

# Sistemas Multimedia: PaintCompleto

---

Documentación sobre la Práctica Final de la  
Asignatura

Ramón Sánchez García  
chentaco@correo.ugr.es

[Nota: Aprobado por Evaluación continua, pero realizo la documentación porque he añadido algunas cosas a la práctica final para ver si es posible sacar algo más nota.]

## Índice

1. Introducción.....	2
2. Desarrollo de la aplicación .....	2
3. Planteamiento de requisitos de la aplicación .....	5
3.1. Gráficos .....	6
3.2. Imagen .....	20
3.3. Sonido.....	38
3.4. Video.....	41
3.5. Interfaz.....	44
4. Comentarios y conclusiones .....	51
4.1. ¿Qué he aprendido? .....	51
4.2. ¿Cuál es el futuro de mi aplicación?.....	52
4.3. Otros comentarios.....	52
5. Bibliografía.....	53

## 1. Introducción

El presente documento recoge información sobre el desarrollo de la aplicación de **PaintCompleto** para la asignatura **Sistemas Multimedia de la Escuela Técnica Superior de Informática y Telecomunicaciones** de la **Universidad de Granada**. En dicho documento recogeré tanto la explicación de cómo la he realizado dicha aplicación, como una explicación de las decisiones tomadas.

El documento estará estructurado de la siguiente forma:

- En primer lugar, se explica en qué consiste mi aplicación y el aspecto que presenta.
- Después se pasará a explicar que hace la aplicación, explicando grosso modo lo que se muestra cuando se inicia el programa. También hablo un poco de qué he aprendido durante el curso y como lo he usado para desarrollar mi aplicación.
- Finalmente haré un análisis de requisitos de cada uno de los bloques importantes de dicha aplicación, distribuyendo la explicación de dichos bloques con el siguiente esquema:
  - Análisis de dicho bloque
    - ¿Qué herramientas ofrece Java para implementarlo?
    - ¿Qué problemas aparecen?
    - ¿Cómo lo he solucionado?
  - Diseño de dicho bloque
  - Implementación (esta parte está recogida en el propio **Javadoc** entregado junto al código de la aplicación).
- Adicionalmente, se recojo mis opiniones sobre el desarrollo de la aplicación en el apartado de “conclusiones” y finalizo con una bibliografía sobre sitios que he utilizado a lo largo del cuatrimestre para la implementación de mi aplicación.

Todo el contenido de este documento, junto el propio código de la aplicación, está sujeta al derecho de autor propio, está prohibida su copia, distribución y en general, uso sin permiso del propio autor:

© **Ramón Sánchez García, todos los derechos reservados, Granada, 2018**

Se tomarán medidas a cualquiera que incumpla alguna de las acciones antes descritas. Asimismo la aplicación está localizada en la base de datos de la Universidad de Granada, por lo que cualquier copia detectada llevará al suspenso de aquella o aquellas personas que lo copien y lo presenten como suyo.

## 2. Desarrollo de la aplicación

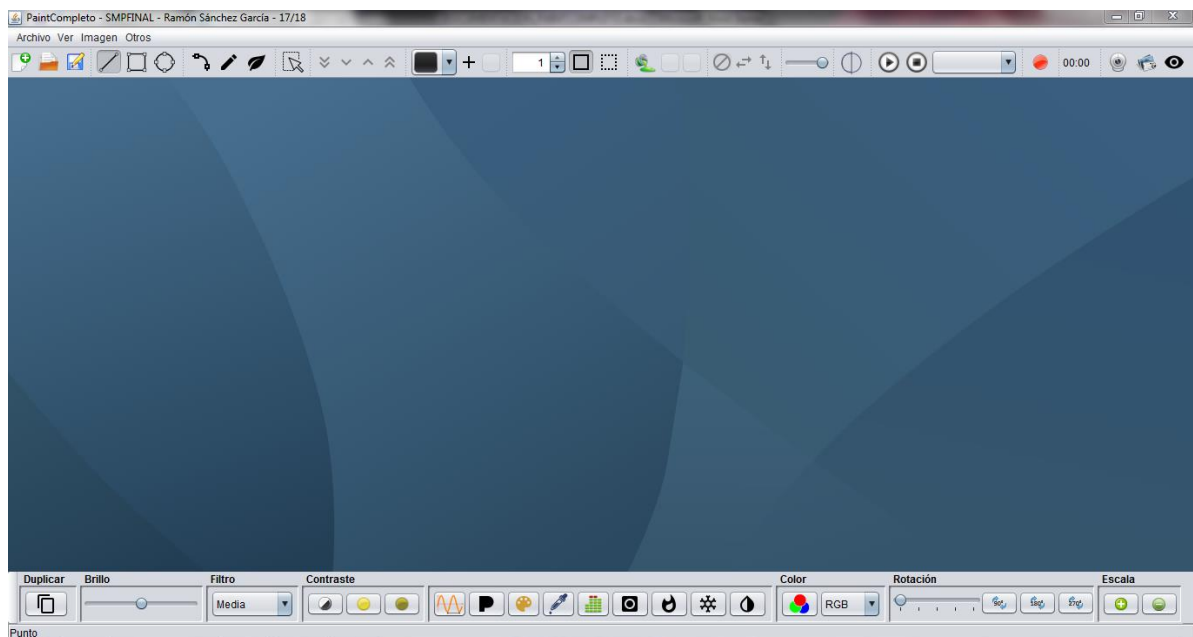
El objetivo de esta asignatura ha sido el de aprender y trabajar con varios elementos del mundo multimedia. Entre ellos, hemos usado gráficos (para pintar figuras, como Rectángulos, líneas o elipses, y sus correspondientes atributos), imágenes (manipulado su tamaño, brillo,

color, etc.), sonido y video. Para ello, hemos utilizado el lenguaje de programación Java, y varias de sus bibliotecas que nos han permitido jugar con estos elementos, y crear una especie de “Paint” multimedia, como el de Windows. Dicho Paint nos ha permitido realizar acciones de pintado, manipular imagen, audio y sonidos. La propia implementación de este programa ha sido a través de Java.

Para poder implementar y utilizar estas acciones, hemos aprendido también a implementar la interfaz correspondiente del programa, la cual se ha hecho gracias a la biblioteca que ofrece este lenguaje: Java Swing. Gracias a ella hemos implementado botones, entre otros elementos. Y si hablo de botones, he de hablar de otra de las cosas más importantes que hemos aprendido en el curso, que son los eventos, los cuales se activan cuando se produce una acción en nuestra aplicación y estará ligado a una serie de acciones que se tomarán cuando se produzcan estos eventos. Aunque nuestro entorno de desarrollo ofrece algunos por defecto, hemos visto que no siempre vamos a tener eventos con sus listeners disponibles, por lo que tendremos que ser nosotros quienes lo implementemos. De todo esto se hablará más adelante.

Hemos utilizado el IDE Netbeans para trabajar durante toda la asignatura y realizar nuestra aplicación. Nos ha permitido trabajar cómodamente con una interfaz gráfica que nos permitió añadir cómodamente elementos Swing a nuestra aplicación, entre otras cosas.

El aspecto de mi aplicación final es el siguiente:



Se distinguen varios elementos swing, como botones, deslizadores, menús desplegables, etc. Además recoge acciones de los cuatro bloques con los que hemos trabajado a lo largo del cuatrimestre. Las acciones que permite mi aplicación son las siguientes:

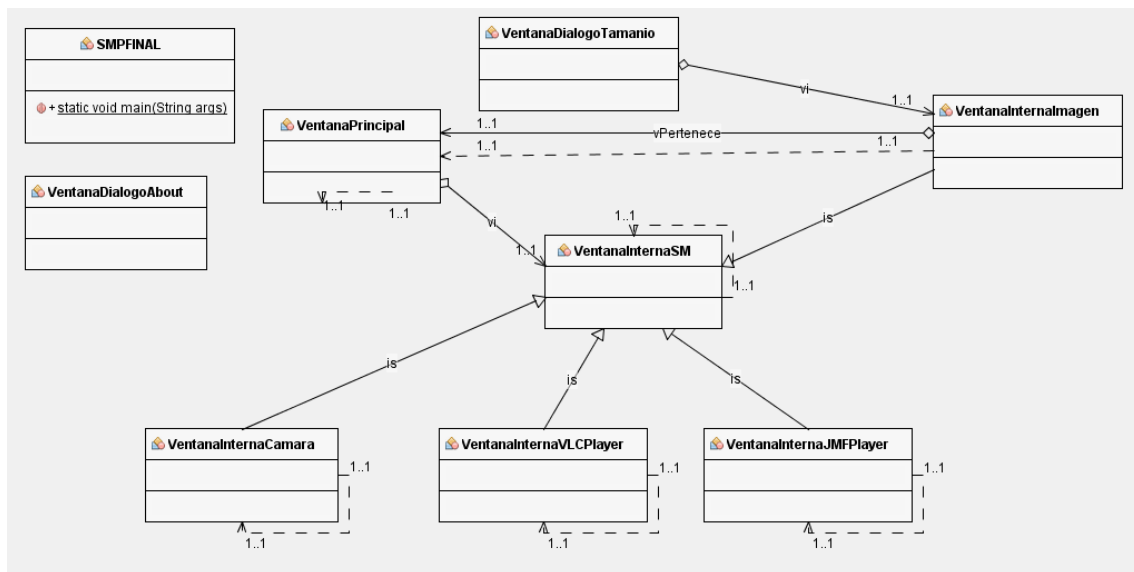
- Abrir, crear o guardar una imagen.
- Abrir audio, reproducirlo e incluso grabar un nuevo audio.

- Abrir video y webcam, de los cuales puedo tomar capturas de pantalla y, en el caso de la webcam, activar un sistema de vigilancia que hará capturas en caso que detecte movimiento.
- Modificar el tamaño o escalar dicha imagen o lienzo de la imagen.
- Pintar una de las figuras disponibles.
- Mover una figura pintada, ya sea por el lienzo, como su posición respecto a las otras del lienzo (detrás o delante del resto de figuras).
- Cambiar el atributo de la figura seleccionada o pintar una con los atributos seleccionados.
- Duplicar una imagen.
- Aplicar a una imagen diversos filtros programados.
- Rotar la imagen.
- Escalar la imagen.
- Obtener sus bandas, dependiendo del tipo de espacio de colores que tenga dicha imagen.

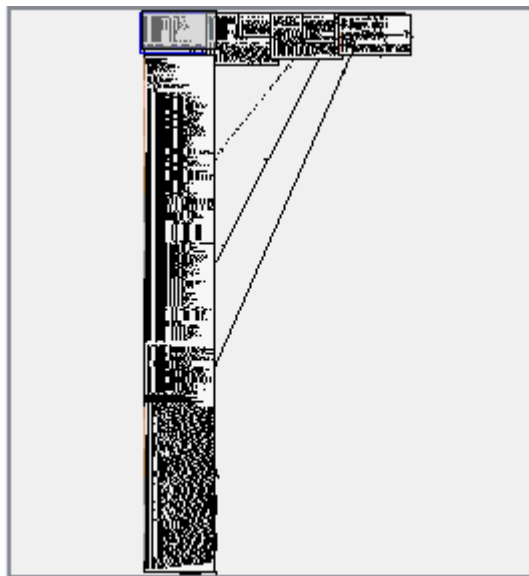
Para poder hacer todo esto posible, he utilizado todo lo que Java me ofrece. Sin embargo, como hemos visto a lo largo de la asignatura, las bibliotecas que ofrece Java, los métodos y demás, no son del todo correctos: les falta personalización. Y este ha sido otro de los objetivos de esa aplicación, el ofrecer nosotros clases propias que permitan realizar dicha personalización. Veremos más adelante que hemos usado esto en el caso de figuras, creando para ello figuras propias.

Mi aplicación se divide en los siguientes elementos:

- Ventana Principal: Será la que albergue todo lo relacionado con los botones, deslizadores y demás elementos swing. Para dibujar alguna figura o trabajar sobre una imagen, se encargará de abrir Ventanas Internas, de diferente tipo.
- Ventana Interna SM: Las Ventanas Internas que albergarán el lienzo donde pintaremos, la imagen que queremos editar, el video o incluso la webcam. Dichos tipos de Ventana Interna son clases hijas de este elemento:
  - Ventana Interna Imagen: Lienzo e Imagen.
    - Tendrá un atributo Lienzo2D con la información de los atributos del lienzo actual y/o de la figura seleccionada.
  - Ventana Interna Video: hay de dos tipos, la VLC y la JMF. Aunque están ambas implementadas, la aplicación usa la primera.
  - Ventana Interna Webcam.
- Ventanas de Dialogo. Hay dos, una primera asignada a la ventana de imagen, para cambiar el tamaño de dicho lienzo que se muestra en esta ventana interna; una segunda ventana de dialogo que se llama al pulsar el botón "Sobre el programa..." en el menú "Otros".



**NOTA:** Dado el tamaño del código y, por ende, de su diagrama UML, se han omitido el nombre de los métodos y atributos. Además no se van a incluir el diagrama de clases en esta documentación, por lo que para la consulta de dichos métodos, consulte el código que se entrega junto a esta documentación. Como prueba del tamaño, esta es la miniatura del diagrama de clases:



Todo esto solo es un vistazo preliminar de la aplicación. En el siguiente paso, entraré en profundidad. Dividiré en bloques cada una de las cosas que he aprendido y como las he incorporado a mi aplicación. Irán acompañadas de un análisis de requisitos funcionales.

