



Universidad
Zaragoza

Trabajo Fin de Grado

Programa para la utilización de la metodología
verbotonal en la formación musical de niños con
problemas auditivos

Software for the use of the verbotonal methodology
in the musical training of children with hearing
problems

Autor

Irene Calabuig Adobes

Director

José Ramón Beltrán Blázquez

Agradecimientos

En primer lugar, dedico este trabajo a mis padres, Vicente y M^a Carmen, y a mis dos hermanos, Ana y Vicente, ya que gracias a ellos soy quien soy. También les doy las gracias por apoyarme en todos los aspectos durante toda mi etapa académica y por la gran paciencia que han tenido conmigo.

También quiero agradecer a aquellos profesores del grado de Ingeniería Electrónica y Automática que me han ayudado con su docencia y gracias a ellos estoy donde estoy.

A mi tutor de este proyecto, José Ramón, no sólo por todo su esfuerzo y dedicación dirigiendo este proyecto, sino también por haberme transmitido ese entusiasmo hacia la música que yo desconocía.

Y por último, pero no menos importante, a la profesora de música del Colegio la Purísima y San Antonio, Begoña, que ha sido la impulsora de este proyecto.

Resumen

Este trabajo tiene como objetivo implementar una aplicación que facilite el aprendizaje musical a niños con discapacidades auditivas. Para ello se han seguido las indicaciones pedagógicas de la profesora de música, Begoña Zarza, del Colegio la Purísima y San Antonio, y en consecuencia se han planteado dos tipos de ejercicios musicales para los niños: por una parte los relacionados con elementos rítmicos, cuyo objetivo es afianzar el pulso, y luego por otro lado los que presentan elementos melódicos, cuya finalidad es combinar el ritmo con la altura tonal.

Para la aplicación se han abordado problemas relacionados con la programación de la música tales como la reproducción temporal de notas musicales y la generación de los sonidos asociados a cada una de éstas. El resultado final es generar aleatoriamente pequeñas piezas musicales incrementando la dificultad del problema conforme se aumenta de curso, desde 1º hasta 4º de primaria.

La herramienta principal para la programación ha sido Spyder, donde se han implementado y evaluado los diferentes algoritmos en lenguaje Python.

Índice

1. Introducción	1
1.1. Motivación y Contexto	1
1.2. Objetivos	1
1.3. Alcance	2
1.4. Resumen del contenido	2
2. Contextualización musical	3
2.1. Figuras y silencios musicales	3
2.2. Ritmo	5
2.3. Melodía	6
2.4. Intervalos	7
2.5. Metodología Verbotonal	8
3. Generación de fragmentos musicales	9
3.1. Herramientas	9
3.2. Librerías	10
3.3. Desarrollo	11
3.4. Algoritmo de Generación de Notas	14
4. Descripción de la aplicación	22
4.1. Análisis de requisitos y especificaciones	22
4.2. Diseño	22
4.2.1. Diagrama de uso de la Aplicación	22
4.2.2. Menu principal	24
4.2.3. Elección tipo ejercicios	24
5. Análisis y conclusiones	33
6. Bibliografía	35
Lista de Figuras	37

Anexos	39
A. Programa generación de notas	40
A.1. Generación de notas	40
A.2. Elementos rítmicos	42
A.3. Elementos melódicos	44

Capítulo 1

Introducción

1.1. Motivación y Contexto

La Música en la Educación tiene una gran importancia en la sociedad, ya que existen estudios [1] que señalan que la música tiene efectos positivos en el desarrollo cognitivo, creativo, intelectual y psicológico de los niños. El problema es que dicha materia cada día es menos valorada por la sociedad y no se es consciente de los beneficios que puede aportar.

El trabajo melódico favorece la memorización de textos y la correcta acentuación de las palabras. El trabajo auditivo con melodía y timbre beneficia la capacidad de concentración y el aprendizaje de otras lenguas. Mientras que el trabajo rítmico ayuda a la comprensión de las relaciones matemáticas. Por ello, unos de los objetivos de este trabajo es la generación de piezas musicales donde se incluyen tanto elementos rítmicos como melódicos.

Por otro lado, este proyecto se ha realizado en colaboración con el Colegio La Purísima y San Antonio para niños sordos y está basado en la Metodología Verbotonal para su desarrollo. Me han proporcionado material didáctico del cual he extraído aquellos conceptos y valores necesarios para alcanzar el objetivo final.

1.2. Objetivos

Este proyecto tiene como objetivo principal implementar una aplicación que facilite el aprendizaje musical a niños con discapacidades auditivas, donde estarán incluidos los ejercicios relacionados con elementos rítmicos y los que presentan elementos melódicos.

De este modo, los objetivos propuestos son los siguientes:

- Generación aleatoria de pequeñas piezas musicales que contengan elementos rítmicos.
- Generación aleatoria de pequeñas piezas musicales que contengan elementos melódicos.
- Diseño e implementación de una aplicación.
- Análisis y conclusiones de los resultados obtenidos.

1.3. Alcance

Se han elaborado los algoritmos necesarios para la generación de notas musicales (en lenguaje de programación interpretado Python), donde se incluyen el tipo de compás y la duración y la escala de cada nota. Para la generación de dichos fragmentos musicales, se ha hecho uso de la librería *Music21*. Por otro lado, se ha utilizado el programa *LilyPond* que contiene un conjunto de herramientas que permite traducir los algoritmos mencionados a un módulo de salida a la representación propia de *Music21*, donde luego con una simple llamada permitirá visualizar y reproducir el resultado. Por último, se ha diseñado e implementado la interfaz gráfica de la aplicación mediante el uso de la librería *Tkinter*.

1.4. Resumen del contenido

El contenido de este trabajo se resume de la siguiente manera:

- Capítulo 2 (Contextualización musical): En este capítulo se explicarán algunos términos relacionados con la música que pueden resultar poco familiares en el ámbito de la Ingeniería Electrónica y Automática.
- Capítulo 3 (Generación de fragmentos musicales): En este capítulo se presentará la arquitectura del programa, incluyendo también la descripción de los algoritmos para la resolución de los dos tipos de ejercicios, rítmicos y melódicos.
- Capítulo 4 (Descripción de la aplicación): En este capítulo se explica la implementación y el funcionamiento de la aplicación.
- Capítulo 5 (Análisis y conclusiones): En este capítulo se detallan las conclusiones extraídas del proyecto y se realiza una evaluación general subjetiva del mismo.

Capítulo 2

Contextualización musical

Este trabajo contiene una cierta carga de teoría musical. Debido a que durante el proyecto se van a emplear ciertos conceptos musicales que pueden ser desconocidos para el lector, es conveniente hacer una pequeña introducción a ciertos elementos básicos de la teoría musical.

Este capítulo es una elaboración propia basada en las referencias [2], [3] y [4].

2.1. Figuras y silencios musicales

Una figura musical es un símbolo que representa gráficamente la duración musical de un determinado sonido en una pieza musical. Por otra parte, un silencio musical es lo mismo que una figura musical, pero en este caso lo que representa es la ausencia de sonido durante esas mismas duraciones.

La manera gráfica (Figura 2.1) de indicar la duración relativa de una nota es mediante la utilización del color o la forma de la cabeza de la nota, la presencia o ausencia de la plica, así como la presencia o ausencia de corchetes.



Figura 2.1: Partes de una figura musical

Por otra lado, también podemos encontrar figuras o silencios compuestos, que son aquellas que están acompañadas de un puntillo (Figura 2.2) modificando e incrementando su duración en la mitad.

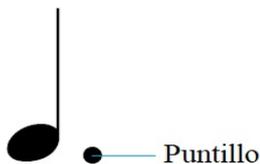


Figura 2.2: Negra con puntillo

A continuación (figura 2.3) se muestran las figuras básicas utilizadas en el proyecto:

Nombre	Figura	Duración	Silencio
Redonda		4	
Blanca con punto		3	
Blanca		2	
Negra		1	
Corchea		1/2	

Figura 2.3: Figuras rítmicas

Aparte de estas figuras, existen otras cuya duración es la misma pero rítmicamente se aprecia la diferencia. En la siguiente figura (Figura 2.4) se observan tres, de las cuales dos de ellas no aparecen en la tabla anterior. La segunda figura sonará con dos sonidos más cortos y la tercera con cuatro sonidos aún más cortos que la precedente, pero la suma de sus duraciones equivaldrán a la duración total de primera figura musical (*negra*).

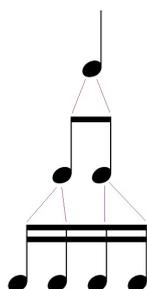


Figura 2.4: Equivalencia negra

2.2. Ritmo

La base de la música se encuentra en el ritmo, que es la repetición de golpes sonoros presentes en una composición musical. Para la formación de estos fragmentos rítmicos, se hace uso de las figuras y silencios musicales de una duración relativa al *tempo* de la pieza.

El tempo, también conocido como BPM (*Beats Per Minute*, o en castellano golpes por minuto), es la duración de las figuras musicales con exactitud. En concreto indica la duración del sonido y cuántas figuras de negra podemos encontrar en un minuto.

Las partituras están divididas en compases, y al principio de la composición se indica el tipo de *compás* de la misma. Éste se representa con dos números colocados en vertical (Figura 2.5).

