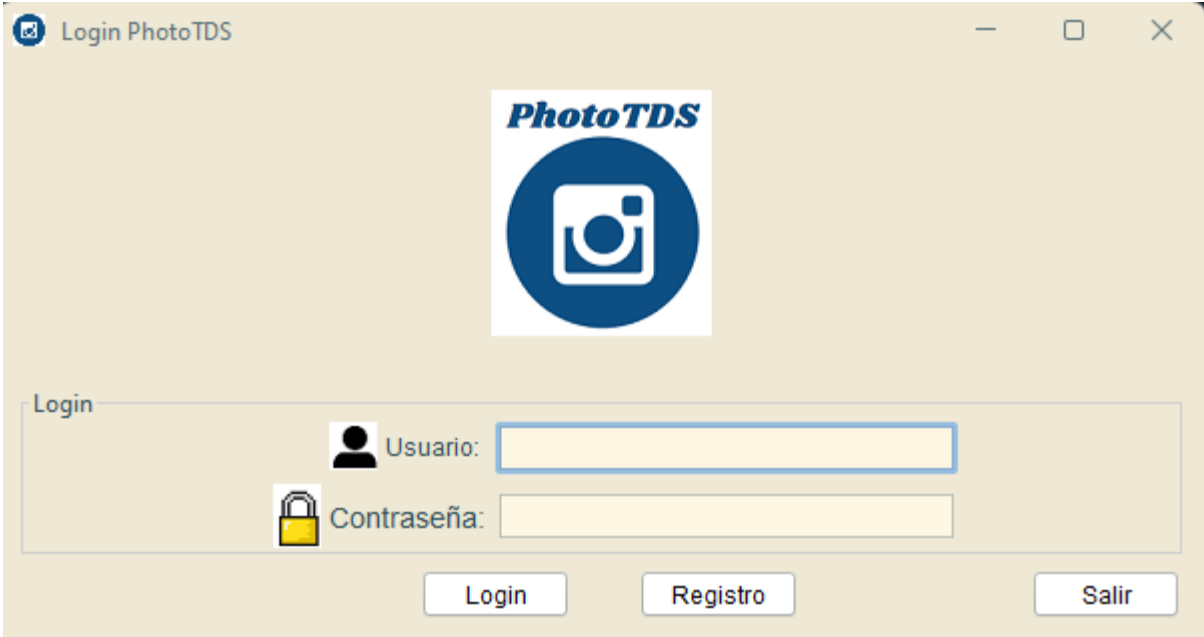
8. Tests Unitarios.

Lamentablemente, debido a restricciones de tiempo en nuestro proyecto de desarrollo de software, no pudimos completar la implementación de los test unitarios como estaba inicialmente planeado. Sin embargo, somos plenamente conscientes de la importancia de los test unitarios en el proceso de desarrollo de software. Reconocemos que los test unitarios son fundamentales para garantizar la calidad del código, detectar posibles errores y facilitar el mantenimiento a largo plazo.

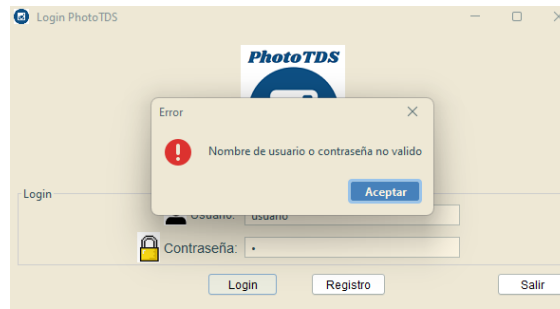
9. Manual de Usuario.

9.1 Ventana login

La ventana de inicio de sesión es la primera interfaz que encontraremos al acceder a la aplicación. Su objetivo es permitirnos acceder a la aplicación si ya estamos registrados. En esta ventana, nos encontramos con un campo para ingresar el nombre de usuario, otro para ingresar la contraseña, y dos botones: el botón de inicio de sesión, el cual nos dará acceso a la aplicación una vez que hayamos completado correctamente los campos de nombre de usuario y contraseña; y el botón de registro, el cual nos llevará a otra ventana para poder registrar un nuevo usuario.



En esta ventana obtendremos un mensaje de error si no hemos conseguido loguearnos correctamente.




