

SMM - Práctica Final

[1.1 Introducción:](#)

[1.2 Vista preliminar:](#)

[1.3 Análisis de requisitos:](#)

[1.4 Diagrama de clases:](#)

[1.5 Lista de archivos:](#)

[1.6 Características adicionales desarrolladas:](#)

[1.7 Explicación del sistema propio de clases:](#)

[1.8 Conclusiones:](#)

[1.9 Bibliografía:](#)

1.1 Introducción:

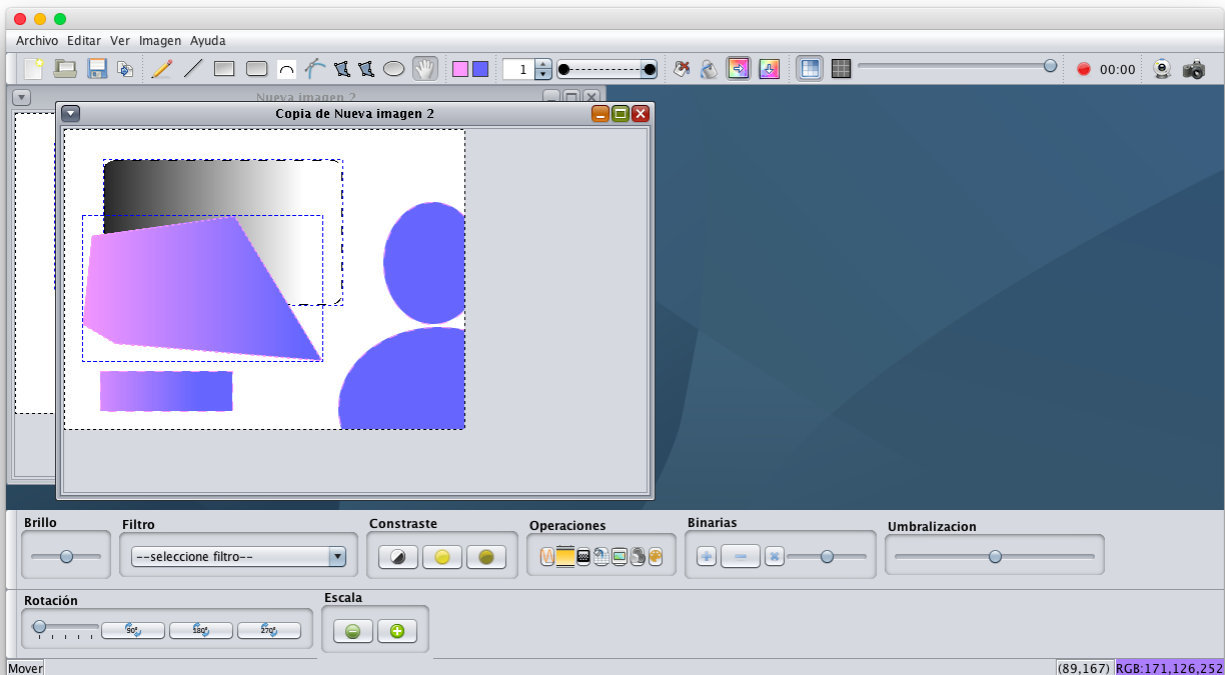
Durante el segundo semestre de 2016 he cursado la asignatura de Sistemas Multimedia en la ETSIIT de la Universidad de Granada, esta asignatura tiene como objetivo enseñarnos diversos conceptos sobre los diferentes sistemas multimedia. Dichos sistemas acceder de una forma natural, usando principalmente los sentidos de la vista y el oído, a contenidos creados, generalmente, por otros seres humanos y tratados/transmitidos por dispositivos digitales. En dicha asignatura se estudian contenidos relacionados con: los dispositivos de adquisición de audio, imágenes y vídeo más usados, su control (resolución, tasas de muestreo) y manejo (incluyendo por ejemplo la edición y el retoque básicos), sus posibilidades (formatos soportados), las principales técnicas de codificación (principalmente compresivas y de transporte, así como la gestión de dichos contenidos desde la plataforma Java utilizando el entorno integrado NetBeans.

Para la evaluación de esta asignatura se ha propuesto la realización de una aplicación en Java que aglutina todo lo aprendido durante el curso, dicha aplicación permitirá trabajar con lo aprendido tanto en imagen como en video y sonido, además en la parte de imagen permitirá por un lado el dibujo de formas geométricas básicas así como operaciones avanzadas sobre imágenes fotográficas.