- Unidad de compás: indica el número de tiempos de que consta el compás, es decir, la cantidad de figuras que caben en cada compás
- Unidad de tiempo: indica el tipo de dicha figura.

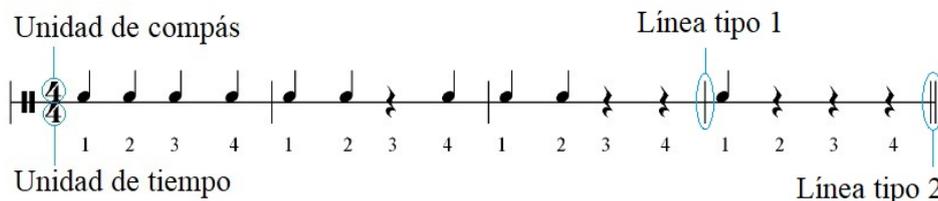


Figura 2.5: Fragmento Rítmico: Partes del compás

Se utilizan unas líneas verticales que atraviesan la composición llamadas *líneas divisorias*. (Figura 2.5).

- Línea tipo 1: representadas mediante una barra simple, representa la separación de compases.
- Línea tipo 2: representadas mediante una doble barra con la última línea más gruesa, señala el final de una obra.

Dependiendo de la unidad de compás, podemos distinguir tres tipos de compases básicos donde en todos ellos se cumple que el primer tiempo de todos los compases es fuerte, y el último de todos los compases es débil. Los tipos de compases son:

- Binario: el compás se representa mediante $\frac{2}{4}$
- Terciario: el compás se representa mediante $\frac{3}{4}$
- Cuaternario: el compás se representa mediante $\frac{4}{4}$ o con el símbolo de la Figura 2.6.



Figura 2.6: Símbolo compás $\frac{4}{4}$

2.3. Melodía

La melodía es una sucesión de notas con una *altura* y *duración* específicas, interpretadas de forma continua en el tiempo para producir una expresión musical. Para ello se utilizan siete notas fundamentales diferentes, y existen dos tipos de notaciones:

Notación latina	Do	Re	Mi	Fa	Sol	La	Si
Notación anglosajona	C	D	E	F	G	A	B

Figura 2.7: Tipos de notaciones

La *altura* citada anteriormente se define como la cualidad que diferencia un sonido agudo de un sonido grave y depende de la frecuencia del sonido. Dicha frecuencia se conoce como *tono*. Existen unas alteraciones de las notas fundamentales conocidas como sostenido y bemol, las cuales no voy a hacer hincapié debido a que no son utilizadas en mi proyecto.

La representación de estos sonidos se hace sobre un pentagrama, que es un conjunto de cinco líneas y cuatro huecos horizontales paralelos y equidistantes sobre el cual se escriben las figuras musicales y demás signos de notación. Cabe esperar que solo se puedan representar nueve notas, pero esto no ocurre, ya que también se pueden utilizar líneas y huecos adicionales por encima y por debajo del pentagrama.

2.5. Metodología Verbotonal

El *método verbotonal* [6] es un modelo de corrección fonética que tiene como objetivo la adquisición inconsciente del sistema fónico de una lengua, argumentando que la intelectualización de la corrección fonética va en perjuicio de la percepción auditiva.

Este método fue creado por Petar Guberina, especialista en las patologías del lenguaje quien elaboró un tratamiento para ayudar a niños con deficiencias auditivas y sus investigaciones clínicas le condujeron a la conclusión de que el elemento principal para percibir bien los sonidos no es el oído, sino el cerebro, por lo tanto, es éste el que hay que estimular. A grandes rasgos, si separamos verbotonal en dos palabras: *verbo* se emplea como sistema verbotonal en el sentido de palabra o habla, mientras que *tonal* responde al hecho de que se recurre al tono como elemento auxiliador en la corrección de los sonidos.

Algunos de los principios de la Metodología Verbotonal en el caso de la deficiencia auditiva o sordera son:

- Independencia directa entre habla, pensamiento y audición.
- Todo nuestro cuerpo actúa como emisor y transmisor de sonidos.
- Búsqueda de la funcionalidad auditiva.
- Búsqueda y desarrollo del campo óptimo de la audición.
- La afectividad y la expresividad son inherentes al lenguaje.

El material [5] proporcionado por la profesora Begoña Zarza está basada y ha servido como base fundamental para la realización de este proyecto en esta metodología donde el fundamento de la estructura del lenguaje está constituido por el ritmo, la entonación, la intensidad, el tempo de la frase, la pausa, la situación-contexto, el gesto y la mímica.

Capítulo 3

Generación de fragmentos musicales

3.1. Herramientas

En este apartado se va a hacer una breve explicación sobre las herramientas utilizadas en el proyecto:

- **Spyder** es un entorno de desarrollo integrado y multiplataforma de código abierto (IDE) para programación en lenguaje interpretado Python. Aquí es donde se ha desarrollado todo el proyecto.
- **LilyPond** [8] es un programa que utiliza una sencilla notación de texto como entrada, y produce una salida en el formato predeterminado (PDF, SVG, PNG y MIDI). Este programa está diseñado para visualizar y reproducir las partituras a partir de una entrada de texto que contiene toda la información musical para su posterior representación.
- **ImageMagick** [9] es un programa para crear, editar y componer imágenes de mapa de bits. Puede leer, convertir y escribir imágenes en una variedad de formatos como GIF, JPEG, PDF, SVG o PNG.
- **Windows Media Player** (conocido en inglés como Windows Media Player y abreviado como WMP) es un reproductor multimedia para el sistema operativo Microsoft Windows. Permite reproducir diversos formatos digitales como Audio CD, MP3 o AVI. También es capaz de interpretar archivos en formato MIDI¹.

¹es un archivo de extensión *.mid que almacena secuencias de dispositivos MIDI, es decir, sintetizadores, donde se recoge qué instrumento interviene, en qué forma lo hace y cuándo

3.2. Librerías

En este apartado se van a explicar las librerías utilizadas en el proyecto:

- **Music21** [7]: conjunto de herramientas que permiten manejar con facilidad ciertos aspectos relacionados con la música. Se basa en marcos y tecnologías preexistentes como Humdrum , MusicXML , MuseData , MIDI y Lilypond, pero este módulo utiliza un esqueleto orientado a objetos que facilita el manejo de datos complejos.

Es de las librerías más importantes del proyecto, ya que se ha usado para la generación, la visualización y la reproducción de partituras.

- **Random** [10]: proporciona acceso a funciones que elijan un elemento aleatorio de una lista.

Se ha usado para la generación aleatoria de las notas musicales, por un lado para saber si hay que crear una figura o un silencio, y por otro lado para saber su tipo de figura.

- **Tkinter** [10]: proporciona las funciones necesarias para el desarrollo de interfaces de usuario.

Junto a la librería *Music21*, es de las más importantes, ya que con ella se ha realizado toda la aplicación.

- **PIL (Python Imaging Library)** [11]: permite la edición de imágenes directamente desde Python. Soporta una variedad de formatos, incluidos los más utilizados como GIF, JPEG y PNG.

Se ha utilizado para abrir imágenes desde un directorio concreto, recortarlas y dimensionarlas.

- **Subprocess** [10]: permite generar nuevos procesos, conectarse a sus tuberías de entrada o salida y obtener sus códigos de retorno.

Se ha utilizado para abrir el archivo MIDI desde el directorio en el que se ha creado.

- **Os** [10]: permite acceder a funcionalidades dependientes del Sistema Operativo. Sobre todo, aquellas que nos refieren información sobre el entorno del mismo y nos permiten manipular la estructura de directorios.

Se ha utilizado para que cuando finalice la reproducción del fragmento, el programa Windows Media Player se cierre.

- **Time** [10]: este módulo proporciona varias funciones relacionadas con el tiempo. Se ha utilizado para que el programa espere un cierto tiempo, en función del compás establecido, para cerrar el programa Windows Media Player.

3.3. Desarrollo

La arquitectura del sistema de la parte relacionada con la música se presenta en la Figura 3.1 y contiene las herramientas citadas en la sección anterior:

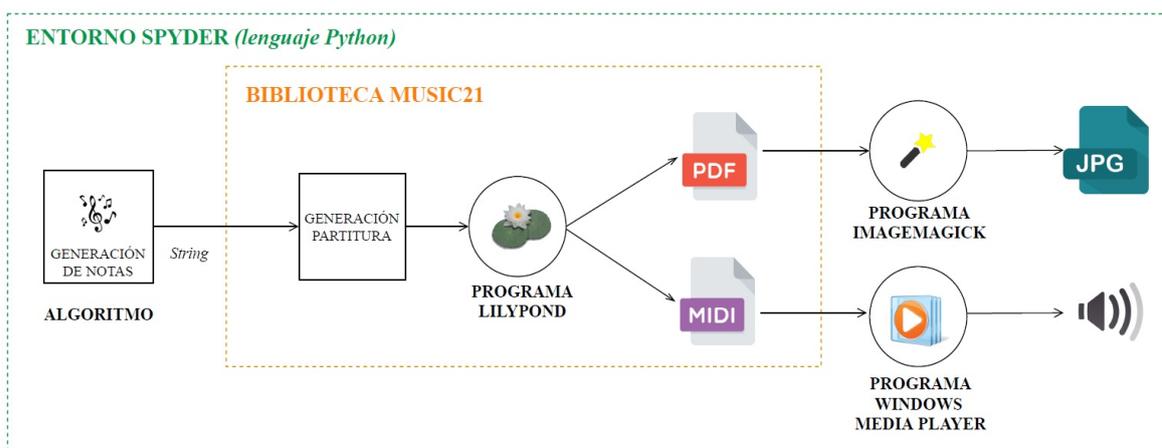


Figura 3.1: Arquitectura del sistema

Para el desarrollo del trabajo, ha sido imprescindible la utilización de la librería *Music21*, ya que contiene un conjunto de herramientas que permiten manejar con facilidad ciertos aspectos relacionados con la música, como la generación de partituras, la visualización y la reproducción.