9.2 Ventana registro

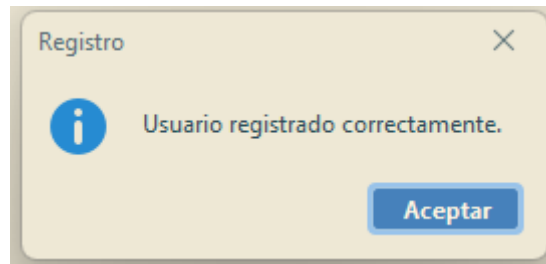
La ventana de registro se puede acceder desde la ventana de inicio de sesión y tiene como propósito permitir el registro de un nuevo usuario en la aplicación. A continuación, se muestran los campos correspondientes en la siguiente figura.

Los campos de arriba son los datos personales y los de abajo los que usaremos en la aplicación tanto como para logearnos, como para subir fotos con dicho username.

The 'Ventana Registro' contains the following fields and controls:

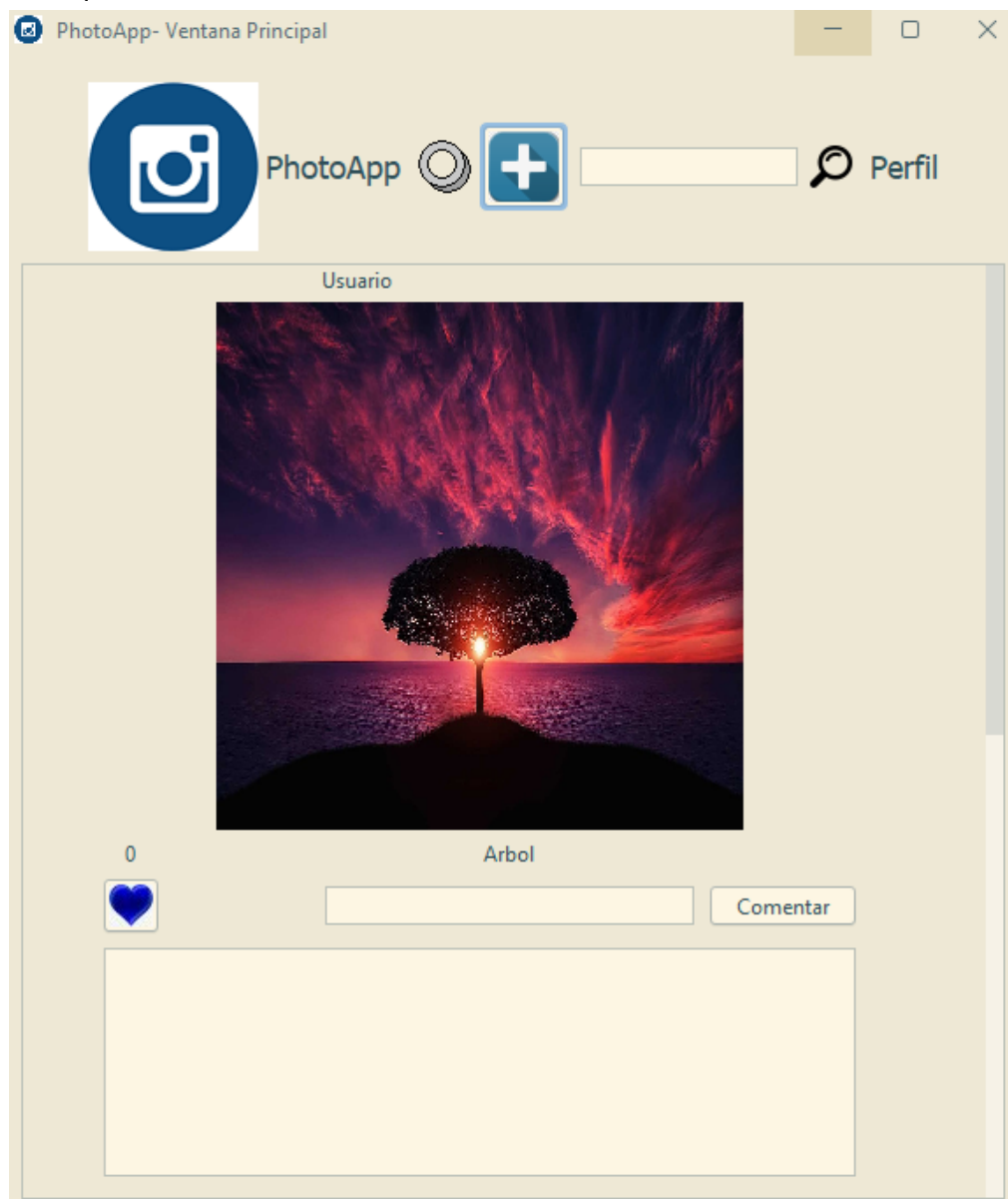
- Datos personales:**
 - Email:
 - Nombre:
 - Apellidos:
 - Fecha Nacimiento: 
- Datos Aplicación:**
 - Nombre Usuario:
 - Contraseña:
 - Repetir Contraseña:
 - Foto de perfil:
- Buttons:**

Si no hemos completado alguno de los campos o no lo hemos completado correctamente, se resaltará en rojo dicho campo indicándonos que nos hemos equivocado. Si todo está correcto nos aparecerá una pestaña indicándonos que el usuario se ha registrado.



