**3º Trimestre**

Jonathan de la Fuente

Pablo Viniegra

Clara Ortiz de la Torre

PROYECTO ENTORNOS DE DESARROLLO



# ÍNDICE

[**Resumen de la aplicación**](#_kwyvs09yfpk4)3

[**Casos de uso**](#_rjcr0hs9aem1)4

[**Pruebas JUnit**](#_nsaheykr4d7x)11

[**Diagrama de clases**](#_xjxq4fgj9tvd)12

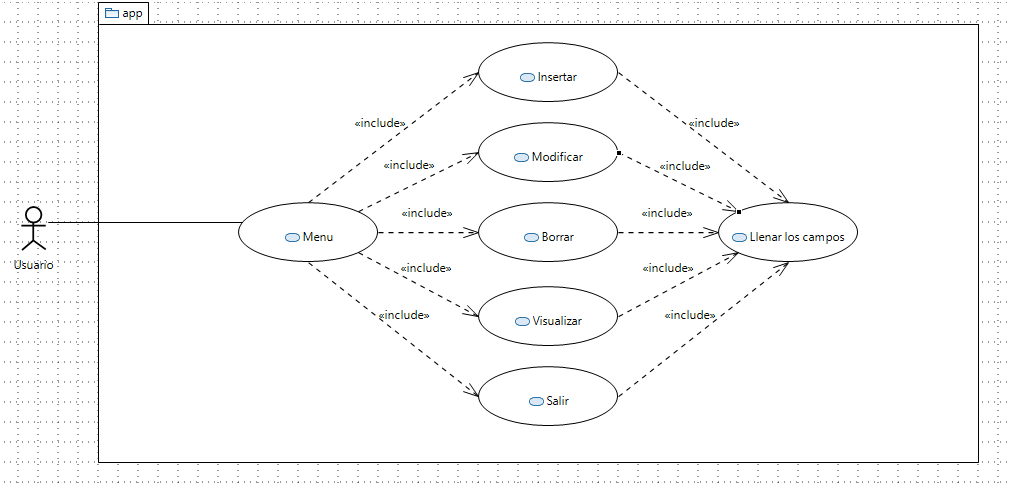
[**Diagramas de flujo y Pseudocódigo**](#_2hip56qjd1wt)13

[**Control de versiones**](#_1d4mxzvagc1o)21

# Resumen de la aplicación

La aplicación se ha planteado como un repositorio de canciones en consola con el lenguaje de programación Java (jdk 1.8). El usuario podrá almacenar en una Base de Datos sus canciones preferidas y generar un documento HTML con toda la lista que ha realizado él mismo. Por tanto, la funcionalidad de la aplicación se dividiría en dos partes fundamentales: en primer lugar, el CRUD (*create read update delete*), esto es, la aplicación permite crear tres objetos distintos: canciones, discos y artistas, tanto discos como artistas se relacionan con la canción a modo de clave foránea; una vez creados serán insertados en la Base de Datos. Para crear una canción, primero debe haber registros de artistas y discos en la Base de Datos. También el usuario podrá, a nivel de aplicación, modificar y borrar dichos objetos . La lectura de la Base de Datos se irá mostrando a medida que el usuario va interactuando con la aplicación, por ejemplo cuando el usuario quiera introducir una canción, a la hora de elegir el artista y el disco, la propia aplicación le mostrará previamente aquellos artistas y discos disponibles en la Base de Datos, cualquier otro artista o disco generaría un error en la aplicación dada la estructura de la Base de Datos. En segundo y último lugar, como se iba diciendo, la aplicación genera un fichero HTML con toda la información que el usuario tiene en la Base de Datos. Este fichero se genera a partir de una clase específica para ello, que hemos llamado Interacción debido a que es un Join de todos los objetos de la aplicación. Vuelca todos los datos de la Base de Datos en un arraylist, y este arraylist queda reflejado en el fichero. El objetivo de este HTML es que el usuario tenga un documento más visual con el que saber su colección de canciones. Puede generar tantos como quiera.