Si nos fijamos en el anterior esquema (Figura 3.1), podemos ver que el primer paso es el uso del algoritmo de generación de notas. El parámetro que nos devuelve va a ser un dato tipo cadena (*String*).

Hay que destacar que previamente a la creación del algoritmo, se ha estudiado el tipo de entrada y notación necesaria para la generación de la partitura. Para ello se ha usado la clase *converter* y la función *parse()* de la biblioteca *Music21* de la siguiente forma (Figura 3.2):

```
fragmentoMusical= music21.converter.parse('tinynotation:'+compas_a+'/'+compas_b+' '+ notas )
##contenedor que contiene todos los datos del fragmento creado
```

Figura 3.2: Generación de la Partitura

En el parámetro de entrada de la función anterior, toma el valor *tinynotation*, que es el tipo de notación utilizado para la creación de las notas, además se indica el tipo de compás a utilizar, y por último, el término *notas* hace referencia a la salida del algoritmo de generación de notas.

Como se ha especificado antes, se ha utilizado la notación *TinyNotation*, la cual permite generar las notas con bastante facilidad aplicando las siguientes reglas:

- Los nombres de las figuras son: **a, b, c, d, e, f y g**, coincidentes con la notación anglosajona (Figura 2.7).
- El nombre de los silencios es: **r**.
- Después del nombre de la figura o silencio, es necesario colocar un número que indique la duración de la nota:
 - * **1**: nota entera (figura Redonda)
 - * **2**: $\frac{1}{2}$ de nota entera (figura Blanca)
 - * **4**: $\frac{1}{4}$ de nota entera (figura Negra)
 - * **8**: $\frac{1}{8}$ de nota entera (figura Corchea)
 - * **16**: $\frac{1}{16}$ de nota entera (figura Semicorchea)
- Si después del número colocamos un punto, '.', indicará que la nota colocada lleva puntillo.

La salida de la generación de la partitura, es un contenedor de datos que contiene toda la información del fragmento creado.

Una vez se tiene dicha información ya podemos pasar a su visualización y reproducción. Para ambos casos se ha utilizado el programa LilyPond, que utiliza una sencilla notación de texto como entrada, y produce una salida en el formato predeterminado.

En cuanto a la visualización, he utilizado como salida el formato PDF ya que de los tipos de salidas disponibles (PDF, SVG Y PNG) es el que mayor calidad presentaba.

```
nombreImagen = 'Partitura'  
pathImagen = 'C:/Users/Usuario/Desktop/Trabajo_Fin_Grado/Partituras/' + nombreImagen  
conv.write(partitura, fmt = 'lilypond', fp=pathImagen, subformats = ['pdf'])
```

Figura 3.3: Generación de la partitura en formato PDF

Una vez que se ha generado el archivo PDF, ha sido necesario pasarlo a formato JPG para su posterior incorporación en la aplicación. Para ello se ha utilizado el programa ImageMagick (Figura 3.4).

```
with im(filename= pathImagen + '.pdf', resolution=500) as image:
    image.compression_quality = 50
    image.save(filename=nombreImagen+'.jpg')

image2=Image.open(nombreImagen+".jpg")
```

Figura 3.4: Conversión de la partitura en formato PDF a JPG

En cuanto a la reproducción he utilizado como salida el formato MIDI siendo la única salida disponible (Figura 3.5).

```
conv2 = music21.converter.subConverters.ConverterMidi()
nombreMusica = 'Musica'
pathMusica = 'C:/Users/Usuario/Desktop/Trabajo_Fin_Grado/Musica/' + nombreMusica+".midi"
```

Figura 3.5: Generación de la partitura en formato MIDI

A continuación se pasará a ejecutar dicho archivo. Este será interpretado por el programa Reproductor Windows Media, y generará el sonido. En la función *CrearSonido()* pasamos el parámetro de entrada *tiempo*, donde en función del compás establecido para el ejercicio el intervalo será mayor o menor. Cuando finalice este, el programa de reproducción será cerrado automáticamente (Figura 3.6).

```
def CrearSonido(tiempo):
    check_output("C:/Users/Usuario/Desktop/Trabajo_Fin_Grado/Musica/Musica.midi", shell=True)
    ##Ejecutar el archivo de ese directorio
    time.sleep(tiempo)
    os.system('TASKKILL /F /IM wmpplayer.exe') # Cerrar el Reproductor Windows Media
    time.sleep(0.25)
```

Figura 3.6: Reproducción del fragmento

3.4. Algoritmo de Generación de Notas

En las siguientes páginas se muestra el diagrama de flujo que hace referencia al algoritmo de generación de notas. Éste se ha dividido en dos hojas debido a su extensión. En la Figura 3.10 aparece la generación aleatoria de figuras y en la Figura 3.11 la generación aleatoria de silencios.

Dicho algoritmo se usa para los dos tipos de ejercicios, y la función que lo implementa se llama *GenerarNotas* (*tipo_figuras*, *tipo_silencios*, *compas_a*, *puntillo*, *escala*). Los parámetros de entrada de la función dependen del curso y de la dificultad elegida y son de tipo *String*. Estos son:

- *tipo_figuras*: indica todos los tipos de notas a usar
- *tipo_silencios*: indica todos los tipos de silencios a usar.
- *compas_a*: indica la unidad de compás a usar.
- *puntillo*: '0' si no la figura no va a llevar puntillo; y '1' en caso contrario.
- *escala*: indica todas las escalas que puede tomar la nota, siguiendo la notación anglosajona (a, b, c, d, e, f y g).

A continuación voy a explicar el diagrama de flujo. Previamente, se introducirán los parámetros de entrada citados anteriormente de la función *GenerarNotas()*. A continuación, de forma aleatoria, se generará una figura (Figura: 3.10) o un silencio (Figura: 3.11).

Debido a que el funcionamiento de la generación aleatoria de figuras y de silencios es muy similar (con la única diferencia de que en el caso de los silencios hay menos tipos de figuras y no existe escala), voy a desarrollar el algoritmo para el caso de que se genere una figura.

Una vez se sabe que se va a generar una figura, de forma aleatoria, se elegirá uno de los caracteres del parámetro de entrada *escala*. El siguiente paso es saber el tipo de figura, donde para ello se tomará el valor del parámetro de entrada, *tipo_figuras* (el cual contiene una cadena de caracteres, que en este caso son números, donde cada uno de ellos hace referencia a una figura siguiendo la notación *TinyNotation*). Para saber el tipo de figura, se tomará un valor de ese parámetro de forma aleatoria. Si por ejemplo, al hacer la aleatoriedad sale valor "2", como la figura blanca es la que hace referencia a dicho número (siguiendo la notación usada), generaríamos ese tipo de figura.

Si nos fijamos en la Figura 3.10 podemos observar las siguientes diferencias para cada tipo de figura:

- Figura redonda: únicamente sale la elección del compás de $\frac{4}{4}$, ya que únicamente va a poder utilizarse en este tipo de compás debido a que su duración es de 4 pulsos. Pero aun así no va a poder colocarse siempre. Esto se explicará un poco más adelante.
- Figura blanca: aunque ésta pueda usarse en los tres tipos de compases, no va a poder colocarse siempre en el pentagrama en función del tipo de compás, por tanto se ha hecho esa diferenciación. Hay que añadir que para el caso de esta figura lleve puntillo, su duración será de 3 pulsos, y no podrá colocarse en un compás de $\frac{3}{4}$.
- Figuras negra, corchea y semicorchea: todas ellas pueden usarse en los tres tipos de compases, por tanto no van a dar ningún problema a la hora de colocarlos sobre el pentagrama.

Como se ha indicado anteriormente, para el caso de las figuras redonda y blanca, puede ocurrir el caso de que se haya generado una figura y luego no se pueda colocar. Esos casos son aquellos en los que la suma acumulada de las duraciones de las figuras supere algún múltiplo de la unidad de compás. Para entenderlo, se van a mostrar varios ejemplos con la figura redonda:

- El primer ejemplo se muestra en la Figura 3.7, donde se han ido generando las notas, y después de generar la figura blanca, se ha generado una figura redonda, la cual no va a caber en ese compás ya que si la añadiésemos superaría a 16 (la unidad de este compás es 4, y 16 es múltiplo de 4).

Figura 3.7: Reproducción del fragmento

- El segundo ejemplo se muestra en la Figura 3.8. Al igual que en el ejemplo anterior, se han ido generando notas, y después de generar la figura blanca, se ha generado una figura redonda, la cual tampoco va a caber en ese compás, ya que superaría a 8.