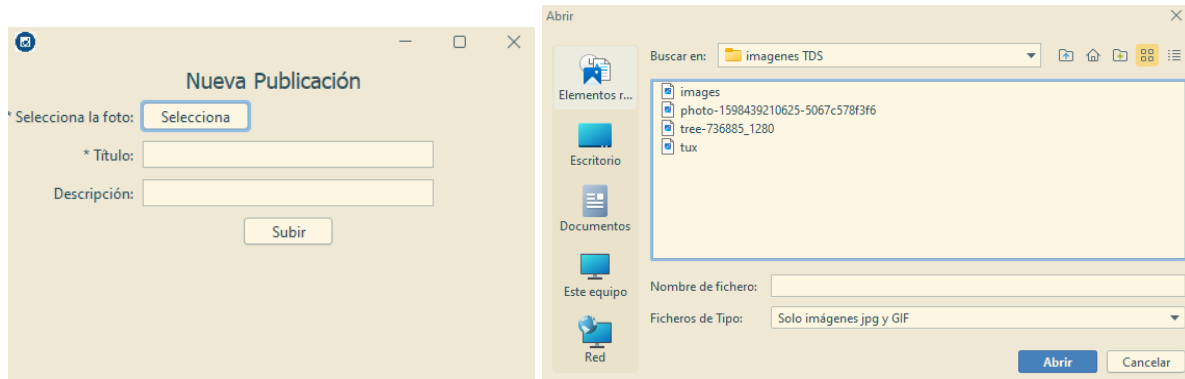
9.3 Ventana inicio

La ventana de inicio se refiere a la página principal de la aplicación, a la cual se accede después de iniciar sesión con un usuario. En esta ventana, se mostrarán todas las publicaciones de los usuarios de PhotoTDS. Si nos vamos de esta vista, para volver a ella solo deberemos hacer click en el logo o en el texto “PhotoApp” que se encontrará siempre arriba a la izquierda.



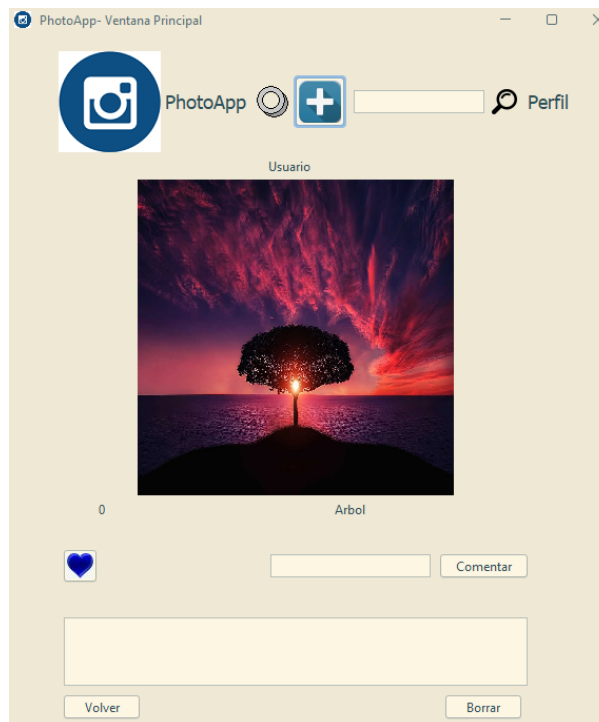
9.4 Subir una foto

Para subir una foto, simplemente debes pulsar el botón de adición en el panel superior. Al hacerlo, aparecerá una nueva ventana donde se te solicitará que ingreses un título y una descripción para la foto que desees subir. En la descripción, podrás proporcionar una descripción detallada de la foto, así como incluir hashtags incrustados en el texto. Los hashtags serán útiles para poder buscar la foto una vez que se haya subido.