# Casos de uso

****

Casos de uso:

**Nombre**: Insertar artista

**ID**:PC2

**Descripción**: El cliente solicita insertar un artista nuevo en el registro de una BBDD mySql, rellenando una serie de campos. El sistema muestra la validación de la operación e inserta el registro y vuelve al menú principal

**Actores**:Cliente

**Precondiciones**: Escoger la opción 1 del menú

**Curso normal del caso de uso**:

1. Usuario escoge la opción 1 del menú
   1. Si opción es correcta
      1. Sistema solicita al usuario el nombre del artista
      2. Usuario introduce el nombre del artista
      3. Sistema solicita el año que empezó/empezaron a tocar el artista o el grupo
      4. Usuario introduce el año
      5. Sistema solicita el número de integrantes del grupo
      6. Usuario introduce el número de integrantes del grupo
      7. Sistema verifica conexión con la BBDD mySQL
         1. Si verifica con éxito
            1. Sistema conecta con la BBDD mySQL
            2. Sistema lanza la consulta a la BBDD my SQL

Si consulta es correcta

Sistema lanza aviso de acción completada con éxito

Sistema cierra conexión con la BBDD mySql

Si hay fallo en la consulta

Sistema lanza aviso de fallo en la insercción del registro

Sistema lanza excepción

Sistema cierra conexión con la BBDD mySql

Vuelve al menú principal

* + - 1. Si sistema detecta fallo de conexion
         1. Sistema lanza excepción
         2. Vuelve al menú principal
  1. Si opción es incorrecta
     1. Sistema devuelve mensaje de error
     2. Sistema pide al usuario pulsar una tecla para continuar
     3. Usuario pulsa una tecla
     4. Vuelve al menú principal

**Postcondiciones**:Sistema inserta registro en la BBDD y vuelve al menú principal

**Nombre**: Insertar disco

**ID**:PC3

**Descripción**: El cliente solicita insertar un disco nuevo en el registro de una BBDD mySql, rellenando una serie de campos. El sistema muestra la validación de la operación e inserta el registro y vuelve al menú principal

**Actores**:Cliente

**Precondiciones**: Escoger la opción 3 del menú

**Curso normal del caso de uso**:

1. Usuario escoge la opción 3 del menú
   1. Si opción es correcta
      1. Sistema solicita al usuario el nombre del disco
      2. Usuario introduce el nombre del disco
      3. Sistema solicita el nombre de la imagen de portada
      4. Usuario introduce el nombre de la imagen
      5. Sistema solicita el precio del disco
      6. Usuario introduce el precio del disco
         1. Si introduce un dato numérico
            1. Sistema verifica conexión con la BBDD mySQL

Si verifica con éxito

Sistema conecta con la BBDD mySQL

Sistema lanza la consulta a la BBDD my SQL

Si consulta es correcta

Sistema lanza aviso de acción completada con éxito

Sistema cierra conexión con la BBDD mySql

Si hay fallo en la consulta

Sistema lanza aviso de fallo en la insercción del registro

Sistema cierra conexión con la BBDD mySql

Sistema solicita pulsar un botón para continuar

Usuario pulsa un botón

Vuelve al menú principal

Si sistema detecta fallo de conexión

Sistema lanza excepción

Vuelve al menú principal

* + - * 1. Si introduce un carácter o separa los números por un punto

Sistema lanza aviso del fallo

Sistema solicita pulsar una tecla para continuar

Usuario pulsa una tecla

Vuelve al menú principal

* 1. Si opción es incorrecta
     1. Sistema devuelve mensaje de error
     2. Sistema pide al usuario pulsar una tecla para continuar
     3. Usuario pulsa una tecla
     4. Vuelve al menú principal

**Postcondiciones**: Sistema inserta registro en la BBDD y vuelve al menú principal

**Nombre**: Borrar artista

**ID**:PC8

**Precondiciones**: Escoger la opción 8 del menú, existencia de un registro de artista en la BBDD mySql

**Descripción**: El cliente solicita borrar un artista en el registro de una BBDD mySql, rellenando una serie de campos. El sistema muestra la validación de la operación, borra el registro y la vuelta al menú principal

**Actores**: Cliente

**Curso normal del caso de uso**:

1. Usuario escoge opción 8 del menú
   1. Si opción es correcta
      1. Sistema conecta con la BBDD mySql
         1. Si conexión tiene éxito
            1. Sistema lanza una consulta a la BBDD mySql