Dado el carácter multiplataforma del lenguaje Java dicha aplicación deberá ejecutarse sin problema en los principales sistemas operativos, estos son Windows, Linux y MacOSX.

1.2 Vista preliminar:

En la captura de pantalla se puede apreciar cuál es el aspecto de la aplicación. Podemos encontrar una barra de menú, una barra de herramientas superior y dos barras para trabajar sobre la imagen, una para efectos y otra para zoom y rotación, se ha separado en dos para reducir el tamaño.



1.3 Análisis de requisitos:

RF1 → Crear ventana principal del programa

RF1.1 → Crear ventana de imagen que contenga un lienzo

RF1.1.1 → Permitir dibujar figuras en el lienzo

RF1.1.2 → Permitir dibujar figuras geométricas básicas (Rectángulo, Elipse, Línea...)

RF1.1.3 → Permitir dibujar figuras avanzadas (Curva, Polígono, Línea...)

RF1.1.4 → Permitir seleccionar una figura y cambiar sus parámetros (color, trazo)

RF1.1.5 → Permitir seleccionar una figura y moverla sobre el lienzo

RF1.1.6 → Permitir abrir imágenes desde archivo

RF1.1.7 → Permitir guardar imágenes en archivo

RF1.1.8 → Permitir realizar efectos sobre la imagen seleccionada (Negativo, Brillo, Relieve...)

RF1.1.9 → Permitir duplicar la ventana de imagen seleccionada

RF1.2 → Crear ventana de visualización de webcam

RF1.2.1 → Permitir realizar una instantánea de la ventana webcam

RF1.3 → Crear ventana de video

RF1.3.1 → Permitir realizar una instantánea de la ventana de video

RF1.3.2 → Permitir controlar la reproducción de video (Play, Stop...)

