Manual Técnico para la Aplicación de Gestión de Archivos Multimedia

Introducción

Este manual detalla las clases principales de la aplicación de gestión de archivos de música, incluyendo sus funcionalidades, atributos y componentes. La aplicación permite a los usuarios crear, gestionar y reproducir playlists de música de manera eficiente, ofreciendo una experiencia de usuario intuitiva.

Clase: VentanaGestionarPlaylists

Descripción General

La clase VentanaGestionarPlaylists extiende JFrame y proporciona una interfaz gráfica que permite a los usuarios crear, borrar y abrir playlists. Utiliza un JTable para mostrar las playlists disponibles y facilita interacciones a través de botones.

Atributos

private JTable tablaPlaylists: Componente de la interfaz que muestra las playlists en forma de tabla.

private DefaultTableModel modeloTabla: Modelo de datos asociado a la tabla que permite gestionar dinámicamente las filas.

private PlaylistManager playlistManager: Instancia de la clase PlaylistManager que gestiona la creación, carga y eliminación de playlists.

private Interfaz interfaz: Referencia a la interfaz principal de la aplicación, utilizada para mostrar la lista de canciones de una playlist seleccionada.

Constructor

```
public VentanaGestionarPlaylists(Interfaz interfaz)
```

Configura la ventana, inicializa los componentes y establece los oyentes de eventos para los botones.

Componentes de la Interfaz

Tabla (JTable):

Muestra los nombres de las playlists, inicializada con una columna llamada "Nombre de Playlist".

Botones:

Crear Playlist: Solicita un nombre para la nueva playlist y la agrega usando PlaylistManager.

Borrar Playlist: Elimina la playlist seleccionada de la tabla y del sistema de archivos.

Abrir Playlist: Carga las canciones de la playlist seleccionada y las muestra en la interfaz principal.

Métodos

cargarPlaylists():

Funcionalidad: Limpia la tabla y carga las playlists desde el archivo de texto usando PlaylistManager.

Comportamiento: Establece el número de filas en el modelo de tabla a cero y luego agrega cada playlist recuperada a la tabla.

Comportamiento de la Interfaz

La ventana se establece con un título, tamaño y operación de cierre.

Se utiliza un panel para agrupar los botones y se añade un JScrollPane alrededor de la tabla para permitir el desplazamiento si hay muchas playlists.

Se llama a cargarPlaylists() en el constructor para que la tabla se llene al abrir la ventana.

Integración y Uso

Esta clase forma parte de un sistema más grande para gestionar archivos de música a través de playlists, proporcionando una forma interactiva de manipular listas de reproducción y mejorando la experiencia del usuario.

Clase: VentanaLetra

Descripción General

La clase VentanaLetra extiende JFrame y proporciona una interfaz gráfica para mostrar la letra de una canción. Utiliza un JTextPane para presentar el texto de manera estilizada y centrada, dentro de un panel con fondo colorido.

Constructor

```
public VentanaLetra(String letra)
```

Configura la ventana, estableciendo su título, tamaño y operación de cierre. También inicializa el panel y el componente para mostrar la letra.

Componentes de la Interfaz

Panel (JPanel):

Utiliza un panel con un diseño de BorderLayout y un fondo de color oscuro (Color RGB: 1, 50, 32).

Texto de la Canción (JTextPane):

No editable, con fondo y texto de colores adecuados para la legibilidad.

Fuente de tipo Arial, en negrita y tamaño 24.

Se coloca la letra de la canción en el componente.

Estilo del Texto

Alineación del Texto: Utiliza StyledDocument para centrar el texto. Se crea un SimpleAttributeSet que establece la alineación del texto a centrado y se aplica a todo el documento.

Desplazamiento

Scroll Pane (JScrollPane): El JTextPane se inserta en un JScrollPane para permitir el desplazamiento vertical de la letra.

Comportamiento de la Ventana

Se centra en la pantalla y se establece para que se cierre al ser desechada.

Integración y Uso

VentanaLetra se utiliza para presentar letras de canciones de manera atractiva dentro de la aplicación. Su diseño centrado y el uso de un color de fondo oscuro mejoran la experiencia visual del usuario.

Clase: VentanaPlaylist

Descripción General

La clase VentanaPlaylist extiende JFrame y proporciona una interfaz gráfica para permitir al usuario agregar una canción a una playlist seleccionada. La ventana muestra una tabla con las playlists disponibles y cuenta con botones para confirmar la acción o cancelar y regresar a la ventana principal.

Atributos

JTable tablaPlaylist: Muestra las playlists disponibles.

