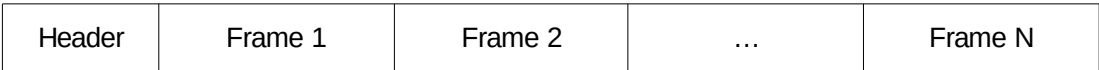


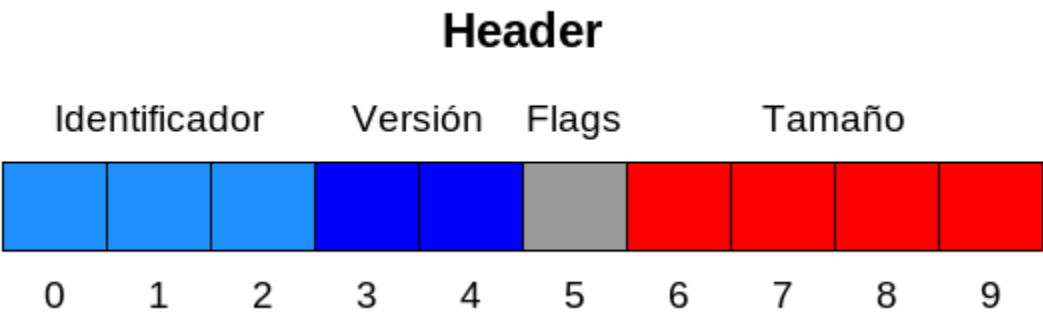
# PACKAGE FileManager

Estructura ID3:

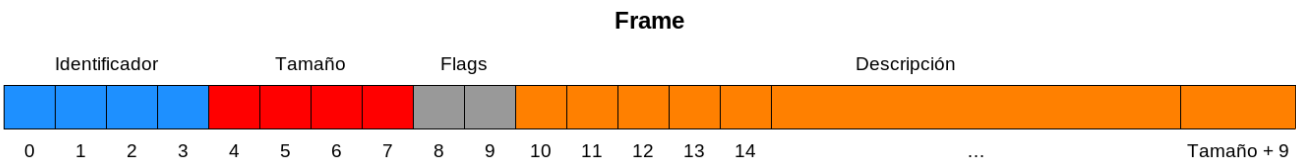
ESTRUCTURA DE ID3



Estructura del Header:



Estructura del Frame:



Lectura de tags en archivo mp3 (ID3)

Header (10 Bytes)	
	Bytes
Identificador del tag = ID3	[0-2]
Versión del TAG	[3-4]
Flags	5
Tamaño del tag	[6-9]

Los frames están en el rango [10, tamaño del tag + 9]

Frames	
	Bytes
Identificador del frame	[0-3]
Tamaño	[4-7]
Flags	[8-9]
Valor del tag	[10, tamaño+9]

## Class MP3File

Retorno	Método y Descripción
String	<b><i>getFileName()</i></b> Retorna el nombre del archivo
String	<b><i>getPath()</i></b> Retorna la ruta absoluta del archivo
String	<b><i>getSong()</i></b> Retorna el nombre de la canción
String	<b><i>getArtist()</i></b> Retorna el artista de la canción
String	<b><i>getAlbum()</i></b> Retorna el álbum al que pertenece la canción
long	<b><i>getSize()</i></b> Retorna el numero de bytes del archivo
byte[]	<b><i>read(int l, int r)</i></b> Retorna el fragmento desde la posición <b>l</b> hasta la posición <b>r</b> en el archivo. Si las posiciones no están en el rango del archivo o si el archivo no se pudo leer retornara null

## Class Directory

Constructor y Descripción
<b><i>Directory</i></b> (File file)

Retorno	Método y Descripción
Vector<MP3File>	<b><i>getMP3Files()</i></b> Retorna todos los archivos mp3 a partir de una carpeta.
void	<b><i>analyze</i></b> (File file) Revisa todos los archivos a partir de ese archivo raíz.

## Class FileWriter (incompleto)

Esta clase permite crear un archivo e inicializarlo con ceros. Permite escribir fragmentos (garantizando que máximo un hilo pueda acceder a cualquier byte).

Usara un Set para guardar las posiciones iniciales de los bloques reservados.

Mantiene dos arboles: En el primero se insertan las posiciones iniciales y en el segundo las finales

Constructor y Descripción
<b><i>FileWriter</i></b> (File file, long size, long currentSize, int blockSize)

Retorno	Método y Descripción
boolean	<b><i>create()</i></b> Llena el archivo con caracteres (0) desde la ultima posición escrita
long	<b><i>nextBlock()</i></b> Retorna la posición del siguiente bloque que puede leer
boolean	<b><i>write</i></b> (byte[] array, int inicio) Sobrescribe ese fragmento del archivo.

	Retorna true si se pudo sobrescribir.
--	---------------------------------------

## Class Tag

Esta clase define una etiqueta, equivalente a un par (clave, valor).

Constructores
Constructor y Descripción
<b>Tag(String name, String value)</b> Crea una etiqueta asignándole nombre y el valor.

Metodos	
Retorno	Descripción
String	<b>getName()</b> Retorna el nombre de la etiqueta.
String	<b>getValue()</b> Retorna el valor de la etiqueta.
int	<b>compareTo(Tag tag)</b> Compara los tag para poder ordenarlos.
boolean	<b>equals(Tag tag)</b> Retorna true si es igual a ese tag.

## Class FileDescription

implements Comparable

Esta clase representa un identificador para un archivo y lo hará por medio de:

- tipo de archivo
- nombre del archivo
- etiquetas

Constructor y Descripción
<b>FileDescription(String type, String fileName, Vector&lt;Tag&gt; tags)</b> Crea un FileDescription asignándole un tipo, el nombre de archivo y una lista de tags.

Metodos	
Retorno	Método y Descripción
String	<b><i>getType()</i></b> Retorna el tipo de archivo.
String	<b><i>getFileName()</i></b> Retorna el nombre del archivo.
int	<b><i>numberOfTags()</i></b> Retorna el numero de etiquetas.
Tag	<b><i>getTag(int index)</i></b> Retorna el tag en la i-esima posición en orden lexicográfico por el nombre del tag.
String	<b><i>getValue(String tagName)</i></b> Retorna el valor asociado a esta etiqueta.
int	<b><i>compareTo(FileDescription fileDescription)</i></b> Compara con otro descriptionFile para saber cual va antes en un ordenamiento. Retorna un numero entero: negativo si es menor 0 si es igual positivo si es mayor
boolean	<b><i>equals(FileDescription fileDescription)</i></b> Retorna true si es igual a ese fileDescription.

## Class FileDescriptionBuilder

Esta clase funcionara como fabrica para crear objetos de tipo FileDescription.

Métodos Estáticos	
Retorno	Método y Descripción
FileDescription	<b><i>createMP3FileDescription(String fileName, String fileSize, String title, String artist, String album)</i></b> Crea un FileDescription para un archivo mp3. El nombre del archivo sera la unión del titulo, artista y album, excepto si estos tres datos son nulos o vacíos en el cual se conservara el mismo nombre.