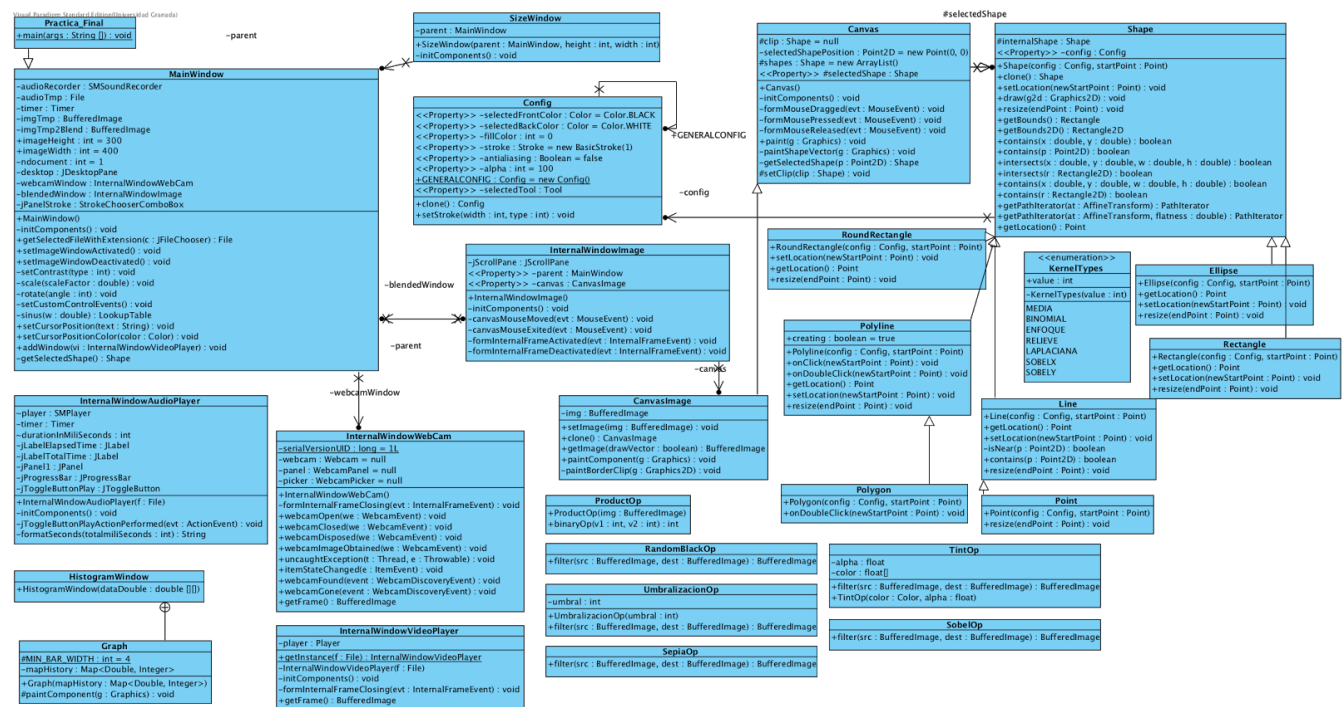
RF1.4 → Crear ventana de audio

RF1.4.1 → Permitir controlar la reproducción de video (Play, Stop...)

RF1.5 → Permitir la grabación de audio

1.4 Diagrama de clases:

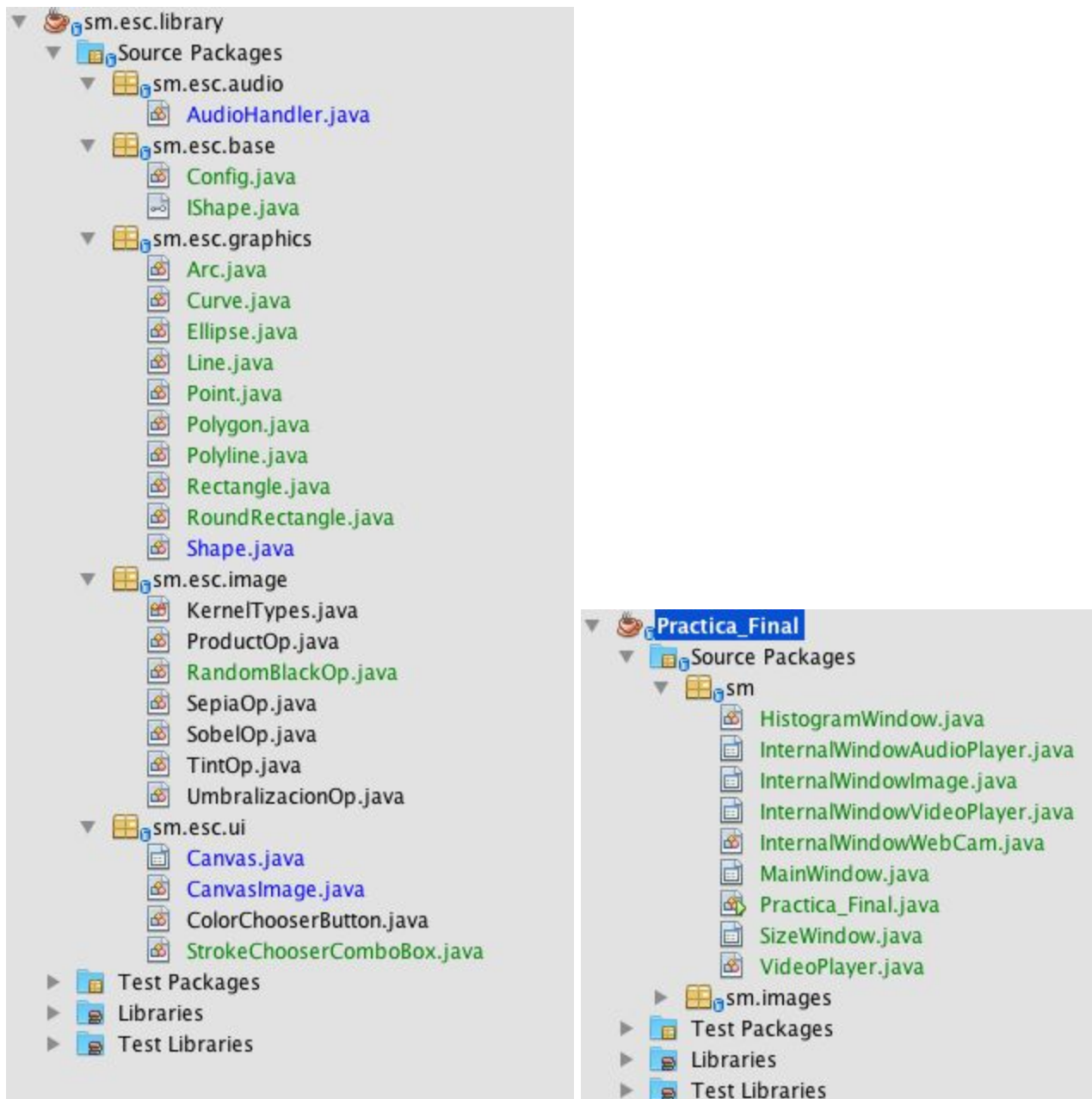
Dada su extensión, y para poder apreciar el diagrama de clases en su totalidad se ha incorporado el fichero “diagrama.png” a la raíz de la entrega.



