

Documentación Paint

Sistemas Multimedia

07/07/2016

Manuel Alonso Braojos

Índice:

Introducción:	2
1.Requisitos Funcionales:	2
2 Requisitos No Funcionales:	14
3 Diseño:	15
3.1 Ventana Principal:	15
3.2 Clases:	19
3.2.1 Lienzo:	19
3.2.2 Clase Figura:	19
3.2.3 Clase Arco:	19
3.2.4 Clase Curva:	19
3.2.5 Clase Elipse:	20
3.2.6 Clase Línea:	20
3.2.7 Clase Polígono:	20
3.2.8 Clase Punto:	20
3.2.9 Clase Rectángulo:	20
3.2.10 Clase Rectángulo Redondeado:	20
3.2.11 MiLinea2D:	20
3.2.12 Lienzo2DImagen:	20
3.3 Ventanas Internas:	21
3.3.1 VentanaInterna:	21
3.3.2 TamañoVentana:	22
3.3.3 VentanaInternaReproductor:	22
3.3.4 VentanaInternaJMFPlayer:	22
3.3.5 VentanaInternaGrabador:	23
3.4 Operaciones de imagen:	23
3.4.1 SepiaOp:	23
3.4.2 UmbralizacionOp:	23
3.4.3 MultiplicacionOp:	23
3.4.4 VisionNocturnaOp:	24
Bibliografía:	24

Introducción:

Esta práctica consiste en el desarrollo de una aplicación multimedia que nos permita reproducir y capturar audio y video, además de poder visualizar, crear y editar imágenes.

La aplicación se ha desarrollado en el lenguaje orientado a objetos java, en el entorno de desarrollo libre NetBeans.

1.Requisitos Funcionales:

Los requisitos funcionales se pueden separar en función de la acción que realice el usuario, lo que es lo mismo, la aplicación multimedia realizará una tarea u otra dependiendo de lo que le indique el usuario.

Como funcionalidades principales del programa destacan las siguientes:

- **RF-1. Funcionalidades generales:** la aplicación permitirá al usuario crear, abrir y guardar archivos de imagen. También permitirá abrir archivos de audio y video.
 - **RF-1.1 Nueva Ventana:** una vez este ejecutada la aplicación, esta permitirá al usuario crear una nueva ventana interna donde el usuario podrá dibujar, mediante el botón "Nuevo" situado en la barra general o en el menú archivo. Una vez seleccionado el botón nos aparecerá una ventana de diálogo en que tendremos que elegir el tamaño de la ventana interna.



Ilustración 1-Icono del botón "Nuevo"

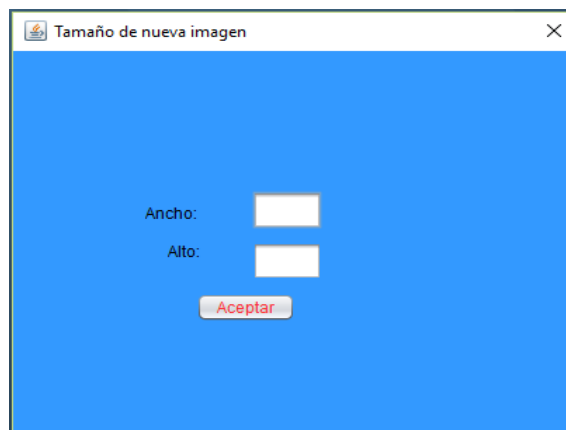


Ilustración 2-Ventana de diálogo de tamaño

- **RF-1.2 Abrir archivo:** la aplicación permitirá al usuario abrir un archivo de imagen, de audio o de video. Todo ello el usuario podrá hacerlo

mediante el botón "Abrir" situado en la barra general o en el menú archivo. Una vez seleccionado nos aparecerá un diálogo en el que seleccionaremos el archivo que el usuario desea abrir.



Ilustración 3-Icono del botón "Abrir"

- **RF-1.3 Guardar imagen:** la aplicación permitirá al usuario guardar un archivo de imagen seleccionando el botón "Guardar" situado en la barra general o en el menú archivo. Una vez seleccionado el botón nos aparecerá un dialogo en el que indicaremos la ubicación, el nombre y el formato que el usuario desee para el archivo.



Ilustración 4-Icono del botón Guardar

- **RF-2. Dibujar:** la aplicación permitirá al usuario elegir la figura que desea dibujar mediante botones, así como los atributos individuales de cada figura como el color del trazo, el color del relleno, el tipo de trazo , el tipo de relleno y la opción de alisado.
También permitirá al usuario elegir el tipo transparencia que desea aplicar a la figura mediante un JSlider y el grosor que desee darle al trazo de la figura mediante un JSpinner.

- **RF-2.1 Seleccionar Trazo:** la aplicación permitirá al usuario elegir entre trazo continuo o discontinuo. Para activar el trazo discontinuo que consistirá en el uso de pequeñas rayas, el usuario deberá seleccionar el botón de dos posiciones pertinente, en caso de que desee que el trazo vuelva a ser continuo deselegionará el botón activado anteriormente. A continuación podemos ver el icono del botón.



Ilustración 5-Icono de botón de trazo discontinuo

- **RF-2.2 Seleccionar Grosor:** la aplicación permitirá al usuario elegir el grosor con el que desea pintar las figuras. El usuario podrá aumentar o disminuir el grosor del trazo median el JSpinner asociado al grosor que irá desde 1 a 99.



Ilustración 6-JSpinner de valor de grosor

- **RF-2.3 Seleccionar Color:** la aplicación permitirá al usuario elegir el color del trazo y el color de relleno que desee.
 - ✓ **RF-2.3.1 Seleccionar Color de trazo:** la aplicación permitirá al usuario elegir el color del trazo de la figura con el que desee dibujar. Para ello el usuario deberá tener seleccionado el botón "Color1" y realizar los requisitos funcionales **2.3.3, 2.3.4, 2.3.5, 2.3.6, 2.3.7, 2.3.8, 2.3.9**, dependiendo del color que desee él

usuario. Cambiará su color de fondo por el del color seleccionado.