Si la consulta tiene éxito

Sistema muestra los resultados de la consulta

Sistema cierra conexión con la BBDD mySql

Sistema solicita el nombre del artista que el usuario desea eliminar

Usuario escribe el nombre del artista que desea eliminar del punto 3.1.1.1.1.1

Sistema conecta con la BBDD mySql

Si la conexión tiene éxito

Sistema lanza una consulta a la BBDD mySql

Si la consulta tiene éxito

Sistema muestra mensaje de validación

Sistema solicita que el usuario pulse una tecla para continuar

Usuario pulsa tecla

Sistema cierra la conexión con la BBDD mySql

Vuelve a menú principal

Si existe fallo en la consulta

Sistema lanza excepción

Sistema cierra la conexión con BBDD mySql

Vuelve al menú principal

Si existe fallo en la consulta

Sistema lanza excepción

Sistema cierra conexión con la BBDD mySql

Vuelve al menú principal

* + - 1. Si hay fallo de conexión
         1. Sistema lanza excepción
         2. Vuelve al menú principal
  1. Si opción es incorrecta
     1. Sistema devuelve mensaje de error
     2. Sistema pide al usuario pulsar una tecla para continuar
     3. Usuario pulsa una tecla
     4. Vuelve al menú principal

/\*La línea sombreada en gris oscuro es un subnivel de 3.1.1.1.1.5.1.1 y la línea sombreada en gris claro es un subnivel de 3.1.1.1.1.1.5.1.2, word no nos deja insertar más subniveles\*/

**Postcondiciones**:Sistema borra registro en la BBDD y vuelve al menú principal

**Nombre**: Salir del programa

**ID**:PC11

**Precondiciones**: Escoger la opción 11 del menú

**Descripción**: El cliente solicita salir del programa

**Actores**: Cliente

**Curso normal del caso de uso**:

1. Usuario escoge opción 11 del menú
   1. Si opción es correcta
      1. Sistema muestra mensaje de despedida
      2. Sistema sale de la aplicación
   2. Si opción es incorrecta
      1. Sistema devuelve mensaje de error
      2. Sistema pide al usuario pulsar una tecla para continuar
      3. Usuario pulsa una tecla
      4. Vuelve al menú principal

**Postcondiciones**: Sistema sale de la aplicación

**Nombre**: Modificar disco

**ID**:PC6

**Descripción**: El cliente solicita modificar un registro de disco en la bbdd mySql

**Actores**: Cliente

**Precondiciones**: Escoger la opción 6 del menú

**Curso normal del caso de uso**:

1. Usuario escoge la opción 6 del menú
   1. Si opción es correcta
      1. Sistema verifica la conexión con la BBDD mySql
         1. Si se produce la conexión
            1. Sistema lanza la consulta a la BBDD mySql

Si no hay error en la consulta

Sistema muestra el resultado de la consulta al usuario

Sistema solicita al usuario introducir un nombre de disco del resultado de la búsqueda de 5.1.1.1.1.1.1

