

# Manual Técnico

# **MANUAL TECNICO**

## **SOFTWARE DE ADMINISTRACIÓN Y GESTION DE ARCHIVOS MULTIMEDIA**

**SERGIO JESUS ALESSANDRO MÉRIDA IXTETELA**  
**CARNÉ: 3090-23-20185**

**UNVIERSIDA MARIANO GALVEZ DE GUATEMALA, SEDE SUCHITEPEQUEZ**  
**INGENIERIA EN SISTEMAS INFROMATICOS Y CIENCIAS DE LA**  
**COMPUTACION**  
**PROGRAMACION II**  
**MAZATENANGO, SUCHITEPQUEZ. NOVIEMBRE DE 2024**

## **TABLA DE CONTENIDO**

1. OBJETIVOS .....	1
2. INTRODUCCION .....	2
3. REQUERIMIENTOS TÉCNICOS. ....	3
3.1 REQUERIMIENTOS MÍNIMOS DE HARDWARE .....	3
3.2 REQUERIMIENTOS MÍNIMOS DE SOFTWARE .....	3
4. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO .....	3
4.1 INSTALACION DE NETBEANS .....	4
4.2 INSTALACION DE LIBRERIAS.....	4
5. MODELO RELACION CLASES .....	6
6. DIAGRAMA DE FLUJO .....	7
7. COMENTARIOS EN EL CODIGO Y ALGUNAS EXPLICACIONES DE BLOQUES DE CODIGO .....	8

## **1. OBJETIVOS**

### **General:**

Evitar la acumulación de archivos en la memoria o disco de nuestras laptops o computadoras.

### **Específicos:**

- Desarrollar un software de gestión de archivos multimedia que permita al usuario organizar, visualizar y reproducir archivos de música, video e imágenes de forma eficiente y sin conexión a internet.
- Promover la eficiencia en el uso del almacenamiento local, al permitir que el usuario gestione de forma integral y organizada sus archivos multimedia, reduciendo el desorden y mejorando la accesibilidad

## **2. INTRODUCCION**

El hábito de descarga puede llevar con el tiempo a la acumulación desorganizada de archivos, ocupando un espacio de almacenamiento considerable y dificultando la búsqueda de contenido debido a la existencia de archivos duplicados y de gran tamaño.

Para resolver este problema se ha propuesto desarrollar un software que permita la gestión de archivos multimedia, optimizando su organización y uso eficiente en una computadora. Lo que nos lleva al presente manual técnico el cual se enfoca en brindar todo el contexto general sobre cómo fue desarrollado el programa para solventar esta necesidad detallando requerimientos y características del software.

### 3. REQUERIMIENTOS TÉCNICOS.

A continuación, se detallarán todos los requerimientos que se necesitan para poder ejecutar el desarrollo del software:

#### 3.1 REQUERIMIENTOS MÍNIMOS DE HARDWARE

- **Procesador:** 1 GHz o más rápido con dos o más núcleos en un procesador o un sistema en un chip (SoC) compatible de 64 bits, recomendado. Intel Core i3 -14100 14th gen.
- **Memoria RAM:** Mínimo: 4 Gigabytes (GB)
- **Disco Duro:** 64 GB o más.

#### 3.2 REQUERIMIENTOS MÍNIMOS DE SOFTWARE

- **Privilegios de administrador**
- **Sistema Operativo:** Windows 10  
Home/Pro/Education/Enterprise/Enterprise LTSC/loT/Pro  
Education/Pro for Workstations

### 4. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO

- **NetBeans:** Es un entorno de desarrollo integrado (IDE) de código abierto utilizado principalmente para la programación en Java, aunque también soporta otros lenguajes como PHP, C++ y HTML. Ofrece herramientas avanzadas para el desarrollo de aplicaciones de escritorio, web y móviles, con características que incluyen un editor de código, depuración y gestión de proyectos.
- **draw.io:** Es una aplicación de diagramación en línea gratuita que permite a los usuarios crear diagramas de flujo, diagramas UML, organigramas y otros tipos de gráficos. Es ampliamente utilizada para documentar procesos y diseñar esquemas visuales de proyectos de software y estructuras organizativas.
- **GitHub:** Es una plataforma de desarrollo colaborativo basada en Git, que permite a los desarrolladores alojar, revisar y gestionar proyectos de software. Ofrece herramientas para el control de versiones, colaboración en equipo, seguimiento de problemas y despliegue de aplicaciones, siendo fundamental para proyectos de código abierto y desarrollo profesional.

