



---

TÍTULO DE GRADO EN  
INGENIERÍA INFORMÁTICA

# **Tecnologías de Desarrollo de Software**

Práctica final

**Documentación**

CURSO 2018 / 2019

CONVOCATORIA DE ENERO

---

Grechyshkin Vladyslav – vladyslav.grechyshkin@um.es

Valero Leal Enrique – enrique.valerol@um.es

Grupo 2.2

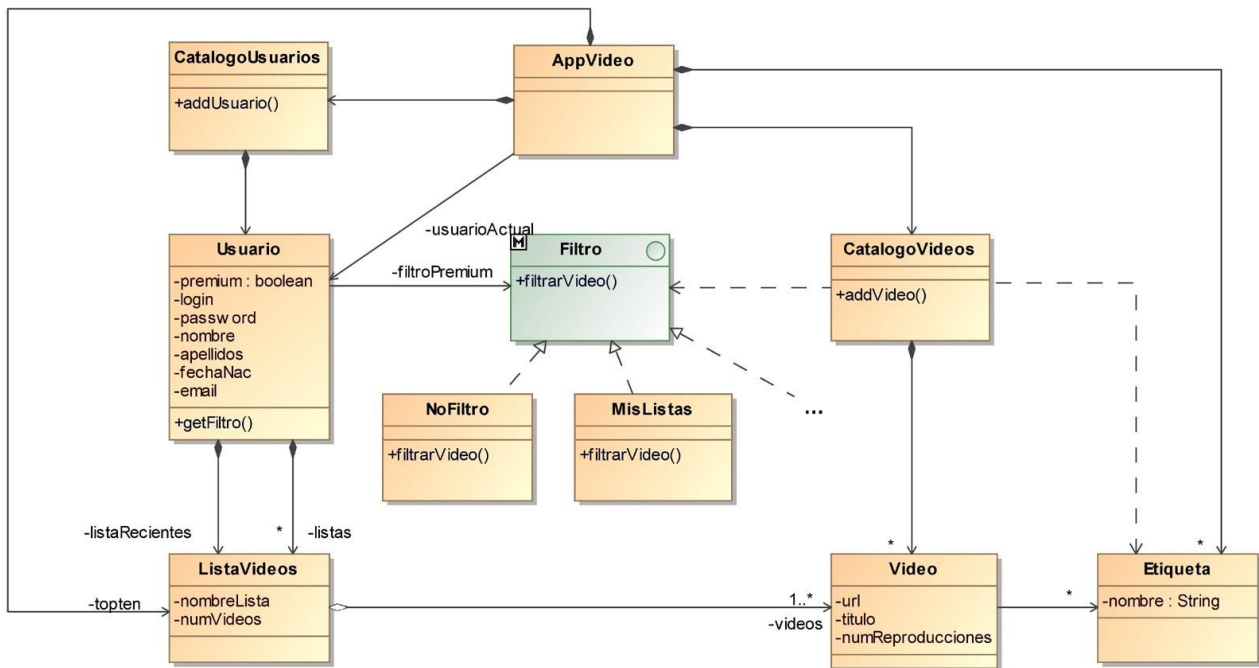
# Índice general

I. Diagrama de clases del dominio .....	2
II. Diagrama de secuencia para añadir un vídeo .....	3
III. Arquitectura de la aplicación .....	4
IV. Patrones de diseño utilizados .....	5
4.1 – Patrón Factoría Abstracta .....	5
4.2 – Patrón Adaptador .....	5
4.3 – Patrón Singleton .....	5
4.4 – Patrón Fachada .....	5
V. Componentes utilizados .....	6
5.1 – Componentes de la librería JFoenix .....	6
5.2 – Componente Buscador de Vídeos .....	6
5.3 – Biblioteca iTextPDF .....	7
5.4 – Componente VideoWeb .....	7
VI. Tests unitarios .....	7
VII. Manual de usuario .....	8
7.1 – Log in .....	8
7.2 – Registro .....	9
7.3 – Recientes .....	10
7.4 – Mi perfil .....	11
7.5 – Explorar .....	12
7.6 – Mis listas .....	13
7.7 – Premium .....	15
7.8 – Reproducción de vídeos .....	16
VIII. Observaciones finales .....	17



## I. Diagrama de clases del dominio

Adjuntamos el diagrama de clases propuesto por el profesorado y utilizado en el desarrollo de nuestra práctica. La única modificación respecto al diagrama indicado es que la clase AppVideo contiene un agregado de los filtros del sistema para facilitar al usuario su uso (*permitiéndole ver qué filtros hay y seleccionarlos*).