Usuario introduce nombre de disco

Sistema solicita a usuario nuevo nombre de disco

Usuario escribe nombre de disco

Sistema solicita a usuario nueva imagen de portada de disco

Usuario escribe nueva imagen de portada de disco

Sistema solicita a usuario nuevo precio

Usuario escribe precio

Si precio está en formato correcto

Sistema verifica conexión con la BBDD mySQL

Si verifica con éxito

Sistema conecta con la BBDD mySQL

Sistema lanza la consulta a la BBDD my SQL

Si consulta es correcta

Sistema muestra a usuario mensaje de validación del proceso

Sistema cierra conexión con la BBDD mySql

Vuelve a menú principal

Si se produce un error en la consulta

Sistema muestra la excepción al usuario

Vuelve al menú principal

Si hay un error de conexión

Sistema muestra al usuario que ha habido un error en la conexión

Vuelve al menú principal

Si precio es un carácter o los decimales van separados por un punto

Sistema lanza aviso del fallo

Sistema solicita pulsar una tecla para continuar

Usuario pulsa una tecla

Sistema vuelve al menú principal

Si hay un error de en la consulta

Sistema muestra la excepción al usuario

Vuelve al menú principal

* + - * 1. Si hay un error en la consulta

Sistema muestra la excepción al usuario

Vuelve al menú principal

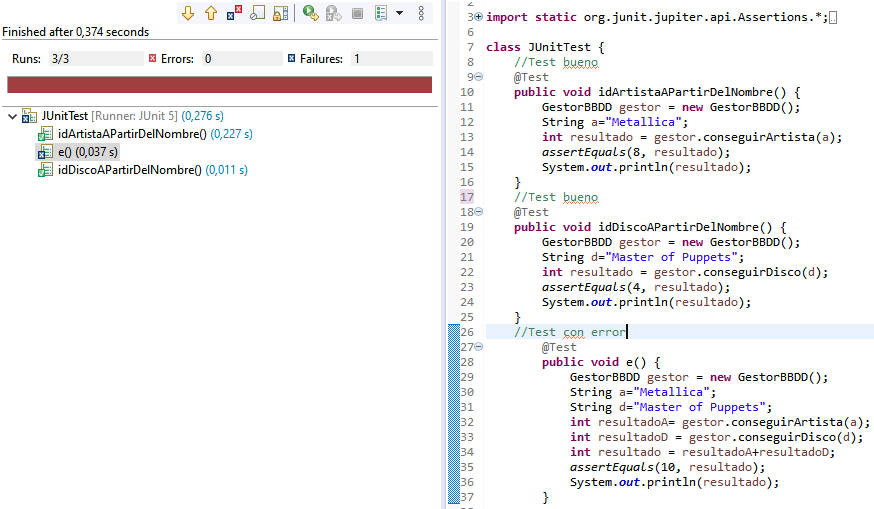
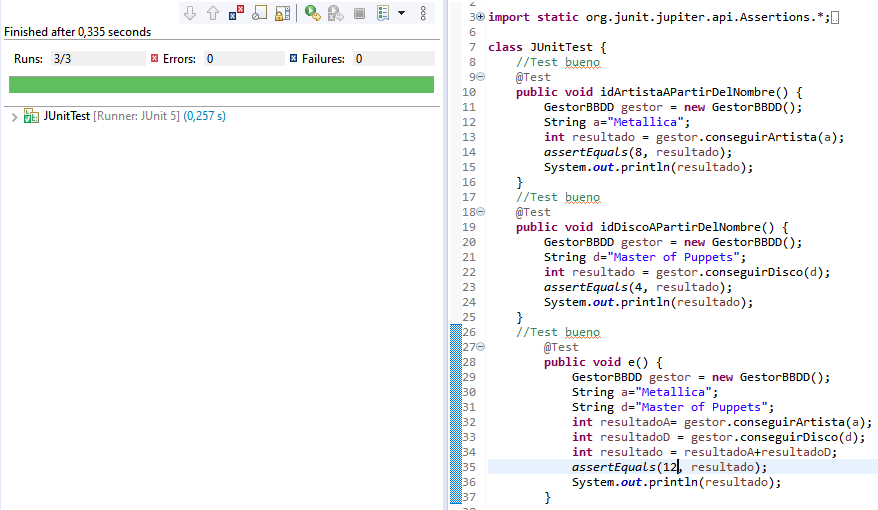
* + - 1. Si existe fallo en la conexión
         1. Sistema muestra al usuario que ha habido un error en la conexión
         2. Vuelve al menú principal
    1. Si existe fallo en la conexión
       1. Sistema muestra al usuario que ha habido un error en la conexión
       2. Vuelve al menú principal
  1. Si opción es incorrecta
     1. Sistema devuelve mensaje de error
     2. Sistema pide al usuario pulsar una tecla para continuar
     3. Usuario pulsa una tecla
     4. Vuelve al menú principal

