

Dagda - Software de Auxílio ao Ensino de Música na Educação Básica

Dorgival da Silva Pereira Netto¹, Francesco Pagani Galvão¹, Luiz Felipe de Souza Jimenez¹, Renan Augusto Monteiro Duarte¹

¹Instituto Federal de Mato Grosso do Sul

Campus Corumbá

Abstract. *Law nº 11.769/ 2008 provides for the mandatory teaching of music in primary education. Even after 12 years of sanctioning the referred law, many schools have not been able to implement it due to the lack of structure they present or the superficiality with which music teaching is treated in Brazil. Thus, this article presents the development of software, which can be used to aid in the teaching-learning process of music in Brazilian primary education schools. The software can recognize and reproduce elements relevant to music theory from sheet music in PDF formats, such as a note, pause, measure, duration of notes, and music time.*

Resumo. *A Lei nº 11.769/2008 dispõe sobre a obrigatoriedade do ensino da música na educação básica. Mesmo após 12 anos da sanção da referida lei, muitas escolas não puderam efetivar sua implementação, pela falta de estrutura que apresentam ou pela superficialidade com que o ensino de música é tratado no Brasil. Dessa forma, este artigo apresenta o desenvolvimento de um software, que possa ser utilizado como auxiliar no processo de ensino-aprendizagem da música em escolas brasileiras de educação básica. O software é capaz de reconhecer e reproduzir elementos pertinentes à teoria musical a partir de partituras em formato PDF, tais como uma nota, pausa, compasso, duração das notas e andamento da música.*

1. Introdução

O processo de ensino-aprendizagem da música pode ser dado de diversas maneiras, variando de uma cultura para outra. Entretanto, mesmo possuindo aspectos diferentes em cada cultura, esse processo possui um objetivo em geral: transmitir todos os costumes e crenças através da música e do seu ensino. Em seu trabalho, Ilari (2014) apresentou como a educação musical poderia estar relacionada com o desenvolvimento cerebral de crianças de 5 a 14 anos de idade beneficiando vários sistemas do neurodesenvolvimento, como o controle de atenção, a memória, a orientação espacial, a movimentação, entre outros. Em outro viés, Kater (2014) afirma que a educação musical auxilia o estudante a esclarecer a realidade na qual se encontra, estimulando uma visão crítica e criativa de mundo, possibilitando que o mesmo possua conhecimento de aspectos pouco perceptíveis no cotidiano, o que contribui para formar a sua consciência de mundo.

No Brasil, esse temário é reconhecido pela Lei 11.769/2008 [Brasil, 2008], que afirma que a música, juntamente com a dança e o teatro, devem fazer parte da unidade curricular da educação básica, promovendo o desenvolvimento da cidadania do educando. Assim, com a sanção da Lei 11.769/2008, cria-se terreno fértil para que políticas públicas relacionadas ao tema venham à tona [Kleber, 2012]. Entretanto,

mesmo após a sanção desta lei, que definiu a música como componente obrigatório da grade curricular, o que se pode observar são muitos obstáculos relacionados aos valores culturais e sociais da sociedade brasileira, que consideram o ensino da arte irrelevante [Falcão e Nascimento, 2010; Kleber, 2012].

Ainda assim, mesmo havendo obstáculos para a implementação dos processos de ensino-aprendizagem da música nas salas de aulas, há muitos projetos no âmbito da informática que buscam desenvolver programas de software educacional voltados para o ensino musical, programas que se tornam importantes para o domínio desse tipo de conhecimento [Flores, 2002]. Por exemplo, Millete et al. [2004] defende que a utilização de tecnologias para o ensino musical gera uma integração entre dois mundos distintos, levando a um ensino multidisciplinar e interdisciplinar, dando acesso a diversas teorias, modelos e práticas desses mundos. Podemos observar essa integração em Freitas (2012) que, ao se utilizar de autômatos, pôde reconhecer a formação de acordes em músicas no formato *MDI (Music Instrument Digital Interface)*.

Em virtude disso, esta pesquisa tem por objetivo desenvolver um software que seja utilizado como auxiliar no processo de ensino-aprendizagem da música em escolas brasileiras de educação básica, possibilitando que o aluno compreenda a forma de execução de uma partitura, mesmo sem a presença de um professor. O software reconhece e reproduz notação musical a partir de partituras em formato *PDF*, retirados do site *IMSLP*¹ e convertidos pelo programa *Audiveris*, executados através de um *Application Programming Interfaces (API)* do navegador *Google Chrome*, além de organizar os dados obtidos do reconhecimento de notação musical a partir de partituras em arquivos *PDF*.