**Ilustración 1. Diagrama de clases del dominio utilizado**

Nótese que ignoramos el diagrama de clases diseñado para la tarea parcial propuesta sobre este mismo, puesto que no es el utilizado en la práctica.

## II. Diagrama de secuencia para añadir un vídeo

Adjuntamos a continuación el diagrama realizado en *MagicDraw*, mostrando la secuencia de invocación de cada una de las clases involucradas para añadir vídeos a una lista.

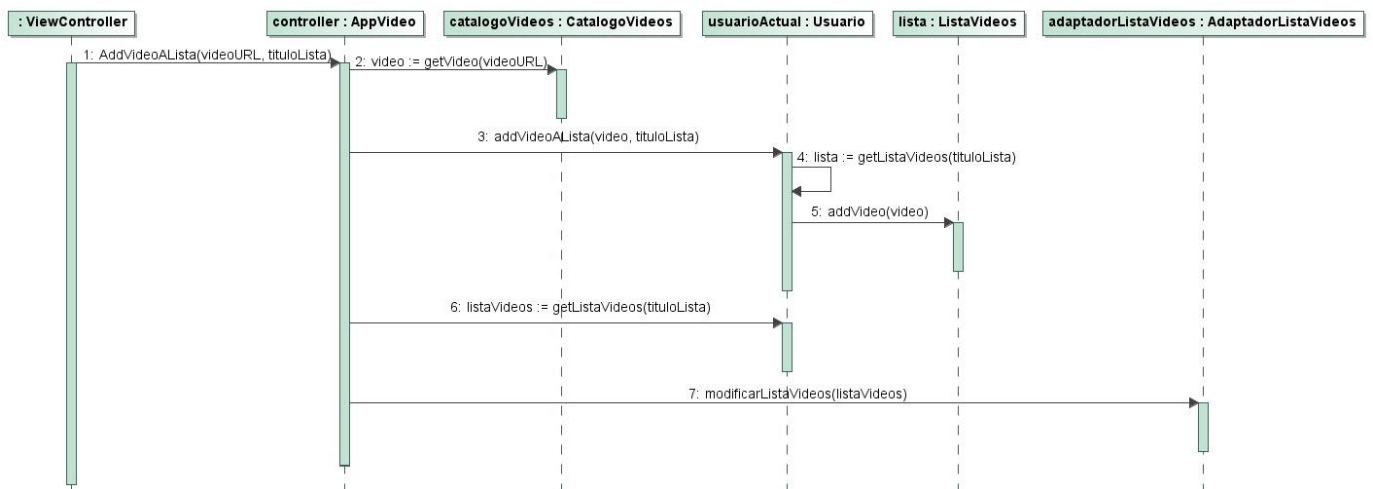


Ilustración 2. Diagrama de secuencia para añadir un vídeo

Nótese que en el diagrama no se visualiza el manejo de errores y situaciones inesperadas (*esto es, que el vídeo o la lista de vídeos no exista o que alguno de los pasos no devolviese el resultado esperado*).

Nótese también que estas situaciones inesperadas no pueden producirse en la estructura de nuestro código puesto que los parámetros para la llamada de **AddVideoALista** del controlador son siempre válidos, al ser extraídos del propio controlador. No obstante, dejamos las comprobaciones en cuestión por completitud de código, pues consideramos más correcto evitar llamadas al método con URLs o listas inexistentes.

---

### III. Arquitectura de la aplicación

La arquitectura general de la aplicación está basada directamente en la arquitectura de tres capas estudiada en la asignatura (*Modelo-Vista-Controlador*) y el principio de «separación modelo-vista». Excluimos cualquier explicación de dicho patrón arquitectural dado que no es el propósito de esta documentación. No obstante, resaltamos que, en la jerarquía de directorios y paquetes de la aplicación, existen, además de cada una de las tres capas, un paquete para las clases relacionadas con la persistencia para proporcionar mayor legibilidad y separación. Asimismo, incluimos un directorio con todos los recursos visuales utilizados para el desarrollo de la interfaz gráfica.

Sobre las decisiones de diseño, destacamos fundamentalmente el uso de las librerías JavaFX y JFoenix (*explicada en el apartado de componentes de este mismo documento*) de Java para la GUI. Asimismo, a diferencia de Swing, donde los elementos de la interfaz creados son añadidos estrictamente como código Java, el uso de la librería JavaFX y la herramienta visual JavaFX Scene Builder nos permite generar ficheros «FXML» que contienen la jerarquía de ventanas y elementos de la aplicación en un lenguaje de marcas similar a XML, separando de manera efectiva el control de la vista de su creación.