- ✓ **RF-2.3.2 Seleccionar Color de relleno:** la aplicación permitirá al usuario elegir el color que desee para el relleno de la figura. Para ello el usuario deberá tener seleccionado el botón "Color2" y realizar los requisitos funcionales **2.3.3, 2.3.4, 2.3.5, 2.3.6, 2.3.7, 2.3.8, 2.3.9**, dependiendo del color que desee el usuario. Cambiará su color de fondo por el del color seleccionado.
- ✓ **RF-2.3.3 Seleccionar color negro:** la aplicación permitirá al usuario seleccionar el color negro siempre y cuando tenga abierta una imagen o un lienzo de dibujado.
- ✓ **RF-2.3.4 Seleccionar color rojo:** la aplicación permitirá al usuario seleccionar el color rojo siempre y cuando tenga abierta una imagen o un lienzo de dibujado.
- ✓ **RF-2.3.5 Seleccionar color azul:** la aplicación permitirá al usuario seleccionar el color azul siempre y cuando tenga abierta una imagen o un lienzo de dibujado.
- ✓ **RF-2.3.6 Seleccionar color blanco:** la aplicación permitirá al usuario seleccionar el color blanco siempre y cuando tenga abierta una imagen o un lienzo de dibujado.
- ✓ **RF-2.3.7 Seleccionar color amarillo:** la aplicación permitirá al usuario seleccionar el color amarillo siempre y cuando tenga abierta una imagen o un lienzo de dibujado.
- ✓ **RF-2.3.8 Seleccionar color verde:** la aplicación permitirá al usuario seleccionar el color verde siempre y cuando tenga abierta una imagen o un lienzo de dibujado.
- ✓ **RF-2.3.9 Seleccionar color de paleta:** la aplicación permitirá al usuario seleccionar el botón de la paleta siempre y cuando tenga abierta una imagen o un lienzo de dibujado. Una vez seleccionado dicho botón aparecerá un diálogo en el que el usuario seleccionará un color de una amplia gama de colores, como podemos ver en la siguiente ilustración.

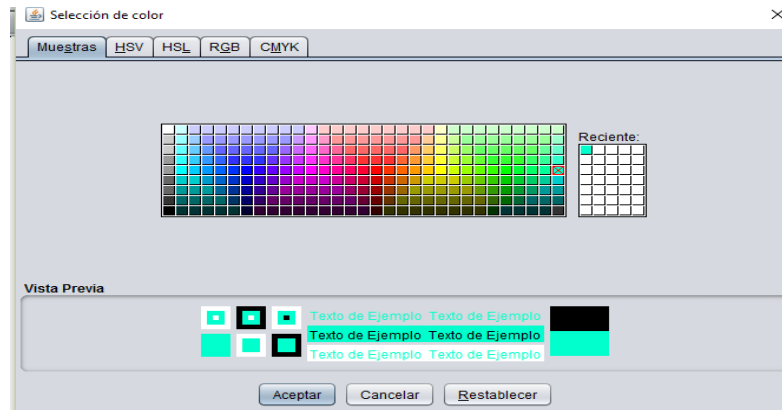


Ilustración 7-Paleta de colores



Ilustración 8-Panel de colores

- **RF-2.4 Seleccionar Relleno:** una vez hecho el **RF-2.3**, la aplicación permitirá al usuario elegir mediante botones el tipo de relleno que desee. El tipo de relleno podrá ser liso, con degradado horizontal o con degradado vertical. El degradado será desde el color de relleno elegido hasta el color del trazo elegido.
 - ✓ **RF-2.4.1 Seleccionar Relleno Liso:** la aplicación permitirá al usuario crear una figura con un color liso seleccionando el botón de relleno cuyo icono lo vemos en la siguiente imagen.



Ilustración 9-Icono de botón de relleno

- ✓ **RF-2.4.2 Seleccionar Relleno con degradado horizontal:** Una vez el usuario haya realizado el requisito funcional **2.4.1**, seleccionará el botón "degraH" para dibujar la figura con degradado horizontal.
 - ✓ **RF-2.4.2 Seleccionar Relleno con degradado vertical:** Una vez el usuario haya realizado el requisito funcional **2.4.1**, seleccionará el botón "degraV" para dibujar la figura con degradado vertical.



Ilustración 10-Tipos de Relleno

- **RF-2.5 Selección de transparencia:** la aplicación permitirá al usuario establecer un nivel de transparencia con el que dibujar la figura, siempre y cuando tenga abierta una ventana de imagen o un lienzo de dibujado. El usuario editará el grado de transparencia mediante un JSlider que irá desde el estado opaco hasta totalmente transparente.
- **RF-2.6 Selección de Alisado de bordes:** la aplicación permitirá al usuario activar o desactivar la mejora en el proceso de renderizado que corresponder al alisado de bordes, seleccionando el botón pertinente cuyo icono vemos a continuación.



Ilustración 11-Icono del botón de alisado de bordes

- **RF-2.7 dibujado de Figuras:** la aplicación permitirá al usuario elegir entre una serie de figuras para dibujar. Para el dibujado de figuras debe de estar abierto el lienzo de dibujado o el lienzo de una imagen, donde se utilizaran los eventos del ratón **moussePressed** que indica el punto donde se ha presionado con el puntero del ratón, **mousseDragged** que corresponderá con el punto donde se encuentre el puntero del ratón sin que se suelte, **mousseReleased** que corresponde al punto donde se encuentra el puntero del ratón cuando se suelta el botón. Cada figura se dibujará seleccionando su botón correspondiente y una vez seleccionado con los eventos vistos y los puntos necesarios para cada figura.
 - ✓ **RF-2.7.1 Punto:** el usuario seleccionará el botón correspondiente y dibujará la figura punto, que se realizara donde se haga el **moussePressed**.



Ilustración 12-Icono figura punto

- ✓ **RF-2.7.2 Línea:** el usuario seleccionará el botón correspondiente y dibujará la figura línea. El primer punto de la figura será cuando se presione el botón del ratón y el punto final será cuando se suelte, es decir, el evento de ratón **mousseReleased**. Se podrá ir viendo la creación de la línea dinámicamente con el evento de ratón **mousseDragged**.



Ilustración 13-Icono figura línea

- ✓ **RF-2.7.3 Rectángulo:** el usuario seleccionará el botón correspondiente y dibujará la figura rectángulo. El primer punto de la figura será donde el usuario presione el botón del ratón, el punto final será el extremo de la diagonal del rectángulo que se determinará con el evento de ratón `mouseReleased`. Se podrá ir viendo la creación del rectángulo dinámicamente con el evento de ratón `mouseDragged`.