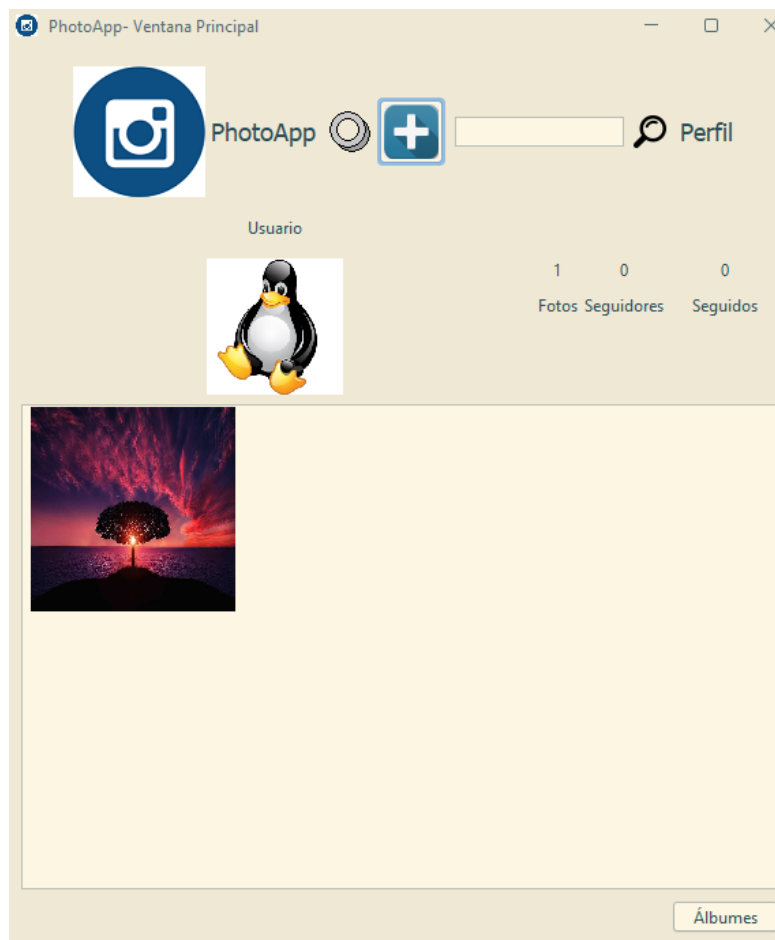
9.5 Ver foto

Al hacer clic en una imagen, ya sea buscándola mediante un hashtag, desde la ventana de inicio o desde el perfil, se abrirá la foto en cuestión. En esta vista, se mostrará el usuario propietario de la foto, el título de la foto, la propia imagen, el número de "me gusta", un botón para darle "me gusta" a la foto, un cuadro para publicar comentarios sobre la foto y todos los comentarios previamente publicados. Además, debajo de los comentarios, encontrarás un botón que te permitirá borrar la foto.



9.6 Perfil

Al hacer clic en el botón de perfil, accedemos a la página de perfil del usuario. En esta página, se mostrará la foto de perfil del usuario, su nombre de usuario, un botón para editar cierta información del perfil, el número de seguidos y seguidores, un botón para cerrar sesión, el nombre completo del usuario y su descripción. Además, se listarán todas las imágenes que el usuario ha subido a la aplicación. Debajo de estas imágenes, encontraremos un botón que, al pulsarlo, nos llevará a los álbumes subidos por el usuario.