## 4.1 INSTALACION DE NETBEANS

### **Descargar NetBeans:**

Ve al sitio web oficial de NetBeans.

Selecciona la versión que prefieras y haz clic en el enlace de descarga adecuado para tu sistema operativo (Windows).

### **Instalar JDK:**

NetBeans requiere tener el JDK (Java Development Kit) instalado. Si no lo tienes, descárgalo desde el sitio oficial de Oracle o usa una distribución de OpenJDK.

Instala el JDK siguiendo las instrucciones del instalador.

### **Ejecutar el instalador de NetBeans:**

Una vez que se haya descargado el archivo de instalación de NetBeans, haz doble clic en él para ejecutarlo.

Sigue las instrucciones del asistente de instalación, acepta los términos y condiciones, y selecciona las opciones de instalación deseadas.

### **Configurar la instalación:**

Elige la carpeta de instalación y verifica que el instalador detecte la ubicación del JDK.

Haz clic en "Instalar" y espera a que el proceso termine.

### **Completar la instalación:**

Puedes empezar a crear proyectos seleccionando "File" > "New Project".

## 4.2 INSTALACION DE LIBRERIAS

Para desarrollar el proyecto de gestor de archivos se hizo el uso de las siguientes librerías:

- Absolute Layout
- AbsoluteLayout.jar
- jaudiotagger-2.0.3.jar
- basicplayer-3.0.0.0.jar

- slf4j-api-1.7.12.jar
- jlayer-1.0.1.4.jar
- junit-3.8.2.jar
- SLF4J - slf4j-api-2.0.16.jar
- javafx-swt.jar
- javafx.base.jar
- javafx.controls.jar
- javafx.fxml.jar
- javafx.graphics.jar
- javafx.media.jar
- javafx.swing.jar
- javafx.web.jar
- JDK 22 (Default)

A continuación, se hace una guía breve de como instalar una librería en Netbeans:

### **Descargar la librería:**

Descarga la librería que desees usar (por ejemplo, un archivo .jar) desde su sitio oficial o repositorio confiable.

### **Agregar la librería al proyecto:**

Abre NetBeans y el proyecto en el que desees usar la librería.

Haz clic derecho en el nombre del proyecto en el panel de "Projects" y selecciona "Properties".

En la ventana de propiedades, selecciona la opción "Libraries".

Haz clic en "Add JAR/Folder".