Este fichero FXML (*en nuestro caso, «Root.fxml»*) se carga en la inicialización de la aplicación junto a la hoja de estilo utilizada, en formato .css, y el controlador de la vista (*especificado dentro del fichero FXML*).

En cuanto a la estructura de nuestra GUI, destacamos el uso de un único fichero FXML (*bajo recomendación del profesorado*) para representar todas las ventanas indicadas en la especificación de la práctica. Esto facilita su implementación, reduce la complejidad y número de archivos, pero aumenta drásticamente el tamaño del controlador de la vista y no resulta de gran escalabilidad.

De este modo, la aplicación consta, en su nodo raíz (*a nivel de ventanas*), de un *BorderPane* que, en su región central, contiene un *StackPane* con cada una de las ventanas (*Explorar, Mis Listas, Recientes, etc...*) de la aplicación, inicialmente invisibles. Para “abrir” cualquiera de ellas, simplemente volvemos invisible la ventana que estuviese mostrada, traemos la deseada al frente y la volvemos visible.

Como matices adicionales de elección de diseño, subrayamos la inclusión de la ventana «Nueva Lista», indicada en la especificación, dentro de la ventana de «Mis listas». Consideramos oportuno este cambio puesto que los elementos de cada una de las ventanas son casi idénticos (*en cuanto a la visualización de listas y vídeos*). En vez de duplicarlos, por la falta de escalabilidad que hemos indicado antes, los fusionamos. Esta modificación queda reflejada en la capacidad de manejo total de las listas del usuario en una única ventana, en vez de separar la creación, edición y borrado de la reproducción.

Asimismo, incluimos una ventana adicional llamada «Mi perfil» que permite la modificación y consulta de la información del usuario actual de manera más extensa que la pedida en la especificación.

---

## IV. Patrones de diseño utilizados

Detallamos, en diferentes sub-apartados, cada uno de los patrones de diseño utilizados en el desarrollo de la práctica.

### 4.1 – Patrón Factoría Abstracta

---

El uso de este patrón reside en la creación de cada uno de los adaptadores DAO utilizados para el manejo de la persistencia. Mediante la clase *TDSFactoriaDAO* conseguimos desacoplar cada uno de los adaptadores específicos, ofreciendo una interfaz genérica de adaptadores en su lugar.

### 4.2 – Patrón Adaptador

---

Las clases adaptadoras indicadas en el subapartado anterior son una aplicación directa del patrón estructural adaptador. Su uso nos permite transformar la interfaz del servicio de persistencia utilizado en una interfaz genérica para el programador, ofreciendo mayor compatibilidad entre el servicio y la aplicación.

### 4.3 – Patrón Singleton

---

Encontramos el uso de este patrón en la creación de diversos elementos de la aplicación, como los catálogos de usuarios y vídeos, los adaptadores previamente explicados, la factoría DAO, y la clase controladora de la aplicación (*AppVideo*). Con este patrón garantizamos el acceso de una única instancia para cada una de estas clases y evitamos el uso de variables globales.

### 4.4 – Patrón Fachada

---

El controlador de nuestra aplicación es un ejemplo claro del uso de este patrón. Con él, conseguimos proporcionar una única interfaz que combina el manejo de la capa de negocio y la capa de persistencia. De este modo, el usuario no necesita, por ejemplo, para operaciones como crear una lista de vídeos, trabajar con el catálogo y los adaptadores involucrados. En su lugar, proveemos un único método, *crearListaVideos*, abstrayéndonos a un nivel significativamente más alto.

---

## V. Componentes utilizados

Destacamos el uso de los siguientes componentes en el proceso de desarrollo de la aplicación:

### 5.1 – Componentes de la librería JFoenix

---

Tal y como hemos indicado en el apartado de la arquitectura de la aplicación, la mayoría de los elementos visuales de nuestra aplicación son de la librería JFoenix. Puesto que, de manera general, estos elementos son en esencia los mismos que los de JavaFX pero con un aspecto visual minimalista y animado, únicamente puntualizamos los de mayor interés o peculiaridad. Estos son:

#### ❖ JFXDatePicker

- Este componente es el de nuestra elección para la introducción de la fecha de nacimiento en la ventana de registro. Aparte de proporcionar un panel bastante sofisticado para elegir una fecha, permite fácilmente obtener un objeto del tipo *LocalDate* con el método *getValue*.

#### ❖ JFXMasonryPane

- Utilizamos este tipo de panel para la visualización de las miniaturas de los vídeos en las ventanas de «Explorar» y «Mis Listas». Su funcionamiento es similar a un *FlowPane*, con el *bonus* añadido de ofrecer una animación cada vez que se añade un elemento o se redimensiona la ventana.