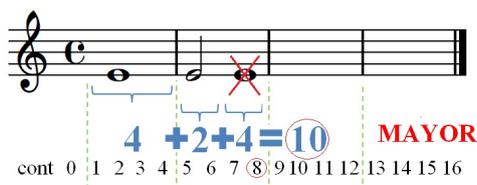


Figura 3.8: Reproducción del fragmento

En ambos casos, no se llegaría a añadir la nota, sino que se volvería al principio del diagrama y se volvería a generar otra figura o silencio hasta que se cumpliese la condición.

- El último ejemplo se muestra en la Figura 3.9. Aquí se generarían las notas sin ningún problema, ya que en ningún caso se supera el contador al final de cada compás

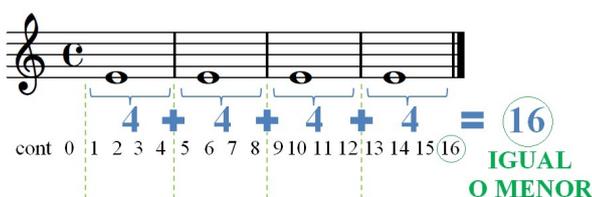


Figura 3.9: Reproducción del fragmento

En conclusión, la figura redonda únicamente se podrá colocar cuando el contador sea 0, 4, 8 ó 12, coincidente con el inicio de cada compás.

Para el resto de figuras se ha realizado el mismo análisis.

Otro aspecto a resaltar, es hablar del puntillo, que únicamente se va a poder añadir cuando la figura sea blanca.

Hay un parámetro de entrada llamado *puntillo*, el cual indicará si va a poder existir figura con puntillo o no. En caso afirmativo, y sólo para el caso que se haya generado una figura blanca, de forma aleatoria, se generará dicha figura con o sin puntillo.

Cuando se haya verificado que la nota se puede colocar en el fragmento, ésta se añadirá a una cadena de caracteres donde incluirá primero la escala, y seguidamente el valor del tipo de figura.

Para finalizar, cada vez que se genere una nota se verificará si el fragmento está completo o no. Para ello se comparará el contador actual, con el número total de pulsos que tenga el compás. Para calcular el número total, se multiplica la unidad de compás (númerador del compás) con la unidad de tiempos (denominador del compás).

El resultado sería:

- Compás $\frac{2}{4}$: 8 pulsos
- Compás $\frac{3}{4}$: 12 pulsos
- Compás $\frac{4}{4}$: 16 pulsos

El programa implementado para la generación de notas se encuentra en el Anexo A.1.

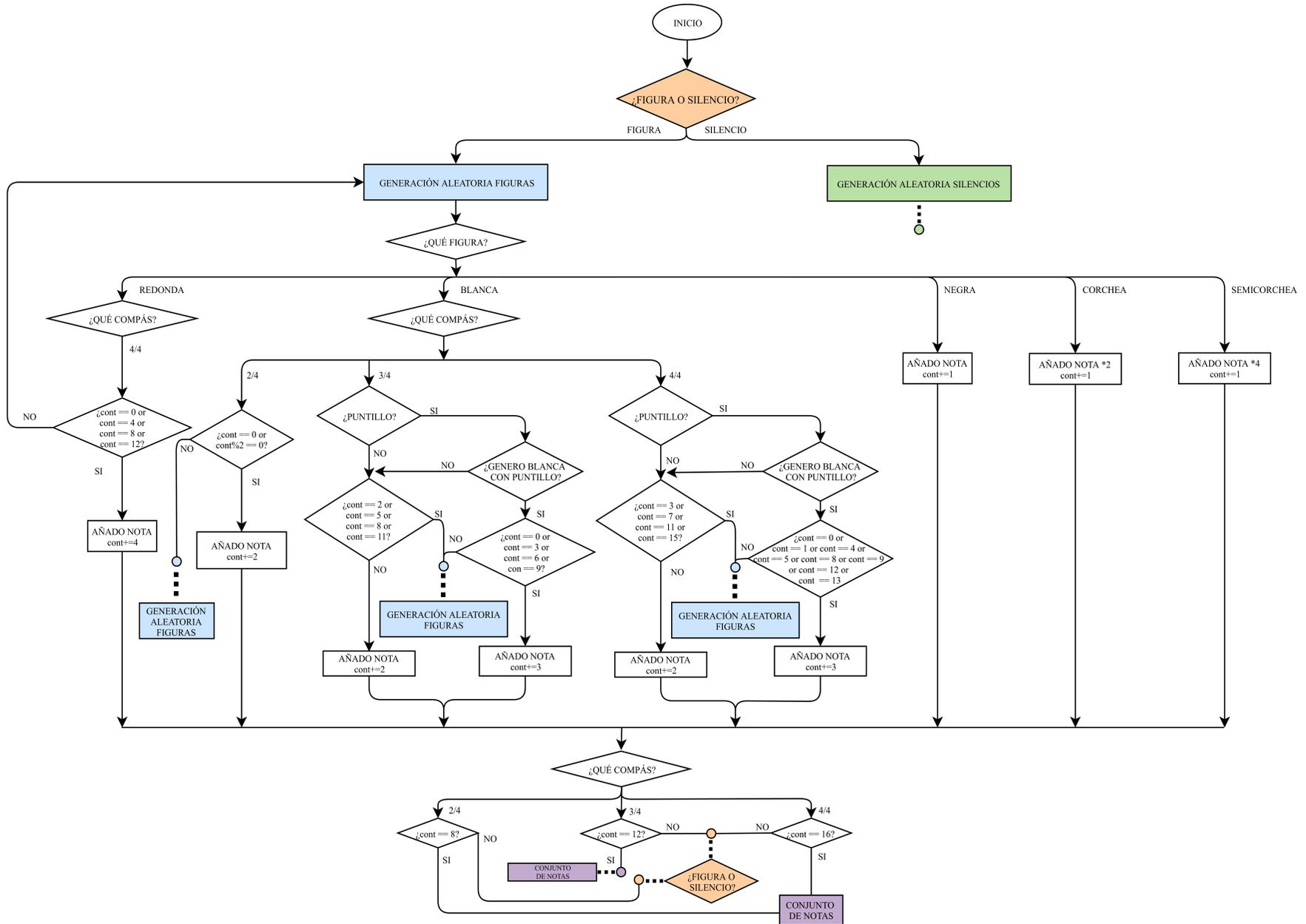


Figura 3.10: Diagrama de flujo (generación aleatoria de figuras)

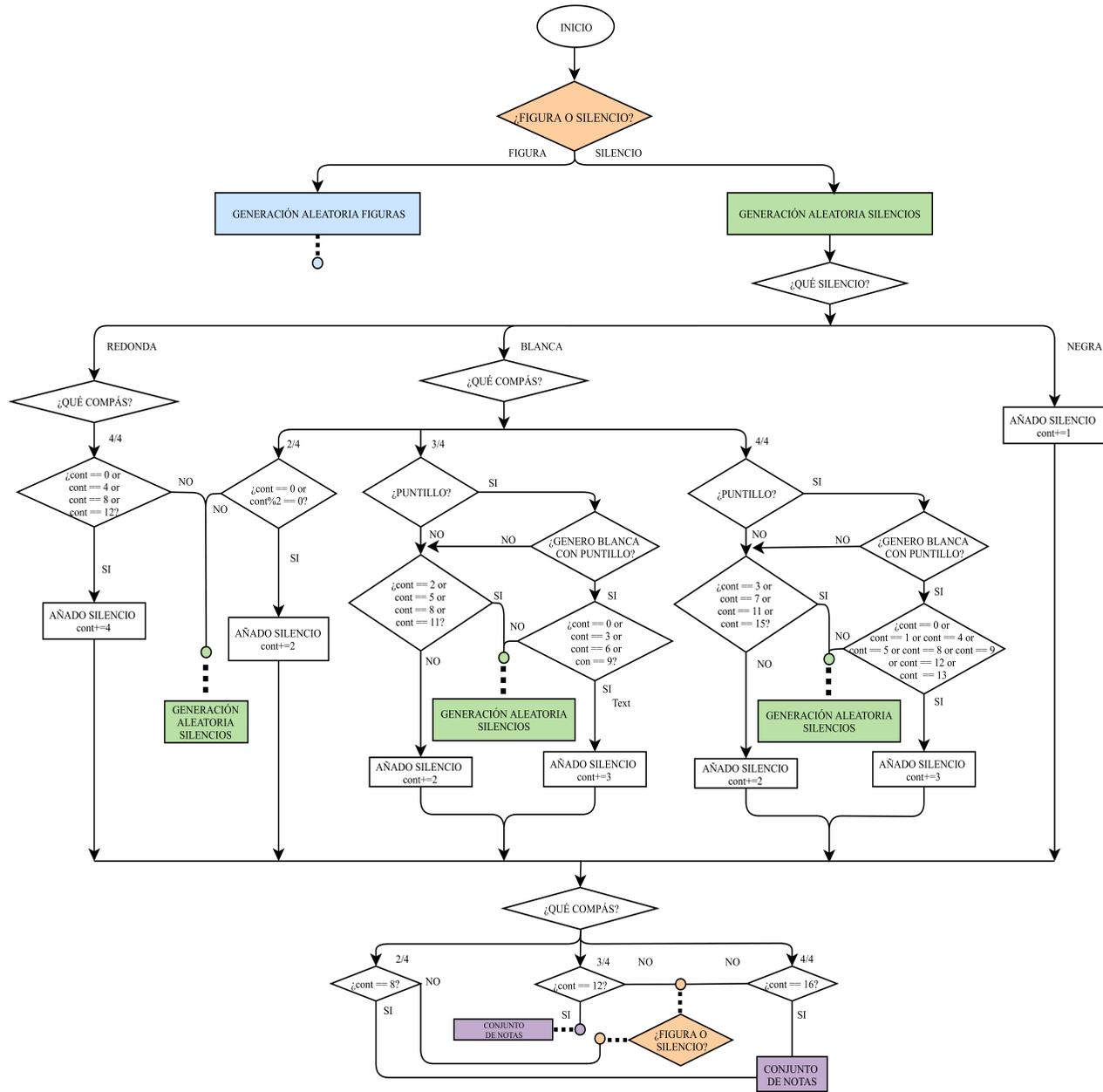


Figura 3.11: Diagrama de flujo (generación aleatoria de silencios)

El resultado del estudio ha sido dividir el problema en otros más pequeños, en función del curso, desde 1° de Primaria hasta 4° de Primaria, donde los cursos 3° y 4° de Primaria aparecen juntos tal y como aparece en la estructura del colegio.