**Postcondiciones**: Sistema modifica registro en la BBDD y vuelve al menú principal

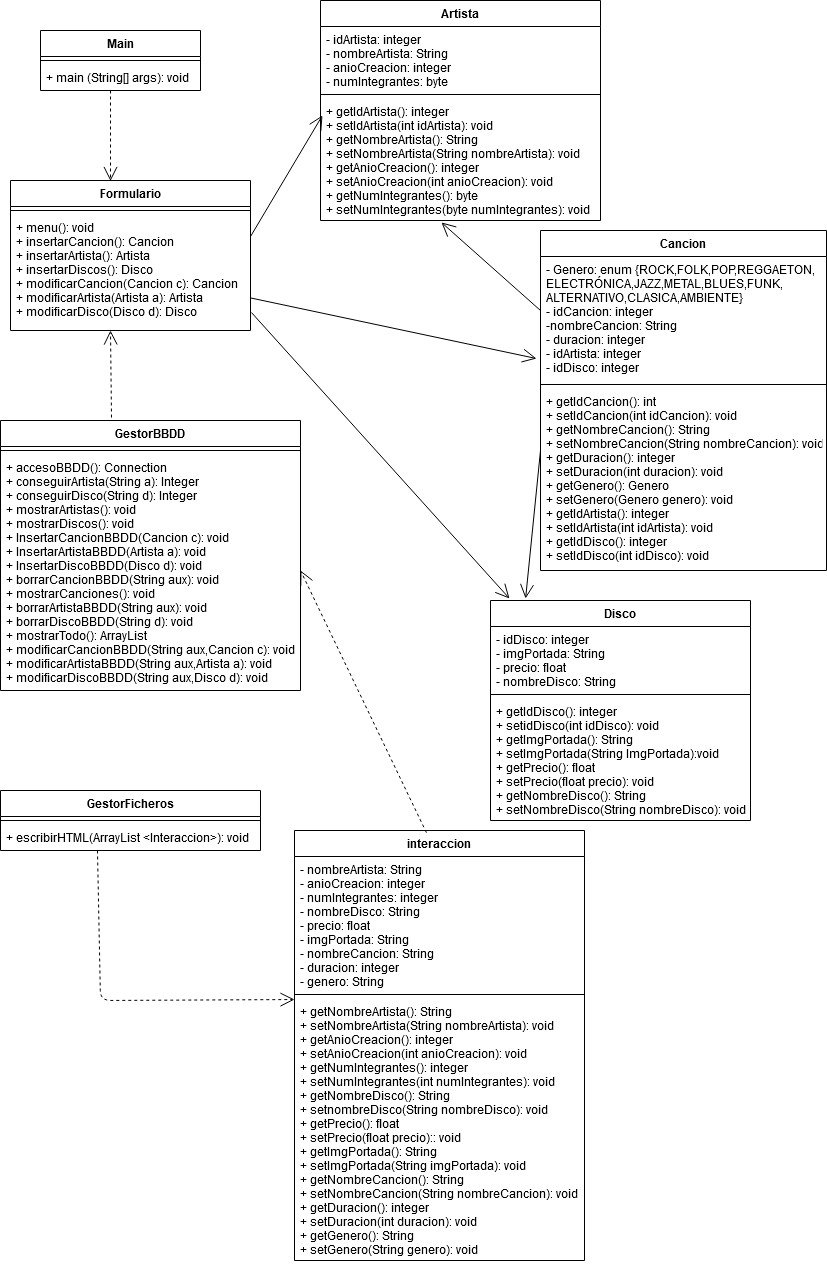
# Pruebas JUnit





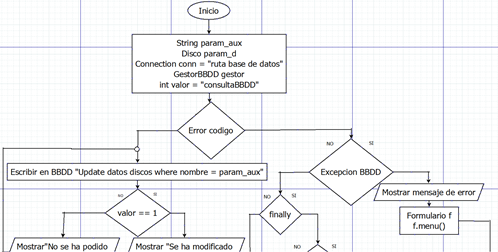
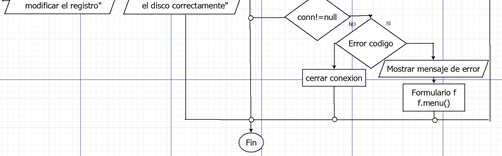


# Diagrama de clases



# Diagramas de flujo y Pseudocódigo

**Método GestorBBDD modificarDisco**



INICIO

String param\_aux

Disco param\_d

Connection conn = “ruta base de datos “

GestorBBDD gestor

Int valor = “consultaBBDD”

Si hay un error en el código Entonces

Si es un error de BBBDD Entonces

Mostrar mensaje de error

Formulario f

f.menu()

Fin si

Si no finally Entonces

Si conn!=null Entonces

Si hay error en código Entonces

Mostrar mensaje de error

Formulario f

f.menu()

Si no Entonces

Cerrar conexión con BBDD

Fin Si

Si no Entonces

Escribir en BBDD “Update datos discos where nombre = param\_aux”

Si valor == 1 Entonces

Mostrar “Se ha modificado el disco correctamente”