#### ❖ JFXNodesList

- Este componente proporciona un menú desplegable animado en forma de botones (*generalmente circulares*). Lo utilizamos en la ventana de «Mis Listas» para organizar cada una de las funcionalidades disponibles sobre una lista (*borrar, modificar, añadir, etc...*).

### 5.2 – Componente Buscador de Vídeos

---

Este componente es el que hemos desarrollado siguiendo las instrucciones y código fuente proporcionado para la creación de un componente JavaBean. Puesto que las propias instrucciones contienen una explicación extensa sobre el componente, nos limitaremos únicamente a indicar su uso en nuestra aplicación.

La clase que importa el componente es *ViewController*, la clase controladora de la GUI. Esta, almacena la clase controladora, *AppVideo*, como oyente. Cuando se presiona el botón que hemos proporcionado para cargar vídeos (*dentro de la ventana de «Mi perfil»*), se pide al usuario un fichero y se notifica al oyente, y este se encarga de procesar y añadir los vídeos cargados por el componente.



---

### 5.3 – Biblioteca iTextPDF

---

Para la implementación de la funcionalidad Premium de generar un archivo .pdf con las listas de vídeos del usuario, utilizamos el componente *PdfWriter*, dentro de la clase controladora. No indagamos en profundidad en las diferentes opciones de esta herramienta, limitándonos únicamente a crear el PDF en un formato sencillo.

### 5.4 – Componente VideoWeb

---

Para la reproducción de los vídeos de la aplicación utilizamos el componente proporcionado en los recursos de la asignatura. Como su explicación y forma de uso ya está explicada, únicamente matizamos su implementación en nuestro código.

Todos los vídeos se muestran dentro de una ventana emergente del tipo *JFXDialog*. Esta ventana contiene todos los elementos de la especificación, como las etiquetas y las reproducciones del vídeo. Parametrizamos la creación de este diálogo en un método auxiliar llamado *showVideoDialog* para la reutilización del código en las diversas ventanas donde se puede reproducir un vídeo.

<h2>VI. Tests unitarios</h2>
------------------------------

Incluimos en la entrega de la práctica, dentro del paquete *test*, dos clases Test sobre la capa del modelo, concretamente para las clases Usuario y ListaVideos.

## VII. Manual de usuario

Desglosamos el manual de usuario para cada una de las ventanas disponibles de la aplicación, proporcionando una descripción de los elementos de estas y sus funcionalidades:

### 7.1 – Log in

The screenshot shows a web application window titled 'Log in'. On the left is a sidebar with five items: 'Log in' (with a key icon), 'Explorar' (with a magnifying glass icon), 'Mis listas' (with a list icon), 'Recientes' (with a clock icon), and 'Premium' (with a diamond icon). The main content area has two input fields: 'Usuario' and 'Contraseña'. Below these is a red button labeled 'LOG IN'. Under the 'LOG IN' button is a link that says '¿No tienes cuenta? Regístrate'. A blue arrow points from the text 'Acceso a la ventana de registro' to this link.

Ilustración 3. Ventana de *login* de la aplicación

La primera ventana que se carga cuando se enciende la aplicación es la ventana de *login* o autenticación. Desde esta, el usuario puede identificarse con su cuenta, introduciendo su nombre de usuario y contraseña. Si alguno de los campos está vacío o la combinación es inválida, el sistema notificará de ello mediante un texto o ventana de emergente.

Si el usuario no dispone de una cuenta, puede registrarse en el sistema a través de la ventana de registro, accesible haciendo click sobre el texto de «¿No tienes cuenta? Regístrate».

## 7.2 – Registro

The screenshot shows a registration window with a sidebar on the left containing the following options: **Log in**, **Explorar**, **Mis listas**, **Recientes**, and **Premium**. The main content area includes the following fields and buttons:

- \*Usuario** (required)
- \*Contraseña** (required)
- \*Repetir contraseña** (required)
- \*Nombre** (required)
- Apellidos**
- \*Fecha de nacimiento** (required, with a calendar icon)
- Email**
- Registrar** button
- Cancelar** button

Two blue arrows point to the text **Vuelta a la ventana de login** at the bottom of the window. One arrow originates from the **Log in** button in the sidebar, and the other originates from the **Cancelar** button.