Para cada curso se ha extraído el conjunto de notas a usar, luego se ha mirado el tipo de compases manejaban y por último, se ha establecido el número de niveles de dificultad en función de los tipos de figuras y silencios musicales que aparezcan en el fragmento a generar. El resultado para cada tipo de ejercicio ha sido este:

– Elementos Rítmicos:

<u>Curso: 1° Primaria</u>	<u>Curso: 2° Primaria</u>	<u>Cursos: 3° Primaria y 4° Primaria</u>
Notas: 	Notas: 	Notas: 
<ul style="list-style-type: none"> • Compás: $\frac{2}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2:  	<ul style="list-style-type: none"> • Compás: $\frac{3}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2:  	<ul style="list-style-type: none"> • Compás: $\frac{3}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2: 
<ul style="list-style-type: none"> • Compás: $\frac{4}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2:  	<ul style="list-style-type: none"> • Compás: $\frac{4}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2:  - Nivel 3:  	<ul style="list-style-type: none"> • Compás: $\frac{4}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2: 

Figura 3.12: Elementos Rítmicos

Para este tipo de ejercicios, el parámetro escala de la función *GenerarNotas()*, *escala*, tomará siempre el mismo valor.

El programa implementado para los elementos rítmicos está detallado en el Anexo A.2.

– Elementos Melódicos:

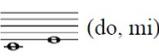
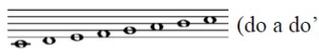
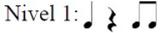
<u>Curso: 1º Primaria</u>	<u>Curso: 2º Primaria</u>	<u>Cursos: 3º Primaria y 4º Primaria</u>
Notas:  Escala:  (do, mi)	Notas:  Escala:  (do a do')	Notas:  Escala:  (do a do')
<ul style="list-style-type: none"> • Compás: $\frac{2}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2:  - Nivel 3:  - Nivel 4:  	<ul style="list-style-type: none"> • Compás: $\frac{4}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1: escala ascendente y descendente - Nivel 2: salto tercera ascendente seguido de salto segunda descendente - Nivel 3: salto tercera descendente seguido de salto segunda ascendente - Nivel 4: salto tercera ascendente y de salto tercera descendente - Nivel 5: salto tercera ascendente y (modificado) de salto tercera descendente - Nivel 6: saltos tercera y saltos segunda alternos - Nivel 7: saltos tercera ascendentes - Nivel 8: saltos tercera descendentes - Nivel 9: escala ascendente y descendente - Nivel 10: escala ascendente y descendente (modificado) 	<ul style="list-style-type: none"> • Compás: $\frac{4}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2:  - Nivel 3:  - Nivel 4: 
<ul style="list-style-type: none"> • Compás: $\frac{4}{4}$ • Dificultad: <ul style="list-style-type: none"> - Nivel 1:  - Nivel 2:  - Nivel 3:  		

Figura 3.13: Elementos Melódicos

El curso 2º de Primaria, para este tipo de ejercicio, no sigue la misma estructura que el resto de cursos, ya que todos los ejercicios propuestos en la programación tenían una estructura muy concreta y vi conveniente copiar tal cual todos los tipos de ejercicios propuestos. Por tanto, aquí no se ha aplicado el algoritmo. El programa implementado para los elementos melódicos está detallado en el Anexo A.3.

Los anexos A.2 y A.3 hacen referencia a la generación de elementos rítmicos y melódicos. Aquí es donde se detallan los distintos niveles de dificultad. En estas funciones los parámetros de entrada son:

- *ncurso*: indica el curso que se desea
- *compas_a*: indica la unidad de compás que se desea
- *ndificultad*: indica la dificultad del ejercicios que se desea

Dentro de estas funciones están definidos los valores para la generación de notas (tipo de figuras, tipo de silencios, si va a poder tener puntillo la figura o no y los tipos de escala) siguiendo las figuras 3.12 y 3.13 en función de estos parámetros de entrada descritos.

Capítulo 4

Descripción de la aplicación

4.1. Análisis de requisitos y especificaciones

Se definieron los siguientes requisitos para la creación de la aplicación:

- Debía ser una aplicación de escritorio.
- Uso sencillo para evitar confusiones a los usuarios.
- Interfaz colorida y botones grandes, ya que la aplicación está orientada a niños.
- Dos tipos de ejercicios (elementos rítmicos y elementos melódicos).

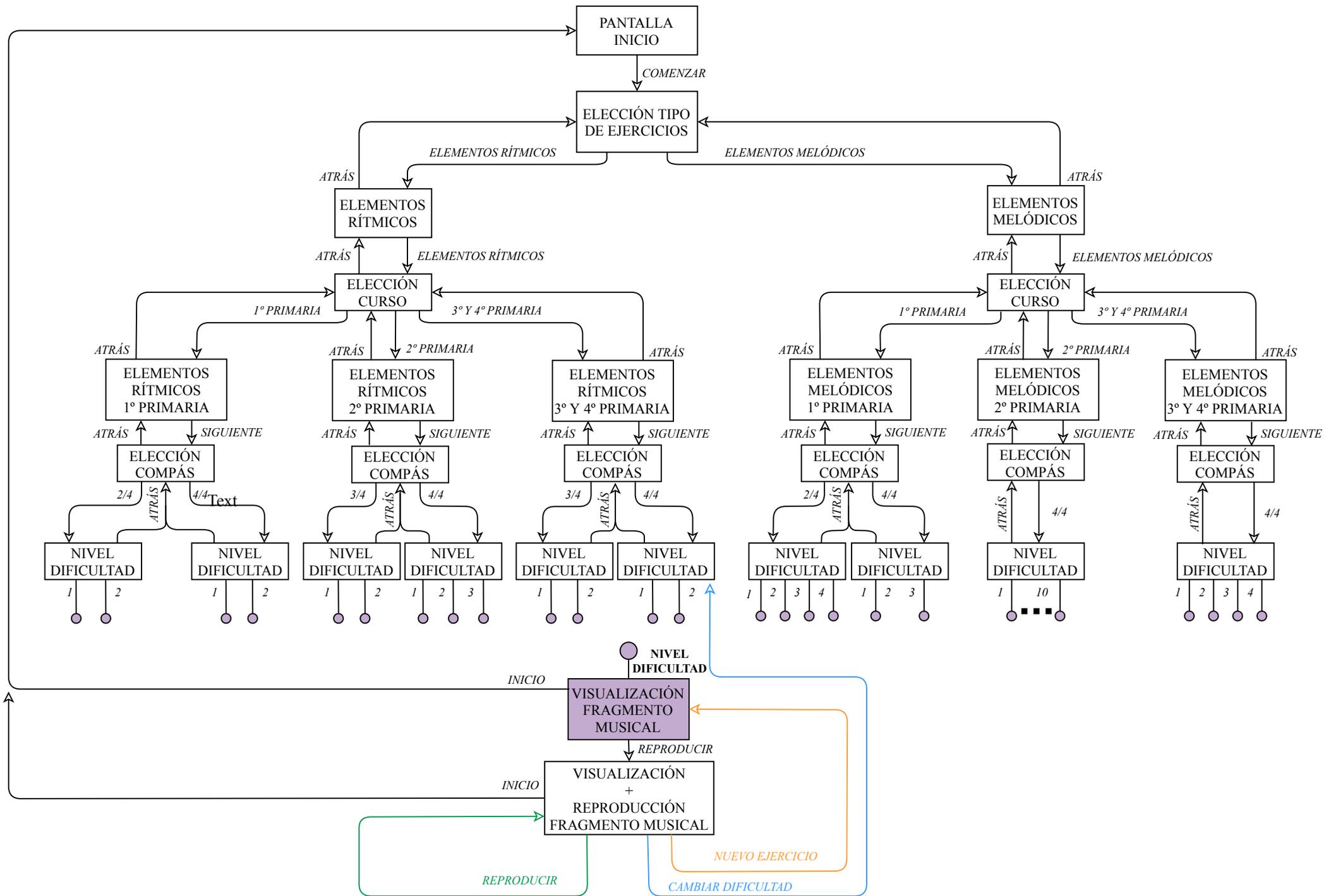
4.2. Diseño

4.2.1. Diagrama de uso de la Aplicación

A lo largo de esta sección se explicará de manera detallada todo el funcionamiento de la aplicación.