Ilustración 14-Icono de la figura Rectángulo

- ✓ **RF-2.7.4 Elipse:** el usuario seleccionará el botón correspondiente y dibujará la figura elipse. El primer punto de la figura será donde el usuario presione el botón del ratón, el punto final será el extremo de la diagonal del rectángulo que contiene la elipse que se determinará con el evento de ratón `mouseReleased`. Se podrá ir viendo la creación del rectángulo dinámicamente con el evento de ratón `mouseDragged`.



Ilustración 15-Icono de la figura Elipse

- ✓ **RF-2.7.5 Rectángulo Redondeado:** el usuario seleccionará el botón correspondiente y dibujará la figura rectángulo con las esquinas redondeadas. El primer punto de la figura será donde el usuario presione el botón del ratón, el punto final será el extremo de la diagonal del rectángulo que se determinará con el evento de ratón `mouseReleased`. Se podrá ir viendo la creación del rectángulo redondeado dinámicamente con el evento de ratón `mouseDragged`.



Ilustración 16-Icono de la figura Rectángulo Redondeado

- ✓ **RF-2.7.6 Arco:** el usuario seleccionará el botón correspondiente y dibujará la figura arco. El primer punto de la figura será donde el usuario presione el botón del ratón, el punto final será el extremo de la diagonal del rectángulo que contiene el arco que se determinará con el evento de ratón `mouseReleased`. Se podrá ir viendo la creación del rectángulo dinámicamente con el evento de ratón `mouseDragged`.



Ilustración 17-Icono de la figura Arco

- ✓ **RF-2.7.6 Curva con un punto de control:** el usuario seleccionará el botón correspondiente y dibujará la figura curva con un punto de control. El usuario indicará el punto inicial haciendo clic con el ratón, una vez tenemos el primer punto presionamos el botón

del ratón donde queramos que se encuentre el punto final de la figura y sin soltar movemos el ratón y la curva se irá generando dinámicamente con el evento de ratón MousseDragged. Una vez el usuario tiene la figura como quiere suelta el botón del ratón y la figura quedará creada.



Ilustración 18-Icono de la figura Curva con un punto de control

- ✓ **RF-2.7.7 Polígono:** el usuario seleccionará el botón correspondiente y dibujará la figura polígono. El usuario podrá realizar un polígono de tantos vértices cuantos quiera, el usuario indicará donde quiere que se encuentre cada vértice presionando el botón del ratón, cuando quiera terminar la figura realizará un doble clic indicando que es el último punto. El doble clic se contará con el evento de ratón mousseClicked.



Ilustración 19-Icono de la figura polígono

- **RF-2.8 Editar atributos de figura:** una vez realizado el **RF-2.7** el usuario podrá editar los atributos de cada figura y moverla de posición. El usuario seleccionará el botón de editar y con el evento de ratón moussePressed seleccionará la figura a editar, esta figura quedará rodeada mediante un rectángulo (boundingbox). La figura seleccionada se le podrá modificar sus atributos (color, trazo, relleno, etc.) realizando los requisitos funcionales **2.1, 2.2, 2.3, 2.4, 2.5, 2.6**.
- **RF-3 Tratamiento de imágenes:** la aplicación permitirá al usuario una vez realizado el **RF-1.2**, realizar varias operaciones sobre la imagen cargada en el lienzo. Las operaciones de imagen se encuentran en la barra de operaciones de imagen situada en la parte inferior de la aplicación.
 - **RF-3.1 Duplicado de imagen:** la aplicación permitirá al usuario duplicar la imagen abierta mediante el **RF-1.2** en una nueva ventana. Para ello el usuario deberá seleccionar el botón duplicar cuyo icono vemos a continuación:



Ilustración 20-Icono de la operación de Duplicado

- **RF-3.2 Modificar brillo de imagen:** la aplicación permitirá al usuario modificar el brillo de una imagen cargada previamente mediante el **RF-1.2**, para modificar el brillo el usuario dispondrá de una barra deslizadora en la barra de operaciones de imagen con la que podrá ajustar el brillo según desee. Una vez se seleccione otra operación sobre la imagen el brillo quedará aplicado de forma definitiva.

- **RF-3.3 Filtros:** la aplicación permitirá al usuario aplicar filtros sobre la imagen cargada previamente mediante el **RF-1.2**, para ello el usuario dispondrá de un menú deslizante situado en la barra de operaciones de imagen. El usuario dispondrá de los filtros Media, binomial, enfoque, relieve y laplaciano.

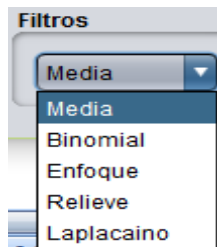


Ilustración 21-Menú deslizante de filtros

- **RF-3.4 Contraste:** la aplicación permitirá al usuario aplicar tres contrastes diferentes (normal, iluminado y oscurecido) a la imagen previamente cargada. Para ello el usuario seleccionará el botón dependiendo del contraste que desee aplicar a la imagen.



Ilustración 22-Tipos de contrastes

- **RF-3.5 Negativo:** la aplicación permitirá al usuario aplicar el negativo a una imagen previamente cargada en el **RF-1.2**, para ello el usuario seleccionará el botón de negativo que se encuentra en la barra de operaciones de imagen y cuyo icono veremos a continuación.



Ilustración 23-Icono de la operación negativo

- **RF-3.6 Transformación a niveles de gris:** la aplicación permitirá al usuario transformar cualquier imagen previamente cargada de color a una imagen en niveles de grises. Para ello el usuario seleccionará el botón Escala de grises situado en la barra de operaciones de imagen y cuyo icono veremos a continuación:



Ilustración 24-Icono de la Transformación a nivel de grises

- **RF-3.7 Giro libre:** la aplicación permitirá al usuario rotar una imagen previamente cargada. Para ello el usuario dispondrá de una barra deslizadora en la barra de operaciones de imagen con la que podrá darle el ángulo de rotación que desee.
- **RF-3.8 Giro 90°:** la aplicación permitirá al usuario girar 90 grados una imagen previamente cargada. Para ello el usuario dispondrá de un botón en la barra de operaciones de imagen cuyo icono lo vemos a continuación.



Ilustración 25-Icono de la operación giro 90°

- **RF-3.9 Giro 180°:** la aplicación permitirá al usuario girar 180 grados una imagen previamente cargada. Para ello el usuario dispondrá de un botón en la barra de operaciones de imagen cuyo icono lo vemos a continuación.



Ilustración 26-Icono de la operación giro 180°

- **RF-3.10 Giro 270°:** la aplicación permitirá al usuario girar 270 grados una imagen previamente cargada. Para ello el usuario dispondrá de un botón en la barra de operaciones de imagen cuyo icono lo vemos a continuación.