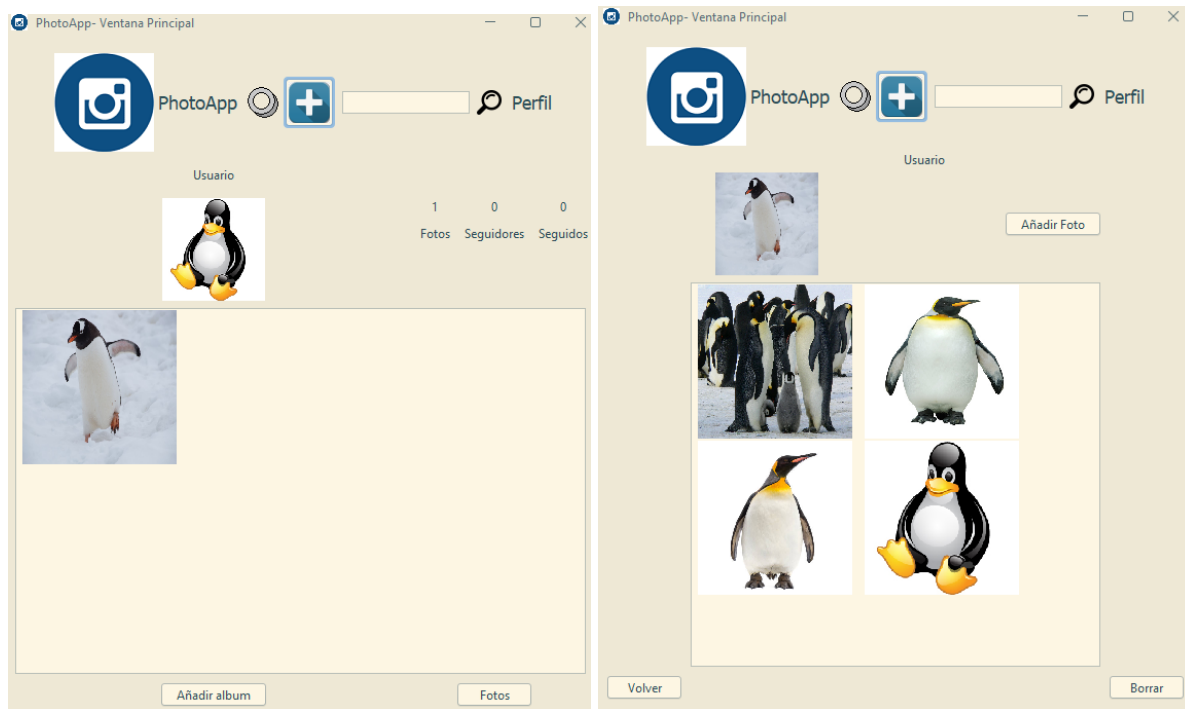


9.7 Álbumes

En el perfil, al pulsar el botón de "Álbumes", nos dirigimos a la ventana donde se muestran todos los álbumes del usuario.

En esta ventana, encontraremos una lista de todos los álbumes que el usuario ha subido. Al hacer clic en cualquier álbum, se abrirá y mostrará las imágenes que contiene.

En la parte inferior de la ventana, encontraremos dos botones. El botón de la derecha, al pulsarlo, nos devolverá a la pestaña donde se encuentran las fotos que ha subido el usuario. Y a la izquierda, hay un botón para subir un nuevo álbum.



9.8 Crear Álbumes

Para crear un nuevo álbum, desde la pestaña de álbumes, simplemente pulsamos el botón "Añadir álbum" y se abrirá una nueva pestaña similar a la de subir imágenes.

En esta pestaña, debemos proporcionar un título y una descripción para el álbum, además de seleccionar una foto que será la primera en añadirse al álbum. Una vez completados estos campos, al pulsar el botón "Subir", el álbum se subirá y estará listo para su uso.