Si no Entonces

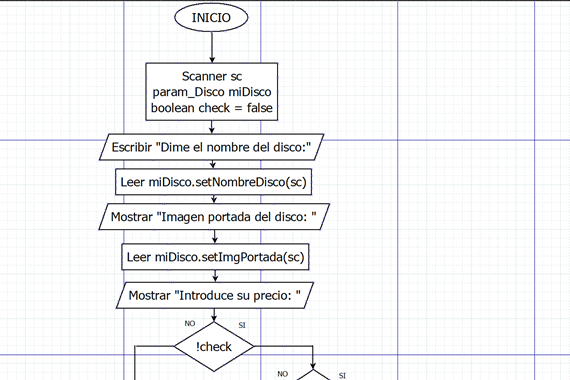
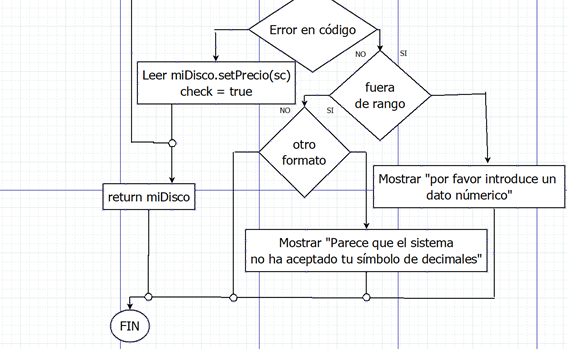
Mostrar “No se ha podido modificar el registro”

Fin si

Fin si

FIN

**Método InsertarDisco**



INICIO

Scanner sc

param\_Disco miDisco

boolean check = false

Mostrar “Dime el nombre del disco: “

Leer miDisco.setNombreDisco(sc)

Mostrar “Imagen portada del disco: “

Leer miDisco.setImgPortada(sc)

Mostrar “Introduce su precio “

Si !check Entonces

Si existe error en el código Entonces

Si esta fuera de rango Entonces

Mostrar “por favor introduce un dato númerico”

Si no Entonces

Mostrar “Parece que el sistema no ha aceptado tu símbolo de decimales”

Fin si

Si no Entonces

Leer miDisco.setPrecio(sc)

Check = true

return miDisco

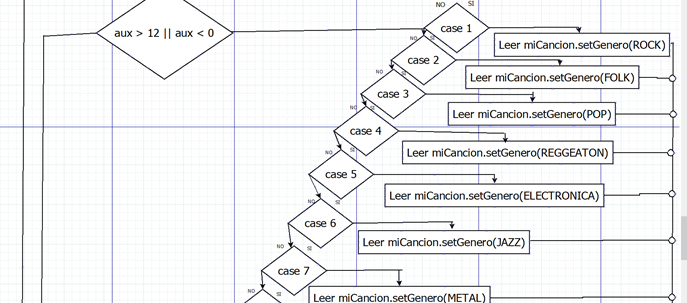
Fin si

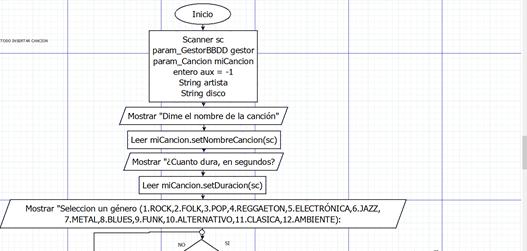
Si no Entonces

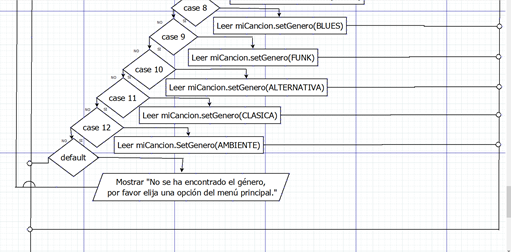
return miDisco

Fin si

FIN

**Método InsertarCancion**





# 

# 

# 

# 

# 

# 

# 

INICIO

Scanner sc

param\_GestorBBDD gestor

param\_Cancion miCancion

entero aux = -1

String artista

String disco

Mostrar “Dime el nombre de la canción”

Leer miCancion.setNombreCancion(sc)

Mostrar “¿Cuánto dura, en segundos?”