Ilustración 27-icono de la operación giro 270°

- **RF-3.11 Aumentar Escalado:** la aplicación permitirá al usuario aumentar el escalado a una imagen previamente cargada. Para ello el usuario deberá de seleccionar el botón en la barra de operaciones de imagen cuyo icono lo vemos a continuación.



Ilustración 28-Icono de la operación de aumento de escalado

- **RF-3.12 Disminuir Escalado:** la aplicación permitirá al usuario disminuir el escalado de una imagen previamente cargada. Para ello el usuario deberá de seleccionar el botón en la barra de operaciones de imagen cuyo icono lo vemos a continuación.



Ilustración 29-Icono de la operación de disminución de escalado

- **RF-3.13 Sepia:** la aplicación permitirá al usuario aplicar el filtro sepia a una imagen que el usuario haya cargado previamente. El usuario aplicará este filtro a las imágenes haciendo que tengan un color rojo anaranjado oscuro. Para realizar esta operación el usuario dispondrá de un botón en la barra de operaciones de imagen y que tiene el icono que vemos a continuación.



Ilustración 30-Icono de la operación de imagen Sepia

- **RF-3.14 Operador sinoidal:** la aplicación permitirá al usuario aplicarle a una imagen previamente cargada la función seno $f(x) = |\sin(w \cdot x)|$ donde w representa la velocidad angular. Para llevar a cabo esta operación el usuario deberá de seleccionar el botón correspondiente situado en la barra de operaciones de imagen y cuyo icono podemos ver en la siguiente imagen.



Ilustración 31-Icono de la operación sinoidal

- **RF-3.15 Suma Binaria:** la aplicación permitirá al usuario realizar la suma binaria de dos imágenes previamente cargadas mediante el **RF-1.2**, esta

operación toma el valor de cada pixel de cada una de la imagen y realiza la suma de estos, el resultado se verá en una nueva ventana. Para llevar a cabo esta operación el usuario seleccionará el botón "suma" situado en la barra de operaciones de imagen y cuyo icono podemos ver a continuación.



Ilustración 32-Icono de la operación binario suma

- **RF-3.16 Resta Binaria:** la aplicación permitirá al usuario realizar la resta binaria de dos imágenes previamente cargadas mediante el **RF-1.2**, esta operación toma el valor de cada pixel de cada una de la imagen y realiza la resta de estos, el resultado se verá en una nueva ventana. Para llevar a cabo esta operación el usuario seleccionará el botón "resta" situado en la barra de operaciones de imagen y cuyo icono podemos ver a continuación.



Ilustración 33-Icono de la operación binaria resta

- **RF-3.17 Multiplicación Binaria:** la aplicación permitirá al usuario realizar la multiplicación binaria de dos imágenes previamente cargadas mediante el **RF-1.2**, esta operación toma el valor de cada pixel de cada una de la imagen y realiza la multiplicación de estos, el resultado se verá en una nueva ventana. Para llevar a cabo esta operación el usuario seleccionará el botón "multiplicación" situado en la barra de operaciones de imagen y cuyo icono podemos ver a continuación.



Ilustración 34-Icono de la operación binaria Multiplicación

- **RF-3.18 Mezcla de dos imágenes:** la aplicación permitirá al usuario mezclar dos imágenes previamente cargadas mediante el **RF-1.2**. Para realizar la mezcla de dos imágenes el usuario dispondrá de una barra deslizador situada en la barra de operaciones de imagen, con dicha barra deslizador se irá realizando una suma binaria interactiva de las imágenes.
- **RF-3.19 Tintado:** la aplicación permitirá al usuario "tintar" una imagen previamente cargada mediante el **RF-1.2** del color que se haya seleccionado en el **RF-2.3.1**. Para realizar esta operación el usuario deberá de seleccionar el botón de tintado situado en la barra de operaciones de imagen y que tiene el siguiente icono.



Ilustración 35-Icono de la operación tintado

- **RF-3.20 Umbralización:** la aplicación permitirá al usuario realizar la operación de umbralización basada en intensidad sobre una imagen previamente cargada en el **RF-1.2**. Para llevar a cabo esta operación el usuario dispone de una barra deslizadora situada en la barra de operaciones de imagen. Si los píxeles de la imagen supera el valor de la barra deslizadora se les asignará el valor 255 y al resto de píxeles se les asignará el valor 0.
- **RF-3.21 Visión Nocturna (operación propia):** la aplicación permitirá al usuario aplicarle un filtro de visión nocturna a una imagen previamente cargada en el **RF-1.2**. El usuario podrá llevar a cabo esta operación seleccionando el botón correspondiente situado en la barra de operaciones de imagen y cuyo icono podemos ver a continuación.



Ilustración 36-Icono de la operación de visión nocturna

- **RF-4 Audio y video:** la aplicación permitirá al usuario reproducir audio, grabar sonidos del exterior mediante el micrófono y reproducir videos. Para abrir un archivo de audio y de video el usuario deberá de realizar el **RF-1.2** abriéndose una ventana de tipo `VentanaInternaReproductor` y `VentanaInternaJMFPlater` respectivamente.
 - **RF-4.1 Reproducir:** Una vez el usuario haya abierto un archivo de audio o de video mediante el **RF-1.2** podrá reproducirlos mediante el botón "play" situado en la barra de audio/video.



Ilustración 37-Icono de botón Play

- **RF-4.2 Grabar audio:** la aplicación permitirá al usuario grabar un archivo de audio mediante el micrófono. Para grabar el archivo de audio el usuario seleccionará el botón grabar, una vez selecciona haya seleccionado el botón aparecerá un diálogo en el que deberá de seleccionar el archivo de audio sobre el que va a realizar la grabación. Una vez seleccionado el archivo se abrirá una ventana de tipo `VentanaInternaGrabador` y la aplicación comenzará a grabar automáticamente.



Ilustración 38-Icono de botón Grabar

- **RF-4.2 Stop:** una vez el usuario haya realizado el **RF-4.1** o el **RF-4.2**, el usuario podrá parar la reproducción de un audio o video en caso de estar reproduciéndolos, o para la grabación de un archivo de audio.



Ilustración 39-Icono de botón Stop

- **RF-5 WebCam:** la aplicación permitirá al usuario la posibilidad de lanzar la WebCam, para ello el usuario deberá de seleccionar el botón de la cámara situado en la barra de Audio/video. Una vez seleccionado el botón se abrirá una ventana de tipo VentanaInternaCamara en la que aparecerá lo que esté captando a tiempo real la WebCam.