DefaultTableModel modeloTabla: Modelo de datos para la tabla que permite la manipulación de las filas.

JButton btnConfirmar y btnCancelar: Botones para confirmar la adición de la canción o cancelar la operación.

PlaylistManager playlistManager: Utilizado para gestionar las playlists.

String nombreArchivo y rutaArchivo: Almacenan el nombre y la ruta del archivo de la canción que se desea agregar a la playlist.

Constructor

```
public VentanaPlaylist(String nombreArchivo, String rutaArchivo)
```

Configura la ventana, establece su título y tamaño, y define la disposición de los componentes. Inicializa el PlaylistManager y la tabla. Carga las playlists disponibles al iniciar.

Componentes de la Interfaz

JScrollPane:

Permite el desplazamiento de la tabla que muestra las playlists.

Panel de Botones (JPanel): Contiene los botones para confirmar o cancelar la operación.

Funcionalidades

cargarPlaylists():

Limpia la tabla y llena el modelo con las playlists disponibles, obtenidas a través de playlistManager.

agregarAPlaylist():

Permite al usuario agregar la canción seleccionada a la playlist elegida. Verifica la selección y notifica al usuario sobre el éxito o error.

regresarAPrincipal():

Cierra la ventana actual, permitiendo al usuario volver a la interfaz principal de la aplicación.

Integración y Uso

VentanaPlaylist se integra en la aplicación de gestión de música, facilitando a los usuarios la adición de canciones a playlists de manera intuitiva.

Clase: VentanaReproductor

Descripción General

La clase VentanaReproductor extiende JFrame y proporciona una interfaz gráfica para la reproducción de videos. Incluye controles para reproducir, pausar y detener el video.

Atributos

Reproductor reproductor: Instancia de la clase Reproductor que se encarga de la reproducción de videos.

Constructor

```
public VentanaReproductor(String rutaArchivo)
```

Inicializa la ventana configurando su título, tamaño y operación de cierre. También crea la instancia del reproductor y configura los controles de la interfaz.

Componentes de la Interfaz

JPanel panelControles:

Panel que contiene los botones de control.

JButton btnReproducir, btnPausar, btnDetener: Botones que permiten al usuario controlar la reproducción del video.

Funcionalidades

Acciones de los Botones:

Reproducir: Invoca el método reproducir del reproductor con la ruta del archivo de video.

Pausar: Invoca el método pausar del reproductor.

Detener: Invoca el método detener del reproductor.

Disposición de Componentes

Se agregan el componente de video del reproductor y el panel de controles utilizando BorderLayout.

Reproducción Automática

Se llama al método reproducirVideo al final del constructor para iniciar la reproducción automáticamente.

Cierre de Ventana

Se sobreescribe el método dispose() para detener la reproducción del video al cerrar la ventana.

Integración y Uso

VentanaReproductor se integra en la aplicación como un componente para la visualización y control de videos, mejorando la experiencia del usuario al permitir interacciones sencillas con la reproducción de contenido multimedia.

Clase: BusquedaArchivos

Descripción General

La clase BusquedaArchivos permite buscar archivos de imagen en un directorio dado, calcular el espacio total ocupado por ellos, identificar duplicados, extraer metadatos y realizar búsquedas en los resultados.

Atributos

List<File> listaArchivosImagen: Almacena la lista de archivos de imagen encontrados.

List<File> resultadosBusqueda: Almacena los archivos que coinciden con las búsquedas realizadas.

Constructor

```
public BusquedaArchivos()
```

Inicializa las listas de archivos de imagen y resultados de búsqueda.

Métodos Principales

public void buscarArchivosImagen(File directorio):

Limpia la lista de archivos de imagen y comienza la búsqueda recursiva en el directorio especificado.

private void buscarRecursivamente(File directorio):

Método privado que busca archivos de imagen en el directorio y en sus subdirectorios.

public void calcularEspacioTotal():

Calcula el espacio total ocupado por todos los archivos de imagen encontrados.

public void mostrarDuplicados():

Identifica y muestra los archivos duplicados en la lista.

Integración y Uso

BusquedaArchivos se utiliza para realizar búsquedas de archivos de imagen y gestionar la información resultante, incluyendo metadatos, espacio ocupado y duplicados.

Clase VentanaImagen

Paquete: paqueteimagenes

Descripción La clase VentanaImagen extiende JFrame y está diseñada para mostrar una imagen en una ventana. Esta ventana se adapta al tamaño del contenido visualizado, redimensionando la imagen original mientras se mantiene su relación de aspecto.