En la siguiente página se encuentra el diagrama de uso de la aplicación. Para entender como funciona, se han aplicado las siguientes reglas:

- Las palabras en cursiva que se encuentran próximas a las flechas indican el botón que aparece en la pantalla.
- Los bloques que aparecen después de la punta de una flecha indican el cambio de pantalla al pulsar el botón.
- Las flechas que apuntan hacia abajo indican que se va a pasar a la siguiente pantalla; y las flechas que apuntan hacia arriba indican que se va a retroceder de pantalla.
- Tal y como indica la figura, el recuadro con línea discontinua en color morado, se aplica cada vez que se haya escogido la dificultad (después de cada "*bola morada*").



4.2.2. Menu principal

La primera pantalla que ve el usuario al lanzar la aplicación es esta:



Figura 4.1: Pantalla Inicio

Pulsando sobre el icono *COMENZAR* pasaremos a la siguiente pantalla, *Elección tipo ejercicios* (Figura 4.2).

4.2.3. Elección tipo ejercicios

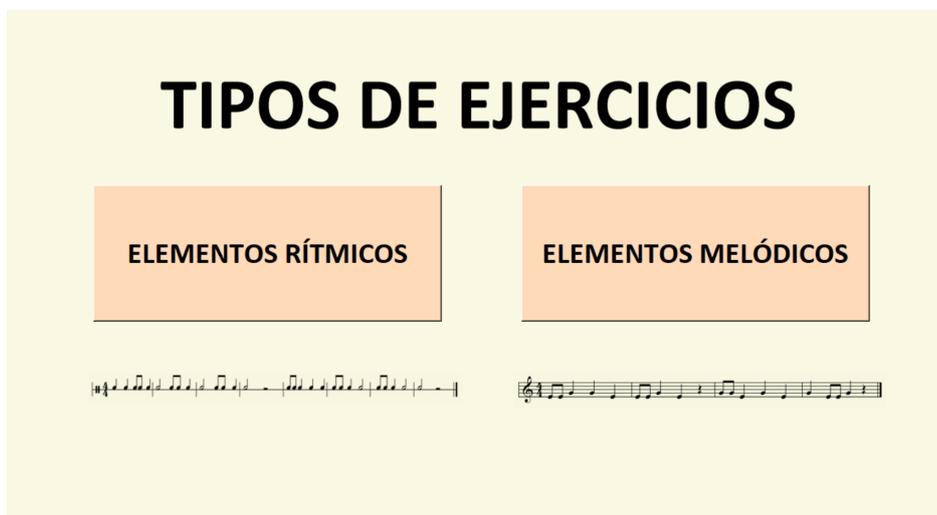


Figura 4.2: Elección tipos de ejercicios

En esta pantalla aparecen los tipos de ejercicios que se han desarrollado para la aplicación. Presionando sobre cualquiera de los dos botones, se retirará dicha pantalla, pasando a la siguiente asociada a cada uno de ellos. Pulsando al botón de la izquierda, *ELEMENTOS RÍTMICOS*, o al de la derecha, *ELEMENTOS MELÓDICOS*, se pasará a dicha actividad (Figura 4.3 o Figura 4.11 respectivamente).

Cabe destacar, que en las siguientes pantallas, existe un botón que pone “ATRÁS” y permitirá retroceder en caso de equivocación o cambio de actividad.

4.2.3.1. Elementos rítmicos

Esta pantalla (Figura 4.3) indica que se ha escogido la actividad **ELEMENTOS RÍTMICOS**.



Figura 4.3: Elementos Rítmicos

En esta subsección se van a ir explicando los pasos a seguir hasta llegar al objetivo principal del ejercicio, generar una pequeña pieza musical con elementos rítmicos. Para este tipo de ejercicio, debido a que el ritmo es la repetición de golpes sonoros, se ha tomado la decisión de elegir el tambor como instrumento para reproducir el fragmento. Por ello, con la finalidad de hacer la interfaz más infantil y llamativa, se ha añadido el dibujo de dicho instrumento.

Para mostrar todas las pantallas de este tipo de ejercicio, se ha escogido como ejemplo:

- Curso: 1º Primaria
- Compás: $\frac{2}{4}$
- Dificultad: 1

Si se pulsa sobre la tecla siguiente, se pasará a la siguiente pantalla, *Elección curso* (Figura 4.4).

Elección curso

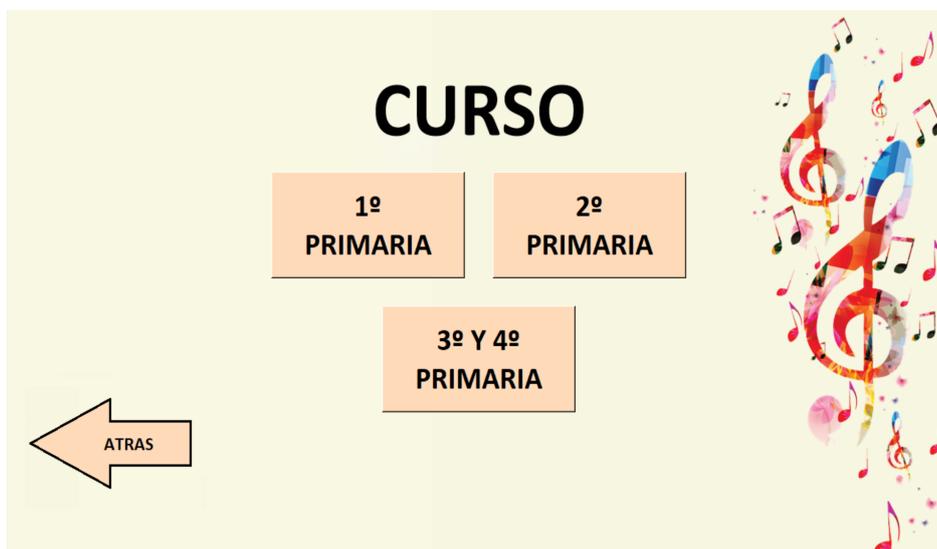


Figura 4.4: Elección Curso (Ejercicio Rítmico)

Esta pantalla permite elegir el curso.

Interfaz para elementos rítmicos con su respetivo curso

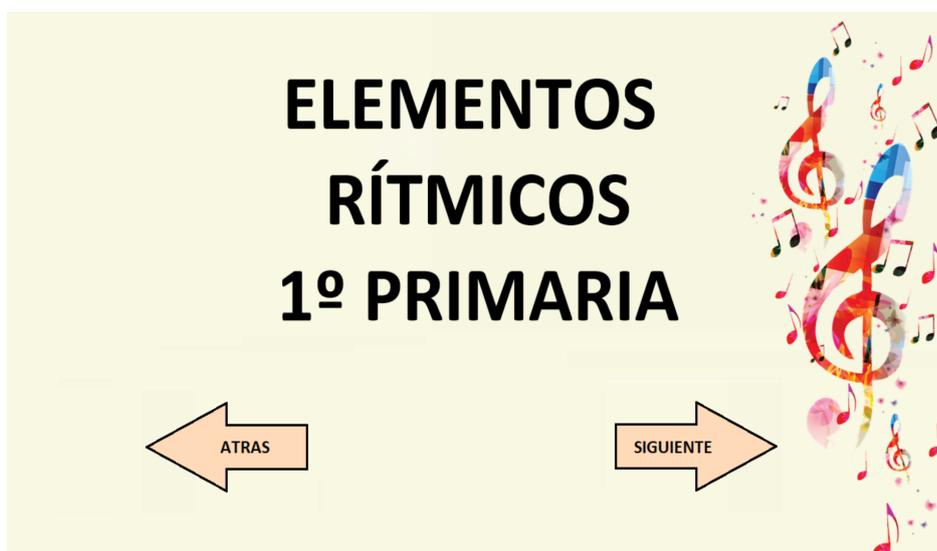


Figura 4.5: Elementos Rítmicos 1º Primaria

Como se ha dicho antes, para el ejemplo se ha cogido el curso 1º de Primaria, pero en caso de escoger otro, aparecería la misma pantalla pero cambiando el número. Si se pulsa sobre la tecla siguiente, se pasará a la siguiente pantalla, *Elección compás* (Figura 4.6).

Elección compás

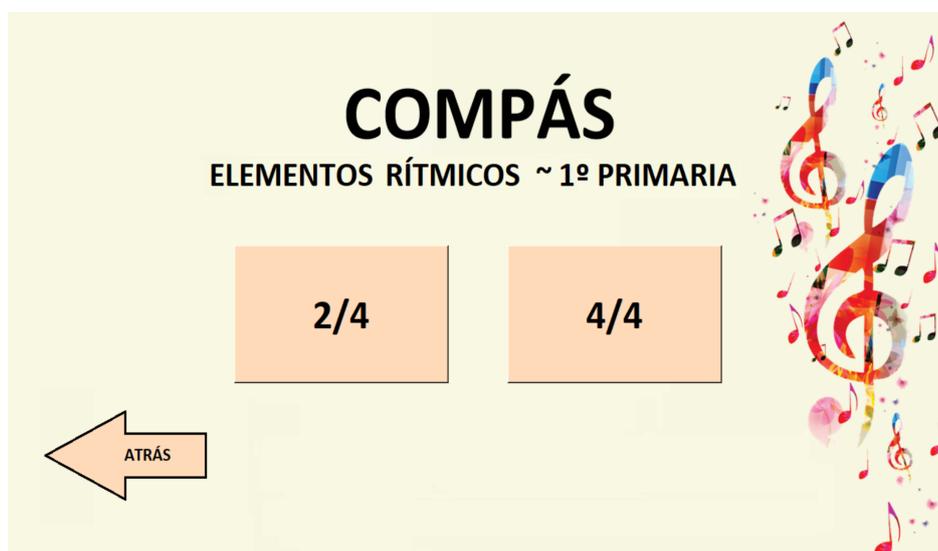


Figura 4.6: Elección compás (Ejercicio Rítmico)

Aquí se elegirá el tipo de compás. Una vez se haya pulsado alguno de los botones, pasaremos a la siguiente pantalla, *Elección dificultad* (Figura 4.7).