Ilustración 40-Icono del botón de la WebCam

- **RF-6 Captura:** la aplicación permitirá al usuario realizar una captura de un video previamente abierto en el **RF-1.2** o de la WebCam previamente abierta en el **RF-5**. Para ello el usuario deberá de seleccionar el botón "Captura" situado en la barra de Audio/video y cuyo icono podemos ver a continuación.



Ilustración 41-Icono de botón Capturar

- **RF-7 Menú Archivo:** desde este menú situado en el menú principal en la parte superior de la aplicación, el usuario podrá crear, abrir y guardar archivos de imagen, también permitirá abrir archivos de audio y video como hemos visto en los **requisitos funcionales 1.1, 1.2, 1.3**.

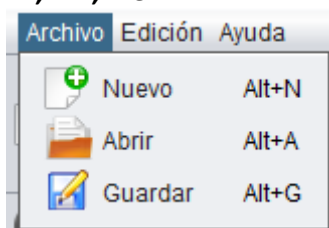


Ilustración 42-Menú archivo

- **RF-8 Menú Edición:** desde este menú situado en el menú principal en la parte superior de la aplicación, el usuario podrá ocultar o visualizar las diferentes barras de herramientas.

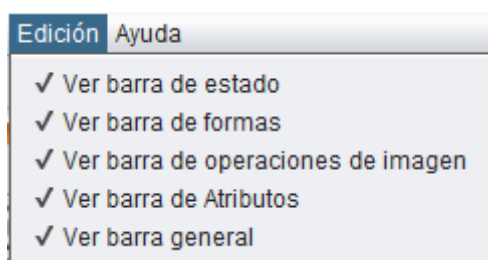


Ilustración 43-Menú edición

- **RF-9 Menú ayuda:** en este menú situado en el menú principal en la parte superior de la aplicación, el usuario encontrará opción **Acerca de**, una vez

seleccionada esta opción se mostrará una ventana de diálogo en la que se mostrará información de la aplicación y del autor.

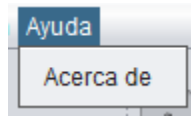


Ilustración 44-Menú ayuda

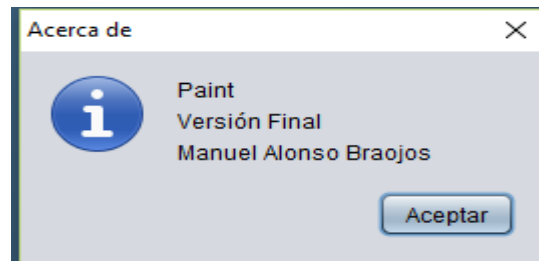


Ilustración 45-Ventana de diálogo

2 Requisitos No Funcionales:

- **RNF-1:** los archivos multimedia que la aplicación abrirá serán solo los que tenga un formato establecido previamente.
 - ✓ Imagen: jpg, jpeg, png.
 - ✓ Audio: au, wav, mp3
 - ✓ video: avi
- **RNF-2:** Cada figura tendrá sus propios atributos.
- **RNF-3:** la barra de estados mostrará el nombre de la figura que se esté dibujando.
- **RNF-4:** se mostrará las coordenadas en los ejes X e Y donde se encuentre el puntero del ratón.
- **RNF-5:** se mostrará el valor de los componentes RGB del pixel donde se encuentre el puntero del ratón.
- **RNF-6:** las operaciones de dibujado y de imagen no podrán ser usadas cuando lo que haya abierto sea un audio, un video, una ventana de grabación o la webcam.

3 Diseño:

3.1 Ventana Principal:

Es la ventana principal de la aplicación y hereda de JFrame. En esta clase se encontrarán todos los elementos necesarios para interactuar con el programa, ya que está compuesta de menús, barras de herramientas que heredan de JToolBar. En esta clase también se podrá ir creando ventanas internas que heredan de JInternalFrame. A continuación veremos una descripción de los elementos que la componen:

Menú principal: es el menú principal de tipo JMenuBar, está situado en la parte superior de la aplicación. Cuenta con cuatro menús:

- **Archivo:** menú de tipo JMenu en el que se encuentra las funcionalidades generales.
 - **Nuevo:** opción de tipo JMenuItem con la que el usuario podrá crear un nuevo objeto de la clase "**VentanaInterna**", dicho nuevo objeto será el lienzo en blanco donde el usuario podrá dibujar. Al seleccionar esta opción aparecerá un diálogo de la clase "**TamañoVentana**" donde el usuario podrá elegir el tamaño de la imagen en blanco sobre la que se dibujará.
 - **Abrir:** opción de tipo JMenuItem, cuando el usuario la selecciona mostrará un diálogo de selección JFileChooser. Una vez elegido un archivo que tenga un formato aceptado por la aplicación se creará la correspondiente ventana interna.
 - **Guardar:** opción de tipo JMenuItem, con esta opción el usuario podrá guardar una imagen que esté en la ventana interna seleccionada. Cuando el usuario la selecciona mostrará una diálogo de guardado JFileChooser, el usuario deberá de elegir el nombre y seleccionar el formato en el menú desplegable de dicho diálogo.
- **Edición:** menú de tipo JMenu, en este se encuentran cinco opciones de tipo JCheckBoxMenuItem con las que se podrá visualizar u ocultar las distintas barras de herramientas de la aplicación.
- **Ayuda:** menú de tipo JMenu que solo contiene la opción "Acerca de" de tipo JMenuItem. Cuando el usuario seleccione esta opción se mostrará un diálogo de tipo JOptionPane que mostrará información de la aplicación.

Barra de Herramientas General: barra de herramienta de tipo JToolBar, está situada en la parte superior de la aplicación, en ella se encuentran las funcionalidades generales.

- **Nuevo:** botón de tipo JButton con la que el usuario podrá crear un nuevo objeto de la clase "**VentanaInterna**", dicho nuevo objeto será el lienzo en blanco donde el usuario podrá dibujar. Al seleccionar este botón aparecerá un diálogo de la clase "**TamañoVentana**" donde el usuario podrá elegir el tamaño de la imagen en blanco sobre la que se dibujará.
- **Abrir:** botón de tipo JButton, cuando el usuario lo seleccione mostrará un diálogo de selección JFileChooser. Una vez elegido un archivo que tenga un formato aceptado por la aplicación se creará la correspondiente ventana interna.
- **Guardar:** botón de tipo JButton, con esta opción el usuario podrá guardar una imagen que esté en la ventana interna seleccionada. Cuando el usuario seleccione el botón mostrará una diálogo de guardado JFileChooser, el usuario deberá de elegir el nombre y seleccionar el formato en el menú desplegable de dicho diálogo.