Leer miCancion.setDuracion(sc)

Mostrar “Selecciona un género (1.ROCK, 2.FOLK, 3.POP, 4.REGGAETON, 5.ELECTRÓNICA, 6.JAZZ, 7.METAL, 8.BLUES, 9.FUNK, 10.ALTERNATIVO, 11.CLASICA, 12.AMBIENTE)”

Hacer

Leer aux

Según aux Hacer

caso 1 :

Leer miCancion.setGenero(ROCK)

caso 2:

Leer miCancion.setGenero(FOLK)

case 3:

Leer miCancion.setGenero(POP)

case 4:

Leer miCancion.setGenero(REGGEATON)

case 5:

Leer miCancion.setGenero(ELECTRÓNICA)

case 6:

Leer miCancion.setGenero(JAZZ)

case 7:

Leer miCancion.setGenero(METAL)

case 8:

Leer miCancion.setGenero(BLUES)

case 9:

Leer miCancion.setGenero(FUNK)

case 10:

Leer miCancion.setGenero(ALTERNATIVO)

case 11:

Leer miCancion.setGenero(CLASICA)

case 12:

Leer miCancion.setGenero(AMBIENTE)

de otro modo

Mostrar “No se ha encontrado el género, por favor elija una opción del menú principal”

Fin Según

Mientras aux > 12

gestor.mostrarArtistas

Mostrar “Dime el artista”

artista = sc

miCancion.setIdArtista(gestor.conseguirArtista(artista))

gestor.mostrarDiscos

Mostrar “Dime el disco al que pertenece la canción”

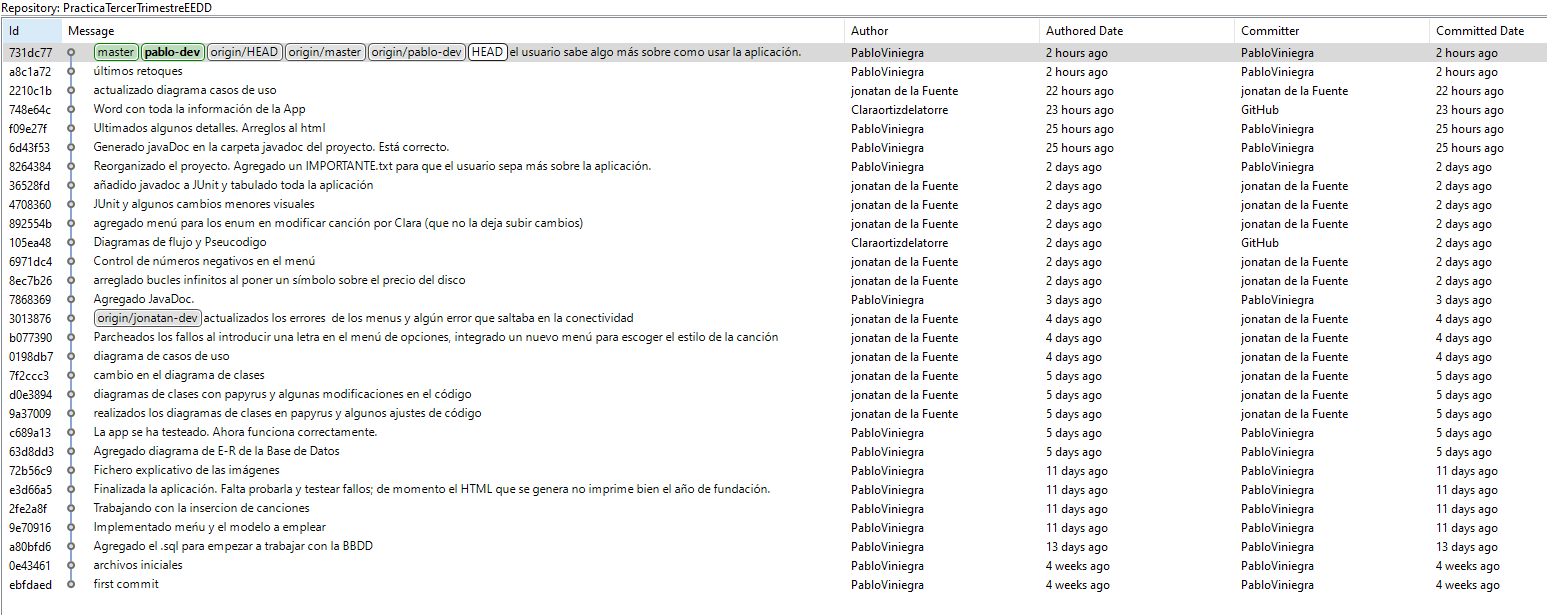
disco = sc

miCancion.setIdDisco(gestor.conseguirDisco(disco)

return miCancion

FIN

# Control de versiones



# DIAGRAMA E-R UTILIZADO

Imagen que contiene texto, mapa

Descripción generada automáticamente