Elección dificultad

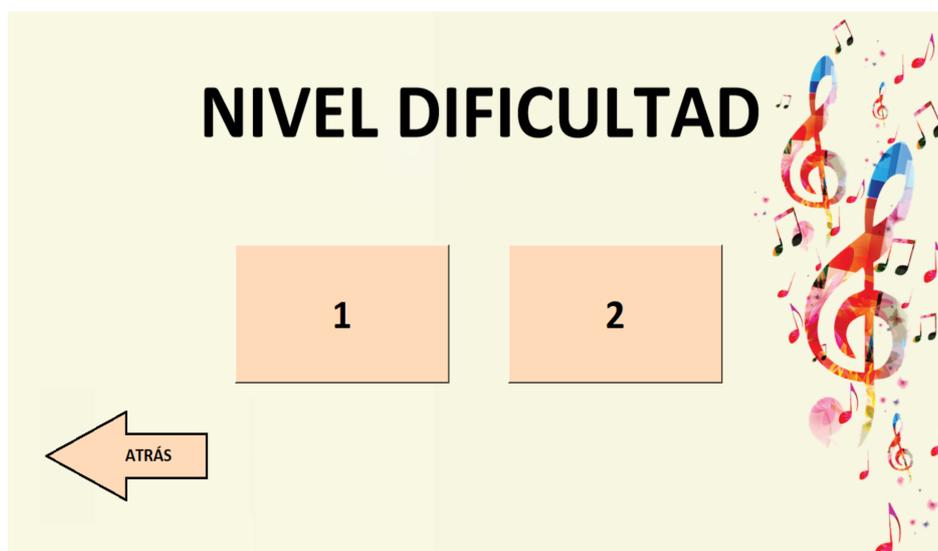


Figura 4.7: Elección dificultad (Ejercicio Rítmico)

En esta pantalla se elegirá en nivel de dificultad. Al seleccionar uno de ellos, se generará el fragmento musical y se pasará a la siguiente pantalla (Figura 4.8).

(Recordad que en el ejemplo se ha elegido *compás*: $\frac{2}{4}$ y *dificultad*: 1).

Generación fragmento musical rítmicos



Figura 4.8: Generación partitura (Ejercicio Rítmico)

Esta es la primera pantalla en la que aparece el fragmento musical representado de manera gráfica. En el momento que se presione el botón *REPRODUCIR* la tecla cambiará a color morado (Figura 4.9).

Reproducción del fragmento (Ejercicio Rítmico)



Figura 4.9: Reproducción de la partitura

Justo en ese momento comenzará a sonar el fragmento, la tecla permanecerá de color morado hasta que finalice la reproducción. Una vez termine la primera reproducción, aparecerá automáticamente la siguiente pantalla (Figura 4.10)

Fin de la reproducción



Figura 4.10: Fin del ejercicio (Ejercicio Rítmico)

En esta pantalla se permitirá hacer tres acciones: volver a reproducir el fragmento pulsando sobre el botón *REPRODUCIR*, generar un nuevo ejercicio pulsando sobre el botón *NUEVO EJERCICIO* o volver a la pantalla de Elegir Dificultad (Figura 4.7) pulsando sobre el botón *CAMBIAR DIFICULTAD*.

Si nos fijamos, en el momento que sobre la pantalla sale representado el fragmento musical (Figuras: 4.8, 4.9 y 4.10), en la parte superior izquierda sale un icono de una casa y un botón que pone *INICIO*. Si se pulsara sobre éste, nos llevaría a la primera pantalla del programa (Figura 4.1).

4.2.3.2. Elementos melódicos

Esta pantalla (Figura 4.11) indica que se ha escogido la actividad **ELEMENTOS MELÓDICOS**.

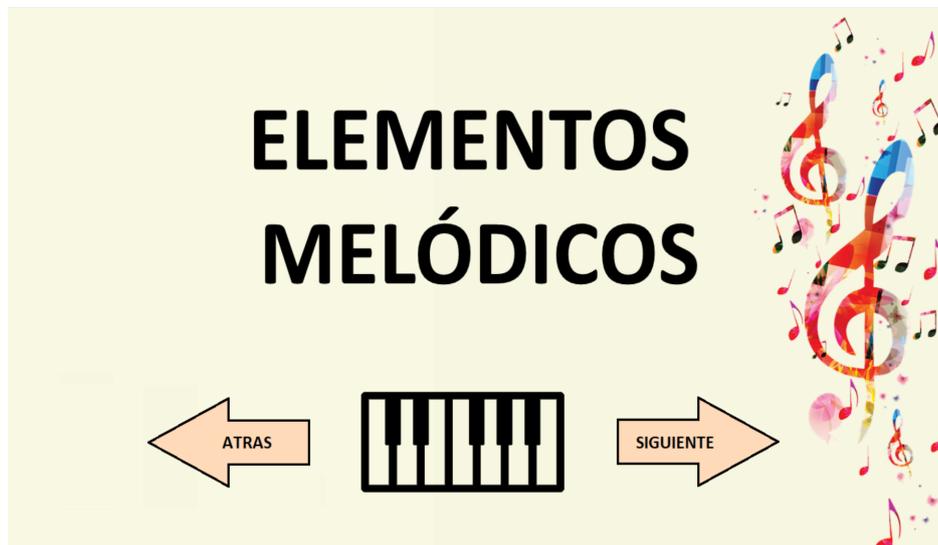


Figura 4.11: Elementos Melódicos

En esta subsección no se van a explicar los pasos a seguir, ya que son iguales a los de la sección anterior, sólo se van a mostrar las pantallas. Únicamente se van a resaltar aquellos aspectos visuales distintos.

El propósito es llegar al objetivo principal del ejercicio, generar una pequeña pieza musical con elementos melódicos.

Para este tipo de ejercicio, se ha elegido el piano como instrumento para reproducir el fragmento. Al igual que en la pantalla principal para el ejercicio de elementos rítmicos, la finalidad es hacer la interfaz más infantil y llamativa añadiendo el dibujo de dicho instrumento.

Para mostrar todas las pantallas de este tipo de ejercicio, se ha escogido como ejemplo:

- Curso: 3º y 4º Primaria
- Compás: $\frac{4}{4}$
- Dificultad: 4

El orden a seguir de las pantallas es de izquierda a derecha y de arriba a abajo.

Al pulsar sobre la tecla *SIGUIENTE* de la Figura 4.11, pasaremos a la siguiente pantalla, *Elección curso* (Figura 4.12a).

Al pulsar sobre alguno de los tres botones, (en este caso sobre *3º Y 4º PRIMARIA*), pasaremos a la siguiente pantalla (Figura 4.12b).

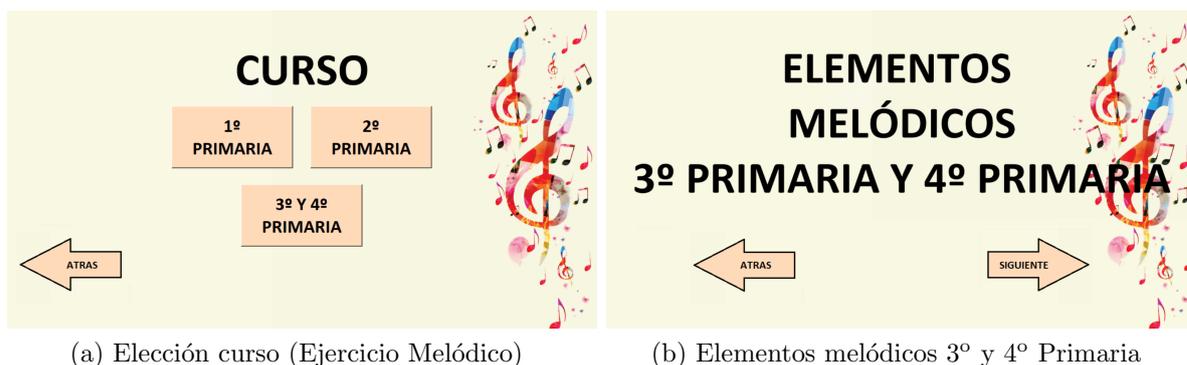


Figura 4.12: Interfaces gráficas para elementos melódicos (Elección curso)

En el momento que se pulse sobre la flecha *SIGUIENTE*, la pantalla cambiará para que se pueda elegir el compás.