Barra de formas: barra de herramientas de tipo JToolBar, está situada en la parte superior de la aplicación junto a la barra de herramientas General, en esta barra se encuentran los botones que corresponde a cada figura y el botón de editar, estos botones serán de dos posiciones JToggleButton y se encuentran todos en un mismo grupo de botones.

Barra de atributos: barra de herramientas de tipo JToolBar, está situada en la parte superior de la aplicación junto a la barra de formas, en esta barra se encuentran las diferentes opciones para asignarle atributos a las figuras que el usuario dibuje como los botones de selección de colores de tipo JToggleButton, la selección de color que se hace mediante un JSpinner que va de 1 a 99, el botón de tipo JToggleButton que corresponde al tipo de trazo, los botones del tipo de relleno que son de tipo JToggleButton y una barra deslizador del tipo JSlider con la que se determinará la transparencia de las figuras que el usuario dibuje.

Barra de operaciones de imagen: barra de herramientas de tipo JToolBar, está situada en la parte inferior de la aplicación. En esta barra encontramos las diferentes paneles de tipo JPanel con las operaciones que el usuario puede aplicarle a una imagen.

- **Brillo:** el usuario dispondrá de una barra deslizador de tipo JSlider con la que podrá modificar el brillo. La operación de brillo se realizará sobre la imagen original.
- **Filtros:** el usuario dispondrá de un JComboBox con la que el usuario podrá elegir diferentes filtros.

- **Contraste:** en el panel de contraste el usuario dispondrá de tres botones de tipo JButton con los que el usuario podrá aplicar a la imagen tres contrastes diferentes: normal, iluminado, oscurecido.
- **Operaciones:** en este panel el usuario dispondrá de siete botones de tipo JButton con los que podrá realizar diferentes operaciones sobre la imagen, tales como: operación sinoidal, operación sepia, negativo, escalado de grises, duplicado de imagen, tintado y visión nocturna.
- **Rotación:** en este panel el usuario dispondrá de una barra deslizador con la que el usuario podrá dar a la imagen el ángulo de rotación que desee. Además dispondrá de tres botones de tipo JButton con los que el usuario podrá aplicarle a la imagen una rotación de 90°, 180°, 270°.
- **Escalado:** en este panel el usuario dispondrá de dos botones de tipo JButton con los que el usuario podrá aumentar y disminuir la escala de una imagen.
- **Operaciones Binarias:** en este panel usuario dispondrá de tres botones de tipo JButton con los que el usuario podrá realizar la suma, resta y multiplicación de los pixeles de dos imagenes, para la multiplicación se hará uso de la clase **multiplicacionOP**. El resultado de aplicar las diferentes operaciones de imagen se mostrará en una nueva ventana interna de tipo **VentanaInterna**.
- **Deslizador Binario:** el usuario dispondrá de una barra deslizador de tipo JSlider con la que llevará a cabo la suma binaria interactiva de dos imágenes.
- **Umbralización:** el usuario dispondrá de una barra deslizador de tipo JSlider con la que el usuario podrá aplicar la umbralización que desee a la imagen seleccionada.

Barra de audio/video: barra de herramientas de tipo JToolBar, en esta barra se encuentran las opciones para la reproducción y pausado de un audio, video y grabación, además de la opción de abrir la webCam y de tomar una captura.

- **Play:** botón de tipo JButton con el que el usuario teniendo una ventana interna de la clase VentanaInternaReproductor o VentanaInternaJMFPlayer podrá iniciar la reproducción de un archivo de audio o de video, utilizando en caso del audio un objeto de la clase **SMClipPlayer** y en caso del video un objeto de la clase **Player**.
- **Stop:** botón de tipo JButton con el que el usuario teniendo una ventana interna de la clase VentanaInternaReproductor, VentanaInternaJMFPlayer o VentanaInternaGrabador podrá parar la reproducción de un archivo de audio o de video y la grabación de un archivo de audio. Para ello se utilizará un objeto de la clase **SMClipPlayer** en caso de ser un archivo de audio, un objeto de la

clase **Player** en caso de ser un archivo de video y un objeto de la clase **SMRecorder** en caso de ser una grabación.

- **Record:** botón de tipo JButton con el que el usuario podrá grabar un archivo de audio abriendo una ventana interna de la clase **VentanaInternaGrabador**.
- **WebCam:** botón de tipo JButton con el que el usuario podrá ejecutar la WebCam abriendo una ventana interna de la clase **VentanaInternaCamara** en la que se mostrará la imagen que capte la WebCam.
- **Captura:** botón de tipo JButton con el que el usuario podrá realizar una captura de un video o de la imagen que esté mostrando la WebCam. La captura se mostrará en una nueva ventana interna de la clase **VentanaInterna**.

Barra de Estado: etiqueta JLabel que actualiza su valor mostrando el nombre de la figura seleccionada.

Etiquetas de coordenadas: etiquetas situadas en la parte inferior de tipo JLabel que muestran las coordenadas X e Y de la posición del puntero. Para ello en la clase **VentanaInterna** en el evento de ratón **lienzo2D1MouseClicked** obtenemos la información del punto sobre el que se encuentra el puntero del ratón con el método **MouseInfo.getPointerInfo().getLocation()**.

Etiqueta RGB: etiqueta situada en la parte inferior de la aplicación de tipo JLabel que muestra el valor de los componentes RGB del pixel sobre el que está situado el puntero del ratón. Para ello en la clase **VentanaInterna** en el evento de ratón **lienzo2D1MouseClicked** obtenemos el valor RGB del pixel con el método **getPixelColor(x, y)** de la clase **Robot**.