Atributos

JLabel labelImagen: Componente gráfico que contiene la imagen.

ImageIcon imagenOriginal: Almacena la imagen original cargada desde la ruta especificada.

Timer timer: Temporizador que se utiliza para retrasar el ajuste de la imagen durante el redimensionamiento de la ventana.

Constructor

```
public VentanaImagen(String rutaImagen)
```

Inicializa la ventana, establece su título y tamaño, y centra la ventana en la pantalla. Se carga la imagen desde la ruta proporcionada y se llama al método ajustarImagen() para mostrarla inicialmente. Además, se configura un Timer para ajustar la imagen después de un breve retraso cada vez que se redimensiona la ventana.

Métodos

private void ajustarImagen()

Calcula el tamaño nuevo de la imagen basado en el tamaño actual de la ventana.

Mantiene la relación de aspecto de la imagen al escalarla.

Actualiza el JLabel con la nueva imagen escalada.

Comportamiento

Se añade un ComponentListener que detecta eventos de redimensionamiento de la ventana. Cuando ocurre un redimensionamiento, se reinicia el temporizador para asegurar que la imagen solo se ajuste después de que el usuario termine de redimensionar la ventana, lo que mejora la eficiencia.

Uso Esta clase puede ser utilizada en aplicaciones Java que requieren la visualización dinámica de imágenes, adaptándose automáticamente al tamaño de la ventana para ofrecer una experiencia de usuario mejorada.

Dependencias

Utiliza componentes de Swing (JFrame, JLabel, ImageIcon, Timer) para la interfaz gráfica.

Maneja eventos de componente para ajustar la visualización de la imagen en función de la interacción del usuario.

Resumen Técnico del Código de VentanaInterfaz

La clase VentanaInterfaz en Java, que extiende JFrame, representa una interfaz gráfica para un administrador multimedia. Esta clase incluye componentes que permiten buscar, visualizar y gestionar imágenes en el sistema de archivos.

Componentes de la Interfaz

Panel y Diseño: Utiliza un JPanel con un diseño nulo para organizar los componentes. El panel tiene un color de fondo específico.

Etiquetas: Se incluye un título "Administrador multimedia" con formato de fuente y color.

Caja de Texto: Se utiliza un JTextField para mostrar la ruta seleccionada de la carpeta.

Botones: Se crean varios JButton para realizar acciones como buscar carpetas, ver imágenes, mostrar espacio ocupado, ver archivos duplicados, eliminar archivos, ver archivos más grandes y mover archivos.

Tabla: Implementa un JTable con un DefaultTableModel para mostrar información sobre los archivos, incluyendo columnas como nombre, extensión, ruta, fechas de creación y modificación, tamaño, y otros metadatos.

FileChooser: Se utiliza un JFileChooser para seleccionar directorios, permitiendo la búsqueda de imágenes dentro de una carpeta específica.

Label de Espacio: Un JLabel muestra el espacio total ocupado por los archivos de imagen.

ComboBoxes: Se utilizan dos JComboBox para opciones de búsqueda y agrupamiento de archivos.

Funcionalidades Implementadas

Inicialización: En el constructor, se configuran los componentes y se carga la ruta predeterminada desde un archivo.

Buscar Imágenes: Al seleccionar un directorio, se buscan las imágenes y se muestran en la tabla.

Visualización de Imágenes: Permite abrir una ventana para visualizar la imagen seleccionada en la tabla.

Actualización de Espacio: Calcula y actualiza el espacio total ocupado por los archivos mostrados en la tabla.

Buscar y Filtrar: Funcionalidades para buscar archivos en la tabla por nombre o fecha de creación.

Gestión de Archivos Duplicados: Identifica y muestra archivos duplicados, vaciando la tabla antes de mostrar los resultados.

Mover Archivos: Implementa la opción de mover archivos seleccionados a otra ubicación.

Manejo de Archivos

Se utiliza FileWriter para guardar la última ruta seleccionada en un archivo de texto y Files para leer la ruta al inicio.

La clase hace uso de la gestión de excepciones para manejar errores relacionados con archivos.

Consideraciones

Escalabilidad: La estructura permite agregar fácilmente más funcionalidades relacionadas con la gestión de archivos multimedia.

Uso de Eventos: Cada botón y comboBox tiene un evento asociado que ejecuta acciones específicas, lo que mejora la interactividad de la interfaz.

Modularidad: Se observa un diseño modular donde las funcionalidades de búsqueda de archivos y manipulación de datos se delegan a la clase BusquedaArchivos, promoviendo la separación de responsabilidades en el código.