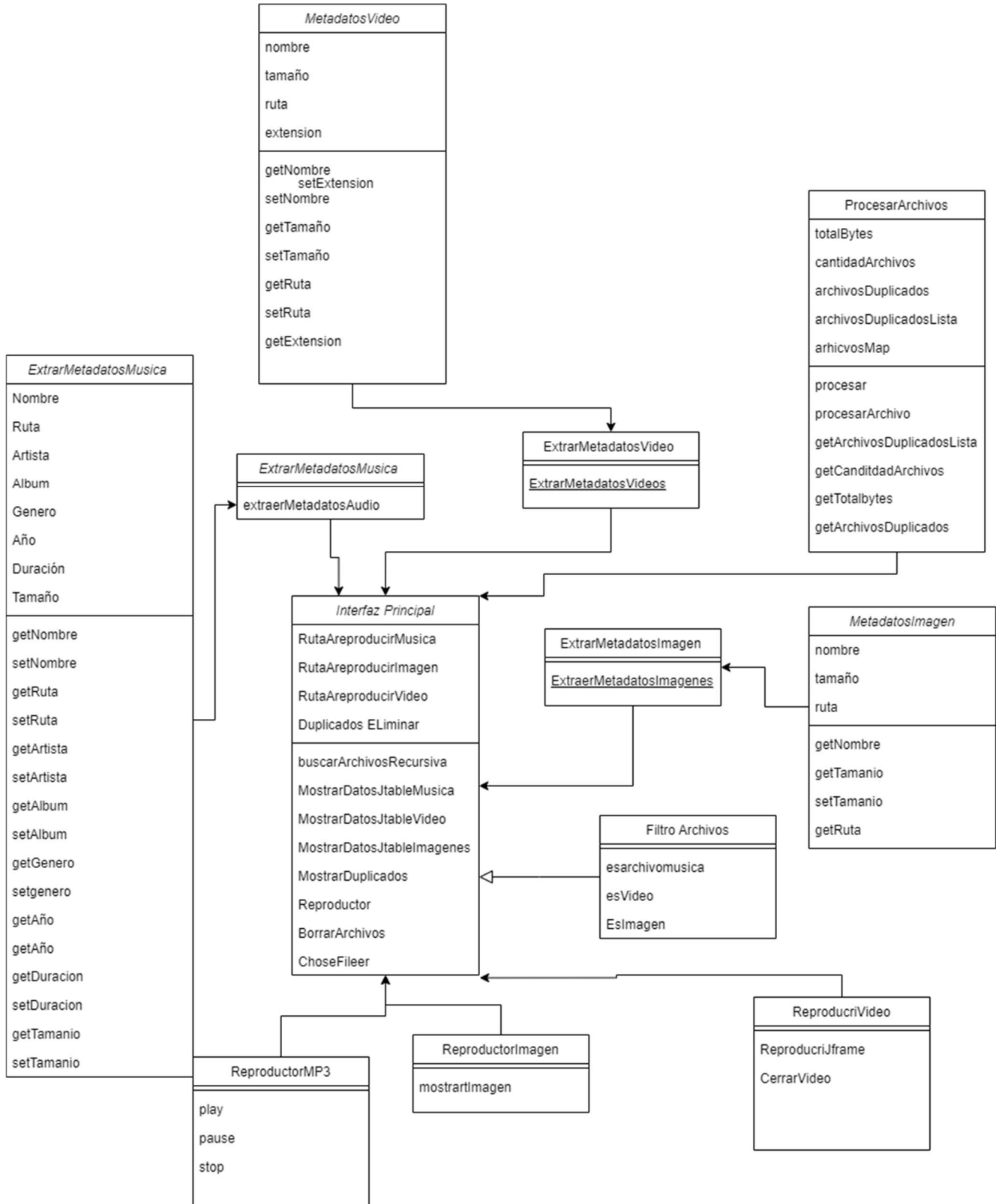
Navega hasta la ubicación donde guardaste la librería .jar, selecciónala y haz clic en "Open".