Ilustración 4. Ventana de registro de la aplicación

Esta ventana, accesible tal y como se indica en el apartado anterior, ofrece al usuario la interfaz necesaria para registrar una cuenta en el sistema. Todos los campos marcados con un asterisco son obligatorios, y, si no se introducen, el sistema notificará al usuario de ello. Asimismo, se debe introducir la misma contraseña dos veces. No dudamos de la capacidad intelectual del usuario, únicamente le instamos a no equivocarse. Para volver a la ventana de *login*, basta con presionar el botón «Cancelar» o volviendo a presionar el botón de «Log in» del panel lateral.

## 7.3 – Recientes



Ilustración 5. Ventana de recientes de la aplicación

Tras un registro o autenticación satisfactoria, el sistema abrirá de manera predeterminada esta ventana. Nótese, no obstante, que también puede ser accedida desde el panel lateral.

En esta ventana, el usuario dispondrá de dos listas de reproducción; una lista de los cinco vídeos reproducidos más recientemente por el usuario, y, si el usuario es premium, una lista con los diez vídeos más reproducidos en todo el sistema.

Tanto una lista como la otra (*si tiene la funcionalidad premium activada*) puede ser reproducida mediante el botón indicado a la derecha del título. Al ser presionado, el sistema preguntará, mediante una ventana emergente, el intervalo de tiempo entre vídeo y vídeo. Una vez introducido el intervalo, se mostrarán los vídeos uno detrás de otro [en la ventana de reproducción diseñada](#). Si el usuario cierra la ventana manualmente, se interrumpirá la reproducción de la lista. Asimismo, haciendo doble click sobre cualquiera de los vídeos lo reproducirá también.

## 7.4 – Mi perfil



Ilustración 6. Ventana de mi perfil de la aplicación

Esta ventana es accesible desde el panel lateral de la aplicación y, mientras que el usuario esté autenticado, sustituirá la ventana de log in. Por tanto, si se desea volver a acceder a las ventanas de registro y log in, será presionando el botón de «Log out».

Dentro de esta ventana, se mostrarán algunos de los datos del usuario, y, se le ofrece, aunque no forme parte de la especificación, la opción de actualizarlos y/o modificarlos en el sistema. Así mismo, dispone de un botón «Cargar vídeos» para, tal y como se indica, cargar vídeos. El sistema mostrará una ventana emergente instando al usuario elegir un fichero en formato .XML con los vídeos a introducir. Si el fichero es erróneo, no se elige o no puede abrirse, el sistema lo indicará mediante un mensaje de error.