### 3. Planteamiento de requisitos de la aplicación

En este apartado voy a analizar cada uno de los requisitos de mi aplicación. Para ello, divido la sección en cada uno de los siguientes bloques:

- RF1: Gráficos
- RF2: Imagen
- RF3: Sonido
- RF4: Video
- RF5: Interfaz

Para cada uno de estos requisitos, haré un análisis haciendo las preguntas que he comentado en el apartado “introducción”. Recordar de nuevo que el apartado “diseño” lo omitiré, dado a que ya hablo de como he diseñado cada clase en el “javadocs” que se ha entregado junto al código de esta práctica.

### 3.1. Gráficos

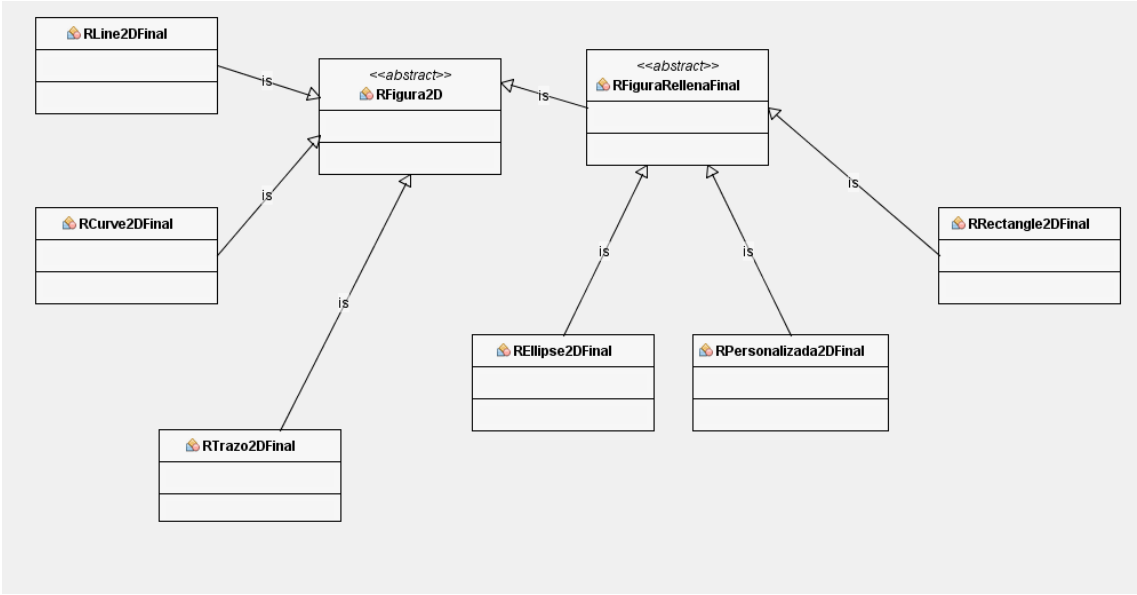


- Análisis:
  - ¿Qué ofrece? Para empezar, Java ofrece la clase Graphics, que permite pintar las figuras básicas, como la línea, el rectángulo, etc. El problema que ofrecía esta clase es que las figuras pintadas no se podían editar. Como vimos al principio de la asignatura, cuando creamos el primer Paint, podíamos crear figuras pero no editarlas: no podíamos moverlas, cambiar su tamaño, su posición, etc. (aunque el color y el formato del texto sí podían cambiar). Debido a esto, pasamos a usar la clase Graphics2D, la cual mejora bastante Graphics, haciendo que cada una de las figuras anteriores tenga una clase asociada, la cual tendrá unos métodos que nos permitirán editar las figuras que pintemos. Graphics2D hereda de Graphics. Además incorpora una nueva serie de atributos que podemos aplicar a nuestra figura, como el tamaño del trazo, el tipo, etc.

Para los atributos, Java ofrece una serie de bibliotecas para permitir editar como se ven mis figuras. Dichas bibliotecas y métodos me han servido para asignar los atributos a mis figuras. No he tenido que crear ninguna biblioteca a parte para los atributos.

- ¿Qué problemas aparecen? A pesar de que usamos Graphics2D y parece la idea, los atributos que tienen son compartidos por todas las figuras. Es decir, a pesar de que cada figura tiene una clase propia, no tiene atributos que almacenen de qué color es cada figura o qué trazo tienen, a pesar de que tengan métodos que lo modifiquen. El problema entonces que todas las figuras tendrán el mismo color, el mismo trazo, y cuando cambiemos una, cambiarán todas. ¿Cómo lo hemos arreglado? En mi caso he creado clases propias, cada una para la figura correspondiente. Para hacer esto, he creado una clase abstracta, llamada “RFigura2D” que contiene todos los atributos de mi geometría, los métodos típicos (de tipo abstracto) y el método Paint, que

será el mismo para todas las figuras, como digo. También añadí un atributo de tipo Shape, que se iniciará en sus clases hijas, dependiendo del tipo de figura que se cree. Dichas clases hijas serán las propias figuras, y sobre estas clases se sobrecargarán los métodos de la clase padre dependiendo del tipo de figura. Además he distinguido entre figuras rellenas y figuras no rellenas, para ello he creado otra clase que hereda sobre la clase padre, a la que he llamado “RFiguraRellenaFinal”, que sobrecargará el método que nos dice si está rellena o no dicha figura. Para más información sobre el código, consulte el “javadocs”. Dichas clases propias tienen la siguiente jerarquía:



**Nota:** De nuevo, omito el nombre de los métodos y atributos por cuestión de espacio. Para consultar esta información, revise el código de la aplicación.

- **Diseño:** Como he comentado en el apartado anterior, he creado mis propias clases propias para cada una de las figuras. Dichas figuras tendrán cada una sus atributos, y podrán editarse si se seleccionan. Para realizar el pintado y editado de las figuras, utilizo botones que he implementado en la propia aplicación. Comentaré lo que hacen estos botones y como lo he diseñado mediante la documentación de “Requisitos funcionales” (haré lo mismo para todos los bloques):

○ RF 1.1. Pintar Línea


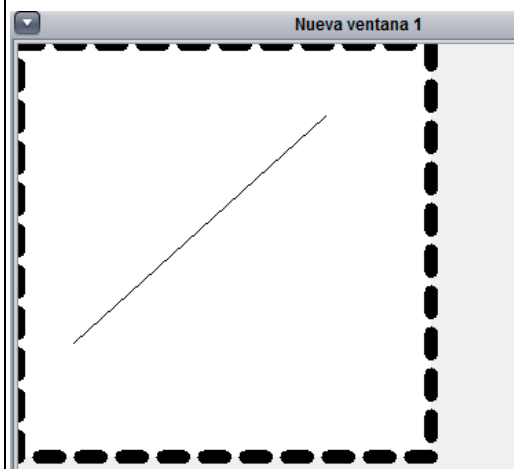
Número de requisito	RF 1.1.		
Nombre de requisito	Pintar Línea		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
Al seleccionar este botón, se podrá pintar una figura “Línea” en la ventana interna. Dicha			



figura tendrá los atributos que actualmente tenga la ventana principal. El primer punto se creará cuando se pulse con el ratón sobre la ventana interna, y el final cuando se suele el botón del ratón. Por defecto, estará seleccionada.



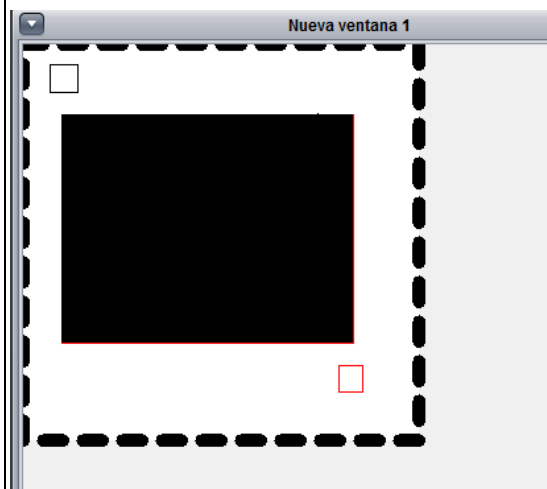
○ RF 1.2. Pintar Rectángulo

Número de requisito	RF 1.2.		
Nombre de requisito	Pintar Rectángulo		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional


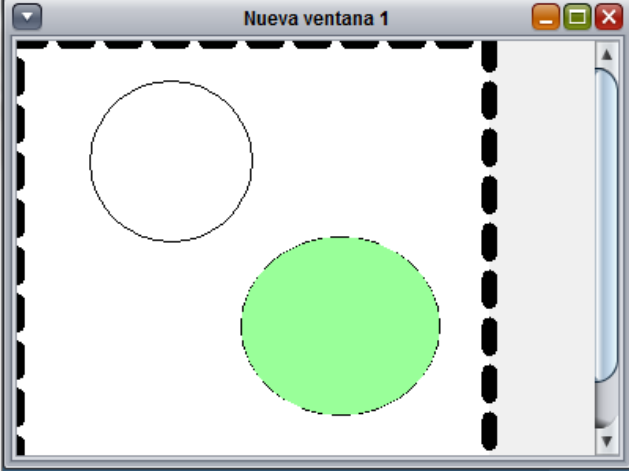
Descripción del requisito:




Al seleccionar este botón, se podrá pintar una figura “Rectángulo” en la ventana interna. Dicha figura tendrá los atributos que actualmente tenga la ventana principal. El primer punto se creará cuando se pulse con el ratón sobre la ventana interna, y el final cuando se suele el botón del ratón. Esta figura puede estar rellena.

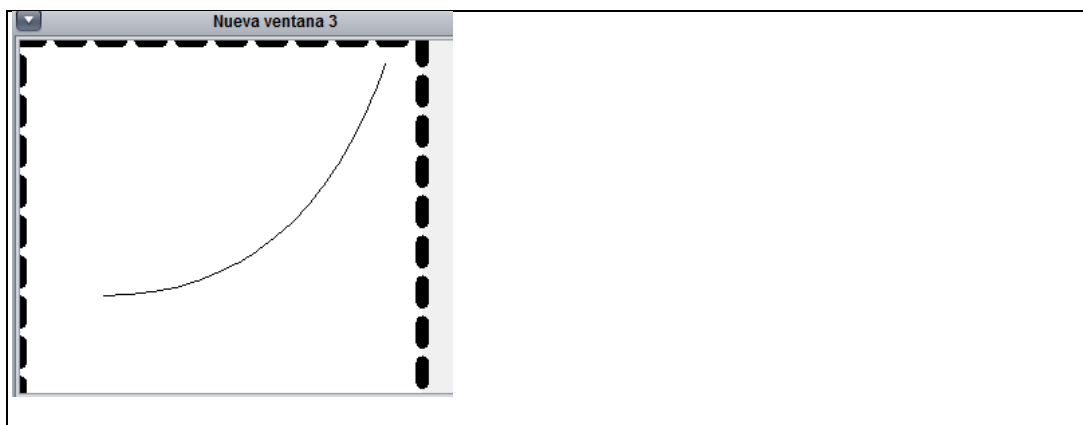


○ RF 1.3. Pintar Elipse



Número de requisito	RF 1.3.		
Nombre de requisito	Pintar Elipse		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Al seleccionar este botón, se podrá pintar una figura “Elipse” en la ventana interna. Dicha figura tendrá los atributos que actualmente tenga la ventana principal. El primer punto se creará cuando se pulse con el ratón sobre la ventana interna, y el final cuando se suelte el botón del ratón. Esta figura puede estar rellena.</p> 			

○ RF 1.4. Pintar Curva

Número de requisito	RF 1.4.		
Nombre de requisito	Pintar Curva		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Al seleccionar este botón, se podrá pintar una figura “Curva” en la ventana interna. Dicha figura tendrá los atributos que actualmente tenga la ventana principal. El primer punto se creará cuando se pulse con el ratón sobre la ventana interna, y el final cuando se suelte el botón del ratón. Tras dibujarla, tendrá que indicarse el punto de control de la curva, es decir, la curvatura de esta. Para ello usará de nuevo el evento de movimiento del ratón junto al clic izquierdo. Una vez que tengamos la curva deseada, lo soltamos.</p>			



○ RF 1.5. Pintar Trazo

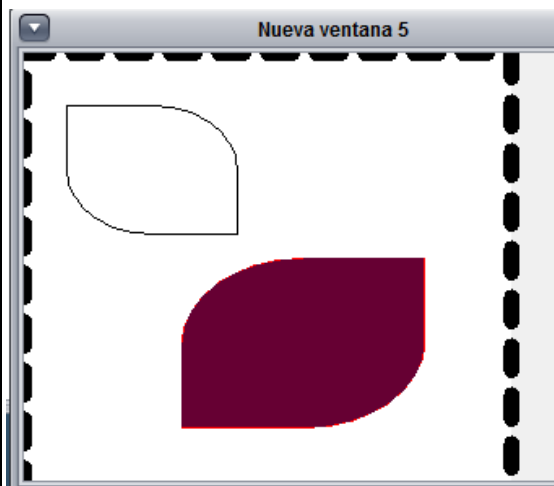
Número de requisito	RF 1.5.		
Nombre de requisito	Pintar Trazo		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Al seleccionar este botón, se podrá pintar una figura "Trazo" en la ventana interna. Dicha figura tendrá los atributos que actualmente tenga la ventana principal. Dibujará un trazo libre desde que se pulse el botón izquierdo del ratón hasta que se suelte, siendo el trazo dibujado mientras se mueva por la ventana interna y se mantenga el botón izquierdo pulsado.</p> 			

○ RF 1.6. Pintar Figura Personalizada: Hoja

Número de requisito	RF 1.6.		
Nombre de requisito	Pintar Figura Personalizada: Hoja		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			




Al seleccionar este botón, se podrá pintar una figura personalizada “Hoja” en la ventana interna. Dicha figura tendrá los atributos que actualmente tenga la ventana principal. El primer punto se creará cuando se pulse con el ratón sobre la ventana interna, y el final cuando se suelte el botón del ratón. Dicha figura puede ser rellena.