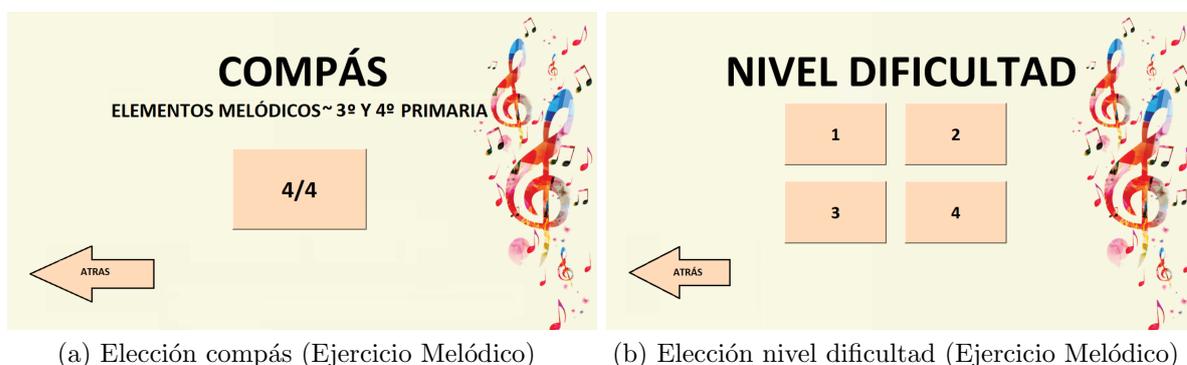


Figura 4.13: Interfaces gráficas para elementos melódicos (Elección compás y dificultad)

En este curso sólo está disponible el de $\frac{4}{4}$ (Figura 4.13a).

A continuación, una vez pulsado dicho botón, se pasará a elegir el nivel de dificultad (Figura 4.13b).

Como se ha indicado al principio de esta subsección, para el ejemplo se ha elegido *dificultad: 4*, por tanto se pulsará sobre ese botón y pasaremos a la generación del fragmento (Figura 4.14).



Figura 4.14: Generación de la partitura (Ejercicio Melódico)

A diferencia del ejercicio anterior, aquí se puede apreciar el cambio de escala de las notas (mi y sol).

Cuando se presione el botón *REPRODUCIR* la tecla cambiará a color morado (Figura 4.15a).



(a) Reproducción de la partitura (Ejercicio Melódico) (b) Fin de la partitura (Ejercicio Melódico)

Figura 4.15: Interfaces gráficas para la reproducción y finalización de la partitura

En el momento que deje de sonar la partitura, se verá la pantalla de la Figura 4.15b, y se tendrán las mismas opciones que en el otro tipo de ejercicio.

Capítulo 5

Análisis y conclusiones

En este proyecto se ha construido una aplicación de ayuda al aprendizaje musical basada en la metodología docente que se sigue en el Colegio La Purísima y San Antonio.

El objetivo principal del proyecto ha sido conseguido satisfactoriamente, por un lado se han abordado los dos tipos de ejercicios a partir de la creación de un algoritmo capaz de generar notas aleatorias cumpliendo los requisitos del usuario y por otro lado, se ha diseñado e implementado una aplicación.

Durante la realización del proyecto se han tenido las siguientes dificultades:

- En cuanto a la visualización de los fragmentos musicales, al incluir la librería music21, inicialmente se instaló el programa *MusicScore*¹, pero no hubo forma de conseguir una salida en un formato fácil de manejar como PDF o JPG , por tanto, al final se optó por usar el programa *LilyPond*. Al principio este programa también dió problemas, pero la solución fue crear un entorno y luego hacer las conversiones necesarias hasta llegar a formato JPG.
- En cuanto a la reproducción de los fragmentos musicales, hubo problemas para conseguir que el archivo generado en formato MIDI se abriera con el reproductor Windows Media. La solución fue establecer que los archivos en formato MIDI se abrieran con dicho reproductor de forma predeterminada. El otro problema fue que no se cerraba el programa, y por tanto al intentar volver a reproducir el fragmento, no funcionaba. Para ello hubo que añadir una instrucción para que se cerrara el programa pasado el tiempo de duración del fragmento.

¹programa de notación musical con soporte completo para reproducir partituras e importar o exportar MusicXML y archivos MIDI estándar.

El desarrollo del proyecto ha resultado muy atractivo e interesante, y gracias a ello he ampliado mis conocimientos sobre la teoría musical. Además, al ser un proyecto realizado en colaboración con un colegio para niños y niñas con necesidades especiales, el trabajo aporta una visión social y de compromiso necesaria también en el desarrollo de proyectos de Ingeniería.

Los conocimientos que he adquirido durante la realización de este proyecto respecto a mi formación en el grado de Ingeniería Electrónica y Automática son la integración de tecnologías de software diferentes y la conexión entre ellas. También he aprendido a manejar distintas librerías, que al final no deja de ser una colección de programas que facilita la ejecución de una serie de funciones relacionadas entre sí. Y por último, el desarrollo una aplicación gráfica, algo totalmente nuevo para mí.

Aunque el balance del proyecto resulta positivo, se podría mejorar en ciertos aspectos:

- En el ejercicio con elementos rítmicos, sería conveniente representar las notas sobre una línea y no sobre un pentagrama, ya que es la notación habitual. Sin embargo el software de generación de partituras que se ha elegido no permite desarrollarlo fácilmente.
- No ha sido probado por los usuarios, tanto estudiantes como profesores, aunque ese es uno de los requisitos fundamentales para comprobar su utilidad.

Las líneas de trabajo futuro son:

- Probar por los usuarios para comprobar su utilidad.
- Se podría incluir en la aplicación un abanico de instrumentos donde el usuario pudiese elegir con qué instrumento quiere escuchar el fragmento.
- Modificar la visualización de los elementos rítmicos, pasando las notas del pentagrama a una línea horizontal.

Capítulo 6

Bibliografía

- [1] La importancia de la música en la educación
<https://www.emagister.com/blog/la-importancia-de-la-musica-en-la-educacion/>
- [2] Rodrigo Martín Prieto. Herramienta para armonización musical mediante Answer Set Programming. Trabajo de fin de Grado de Ingeniería Informática. Universidad Da Coruña, 2016.
- [3] Hugo Riemann. *“Teoría General de la Música”*. Idea Books, 2005.
- [4] Joaquín Zamacois. *“Teoría de la Música”*. Labor, 2007.
- [5] Begoña Zarza Alzugaray. *“Método procesual nivel 1 y 2”* [Material del aula]. Colegio la Purísima y San Antonio, Zaragoza.
- [6] Logopedia educativa: Sistema verbotonál en sordos
<https://www.isep.es/actualidad/logopedia-educativa-sistema-verbotonál-en-sordos/>
- [7] Music21
<https://web.mit.edu/music21/>
Última actualización 08/06/2019
- [8] Lilypond
<http://lilypond.org/index.es.html>
Última actualización 22/03/2019
- [9] ImageMagick
<https://imagemagick.org/index.php>
- [10] Documentación bibliotecas estándar de Python
<https://docs.python.org/3.6/library/>

[11] Biblioteca Pillow

<https://pillow.readthedocs.io/en/stable/>

Lista de Figuras

2.1. Partes de una figura musical	3
2.2. Negra con puntillo	4
2.3. Figuras rítmicas	4
2.4. Equivalencia negra	4
2.5. Fragmento Rítmico: Partes del compás	5
2.6. Símbolo compás $\frac{4}{4}$	6
2.7. Tipos de notaciones	6
2.8. Escala musical	7
2.9. Ejemplo intervalo segunda e intervalo tercera	7
3.1. Arquitectura del sistema	11
3.2. Generacion de la Partitura	11
3.3. Generación de la partitura en formato PDF	12
3.4. Conversión de la partitura en formato PDF a JPG	13
3.5. Generación de la partitura en formato MIDI	13
3.6. Reproducción del fragmento	13
3.7. Reproducción del fragmento	15
3.8. Reproducción del fragmento	16
3.9. Reproducción del fragmento	16
3.10. Diagrama de flujo (generación aleatoria de figuras)	18
3.11. Diagrama de flujo (generación aleatoria de silencios)	19
3.12. Elementos Rítmicos	20
3.13. Elementos Melódicos	21
4.1. Pantalla Inicio	24
4.2. Elección tipos de ejercicios	24
4.3. Elementos Rítmicos	25
4.4. Elección Curso (Ejercicio Rítmico)	26
4.5. Elementos Rítmicos 1º Primaria	26
4.6. Elección compás (Ejercicio Rítmico)	27

4.7. Elección dificultad (Ejercicio Rítmico)	27
4.8. Generación partitura (Ejercicio Rítmico)	28
4.9. Reproducción de la partitura	28
4.10. Fin del ejercicio (Ejercicio Rítmico)	29
4.11. Elementos Melódicos	30
4.12. Interfaces gráficas para elementos melódicos (Elección curso)	31
4.13. Interfaces gráficas para elementos melódicos (Elección compás y dificultad)	31
4.14. Generación de la partitura (Ejercicio Melódico)	32
4.15. Interfaces gráficas para la reproducción y finalización de la partitura	32
A.1. Generación de notas (Parte 1)	40
A.2. Generación de notas (Parte 2)	41
A.3. Generación de notas (Parte 3)	42
A.4. Elementos rítmicos (Parte 1)	42
A.5. Elementos rítmicos (Parte 2)	43
A.6. Elementos melódicos (Parte 1)	44
A.7. Elementos melódicos (Parte 2)	45