## 7.5 – Explorar

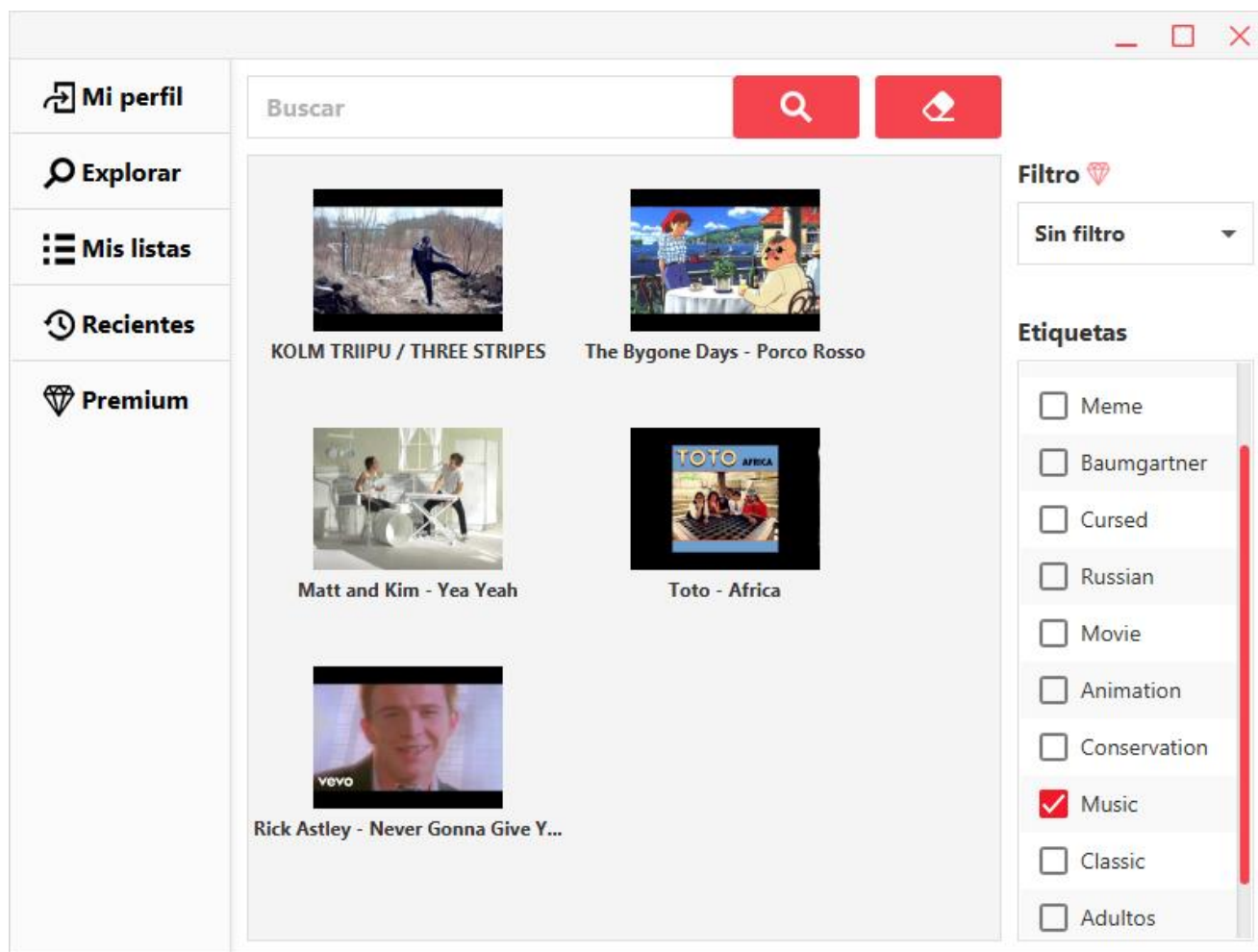


Ilustración 7. Ventana de explorar de la aplicación

Esta ventana contiene la interfaz y funcionalidad necesaria para la búsqueda y reproducción de los vídeos disponibles en el sistema. A través de la barra de búsqueda situada en la parte superior, el usuario podrá introducir títulos de vídeos o partes de ellos y el sistema devolverá todos los vídeos que contengan (*siendo case insensitive*) ese título. De este modo, si no se introduce ningún título, el sistema devolverá todos los vídeos disponibles.

Presionando el botón de búsqueda mostrará los vídeos que contengan el título buscado. Si ya hubiese una búsqueda mostrada, se limpiará y se cargará el nuevo resultado. De manera paralela, el botón de limpieza vaciará la búsqueda actual, si la hay.

Asimismo, el usuario podrá filtrar la búsqueda de vídeos mediante las etiquetas disponibles en el panel lateral derecho de la ventana. Marcando cualquiera de ellas la añadirá a la lista de etiquetas utilizadas para la búsqueda. Al realizar una búsqueda con etiquetas, se buscarán aquellos vídeos que contengan todas las etiquetas marcadas.

Si el usuario es premium, podrá también aplicar cualquiera de los filtros disponibles en el sistema. El filtro elegido se guardará entre sesiones y se aplicará únicamente a las búsquedas de la ventana de explorar. Explicamos, brevemente, cada uno de los filtros disponibles:

**Populares** muestra únicamente los vídeos que tengan más de 5 visitas totales. **En mis listas** muestra únicamente aquellos vídeos que estén en alguna de las listas creadas por el usuario. Finalmente, el filtro **Menores de edad** muestra aquellos vídeos que no tengan la etiqueta “Adultos” si el usuario es menor de 18 años.

## 7.6 – Mis listas



Ilustración 8. Ventana de mis listas de la aplicación

Ofrecemos, en esta ventana, toda la interfaz necesaria para la manipulación completa de listas de vídeos. Destacamos dos elementos principales; el panel lateral con la lista de vídeos seleccionada y las opciones del usuario, y el panel central para la búsqueda de vídeos, similar a la ventana de explorar.

En el panel lateral, el usuario podrá seleccionar cualquiera de las listas que haya creado a través del menú desplegable. Al seleccionar una lista, se mostrarán sus vídeos. De manera predeterminada, haciendo doble click sobre cualquier vídeo lo reproducirá en el formato usual. A la derecha del menú

de las listas, el usuario dispone de un botón desplegable con toda la funcionalidad disponible para la manipulación de las listas.

Explicamos, una a una, las opciones de este botón desplegable:



#### – Editar una lista de vídeos

Al presionar el botón con este icono, se habilitará el panel central para la búsqueda de vídeos y se deshabilitará el resto de botones del panel lateral. En este modo de edición, el usuario podrá buscar vídeos y, haciendo doble click sobre ellos, añadirlos a la lista actual. No se permite, por elección arbitraria, tener vídeos repetidos en la lista, a pesar de que el código sí lo soporta. Haciendo doble click sobre cualquier vídeo ya en la lista, lo eliminará de esta. Para salir de este modo, basta con volver a hacer click sobre el botón de edición.