Nota: En “javadocs” está descrito como he diseñado esta figura.

○ RF 1.7. Seleccionar figura

Número de requisito	RF 1.7.		
Nombre de requisito	Seleccionar figura		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
Si hay alguna figura dibujada y este botón está seleccionado, permite seleccionar la figura más cercana al punto donde pulsemos. Dicha figura estará rodeada por un marco discontinuo. Tras seleccionarse, la ventana principal se actualizará con los atributos de dicha figura. Además permite modificar los atributos de la figura seleccionada. También puede moverse respecto a otras figuras de posición. Se deselectionará al pulsar sobre otra herramienta de figura.			

Nueva ventana 5

La forma del cursor cambia a una cruz con flechas cuando se mueva. Referencia a:

- RF 5.9. Visualización del Cursor

- RF 1.8. Posición de la figura respecto a otras

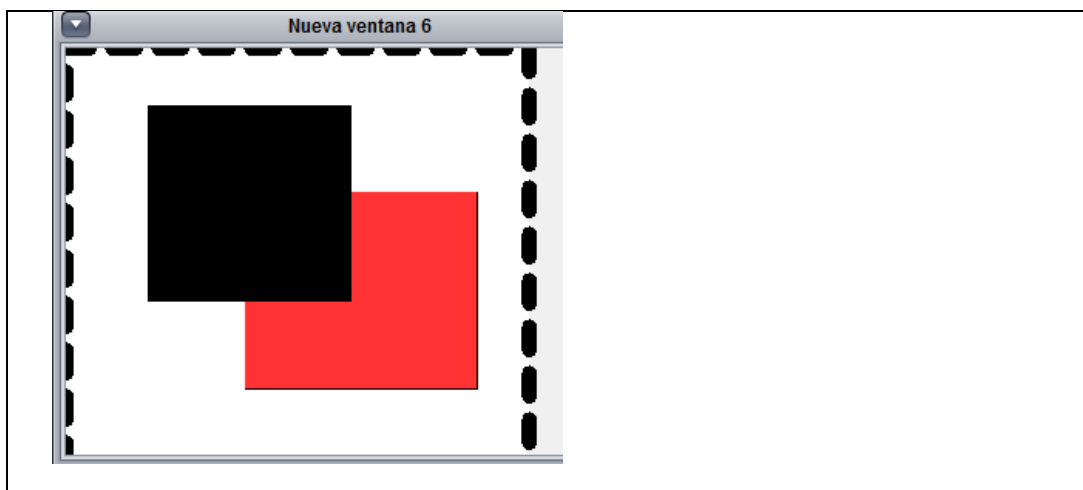
Número de requisito	RF 1.8.		
Nombre de requisito	Posición de la figura respecto a otras		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:

Si hay una figura seleccionada, permite moverla en respecto a otras. Puede estar al frente de todas o al fondo, o subir y bajar a su antojo. Si no puede, estará deshabilitado. Hace referencia a los siguientes requisitos funcionales:

- RF 1.8.1. Enviar al fondo
- RF 1.8.2. Mover atrás
- RF 1.8.3. Mover adelante
- RF 1.8.4. Enviar al frente

Nueva ventana 6



▪ RF 1.8.1. Enviar al fondo

Número de requisito	RF 1.8.1.		
Nombre de requisito	Enviar al fondo		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
Si puede, envía la figura una posición hacia el fondo del todo. Deshabilitado si no se puede o no hay ninguna figura seleccionada.			

▪ RF 1.8.2. Mover atrás

Número de requisito	RF 1.8.2		
Nombre de requisito	Mover atrás		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
Si puede, envía la figura una posición hacia atrás. Deshabilitado si no se puede o no hay ninguna figura seleccionada.			


▪ RF 1.8.3. Mover adelante

Número de requisito	RF 1.8.3.		
Nombre de requisito	Mover adelante		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
Si puede, envía la figura una posición hacia delante. Deshabilitado si no se puede o no hay ninguna figura seleccionada.			

▪ RF 1.8.4. Enviar al frente

Número de requisito	RF 1.8.4.		
Nombre de requisito	Enviar al frente		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
Si puede, envía la figura al frente de todas las demás. Deshabilitado si no se puede o no hay ninguna figura seleccionada.			

○ RF 1.9. Color del trazo


Número de requisito	RF 1.9.		
Nombre de requisito	Color del trazo		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Permite elegir el color del trazo de la figura seleccionada o que se va a dibujar. Para ello elige uno de los colores de la lista, o habilita el poder usar otros más. Hacer referencia a los requisitos:</p> <ul style="list-style-type: none"> <li>• RF 1.9.1. Menú con algunos colores</li> <li>• RF 1.9.2. Permitir más colores</li> <li>• RF 1.9.3. Más colores</li> </ul>			

▪ RF 1.9.1. Menú con algunos colores

Número de requisito	RF 1.9.1.		
Nombre de requisito	Menú con algunos colores		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			

Muestra y permite seleccionar algunos de los colores por defecto que he añadido para el trazo. Cambia el color del trazo de la figura seleccionada o por dibujar. Si el botón permitir colores no está pulsado, estará habilitado.

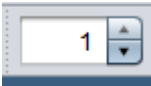
▪ RF 1.9.2. Permitir más colores

Número de requisito	RF 1.9.2.		
Nombre de requisito	Permitir más colores		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Botón que permite la selección de otros colores adicionales de los que ya hay en el requisito anterior. Si está presionado, deshabilitará la lista de colores y habilitará el botón de más colores. En caso contrario será a la inversa.</p>			

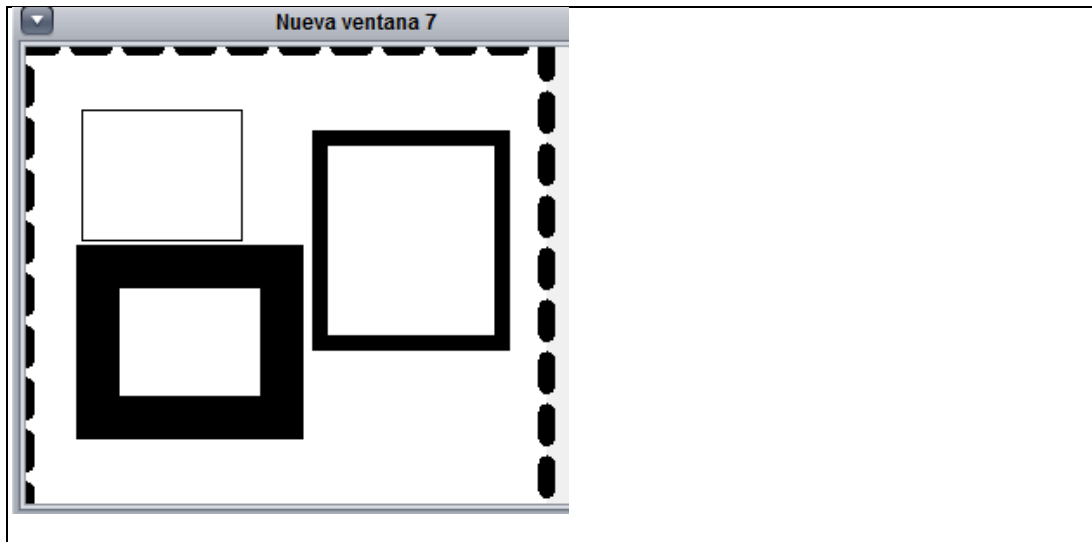
▪ RF 1.9.3. Más colores

Número de requisito	RF 1.9.3.		
Nombre de requisito	Más colores		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
<p>Botón que abre una ventana con más colores para el trazo de la figura que voy a dibujar o ya esté dibujada. Se habilitará si el botón “más colores” (RF 1.9.2) está pulsado.</p>			

○ RF 1.10. Tamaño del trazo

Número de requisito	RF 1.10.		
Nombre de requisito	Tamaño del trazo		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Elige o modifica el tamaño del trazo de la figura. Tiene que crear un objeto del tipo BasicStroke con dicho valor. El valor representa el tamaño del trazo, el cual, su valor mínimo será 1 y el máximo será 100.</p>			





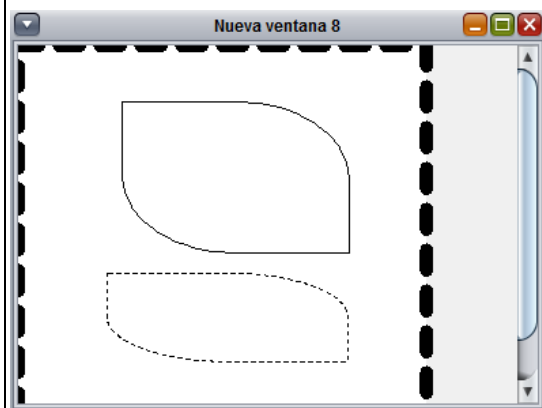
○ RF 1.11. Tipo de trazo

Número de requisito	RF 1.11.		
Nombre de requisito	Tipo de trazo		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:

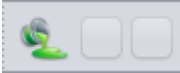


Elige o modifica el tipo de trazo que tendrá la figura. Para ello crea un nuevo objeto BasicStroke dependiendo del tipo de trazo que seleccionemos. No se pueden seleccionar los dos a la vez. Si se selecciona uno, el otro se deselectiona.



Nota: No he encontrado ningún método que me permita modificar el trazo sin tener que crear un nuevo objeto, ya que tendré que crear un objeto "BasicStroke" nuevo con el trazo y el tipo de trazo cada vez que se llame, al no tener un método que lo modifique. Esto ocurre tanto en el "tamaño" como en el "tipo" del trazo.

○ RF 1.12. Relleno

Número de requisito	RF 1.12.		
Nombre de requisito	Relleno		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Habilita el relleno de la figura. Dicho relleno puede ser simple o de degradado. Al seleccionar el botón de relleno, se habilitarán las opciones para seleccionar el color de relleno y el tipo. Hace referencia a los requisitos funcionales:</p> <ul style="list-style-type: none"> <li>• RF 1.12.1. Color de relleno</li> <li>• RF 1.12.2. Color de relleno secundario</li> <li>• RF 1.12.3. Tipo de relleno <ul style="list-style-type: none"> <li>• RF 1.12.3.1. Relleno simple</li> <li>• RF 1.12.3.2. Relleno horizontal</li> <li>• RF 1.12.3.3. Relleno vertical</li> </ul> </li> </ul>			

▪ RF 1.12.1. Color de relleno


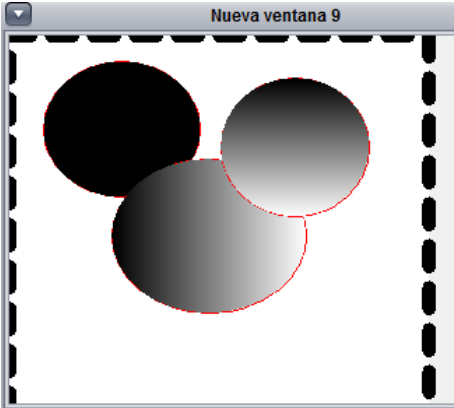
Número de requisito	RF 1.12.1.		
Nombre de requisito	Color de relleno		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
<p>Se habilitará si está presionado el botón de “Relleno”. Abrirá una ventana con diferentes colores para seleccionar el color de relleno de la figura. También actúa como el color primario del degradado, en caso de que se seleccione este tipo de relleno.</p>			

▪ RF 1.12.2. Color de relleno secundario


Número de requisito	RF 1.12.2.		
Nombre de requisito	Color de relleno secundario		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
<p>Abre una ventana para seleccionar el color secundario del relleno. Útil para cuando mi relleno tenga un degradado, horizontal o vertical. Este será el segundo color de dicho degradado. Se habilitará cuando el relleno sea de este tipo.</p>			

▪ RF 1.12.3. Tipo de relleno

Número de requisito	RF 1.12.3.		
---------------------	------------	--	--

Nombre de requisito	Tipo de relleno		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>El programa permite elegir el tipo de relleno que tendrá la figura que voy a dibujar o editar el de la figura seleccionada. Para ello hay tres opciones:</p> <ul style="list-style-type: none"> <li>• RF 1.12.3.1. Relleno simple</li> <li>• RF 1.12.3.2. Relleno horizontal</li> <li>• RF 1.12.3.3. Relleno vertical</li> </ul> 			