1.5 Lista de archivos:

Los ficheros están separados en dos proyectos, el del ejecutable que contiene la función Main() para entrar al programa así como las distintas ventanas (formularios), y la librería **sm.esc.library**, que contiene las diferentes clases que utilizamos en el principal.

El árbol de archivos es el siguiente:



(Nota: Los colores simplemente indican el estado de los archivos en el repositorio GIT)

1.6 Características adicionales desarrolladas:

Aparte de las características básicas solicitadas se han desarrollado las siguientes características:

- Clase **ColorChooserButton**: Se ha implementado un nuevo “control” que hereda de JLabel e implementa un JColorChooser al hacer clic, una vez seleccionado el color lo asigna como color de fondo a dicho label, así podemos ponerlo en la barra de herramientas
- Clase **StrokeChooserComboBox**: Se ha implementando un nuevo “control” que hereda de JComboBox y sobrecarga su pintado para pintar varios tipos de “Stroke”.
- Se ha mostrado un formulario con el Histograma.
- Se ha usado la librería “Sarxos” para que funcione la webcam en múltiples plataformas.
- Se ha implementado la opción de “Duplicar” dibujo.
- Se ha implementado la forma **Polilínea** (polígono sin cierre) y **Polígono** (polígono cerrado).
- Se ha implementado la opción de tintado usando el color primario seleccionado.
- Aparte de esas mejoras se ha adaptado el código para compilarlo sobre Java 1.6, con lo que se puede ejecutar sin problemas en versiones antiguas de Windows y Linux (en los ubuntu 10.04 del aula, sin ir mas lejos), esto se ha conseguido evitando usar el operador Diamond, así como otras características propias de java 1.7 como puede ser la conversión directa de Object a Int.
- Posibilidad de Cortar, Copiar y Pegar figuras de un lienzo a otro.
- Mostrar un panel con el orden de las capas (Shapes) y permitir cambiar el orden de las mismas al gusto del usuario.

1.7 Explicación del sistema propio de clases:

El diseño del sistema es muy sencillo, se parte de una clases llamada Config donde se almacena la configuración de las figuras en un estado de tiempo, dicha instancia se clona cada vez que se crea una figura nueva con lo que cada “Shape” tiene su propia instancia de config, así es muy sencillo alterar las propiedades dependiendo de cual sea la figura seleccionada.

Gracias a este sistema además se permite el poder cargar la lista de “figuras” y moverla de arriba hacia abajo mediante los botones, nótese que cuando se refiere a subir la imagen realmente es bajar en la lista y viceversa.

Además se ha implementado relleno radial y diagonal así como la posibilidad de copiar, cortar y pegar figuras tanto dentro de un mismo lienzo como hacia otros.

1.8 Conclusiones:

A lo largo de esta asignatura hemos profundizado en multitud de conceptos sobre elementos multimedia que si no fuera por el carácter práctico de la asignatura no podríamos haber experimentado de

igual forma, tal y como pueden ser las operaciones sobre imagen o los distintos tipos de compresión tanto de imagen como de video, además el desarrollo de una aplicación de mediana envergadura en un entorno de desarrollo me ha servido para mejorar mi destreza a la hora de trabajar con proyectos, considero un acierto el sistema de evaluación y el enfoque dado a dicha asignatura.

1.9 Bibliografía:

- Documentación oficial de Java: <https://docs.oracle.com/javase/7/docs/api/>
- Búsqueda de errores y ejemplos de uso: <http://stackoverflow.com/questions/tagged/java>
- Wrapper de varios sistemas de conexión para la webcam: <http://webcam-capture.sarxos.pl/>
- Control de selección de color que integra un JColorChooser dentro de un JLabel:
<http://stackoverflow.com/questions/26565166/how-to-display-a-color-selector-when-clicking-a-button/30433662#30433662>
- Control de selección de “trazo” que hereda de JComboBox y muestra los estilos como elementos seleccionables: <http://www.jfree.org/jcommon/api/org/jfree/ui/StrokeChooserPanel.html>