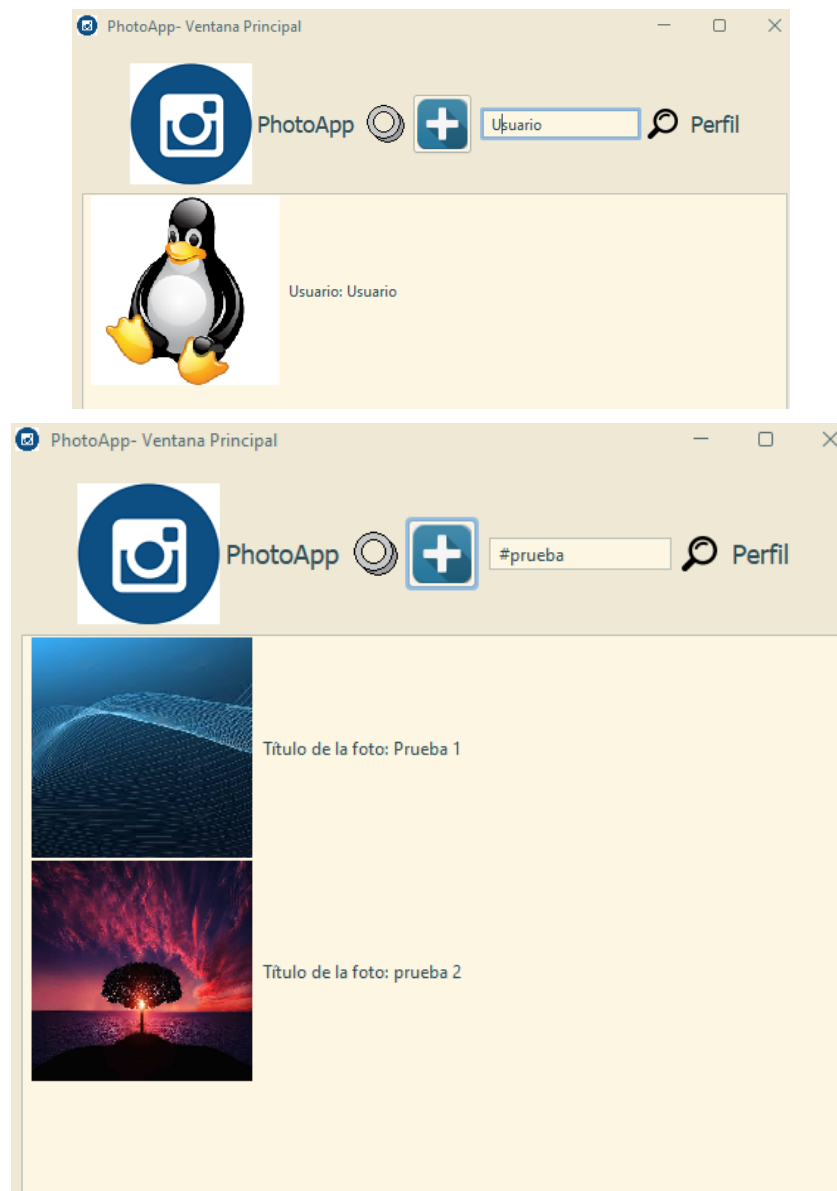
9.9 Subir fotos a álbumes

Si hacemos clic en un álbum, podremos ver la información del creador del álbum, el título del álbum y la primera imagen que se considera como portada del álbum. Además,

encontraremos un botón para añadir fotos al álbum. Si le damos a este botón se desplegará la misma ventana que en el caso anterior y nos permitirá seleccionar una nueva imagen que agregar al álbum.

9.10 Buscar foto o usuario

A la izquierda del botón “Perfil” en la Ventana Principal, tenemos un cuadro de texto junto a una lupa. En él podemos indicar un nombre de usuario o un hashtag y tras hacer click en la lupa nos mostrará el resultado de la búsqueda.



Si hacemos click en el resultado mostrado que queramos, nos dirigirá al perfil del usuario en caso de haber buscado un usuario o al panel de la foto, en caso de haber buscado una foto mediante su hashtag.

10. Uso de GitHub.

El uso de github no ha sido del todo óptimo en nuestro proyecto, sobre todo al principio del mismo ya que era la primera vez que usábamos github para un proyecto medio-grande y hemos tenido fallos que nos han provocado algunos conflictos, como estar trabajando cada uno en una rama diferente y al hacer merge encontrarnos con que una clase la habíamos modificado los dos.

Sin embargo, las semanas previas a la entrega nos hemos coordinado para trabajar los dos en la misma rama, si uno trabajaba y el otro estaba con otra asignatura o sin tocar nada del proyecto, al finalizar hacia el commit y cuando el otro comenzara a trabajar hacia el pull a la rama y tenía el trabajo del otro, y si los dos estábamos a la vez en el proyecto, nos coordinamos por discord para que no trabajamos en las mismas clases, o uno de los dos compartía pantalla para que el otro pudiera ayudar.

Al final, nos ha resultado muy gratificante el uso de github, aunque hayamos tenido algunos errores de novatos, hemos aprendido a trabajar con comunicación y sin cambiar el trabajo del otro compañero.

11. Observaciones finales.

Nos ha parecido muy interesante la práctica, ya que era la primera vez que desarrollamos un código de gran tamaño y nos sirve para experiencias futuras ya sea laborales o académicas, hemos entendido la importancia de una buena comunicación en equipo para el desarrollo de proyectos de este tamaño, y también la importancia de tener en cuenta las buenas prácticas de programación y el uso de patrones GRASP, ya que cuando lo hemos aplicado bien, el desarrollo del proyecto ha sido más fácil.

Aproximadamente, cada integrante del grupo le habremos dedicado entre **60-80h** cada uno, siendo este un valor muy exacto porque al principio, al no haber podido ir a las clases teóricas y prácticas por conflictos en los horarios de asignaturas de cuarto, “perdimos” bastante tiempo viendo diapositivas de la parte teorica y practica sin poder avanzar mucho ya que estábamos bastante perdidos.

Como conclusión, este proyecto nos ha servido para evitar futuros errores a la hora de programar proyectos más grandes y para hacer uso de buenas técnicas de programación.