### **Confirmar la adición:**

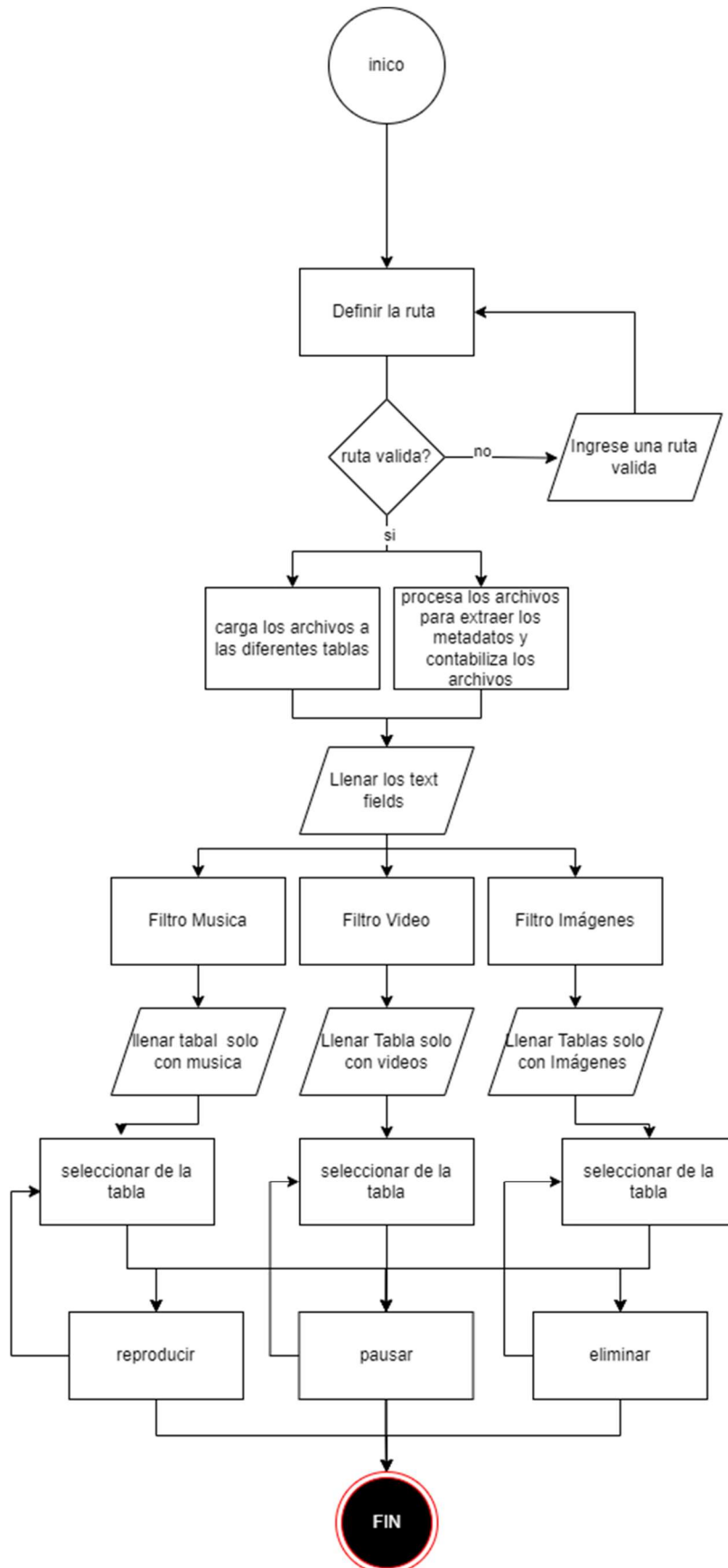
Asegúrate de que la librería aparezca en la lista de librerías del proyecto.