- RF 1.12.3.1. Relleno simple

Número de requisito	RF 1.12.3.1.		
Nombre de requisito	Relleno simple		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Deshabilita el degradado/asigna un relleno normal, si el relleno está activado. Pintará la figura o modificará el relleno sin degradado, relleno normal, con el color de relleno seleccionado.</p>			


- RF 1.12.3.2. Relleno horizontal

Número de requisito	RF 1.12.3.2.		
Nombre de requisito	Relleno horizontal		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			




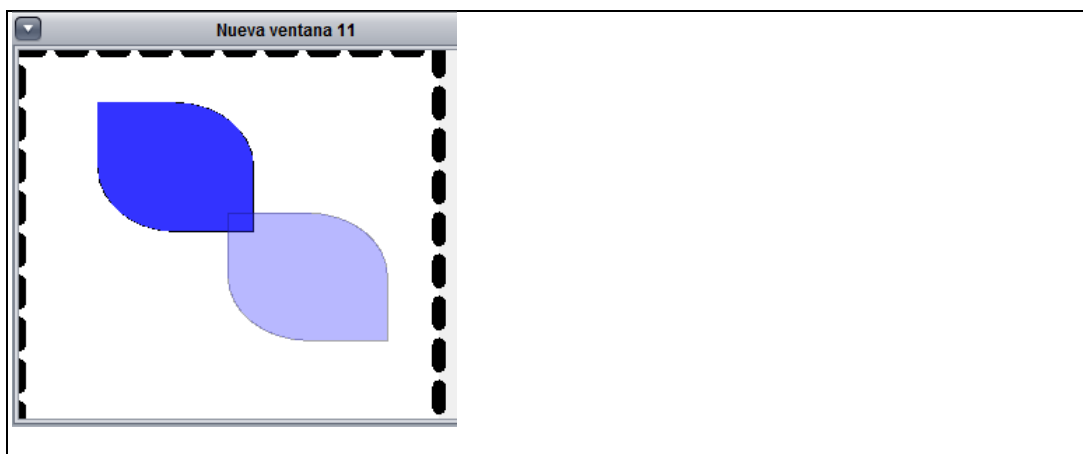
Permite el relleno, en caso de que esté habilitado, de forma horizontal. Para ello, crea un objeto GradientPaint donde los puntos de degradado empieza en la izquierda de la figura y acaba en la derecha de la figura, tanto de la que se vaya a pintar a continuación como de la que esté seleccionada. Utilizará el color secundario como el segundo color del degradado.

- RF 1.12.3.3. Relleno vertical


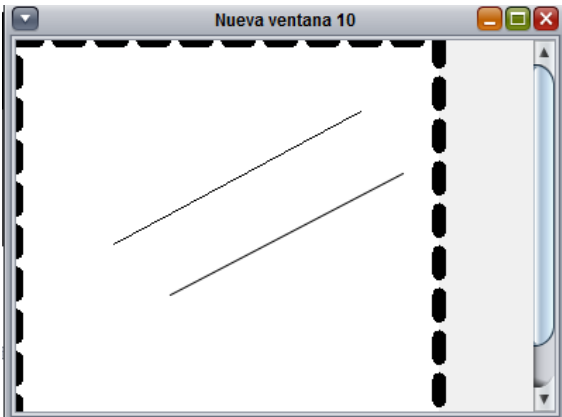
Número de requisito	RF 1.12.3.3.		
Nombre de requisito	Relleno vertical		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
Permite que el tipo de relleno, en caso que esté habilitado, sea del tipo horizontal. Lo realiza igual que el vertical, solo que los puntos que usa es el superior e inferior de la figura en cuestión para crear este nuevo GradientPaint.			

- RF 1.13. Modificar Transparencia

Número de requisito	RF 1.13.		
Nombre de requisito	Modificar Transparencia		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
Elemento deslizable que permite aplicar la cantidad de transparencia que tendrá mi figura. Para ello utiliza un JSlider que manda un valor de transparencia e inicializa el objeto Composite con dicho valor. Modifica la figura seleccionada o crea la siguiente con dicha transparencia.			



- RF 1.14. Habilitar Alisado

Número de requisito	RF 1.14.		
Nombre de requisito	Habilitar Alisado		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Aplicar a la figura seleccionada o a la próxima figura que vaya a dibujar un “filtro” de alisado (RendererHints). Si el botón está seleccionado, estará aplicado. En caso de no estarlo, quitará el alisado.</p>			
			

## 3.2. Imagen



- Análisis:

- ¿Qué ofrece? En clase hemos visto que Java ofrece tres modelos de imagen: modelo productor/consumidor, proveniente de la primera versión de java; modo inmediato, introducido en Java2D; modelo pipeline, introducido en JAI. Nosotros nos hemos quedado con el segundo, el que entró con Java2D, y cuyas clases principales son `BufferedImage`, `BufferedImageOp` e `ImageIO`.

Dicho modelo ofrece gran cantidad de métodos que permiten desde dibujar la imagen hasta obtener sus componentes. Sobre este último hemos trabajado mucho a la hora de aplicar filtros, ya que un objeto `BufferedImage` puede crearse con diferentes parámetros; una opción para crear un objeto de esta clase es usando su `Raster` y `ColorModel`, por ejemplo. El constructor por defecto utiliza el ancho y alto y el tipo de la imagen. Como veremos en el siguiente punto, este último ha de tenerse en cuenta si manipulamos imágenes que tienen un canal alfa, ya que no todas las imágenes tienen solo tres canales (RGB). Utilizaremos el otro constructor para cuando, por ejemplo, queramos hacer la imagen de otra forma, donde queramos utilizar un `Raster` diferente.

A la hora de aplicar filtros, utilizaremos varias formas de hacerlo, todos ellos con ayuda de la biblioteca “`BufferedImageOp`”, la cual nos permite procesar imágenes utilizando desde funciones (`LookupOp`), máscaras (`ConvolveOp`), girarla (`AffineTransformOp`) e incluso aplicar una transformación a cada elemento de la imagen. Podemos aplicar las nuestras propias clases, heredándolas de `BufferedImageOp`. Hablaré un poco sobre esto en el siguiente punto.

Finalmente, `ImageIO` lo hemos usado a la hora de abrir y escribir una imagen. Utiliza para ello los métodos “`read`” y “`write`”, despectivamente. El primero devolverá una variable del tipo `BufferedImage`, con la información de la imagen que hemos abierto.

- ¿Qué problemas aparecen? Hay algunos problemas que aparece a la hora de trabajar con estos elementos. Uno de ellos ya lo comentaba: vigilar el canal alfa. Si mi imagen tiene transparencia (ya sea porque es un PNG o cualquier otra cosa), debo tener en cuenta que habrá 4 canales y no 3, es decir, tendré R, B, G, y Alfa. Por lo que cuando vaya a guardar una imagen con transparencia en otro formato que no sea PNG, debo hacer algo para que la imagen no quede destrozada.

Si pintamos sobre una imagen que hallamos abierto, al guardarla no se pintarán las figuras. Tenemos que hacer que las vuelque, que pinte sobre ellas. El problema está en que si usamos el método “`Paint`” pintará la imagen y luego sobre ella el lienzo sobre las imágenes. Por ello, nosotros hemos llamado al método `paintComponent` en su lugar. Explico esto con más detalle en el javadocs.

Otro de los problemas con los que hemos mediado ha sido a la hora de crear filtros propios. Como comentaba, podemos crear nuestras clases propias de filtros `BufferedImageOp` si para ello la heredamos de esta. Ahí hemos trabajado utilizando dos tipos: filtro pixel a pixel y filtro componente a componente. Explicaré cómo en el siguiente punto.

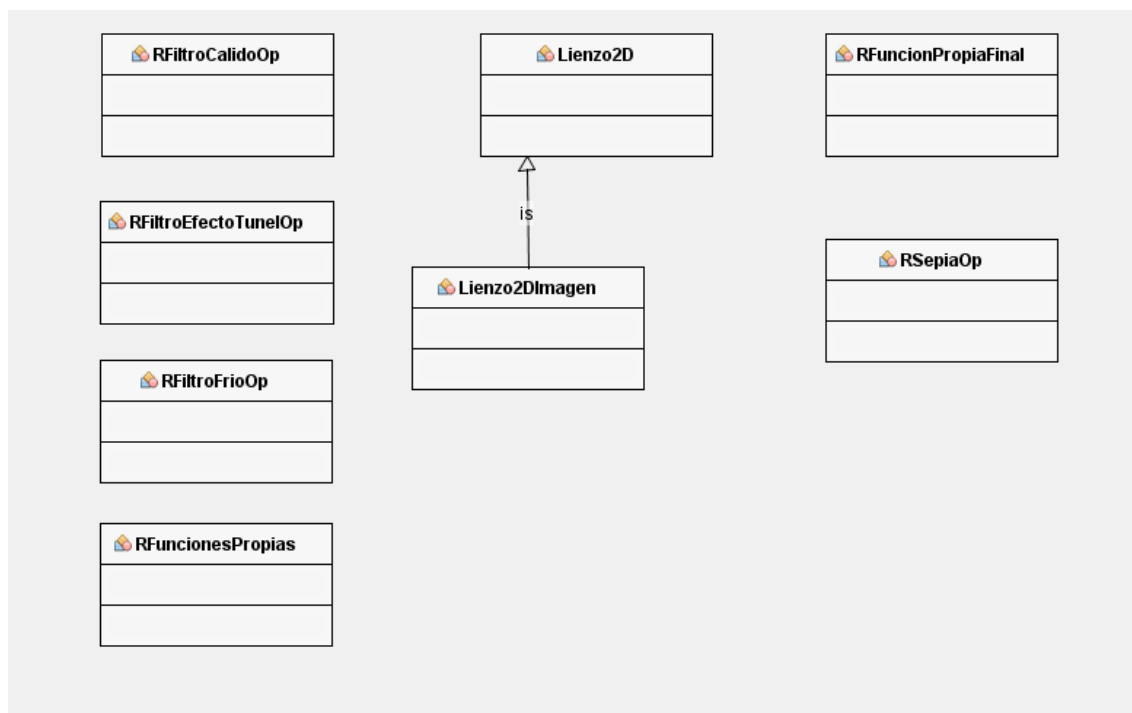
Adicionalmente, tener en cuenta como hemos trabajado con imágenes. Cuando convenía aplicar el filtro directamente sobre la imagen o sobre una copia. Para realizar dicha copia hemos tenido que crear una nueva imagen de tipo `BufferedImage` con los parámetros de `Raster` y `ColorModel` de la imagen original. Para ello, hemos todos estos componentes, lo hemos copiado y hemos montado la imagen nueva a partir de estos nuevos componentes. Será útil cuando queramos girar una imagen o hacer una copia de ésta.

Finalmente, una de las cosas que se me olvidaba. En el caso del lienzo vacío sobre el que pintaremos las formas, tuve en cuenta que se pintase sobre una región “clip”, y que el fondo fuese blanco. También que pintase un borde en dicho lienzo.

- Diseño: En mi caso, tengo una clase “**Lienzo2D**” sobre la que podré pintar todo tipo de imágenes. Con él crearé una imagen vacía de dimensiones 300 x 300, sobre la que dibujaré mis formas. Cuando quiera guardar la imagen, haré un volcado de todo lo que haya en dicho lienzo y lo guardaré en una imagen. Como dije, usamos `ImageIO` para guardar mis imágenes. Tendré que crear una imagen del tipo `BufferedImage` para este tipo de cosas (que así hice).

Adicionalmente, y en mi caso, utilizo `Lienzo2D` para albergar información de los atributos con los que pintaré mi figura. Esto es debido a que cada ventana interna del tipo imagen tiene unos atributos diferentes. Si en la primera ventana elijo la herramienta círculo y el color rosa, y en la segunda el color verde para la herramienta rectángulo, cuando cambie de una venta a otra, los atributos que debe mostrar la ventana principal son los que haya utilizado en último lugar en esa ventana. Es por ello que guardo en mi `Lienzo2D` dicha información.