#### – Reproducir una lista de vídeos

De manera idéntica al botón con el mismo icono de la ventana de «Recientes», al presionar el botón con este icono se preguntará al usuario, a través de una ventana emergente, el intervalo, en segundos, entre vídeo y vídeo. A continuación, se reproducirán cada uno de los vídeos de la lista con el intervalo especificado hasta finalizar la lista o hasta que el usuario cierre manualmente el reproductor de vídeos.

Nótese que, por limitaciones de la aplicación, el reproductor de vídeos tarda aproximadamente un segundo en abrirse, el cual queda contabilizado dentro de este intervalo. No consideramos posible solventarlo puesto que es únicamente la interfaz gráfica la que tarda en cargarse.



#### – Eliminar una lista de vídeos

El botón con este icono sirve para eliminar la lista de vídeos seleccionada actualmente. Al presionarlo, se le preguntará al usuario si está seguro de la operación (*con la opción de no volver a mostrar ese mensaje*).



#### – Crear una lista de vídeos

El botón con este icono permite crear una nueva lista. Al presionarlo, el usuario podrá introducir, en el mismo panel lateral, el título de la nueva lista a crear. Junto a la región para introducir el título, dispondrá de un botón para confirmar la acción. Si no se introduce ningún título de la lista, lo interpretamos como que el usuario no quiere crear ninguna lista.



#### – Generar archivo .PDF con todas las listas de vídeos

Esta funcionalidad, indicada por el icono con forma de diamante, es exclusiva para usuarios premium. Ofrecemos la posibilidad de generar un archivo .PDF en el directorio donde se encuentre la aplicación con todas las listas de vídeos (*y sus respectivos vídeos*) del usuario. Para ello, se preguntará al usuario el título del fichero a crear (*con un máximo de 50 caracteres*). Independientemente del resultado, se notificará mediante una ventana emergente.