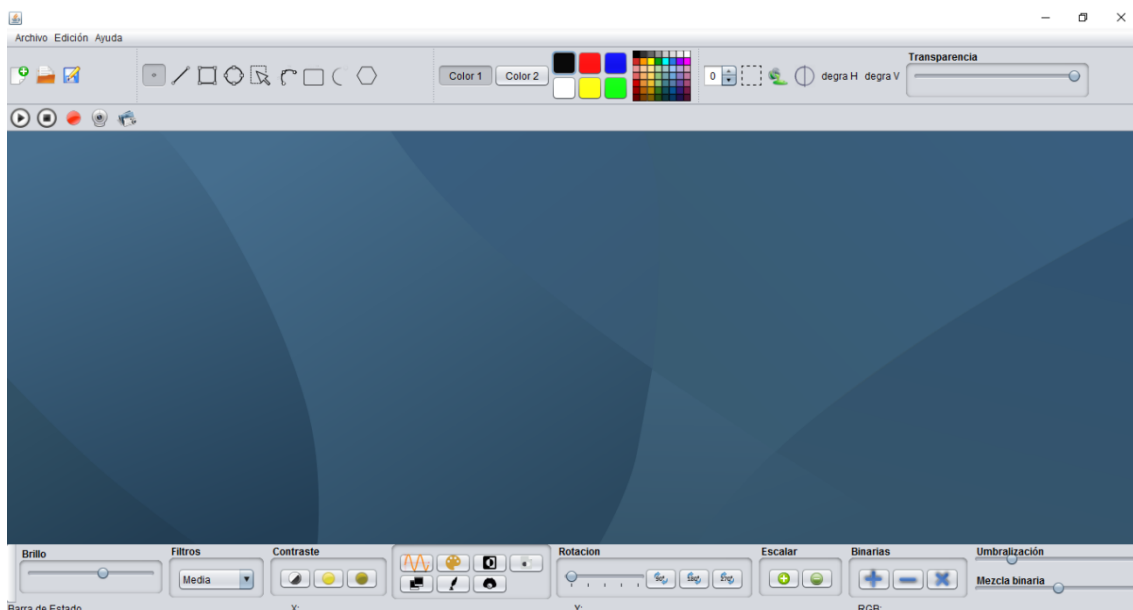


Ilustración 46-Ventana Principal

3.2 Clases:

3.2.1 Lienzo:

La clase lienzo hereda de JPanel y se encarga de dibujar las figuras, en ella se dispone de los atributos de las figuras con sus respectivos métodos get y set para consultar y modificar los atributos.

La creación, modificación y dibujo se realiza con el método `createShape` en el que dependiendo de la figura seleccionada se creará un objeto de la clase de la figura seleccionada y los eventos de ratón `mousePressed`, `mouseDragged` y `mouseReleased`.

Las figuras creadas se irán guardando en un vector de la clase `Figura` por lo que se podrá dibujar más de una figura en el lienzo, cada una con sus atributos. Este vector de figuras será dibujado en el lienzo por el método **`paint()`**, cada figura con sus atributos independientes pudiendo como vemos a continuación editar los atributos por separado.

Para editar una de las figuras seleccionadas se dispone de los métodos **`getSelectedShape(Point2D p)`** que es el encargado de obtener la figura seleccionada y el método **`setLocation(Figura fig, Point2D pos)`** con el que se modifica la posición de la figura seleccionada en el lienzo.

3.2.2 Clase Figura:

La clase `figura` es la clase padre de la que heredan todas las clases de las figuras. Dispone de los atributos de las figuras con sus respectivos métodos get y set para consultar y modificar los atributos de cada figura.

Dispone de unos constructores que serán invocados por los constructores de las subclases, que corresponden a las clases de las figuras, la invocación se realiza con el método `super()` y los parámetros que requiera el constructor.

3.2.3 Clase Arco:

Clase de la figura arco que hereda de la clase **`Figura`**, en su constructor se invoca al constructor de la superclase **`Figura`**. Esta clase dispone del método que actualiza el punto final de creación de la figura.

3.2.4 Clase Curva:

Clase de la figura curva con punto de control que hereda de la clase **`Figura`**, en su constructor se invoca al constructor de la superclase **`Figura`**. Esta clase dispone del método que actualiza el punto final de creación de la figura.

3.2.5 Clase Elipse:

Clase de la figura Elipse que hereda de la clase **Figura**, en su constructor se invoca al constructor de la superclase **Figura**. Esta clase dispone del método que actualiza el punto final de creación de la figura. También dispone de los métodos que devuelven el degradado horizontal y vertical de la figura

3.2.6 Clase Línea:

Clase de la figura Línea que hereda de la clase **Figura**, en su constructor se invoca al constructor de la superclase **Figura**, la línea se crea con la clase propia **MiLinea2D**. Esta clase dispone del método que actualiza el punto final de creación de la figura.

3.2.7 Clase Polígono:

Clase de la figura polígono que hereda de la clase **Figura**, en su constructor se invoca al constructor de la superclase **Figura**.

3.2.8 Clase Punto:

Clase de la figura punto que hereda de la clase **Figura**, en su constructor se invoca al constructor de la superclase **Figura**.

3.2.9 Clase Rectángulo:

Clase de la figura Rectángulo que hereda de la clase **Figura**, en su constructor se invoca al constructor de la superclase **Figura**. Esta clase dispone del método que actualiza el punto final de creación de la figura. También dispone de los métodos que devuelven el degradado horizontal y vertical de la figura

3.2.10 Clase Rectángulo Redondeado:

Clase de la figura Rectángulo redondeado que hereda de la clase **Figura**, en su constructor se invoca al constructor de la superclase **Figura**. Esta clase dispone del método que actualiza el punto final de creación de la figura. También dispone de los métodos que devuelven el degradado horizontal y vertical de la figura

3.2.11 MiLinea2D:

Esta clase hereda de la clase **Line2D**. La clase dispone de los métodos **isNear(Point2D p)** que devuelve si un punto está cerca de la línea y **contains(Point2D p)** que comprueba si un punto está dentro de los límites de la línea.

3.2.12 Lienzo2DImagen:

Esta clase que hereda de Lienzo2D se encarga de cargar y visualizar imágenes. Al heredar de lienzo se pueden dibujar figuras sobre la imagen cargada. Tiene el atributo `img` de la clase **BufferedImage**.

La clase dispone del método **BufferedImage getImage(boolean drawVector)**, en caso de que **drawVector** sea **true** devolverá la imagen con el vector de figuras dibujado sobre la imagen cargada, en caso de que sea **false** devolverá la imagen.

Esta clase también dispondrá del método **coordRaton()** que devolverá la coordenada del ratón.

3.3 Ventanas Internas:

En este apartado veremos los distintos tipos de ventanas internas y explicaremos su funcionamiento.