Para imágenes que abra con el programa y dibuje sobre ellas o cambie el filtro, la cosa irá de una forma parecida. Por ello creé la clase “**Lienzo2DImagen**”, que hereda de “`Lienzo2D`” y tiene varios métodos sobrecargados de la original. En el siguiente modelo represento como he hecho la implementación (de nuevo, omitiendo el nombre de los atributos y de los métodos por temas de espacio):



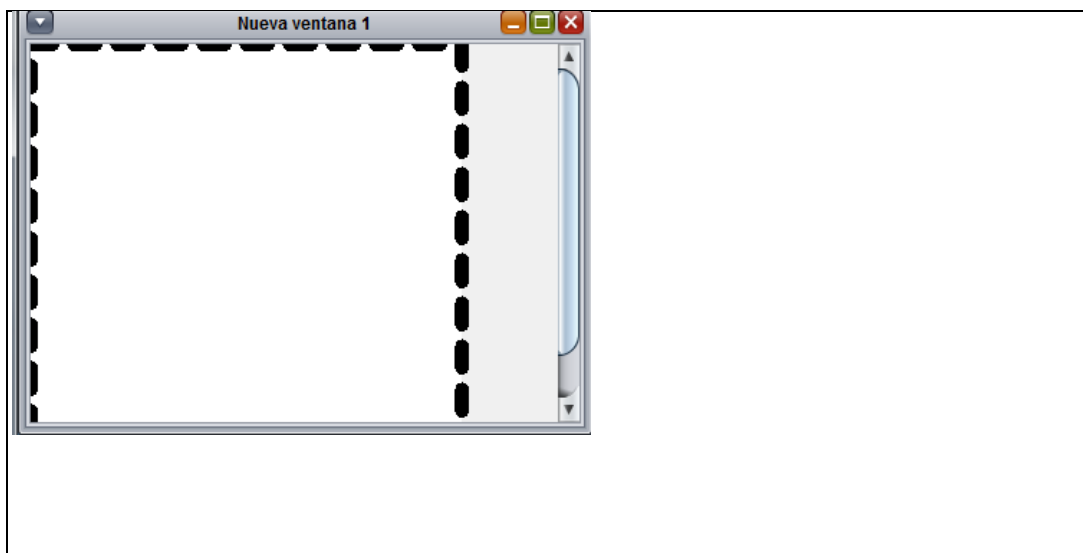
Adicionalmente, muestro las clases que tengo implementadas para los filtros y otras acciones que realizo en este bloque de Imagen.

Ya sea mi lienzo o una imagen abierta, puedo realizar diversas acciones, las cuales explico con los siguientes requisitos funcionales:



- RF 2.1. Nueva Imagen

Número de requisito	RF 2.1.		
Nombre de requisito	Nueva Imagen		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Crea un nuevo lienzo en blanco de tamaño 300 x 300. Dicho tamaño se puede modificar desde el menú Imagen &gt; Tamaño nueva Imagen (atajo de teclado Ctrl + T, referencia al RF 5.3.1.). El título indicará el número de ventanas que se han creado, junto a su título. El atajo para crear una nueva ventana es Ctrl + N. Dicho botón está tanto en el menú Archivo como en la Ventana Principal.</p>			






○ RF 2.2. Abrir Imagen

Número de requisito	RF 2.2.		
Nombre de requisito	Abrir Imagen		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Lanzará una ventana donde podremos elegir la imagen que queremos cargar, cuya extensión debe ser una de las disponibles. Después, carga dicha imagen en un lienzo, cuyo tamaño será el de que tenga la imagen en cuestión. Podré dibujar sobre ella y/o aplicarle uno de los diversos filtros que la aplicación ofrece. Pondrá como título el nombre del archivo. Dicho botón también está en el menú archivo. El atajo de botones es Ctrl + A.</p> 			

○ RF 2.3. Guardar Imagen

Número de requisito	RF 2.3.
Nombre de requisito	Guardar Imagen

Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
 <p>Lanzará de nuevo el dialogo preguntándonos por el nombre con el que queremos guardar nuestra imagen. Se pone el nombre (junto "punto" y las siglas de la extensión) y se elige la extensión abajo. Si todo ha ido bien, habrá guardado la imagen. También está en el menú "Archivo", y además su atajo de teclado es Ctrl + G.</p>		


○ RF 2.4. Escalar Imagen

Número de requisito	RF 2.4. Escalar	
Nombre de requisito	Escalar Imagen	
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
<p>Si hay una imagen abierta, permite que dicha imagen sea escalada (es decir, que su ancho y alto se "ajustarán" a los nuevos). Dicho requisito está explicado con más detalle en el RF 5.3.1., el lugar desde donde se llama.</p>		

○ RF 2.5. Redimensionar Imagen

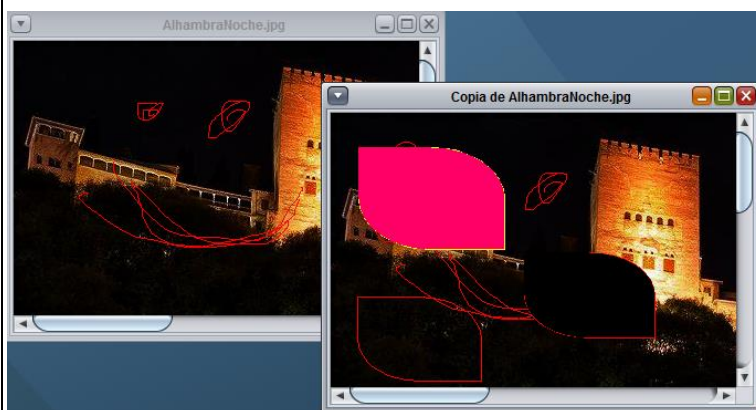
Número de requisito	RF 2.5.	
Nombre de requisito	Redimensionar Imagen	
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
<p>Si hay una imagen abierta, permite que dicha imagen sea redimensionada (será recortada o ampliada, junto al tamaño del lienzo, con los nuevos valores de ancho y alto). Dicho requisito está explicado con más detalle en el RF 5.3.1., el lugar desde donde se llama.</p>		

○ RF 2.6. Duplicar Imagen

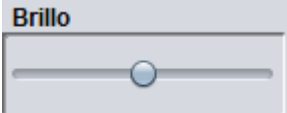
Número de requisito	RF 2.6.	
Nombre de requisito	Duplicar Imagen	
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
 <p>Al pulsar el botón, realizará una copia sobre la imagen actual (si hay alguna). Dicha copia</p>		

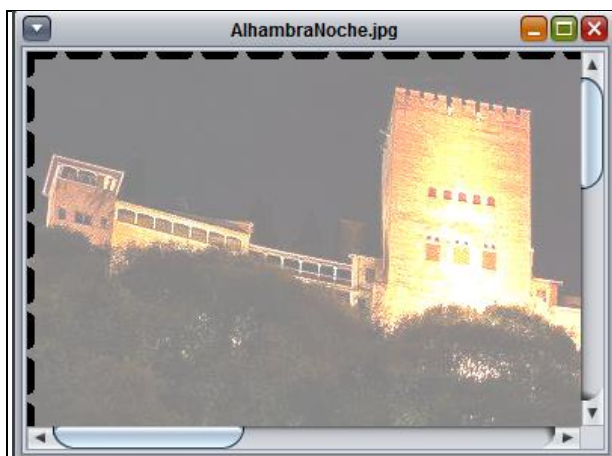
se realizará de la siguiente forma:

- Creará una nueva imagen a partir de la original, copiando hasta las formas que se hayan dibujado en la primera. Para ello hará un volcado de estas imágenes, provocando que éstas estén “incrustadas” en la figura y no puedan volver a modificarse.
- Se realizará el copiado del Raster, el ColorModel, y si existe algún canal alfa.
- Creará una NUEVA imagen, utilizando las copias de los elementos anteriores, además del mismo tamaño y tipo de la imagen original.
- Se añadirá la nueva imagen a una nueva ventana Interna.
- Dicha imagen NO es una referencia a la original. Es una nueva imagen, copia de la primera.



○ RF 2.7. Modificar brillo de la Imagen

Número de requisito	RF 2.7.		
Nombre de requisito	Modificar brillo de la imagen		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
<div> <p><b>Brillo</b></p>  </div> <p>Modifica el brillo de la imagen actual, si hay alguna. Desplazando el botón hacia la derecha, la imagen será más brillante, mientras que si se desplaza hacia la izquierda, la imagen oscurecerá.</p>			

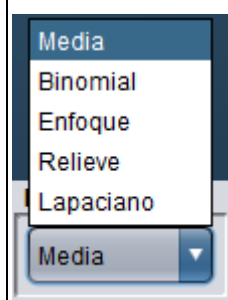


Nota: Debido a la implementación, cuando se pierda el “foco” de la ventana, el valor que tiene este botón deslizable volverá al defecto, pero el brillo permanecerá en la imagen, provocando que mi imagen se quede con dicho brillo definitivamente, no pudiendo volver al original.

#### ○ RF 2.8. Menú Filtros Imagen

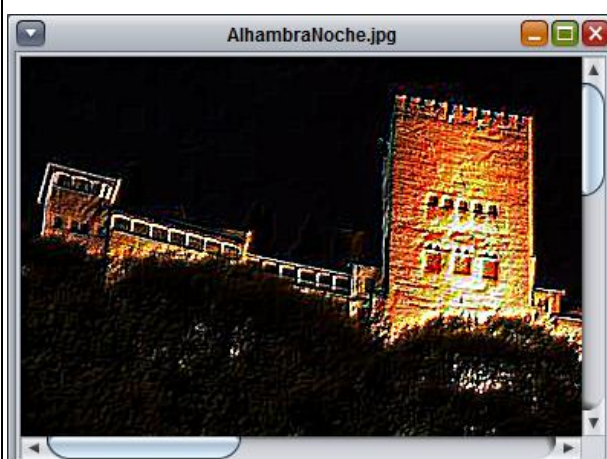
Número de requisito	RF 2.8.		
Nombre de requisito	Menú Filtros Imagen		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:



Aplica uno de los filtros disponibles en este menú a la imagen actual, si hay alguna. Dichos filtros están implementados en el paquete “sm.imagen” aportado por la asignatura. Una

vez que se seleccione uno, se aplica, sin opción de quitarlo.



*Ejemplo de filtro "Relieve"*

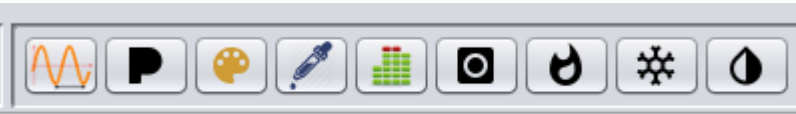
○ RF 2.9. Modificar contraste

Número de requisito	RF 2.9.		
Nombre de requisito	Modificar contraste		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
<div data-bbox="279 1120 587 1256" data-label="Image"> </div> <p>Aplica uno de los tres tipos de contraste a la imagen seleccionada, si hay alguna. Los botones, de izquierda a derecha, realizan las siguientes opciones:</p> <ul style="list-style-type: none"> <li>• El primer botón aplica "contraste" a la imagen.</li> <li>• El segundo botón, ilumina la imagen.</li> <li>• El tercer botón, oscurece la imagen.</li> </ul> <p>Dichos filtros se aplican una vez pulsados, no se pueden rehacer. Al pulsar otro de los tres, este filtro se añadirá además del que previamente se había seleccionado.</p>			



Ejemplo de "Contraste"

○ RF 2.10. Otros filtros

Número de requisito	RF 2.10.		
Nombre de requisito	Otros filtros		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
<p>En este apartado recojo otros de los filtros que he aplicado. Algunos de ellos son filtros propios que he tenido que implementar personalmente. Dichos filtros se recogen en:</p> <ul style="list-style-type: none"> <li>• RF 2.10.1. Función Seno</li> <li>• RF 2.10.2. Función Propia: Seno + Coseno</li> <li>• RF 2.10.3. Filtro Sepia</li> <li>• RF 2.10.4. Tintado</li> <li>• RF 2.10.5. Ecualizador</li> <li>• RF 2.10.6. Efecto Túnel</li> <li>• RF 2.10.7. Efecto calor</li> <li>• RF 2.10.8. Efecto frío</li> <li>• RF 2.10.9. Invertir Colores</li> </ul>			

▪ RF 2.10.1. Función Seno

Número de requisito	RF 2.10.1.		
Nombre de requisito	Función Seno		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			



Aplica a la imagen actual, si hay alguna, el filtro LookupOp con el LookupTable que devuelve aplicar la función seno(x) (en valor absoluto).



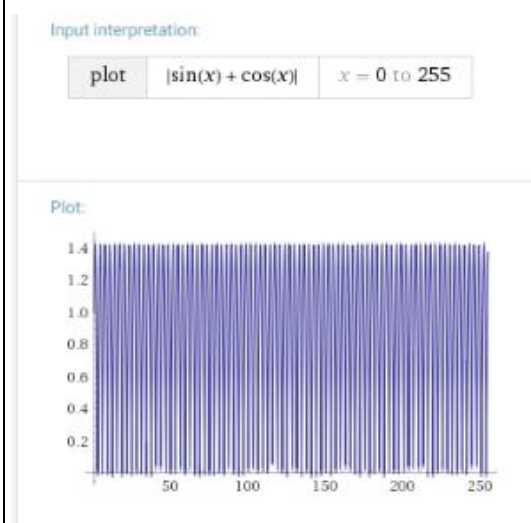
▪ RF 2.10.2. Función Propia: Seno + Coseno

Número de requisito	RF 2.10.2.		
Nombre de requisito	Función Propia: Seno + Coseno		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:



Aplica a la imagen actual, si hay alguna, el filtro LookupOp con el LookupTable que devuelve aplicar la función seno(x) + coseno(x) (en valor absoluto), cuyo máximo es la raíz cuadrada de 2. Dicha función se representa de la siguiente forma:



Para aplicar esta función a la imagen, basta con pulsar el botón:





▪ RF 2.10.3. Filtro Sepia

Número de requisito	RF 2.10.3.		
Nombre de requisito	Filtro Sepia		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:





Aplica a la imagen actual, si hay alguna, un filtro estilo “sepia”. Bastará con pulsar sobre el botón para que la imagen actual se vea modificada con dicho filtro.