Haz clic en "OK" para confirmar.

## 5. MODELO RELACION CLASES

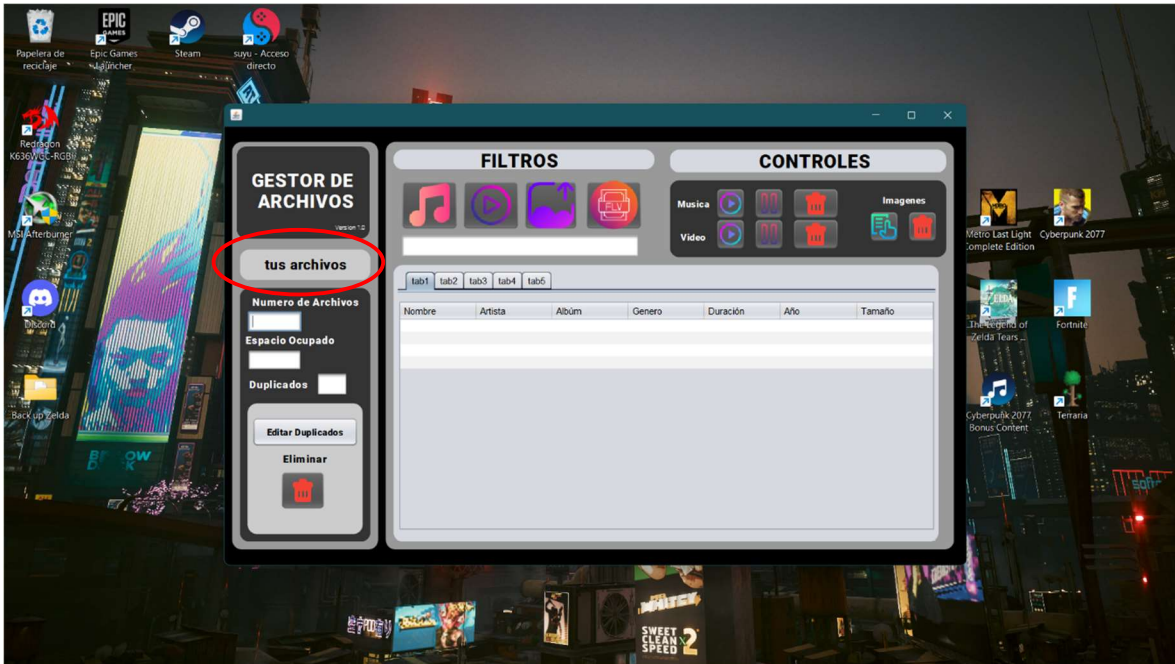


## 6. DIAGRAMA DE FLUJO



## 7. COMENTARIOS EN EL CODIGO Y ALGUNAS EXPLICACIONES DE BLOQUES DE CODIGO

El 80% de la lógica del código se centra en el botón que permite seleccionar la ruta sobre la cual se trabajara



```
private void Origen9MouseClicked(java.awt.event.MouseEvent evt) {  
    JFileChooser FileRuta = new JFileChooser(); //crear el filchooser  
    FileRuta.setMultiSelectionEnabled(b: true); //seleccionar varias cosas  
    FileRuta.setSelectionMode(mode: JFileChooser.FILES_AND_DIRECTORIES); //filtro para carpetas y archivos  
  
    int resultado = FileRuta.showOpenDialog(parent: this); //mostrar el chosefile  
    if (resultado != JFileChooser.CANCEL_OPTION) { //si no cancela  
        File carpeta = FileRuta.getSelectedFile();  
        if (carpeta != null && !carpeta.getName().equals(anObject: "")) { //si la carpeta es valida  
            ProcesarArchivos procesador = new ProcesarArchivos(); //intacia de la que tengo de procesar  
            //FiltroArchivos filtro = new FiltroArchivos(); // crear instancia del filtro  
  
            List<File> listaArchivos = new ArrayList <>(); //crea la lista  
            buscarArchivosRekursiva(carpeta, listaArchivos); //busca los archivos y lo agrega la lista  
            mostrarDatosJtableMusica (archivos: listaArchivos.toArray (File [] ::new)); //convierte la lista a un array de archivos  
            mostrarDatosJtableVideo (archivos: listaArchivos.toArray (File [] ::new));  
            mostrarDatosJtableImagen(archivos: listaArchivos.toArray (File [] ::new));  
  
            procesador.procesar(archivo: carpeta); //procesar los duplicados  
            List<File> archivosDuplicados = procesador.getArchivosDuplicadosList(); //obtener los duplicados  
            mostrarDuplicadosJtable(archivosDuplicados); //mostrar los duplicados  
  
            File[] archivos = carpeta.listFiles(); //guardar en un array  
            for (File archivo : archivos) { //recorred todo el array  
                procesador.procesar(archivo); //llamar al metodo de la clase  
            }  
  
            // obtener los resultados y llenar los text fields tablas, etc  
            NumeroArchivos.setText(t: String.valueOf(i: procesador.getcantidadarchivos()));  
            EspacioOcupado.setText(t: String.format(format: "%.2f GB", args: procesador.gettotalgb()));  
            EspacioDuplicados.setText(t: String.valueOf(i: procesador.getarchivosduplicados()));  
            RutaSeleccionada.setText(t: carpeta.getAbsolutePath());  
        }  
    } else {  
        JOptionPane.showMessageDialog(parentComponent: this, message: "Error al cargar la carpeta");  
    }  
}
```

Dentro de el apartado de reproductor de video se necesita crear un hilo diferente del procesador ósea un subproceso para que el video pueda reproducirse y no se congele la interfaz de usuario, lo cual se ejemplifica a continuación:

```
Platform.runLater(() -> { // Ejecuta en un hilo difere
    Media videoMedia = new Media(string:new File(pathname: rutaVideo).toURI().toString()); //crea objeto media
    MediaPlayer reproductorMultimedia = new MediaPlayer(media: videoMedia); //crea el media player
    MediaView verVideo = new MediaView(mp: reproductorMultimedia); //crea un media view que reenderiz ael video

    verVideo.setPreserveRatio(bln:true); // Mantiene la relación de aspecto del video
    verVideo.setFitWidth(d: 800); // Ancho del video
    verVideo.setFitHeight(d: 600);

    BorderPane ventanaVideo = new BorderPane(); //contenedor para el video
    ventanaVideo.setCenter(node: verVideo); //coloca el video en el centro
    Scene escenai = new Scene(parent: ventanaVideo, d: 800, dl: 600); //crea la escena

    jfxPanel.setScene(scene: escenai); //establece la escena
    reproductorMultimedia.play(); //reproduce
});
```