3.3.1 VentanaInterna:

Esta ventana interna hereda de **JInternalFrame** y en ella está contenido **LienzoImagen**, por lo que esta ventana es la encargada del lienzo de dibujo y de cargar las imágenes. El tamaño de esta ventana podrá definirlo el usuario introduciendo el tamaño que desee en el diálogo de la clase **TamañoVentana** en caso de que se abra un lienzo de dibujo, en caso de que se abra una imagen la ventana tendrá el tamaño de la imagen.



Ilustración 47-Ventana interna imagen

3.3.2 TamañoVentana:

Esta ventana de diálogo que hereda de **JDialog** es la encargada de pedir al usuario que determine el tamaño del nuevo lienzo de dibujado. Esta ventana dispone de dos campos de texto en el que el usuario introducirá el ancho y el alto del nuevo lienzo de dibujado, una vez el usuario haya introducido los datos tendrá un botón de JButton con el que enviará la información.



Ilustración 48-Tamaño ventana interna

3.3.3 VentanaInternaReproductor:

Esta ventana que hereda de **JInternalFrame**, es la encargada de reproducir un archivo de audio que tenga un formato permitido. Para iniciar o pausar la reproducción el usuario interactuará con los botones **play** y **stop** situados en la **barra de audio/video** vista anteriormente.

3.3.4 VentanaInternaJMFPlayer:

Esta ventana que hereda de **JInternalFrame**, es la encargada de reproducir un archivo de video que tenga un formato permitido. Para iniciar o pausar la reproducción el

usuario interactuará con los botones **play** y **stop** situados en la **barra de audio/video** vista anteriormente. También podrá reproducir y pausar la reproducción con los componentes que cuenta la ventana.

3.3.5 VentanaInternaGrabador:

Esta ventana que hereda de **JInternalFrame**, es la encargada de grabar un archivo de audio. Para pausar la grabación el usuario interactuará con el botón **stop** situado en la **barra de audio/video** vista anteriormente.

3.4 Operaciones de imagen:

En este apartado veremos las operaciones de imagen que se han implementado en clases propias.

3.4.1 SepiaOp:

Esta clase que hereda de **BufferedImageOpAdapter**, se encarga de modificar el tono y la saturación para darle un aspecto de “fotografía antigua”. La transformación que producen este tipo de efecto se define en base a la siguiente ecuación:

$$\begin{aligned}\text{sepiaR} &= \min(255, 0.393 \cdot R + 0.769 \cdot G + 0.189 \cdot B) \\ \text{sepiaG} &= \min(255, 0.349 \cdot R + 0.686 \cdot G + 0.168 \cdot B) \\ \text{sepiaB} &= \min(255, 0.272 \cdot R + 0.534 \cdot G + 0.131 \cdot B)\end{aligned}$$

con [R,G,B] el color del pixel original. Debemos de tener en cuenta que en la ecuación anterior hay que tener en cuenta que si el valor obtenido para un componente es superior a 255, hay que truncarlo a 255.

3.4.2 UmbralizacionOp:

Esta clase que hereda de **BufferedImageOpAdapter**, genera como resultado una imagen binaria donde los píxeles que superan un umbral que se establece con la barra deslizadora vista anteriormente, se le asigna un valor de 255 y al resto 0.

3.4.3 MultiplicacionOp:

Esta clase que hereda de **BinaryOp**, genera una nueva imagen como resultado de realizar la multiplicación de los píxeles de dos imágenes. El resultado de la operación

se truncará, siendo el resultado $\text{res}/255$ si el resultado es mayor que 255, y 0 si es inferior a 0.

3.4.4 VisionNocturnaOp:

Esta clase que hereda de **BufferedImageOpAdapter**, aplica la operación a la imagen seleccionada dándole una apariencia de la visión a través de unas gafas de visión nocturna como las que utilizan los militares.

Para llegar a esta apariencia en el método **filter** a la imagen se le aplica un tintado de color magenta con $\text{RGB}=255,0,254$, una vez aplicado el tintado se le aplica la operación de negativo obteniendo una apariencia verdosa de las gafas de visión nocturna.

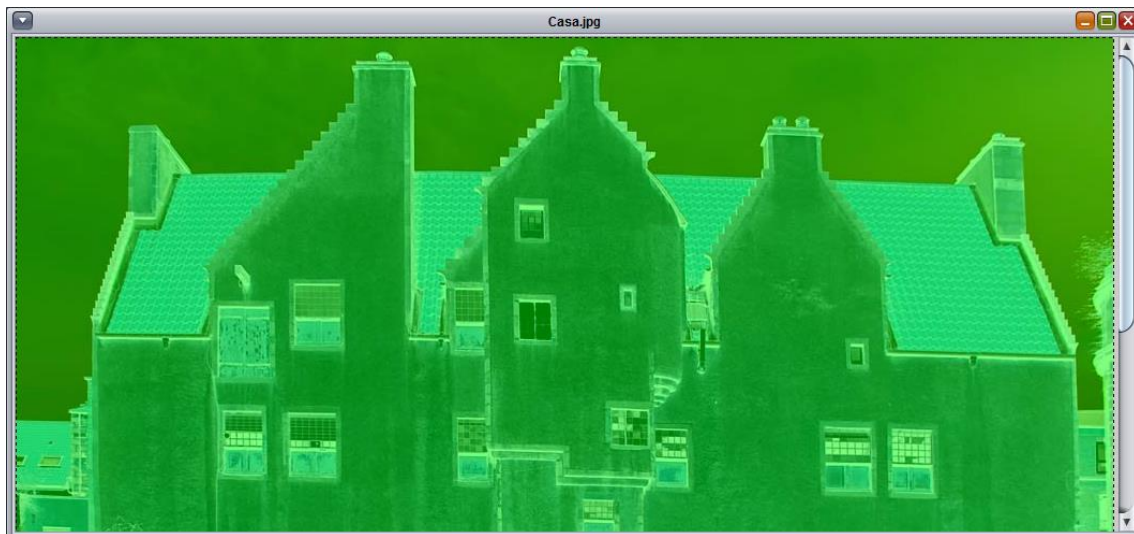


Ilustración 49-Visión Nocturna

Bibliografía:

- <https://docs.oracle.com/javase/7/docs/api/java/awt/Robot.html>
- <https://docs.oracle.com/javase/7/docs/api/java/awt/geom/QuadCurve2D.html>
- <https://docs.oracle.com/javase/7/docs/api/java/awt/Polygon.html>
- <https://docs.oracle.com/javase/7/docs/api/javax/swing/JFileChooser.html>
- <https://docs.oracle.com/javase/7/docs/api/java/awt/geom/RoundRectangle2D.html>