▪ RF 2.10.4. Tintado

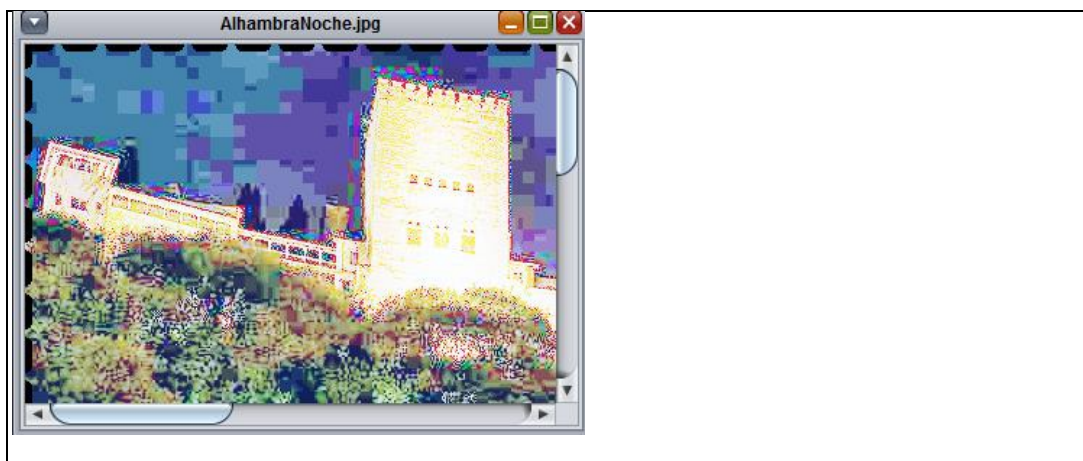
Número de requisito	RF 2.10.4
Nombre de requisito	Tintado



Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
 <p>Aplica a la imagen actual, si hay alguna, un filtro de “tintado” con el color seleccionado. Para hacerlo, elijo el color con el que quiero tintar, y a continuación pulso sobre este botón.</p>  <p><i>Ejemplo de tintado color rojo</i></p>		

▪ RF 2.10.5. Ecualizador

Número de requisito	RF 2.10.5.	
Nombre de requisito	Ecualizador	
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
 <p>Permite ecualizar la imagen actual, si hay alguna. Basta con pulsar el botón y se nos mostrará una nueva imagen con el filtro aplicado.</p>		



▪ RF 2.10.6. Efecto Túnel

Número de requisito	RF 2.10.6.		
Nombre de requisito	Efecto Túnel		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:

Aplica el filtro “efecto túnel” a la imagen actual, si hay alguna. Lo que hace en este filtro es crear la sensación de que la imagen se cómo si estuviera al final de un túnel. Para ello, oscurece cada componente dependiendo de su posición: cuanto más cercano a una esquina, más oscuro. Este filtro es “componente a componente”.

▪ RF 2.10.7. Efecto Calor

Número de requisito	RF 2.10.7.		
Nombre de requisito	Efecto Calor		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:



Aplica el filtro “calor” a la imagen actual, si hay alguna. Dicho filtro aumenta la tonalidad de los colores rojos (banda R) y un poco los verdes (banda G). Disminuye un poco los azules (banda B). Este filtro es “pixel a pixel”.



▪ RF 2.10.8. Efecto Frío

Número de requisito	RF 2.10.8.		
Nombre de requisito	Efecto Frío		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:



Aplica el filtro “frío” a la imagen actual, si hay alguna. Dicho filtro aumenta la tonalidad de los colores azules (banda B) y un poco los verdes (banda G). Disminuye un poco los rojos (banda R).



Nota: Aunque el enunciado de la práctica pedía que aplicásemos un solo filtro personalizado del tipo “pixel a pixel”, y ya implementé el filtro “calor” (RF 2.10.7.), me pareció que tendría que poner también el filtro frío para acompañarlo.

#### ▪ RF 2.10.9. Invertir Colores

Número de requisito	RF 2.10.9.		
Nombre de requisito	Invertir Colores		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:



Invierte los colores de la imagen actual (si hay alguna). Realiza la operación sobre dicha imagen.



#### ○ RF 2.11. Modificar espacio de colores y extraer bandas

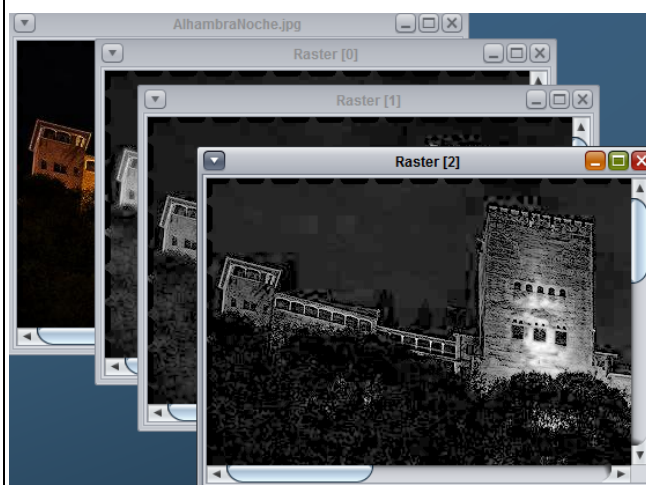
Número de requisito	RF 2.11.
Nombre de requisito	Modificar espacio de colores y extraer bandas

Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

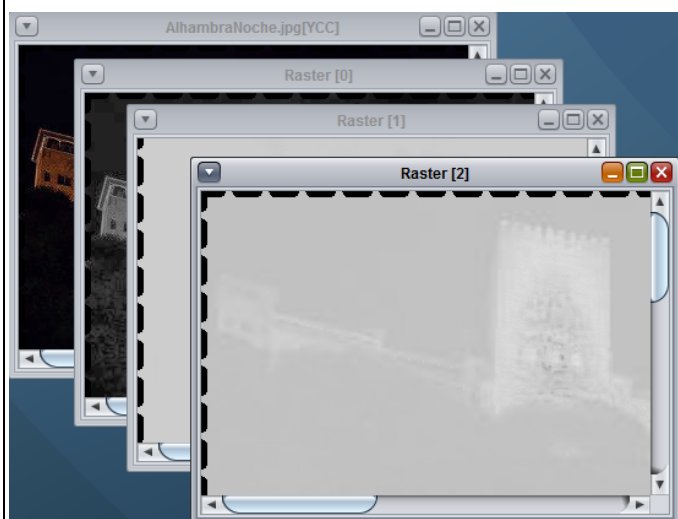
Descripción del requisito:



El menú de la izquierda cambiará el tipo de espacio de colores que tiene mi imagen. El segundo elemento, el botón de la derecha, extrae las bandas de mi imagen. Dependiendo del tipo de espacio de colores, las bandas que extrae serán de una forma u de otra:



*Ejemplo de extracción de bandas RGB*

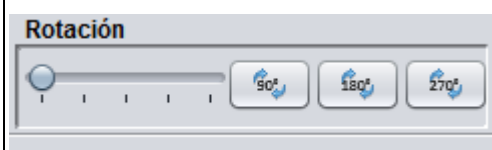


*Ejemplo de extracción de bandas de YCC*

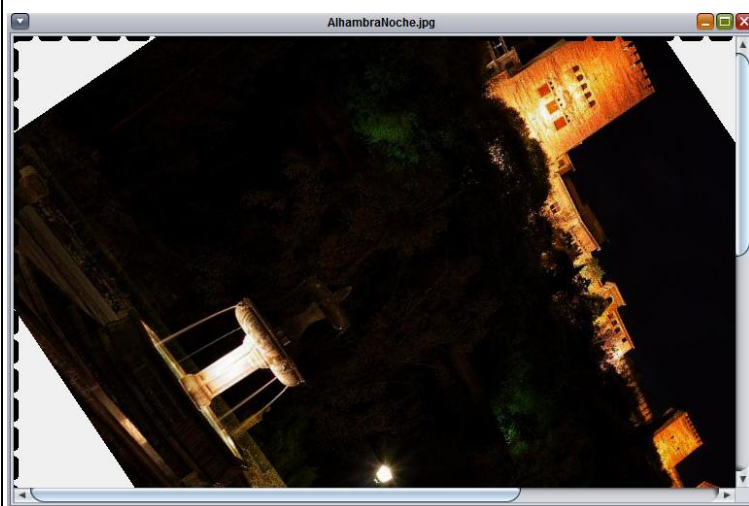
○ RF 2.12. Rotación

Número de requisito	RF 2.12.		
Nombre de requisito	Rotación		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:



Aplica una rotación a la imagen actual (en caso de que haya alguna). Puedo utilizar el botón deslizable para aplicar una rotación personalizada (mínimo 0, máximo 359 grados), o utilizar una de las que están implementadas por defecto (90, 180 u 270 grados).

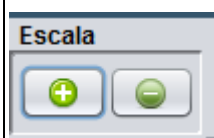


Nota: Este es uno de los casos donde hay que vigilar el canal alfa a la hora de guardar la imagen, ya que cuando se gira la imagen, se crea una transparencia, por ende, hay un cuarto canal. Todo esto está implementado a la hora de guardar la imagen (RF 2.3.)

○ RF 2.13. Escalado

Número de requisito	RF 2.13.		
Nombre de requisito	Escalado		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

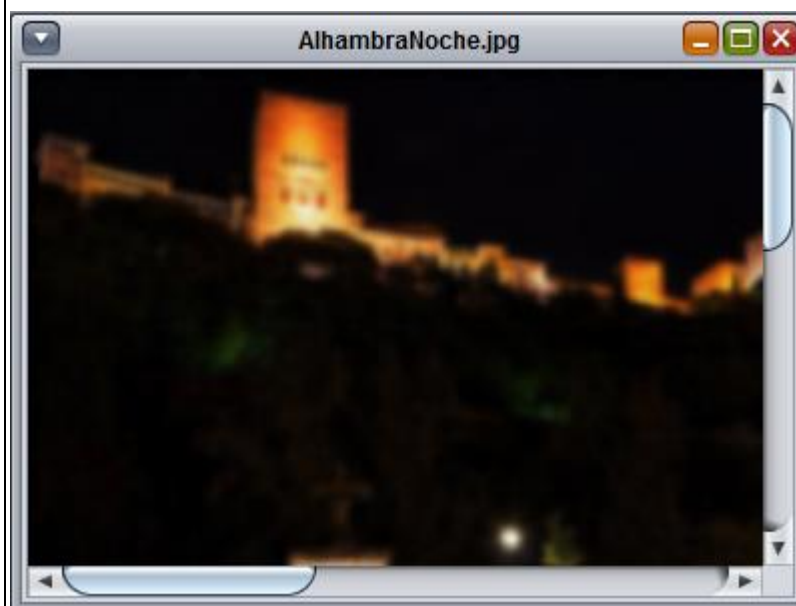
Descripción del requisito:



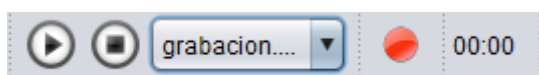
Realiza el efecto “zoom” sobre la imagen, si hay alguna en el lienzo actual. Si pulso sobre el botón “+”, dicho zoom aumentará. El efecto contrario se conseguirá si pulso sobre el botón “-”.



Nota: Si el zoom se aumenta y acto seguido disminuye, o al contrario, la imagen quedará “desenfocada” si se vuelve al tamaño original que tenía la imagen. No tendrá la misma nitidez que cuando se abrió dicha imagen.