## 7.7 – Premium

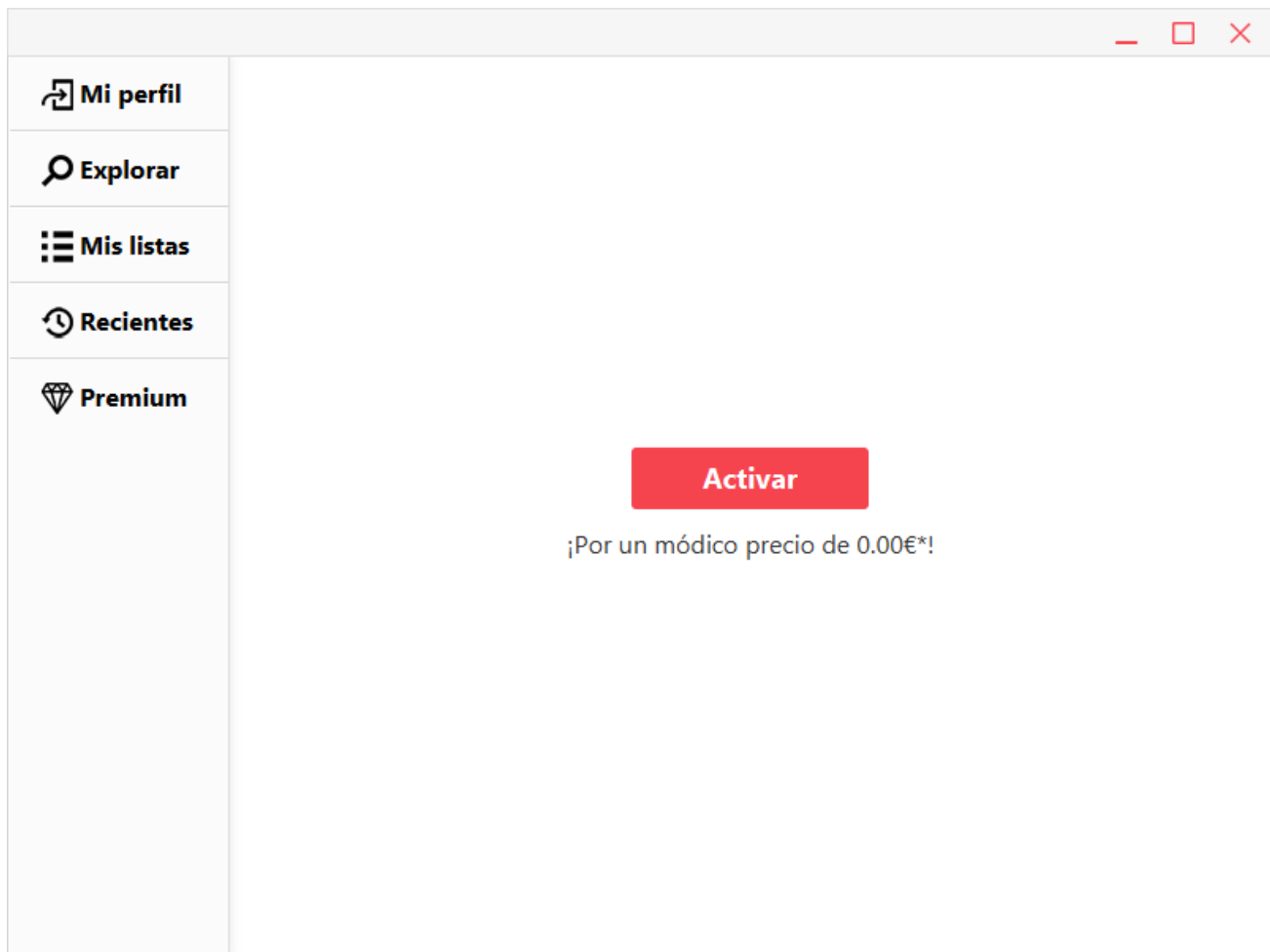


Ilustración 9. Ventana de premium de la aplicación

Esta ventana, en comparación con las anteriores, es de una complejidad conceptual y visual minúscula. Únicamente incluimos un botón para activar la funcionalidad premium. Una vez activada, no se puede dar de baja. Porque es gratis.

## 7.8 – Reproducción de vídeos

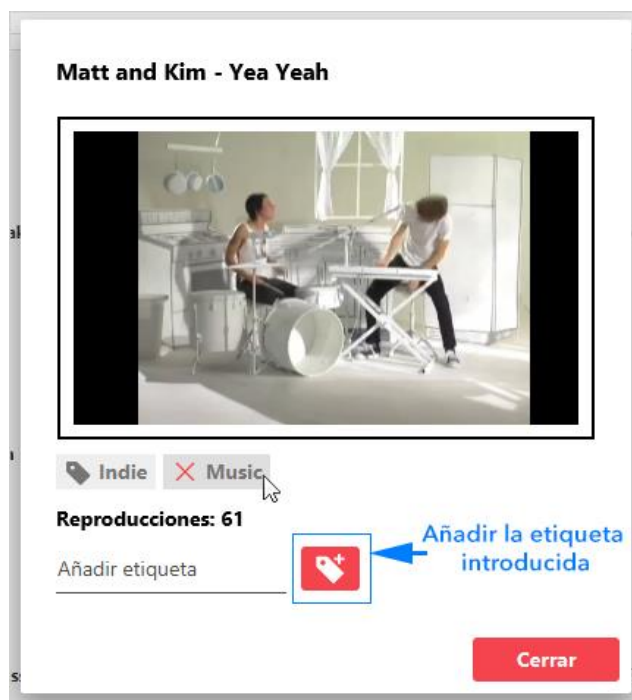


Ilustración 10. Ventana emergente de reproducción de la aplicación

La interfaz para la reproducción de vídeos se muestra en forma de ventana emergente. Esta incluye, en orden vertical, el título del vídeo introducido al sistema, un reproductor embebido de YouTube, un listado con las etiquetas del vídeo, el contador de reproducciones total y, finalmente, un campo de texto para añadir etiquetas al vídeo mediante el botón situado a su derecha.

Asimismo, permitimos eliminar etiquetas de un vídeo haciendo click sobre ellas (*tal y como se muestra en la ventana*). Para cerrar el reproductor es suficiente con presionar el botón correspondiente.

---

## VIII. Observaciones finales

Consideramos esta práctica muy nutritiva en términos de aprendizaje para el desarrollo y organización de un proyecto de tamaño considerable (*en comparación con el resto de asignaturas del grado hasta el momento*). Asimismo, el uso de un sistema de control de versiones (*en nuestro caso, Git*) tiene un impacto muy positivo sobre la dinámica del trabajo, y, de igual importancia, ha sido una práctica sumamente entretenida.

En cuanto al tiempo empleado para la práctica, incluimos un fichero con un seguimiento de horas (*aproximado*) de cada uno de nosotros, desglosado por días y trabajo realizado. Estimamos un total de **105** horas entre los dos miembros.