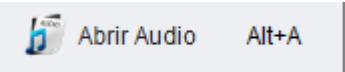


### 3.3. Sonido






- Análisis:
  - ¿Qué ofrece? Java ofrece dos alternativas para reproducir sonido. Una es AudioClip, para reproducción simple. La otra es Java Sound API, que además de reproducción, permite grabación, mezcla, etc. También existe Java Multimedia Framework, que es una herramienta más avanzada. En nuestro caso, nos hemos quedado con la Sound API para reproducir y grabar sonido.
  - ¿Qué problemas aparecen? Aunque Sound API nos permite abrir .wav y .au, no reproduce correctamente .mp3. Este problema no lo hemos solucionado (aunque con VLCj, en el apartado video, podemos abrir mp3).
  - Otra cosa, aunque no es un problema, es que hay que crear manualmente los Listener. Dada la experiencia que hemos conseguido en la asignatura, esto último no es un problema para nosotros.
- Diseño: Para manipular sonido (reproducción y grabación), he añadido a mi aplicación los siguientes elementos:
  - RF 3.1. Abrir audio

Número de requisito	RF 3.1.		
Nombre de requisito	Abrir audio		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
<p>Se encuentra en el menú "Archivo". Abre una ventana para poder elegir un audio (está limitado a archivos con extensión .wav y .au), lo abrirá y lo añadirá al menú desplegable. Se puede llamar con ALT + A.</p>			


- RF 3.2. Iniciar Reproducción

Número de requisito	RF 3.2.		
Nombre de requisito	Iniciar Reproducción		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
<p>Inicia la reproducción del elemento seleccionado del menú desplegable. En caso de estar vacío, no reproducirá nada. El botón se deshabilitará mientras cuando se pulse y se volverá a habilitar cuando termine la reproducción o cuando se pulse el botón "parar".</p>			


- RF 3.3. Parar Reproducción/Grabación

Número de requisito	RF 3.3.
---------------------	---------


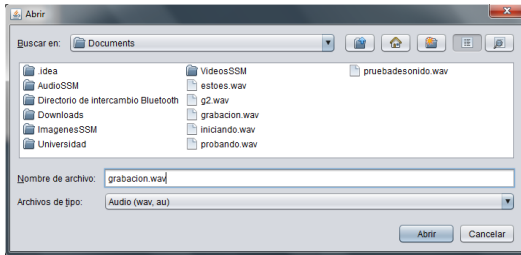


Nombre de requisito	Parar Reproducción/Grabación		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Para la grabación o reproducción, depende de lo que se esté ejecutando en el momento. Al hacerlo, el tiempo de reproducción se pondrá a 00:00. En caso que sea una grabación, abrirá el menú para guardarla</p>			

- RF 3.4. Menú desplegable de audios


Número de requisito	RF 3.4.		
Nombre de requisito	Menú desplegable de audios		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Contiene una lista de los audios “abiertos” en el programa. Cada vez que abramos uno nuevo, o grabemos uno y lo guardemos, se añadirá a esta lista. Podemos seleccionar el que queramos reproducir en su momento desde aquí.</p>			

- RF 3.5. Iniciar Grabación

Número de requisito	RF 3.5.		
Nombre de requisito	Iniciar Grabación		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Botón que inicia la grabación del sonido. La aplicación deshabilitará el botón grabación cuando se active, y el contador de tiempo de grabación empezará a andar. Cuando queramos guardar el sonido, pulsaremos el botón “Parar” (RF 3.3.) y se nos abrirá una ventana para elegir el lugar donde queramos guardarlo y el nombre del fichero.</p>  <p>En caso que lo guardemos, se añadirá al menú desplegable. Si cancelamos, cerrará la ventana y no hará nada.</p>			



- RF 3.6. Duración Reproducción/Grabación

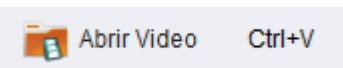
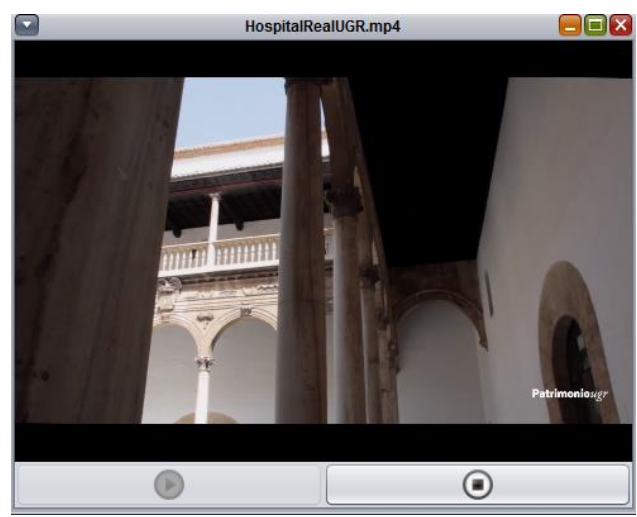
Número de requisito	RF 3.6.		
Nombre de requisito	Duración Reproducción/Grabación		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Campo de texto (JLabel) que muestra la duración de la reproducción/grabación actual. Inicialmente mostrará el tiempo a 00:00. El campo de la derecha muestra los segundos (máximo 59, min 00) y el apartado de la izquierda, los minutos (sin máximo, mínimo 00). Cuando la reproducción del sonido pare, o paremos nosotros la grabación, el tiempo volverá a 00:00. Funciona tanto para reproducción como para grabación.</p>			

### 3.4. Video




- Análisis:
  - ¿Qué ofrece? Para la reproducción de video hay varias API que ofrecen la reproducción del video. Una de ellas es JMF, usada para captura y reproducción de video. Aunque esta cubre ambas opciones, no me gustó usarla en mi aplicación.
  - ¿Qué problemas aparecen? El problema que hay con JMF es que es antiguo, tan antiguo que su última actualización fue en 2003. Eso provoca que no soporte los códec más actuales que se lanzaron después de dicha fecha. También tiene problemas con las arquitecturas de 64 bits, pues fue desarrollado para las de 32. La “solución” para esto, ha sido quedarme con la otra biblioteca, la VLCj, aportada por VLC. Para la captura de video, usaré la API de Webcam Capture. Utilizaré para ambos ventanas internas, donde los implementaré internamente.
- Diseño:
  - RF 4.1. Abrir Video


Número de requisito	RF 4.1.
Nombre de requisito	Abrir Video

Tipo	X Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	X Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
 <p>Abre una ventana de dialogo para elegir el video que quiera abrirse. En mi aplicación, permite abrir videos con formato .wav, .mp4 y .avi. Después de elegir el archivo, se abrirán una nueva ventana interna con dos botones. El comando que utiliza es Ctrl + V.</p> 		



- RF 4.1.1. Reproducir Video

Número de requisito	RF 4.1.1.	
Nombre de requisito	Reproducir Video	
Tipo	X Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	X Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
 <p>Botón para reproducir el video. Mientras se reproduzca el video, estará deshabilitado.</p>		


- RF 4.1.2. Parar reproducción Video

Número de requisito	RF 4.1.2.	
Nombre de requisito	Parar reproducción Video	
Tipo	X Requisito	<input type="checkbox"/> Restricción
Prioridad del requisito	X Alta/Eencial	<input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:		
 <p>Botón para parar la reproducción del video. Estará deshabilitado por defecto. Al pulsarlo, parará la reproducción.</p>		


- RF 4.2. Iniciar Webcam

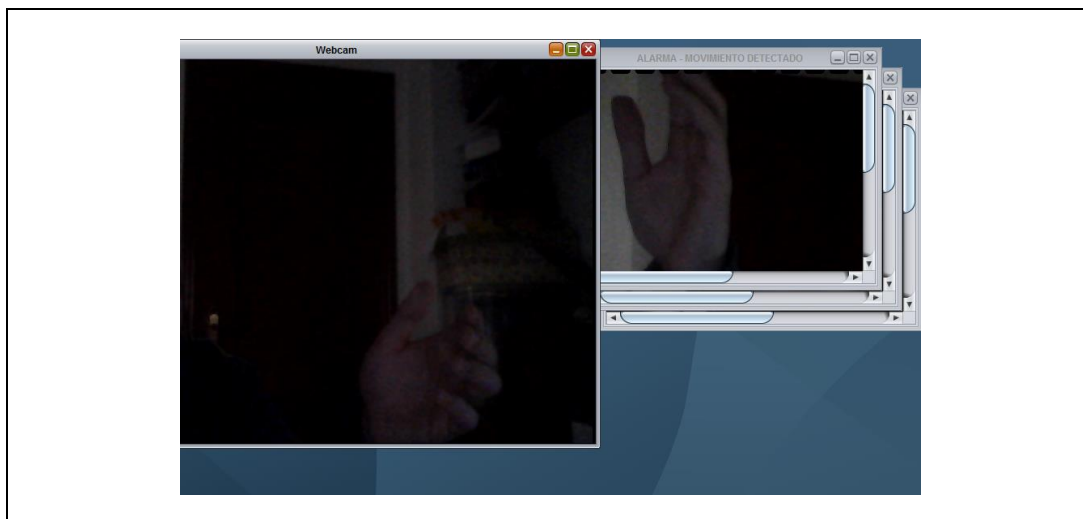
Número de requisito	RF 4.2.		
Nombre de requisito	Iniciar Webcam		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Eseñcial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Abre una nueva ventana interna del tipo webcam donde muestra el contenido que hay por cámara.</p> 			

- RF 4.3. Captura de pantalla video

Número de requisito	RF 4.3.		
Nombre de requisito	Captura de pantalla video		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Eseñcial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Realiza una captura de pantalla del video o la webcam. Tras pulsarlo, crea una nueva ventana interna de tipo Imagen con la captura realizada.</p>			

- RF 4.4. Activar vigilancia


Número de requisito	RF 4.4.		
Nombre de requisito	Activar vigilancia		
Tipo	X Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	X Alta/Eseñcial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Activa la vigilancia si hay una webcam abierta. Cada vez que la webcam detecte movimiento, hará una nueva captura de pantalla y la añadirá al escritorio.</p>			



### 3.5. Interfaz

- Análisis:
  - ¿Qué ofrece? Java ofrece multitud de elementos Swing y eventos que me han permitido el desarrollo de esta aplicación. Además, el uso de la IDE Netbeans ha ayudado a que muchas de estas cosas sean más cómodas, ya que gracias a su interfaz he podido añadir los botones y demás elementos, configurar sus propiedades e incluso trabajar con los eventos, todo cómodamente.
  - ¿Qué problemas aparecen? Más que problemas, hay dos inconvenientes y todos se deben a lo mismo: Netbeans no incluye todos los elementos swing y todos los eventos del mundo. Por eso hay un par que hay un par de elementos que he tenido que crear yo manualmente (como el menú desplegable de algunos colores) y los eventos asignados a audio y video que hemos visto que eran necesarios para algunos botones e incluso el contador de tiempo para el audio. Como digo, esto se ha “solucionando” haciendo yo mi propia implementación.
- Diseño: En este punto de la documentación ya he comentado como he hecho estos botones y eventos a lo largo de la documentación e incluso en el propio javadocs, voy a aprovechar este apartado para hablar como he diseñado aquellos elementos de mi aplicación que no han sido listados en ninguno de los requisitos funcionales que he mencionado.
  - RF 5.1. Menú Archivo

Número de requisito	RF 5.1		
Nombre de requisito	Menú Archivo		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			



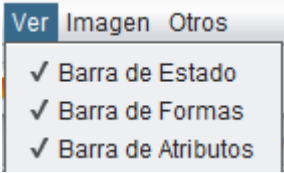
Abre un menú sobre las diferentes opciones de Nueva, Abrir y Guardar Imagen. También permite Abrir Audio y Abrir Video. Hace referencia a los requisitos:

- RF 2.1. Nueva Imagen
- RF 2.2. Abrir Imagen
- RF 2.3. Guardar Imagen
- RF 3.1. Abrir Audio
- RF 4.1. Abrir Video

○ RF 5.2. Menú Ver

Número de requisito	RF 5.2.		
Nombre de requisito	Menú Ver		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:



Menú donde hay 3 botones, seleccionados por defecto, los cuales permiten la visualización de: barra de estado, barra de formas y barra de atributos. Hace referencia a los requisitos siguientes:

- RF 5.2.1. Ver Barra de Estado
- RF 5.2.2. Ver Barra de Formas
- RF 5.2.3. Ver Barra de Atributos

▪ RF 5.2.1. Ver Barra de Estado

Número de requisito	RF 5.2.1.		
Nombre de requisito	Ver Barra de Estado		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Descripción del requisito:

Si está seleccionado, la barra de estado estará visible. En caso contrario, no se verá. Se selecciona y deselecciona haciendo clic sobre ella.

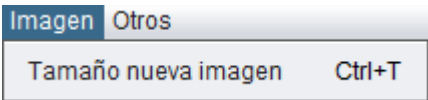
▪ RF 5.2.2. Ver Barra de Formas

Número de requisito	RF 5.2.2.		
Nombre de requisito	Ver Barra de Formas		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
Si está seleccionado, la barra de las figuras estará visible. En caso contrario, no se verá. Se selecciona y deselecciona haciendo clic sobre ella.			

▪ RF 5.2.3. Ver Barra de Atributos

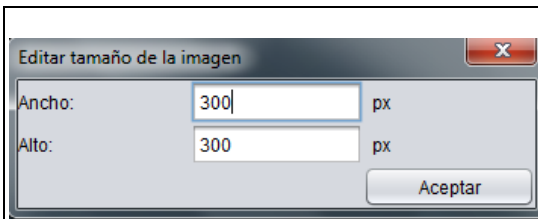
Número de requisito	RF 5.2.3.		
Nombre de requisito	Ver Barra de Atributos		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
Si está seleccionado, la barra de los atributos estará visible. En caso contrario, no se verá. Se selecciona y deselecciona haciendo clic sobre ella.			

○ RF 5.3. Menú Imagen

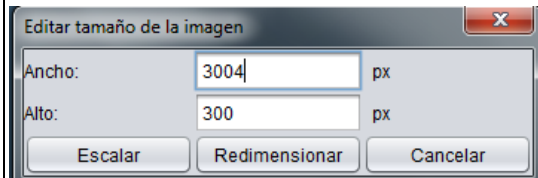
Número de requisito	RF 5.3.		
Nombre de requisito	Menú Imagen		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
 <p>Abre un menú con la opción “Tamaño nueva Imagen”, que me permitirá cambiar el tamaño de la imagen actual, si hay alguna. Se puede abrir pulsando Ctrl + T. Hace referencia al siguiente requisito:</p> <ul style="list-style-type: none"> <li>• RF 5.3.1. Tamaño nueva Imagen</li> </ul>			

▪ RF 5.3.1. Tamaño nueva Imagen

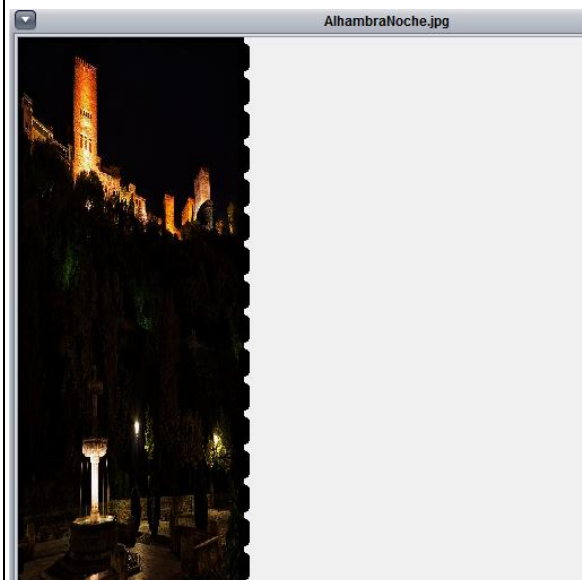
Número de requisito	RF 5.3.1.		
Nombre de requisito	Tamaño nueva Imagen		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			



Abra una ventana de diálogo, y si hay una imagen abierta, informará del ancho y del alto que tiene. Tendrá un botón aceptar que cerrará dicha ventana de diálogo.

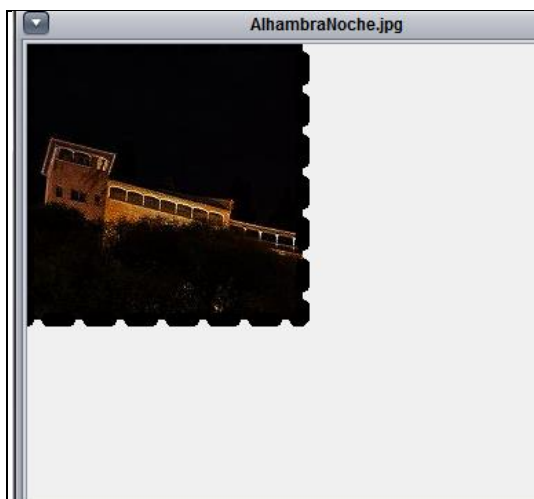


Si añadimos un nuevo número a la ventana de diálogo, ya sea en ancho o alto, aparecerán dos botones que nos indicarán si lo que queremos hacer con esa nueva medida es “escalar” la imagen (la imagen se adaptará al nuevo ancho y alto, “estirándose”) o “redimensionar” esta imagen (el lienzo cambiará de tamaño al nuevo insertado). Para aplicar cualquiera de las dos cosas, hay que pulsar el botón correspondiente.



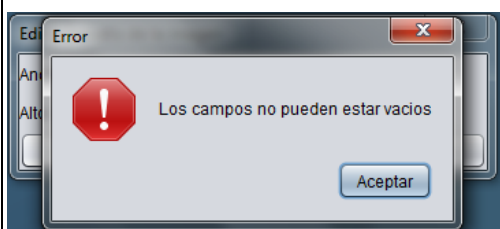
*Ejemplo de Escalado*





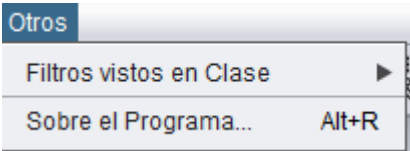
*Ejemplo de redimensionado*

En caso de no querer hacer nada, el botón “cancelar” cerrará la ventana.




Si uno de los campos está vacío, se lanzará una nueva ventana informando de ello.

○ RF 5.4. Menú Otros

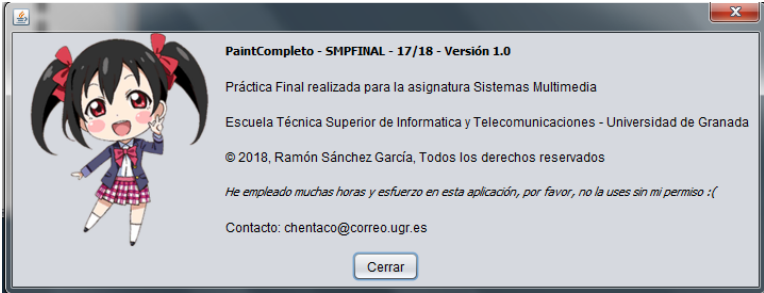
Número de requisito	RF 5.4.		
Nombre de requisito	Menú Otros		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eseñcial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
<p>Abre un menú con un menú desplegable, el cual incluye algunos ejemplos de los filtros vistos en clase, y un botón para abrir un dialogo sobre el programa. Incluye:</p> <ul style="list-style-type: none"> <li>• RF 5.4.1. Filtros vistos en clase</li> <li>• RF 5.4.2. Sobre el programa</li> </ul>			

▪ RF 5.4.1. Filtros vistos en clase

Número de requisito	RF 5.4.1.		
Nombre de requisito	Filtros vistos en clase		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	

Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
<p>Menú desplegable con algunos de los filtros de imagen que hemos visto de ejemplo en clase. Lo he dejado porque ya estaba implementado. Para aplicarlos es necesario tener abierta una imagen. Dichos filtros están implementados de forma más “profesional” como vimos en el bloque de imagen.</p>			

▪ RF 5.4.2. Sobre el programa

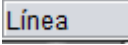
Número de requisito	RF 5.4.2.		
Nombre de requisito	Sobre el programa		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
<p>Abre una ventana de diálogo informando sobre algunos datos del programa y del autor. Dentro de la propia ventana hay un botón “Cerrar” que cierra dicha ventana. Se puede abrir pulsando Alt + R.</p>			

○ RF 5.5. Barra de estado

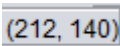
Número de requisito	RF 5.5.		
Nombre de requisito	Barra de estado		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
<p>Barra inferior en la que se mostrará tanto un texto como informando de la herramienta que tenemos seleccionada como las coordenadas donde se sitúa el cursor. Usa los requisitos funcionales:</p> <ul style="list-style-type: none"> <li>• RF 5.5.1. Información de la herramienta</li> <li>• RF 5.5.2. Coordenada del cursor</li> </ul>			

--


▪ RF 5.5.1. Información de la herramienta

Número de requisito	RF 5.5.1.		
Nombre de requisito	Información de la herramienta		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
Muestra un texto informando de la herramienta que tenemos seleccionada.			

▪ RF 5.5.2. Coordenada del cursor

Número de requisito	RF 5.5.2.		
Nombre de requisito	Coordenada del cursor		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
Muestra la información del punto donde se encuentra el cursor actualmente. Muestra tanto la coordenada (X) como la ordenada (Y).			

○ RF 5.6. Minimizar ventana aplicación

Número de requisito	RF 5.6.		
Nombre de requisito	Minimizar ventana aplicación		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
Permite a la ventana principal, y por ende, la propia aplicación ser minimizada.			


○ RF 5.7. Maximizar ventana aplicación

Número de requisito	RF 5.7.		
Nombre de requisito	Maximizar ventana aplicación		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			



Permite a la ventana principal, y por ende, la propia aplicación ser maximizada/expandida.

- RF 5.8. Cerrar ventana aplicación

Número de requisito	RF 5.8.		
Nombre de requisito	Cerrar ventana aplicación		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
			
Permite a la ventana principal, y por ende, la propia aplicación ser cerrada.			

- RF 5.9. Visualización del Cursor

Número de requisito	RF 5.9.		
Nombre de requisito	Visualización del Cursor		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/ Opcional
Descripción del requisito:			
El cursor cambiará de estilo dependiendo del lugar. Si este entra en una ventana interna del tipo Imagen, cambiará a forma de cruz.			
Si este cursor está en modo “Editar” y mueve una figura, cambiará a forma de “cruz con flechas”.			
El cursor volverá a estar como cruz normal si se deja de mover la figura y a flecha si se sale de lienzo imagen.			

## 4. Comentarios y conclusiones

### 4.1. ¿Qué he aprendido?

Desde mi punto de vista, creo que lo que más he aprendido ha sido que no todo lo que existe en el mundo de la informática está bien. Solo hay que echar un ojo a las bibliotecas que hemos usado y ver que los equipos que la han desarrollado no lo hacían con mucho cuidado. A pesar que he mejorado mis conocimientos sobre Java y sus elementos Swing, a diseñar una aplicación, crear clases propias y filtros, creo que lo que he comentado es lo que más me ha llegado y motivado.

Quizás sueño a lo grande, pero me gustaría en un futuro hacer algo parecido a las bibliotecas que hemos utilizado durante el curso, algo tipo “Graphics2Dv2” donde cada figura tenga sus atributos, separados de las otras. Algo más profesional a lo de esta práctica final.

## 4.2. ¿Cuál es el futuro de mi aplicación?

Cuando tenga tiempo libre me gustaría revisar la aplicación, ir botón a botón y ver si tienen algún fallo. También me gustaría arreglar el código, o, si algún día me animo, hacerla desde cero, haciendo todo lo voluntario que no me ha dado tiempo, o incluso realizar una versión “comercial” (dentro de lo que se pueda).

Esta aplicación me ha animado mucho sobre el futuro, y me gustaría trabajar en algo relacionado. Ya no se trata de crear una aplicación propia, sino de crear mis clases, mis bibliotecas propias y que sean otras personas las que la usen, además de que sean ellos los que me saquen fallos y eso les anime a hacer las suyas propias, tal y como me ha pasado a mí con esta asignatura.

Volviendo al tema de mi aplicación, dicho código solo será compartido con empresas y me servirá de “carta de presentación”, tal y como se comentó en clase. No pretendo compartirlo con nadie que no sea de mundo laboral, ni si quiera con compañeros de facultad, ya que he trabajado mucho en ella y no quiero que copien mi trabajo. Por lo que por el momento dicho código y documentación serán privados y por ahora solo dispondrá una copia la ETSIT.

## 4.3. Otros comentarios

Esta aplicación ha sido realizada con sumo cariño, ya que, en primer lugar, me encanta programar en Java, y en segundo lugar, el resultado es bastante conmovedor. Ver que he sido capaz de realizar un “Paint” que realiza tantas cosas, eso me encanta. Me habría gustado haber realizado más cosas opcionales, pero dado el tiempo libre que dispongo, no podré hacerlo. Como dije en el punto anterior, quizás en un futuro retome y complete, o lo haga desde cero con otro diseño.

He intentado que mi aplicación no tenga ningún “bug” y que todo funcione como quiero que funcione, sin embargo hay alguna cosilla que se me ha olvidado “arreglar”, como viene a ser tener en cuenta SIEMPRE el canal alfa, y creo que se me ha pasado en alguno de los filtros personalizados, o sobre los botones que se seleccionen dependiendo de la ventana activa o de la figura seleccionada. También hay algún que otro problema con abrir video. Desde algunos ordenadores funciona sin problema, y en otros no funciona. Puede deberse a que no tengan VLC o no dispongan de la misma versión que se ha usado para el desarrollo de esta aplicación.

Para probar dicha aplicación, he usado a varios seres cercanos para que “jueguen” con ella, y ninguno me ha dicho que haya algo raro, por el contrario, han quedado bastante emocionados.

Acabar diciendo que me ha gustado la asignatura y que me hubiese gustado haber trabajado aún más en ella, ya que como dije, todo este mundo me gusta mucho.

## 5. Bibliografía

La gran mayoría de material usado para la realización tanto de esta práctica final como de todas las prácticas del curso ha sido otorgada por el profesor de la asignatura. No dispongo de ningún enlace a dicho material, ni tengo permiso para distribuirlo.

Los siguientes links hacen referencia al resto del material utilizado a lo largo de esta asignatura que me ha ayudado o he usado:

[1] **Java™ Platform, Standard Edition 8 API Specification** (la “biblia” de esta asignatura, útil para consultar cualquier clase o método que tenga Java) - <https://docs.oracle.com/javase/8/docs/api/>

[2] **The Java™ Tutorials, Trail: 2D Graphics** (tutorial sobre cómo crear y manejar gráficos en Graphics2D) - <https://docs.oracle.com/javase/tutorial/2d>

[3] **Interface ListCellRenderer<E>** (documentación sobre cómo crear un ListCellRenderer personalizado) - <https://docs.oracle.com/javase/8/docs/api/javax/swing/ListCellRenderer.html>

[4] **Caprica Software Limited – VLCj** (web sobre el creador de la biblioteca VLCj, para la reproducción de video) - <http://capricasoftware.co.uk/#/projects/vlcj>

[5] **Webcam Capture API** (repositorio del creador de la API para la captura de Webcam) - <https://github.com/sarxos/webcam-capture>

[6] **Stack Overflow** (foro al que he ido a parar en más de una ocasión) - <https://stackoverflow.com/>

[7] **Netbeans** (IDE utilizada para el desarrollo de la práctica) - <https://netbeans.org/>

Finalmente agradecer a **D. Jesús Chamorro Martínez** por sus enseñanzas en la asignatura de **Sistemas Multimedia** de la ETSIT de la **Universidad de Granada**, sin las cuales no podría haber sido posible el desarrollo de esta aplicación.