2. Fundamentação teórica

Segundo Lacerda (1961), a música é a arte do som, que possui quatro propriedades: duração, intensidade, altura e timbre. A junção destas formam o que conhecemos como som musicais. Dentro de uma partitura, para se representar cada propriedade, é utilizada a representação escrita, como a figura da nota, os sinais de dinâmica, a posição da nota no pentagrama e clave, a indicação da voz ou instrumento que deve executar a música, respectivamente, como mostrado na Figura 1.

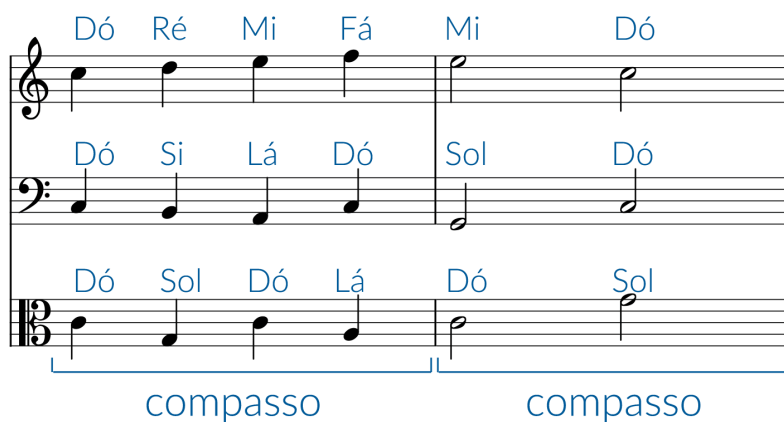


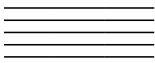



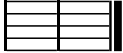
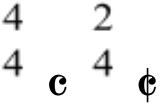
Figura 1. Claves, figuras e notas em dois compassos

¹ Disponível em: <https://imslp.org/wiki/Main_Page> Acesso em: 15 jun, 2020.

Fonte: Academia de Composição (2017).

Cada figura possui um significado diferente dentro da partitura, podendo modificar a forma como a música é executada, a altura das notas, se é uma pausa ou não, indica em qual clave deve ser lida a partitura, em qual fórmula de compasso está. Esses aspectos serão importantes para o programa. O Quadro 1 mostra as definições de cada elemento musical de uma partitura, conforme Lacerda (1961).

Quadro 1. Significado dos elementos musicais de uma partitura

Elemento	Representação	Significado
Pentagrama		Conjunto de 5 linhas e 4 espaços, onde são escritas as notas, quanto mais alto é o som, mais para cima a nota estará no pentagrama.
Nota		É a representação gráfica do som, a figura varia conforme a duração deste, quanto menor a duração mais bandeirolas a figura terá.
Pausa		É o silêncio na música, tomam o nome da nota de duração correspondente.
Clave		As três claves mais utilizadas são a clave de Dó, de Fá, e de Sol. A clave determina a posição da nota de seu nome no pentagrama, servindo de guia para nomear todas as outras notas.
Compasso		Compasso é a divisão da música em pequenas partes iguais ou variáveis. Pode também indicar quantas divisões existem na partitura, ao lado é possível ver dois compassos, separados por uma barra de compasso no centro.
Fórmula de Compasso		São dois números que indicam a unidade de tempo e o número de tempos do compasso, também pode ser escrito por meio do símbolo ao lado.

Fonte: Elaboração dos autores

3. Metodologia

Inicialmente, foi realizada uma análise bibliográfica buscando fundamentar o tema proposto, sucedida de uma síntese dos principais materiais obtidos. Esta etapa constitui o embasamento teórico da pesquisa a partir da análise de trabalhos e artigos nas áreas de Educação Musical, Informática e Tecnologias para o ensino da música.

Ao final desta etapa, foi gerado um levantamento dos principais programas e aplicativos que pudessem auxiliar no desenvolvimento do programa. Observou-se a necessidade de um algoritmo de Reconhecimento Óptico de Caracteres (OCR), capaz de

reconhecer elementos em um documento em formato PDF. Assim, o *Audiveris*², um programa baseado em Java, que tem a capacidade de ler partituras em formato *PDF* e convertê-las para o formato *MXL* (arquivo *MusicXML* compactado), através do programa *Tesseract OCR*, que permite o reconhecimento de notação musical em arquivos *PDF*, foi escolhido para a utilização. A escolha desse programa se deu por este possuir código aberto, o que permite a sua utilização por parte dos autores, além de ser um programa que utiliza o conceito de *Machine Learning*, podendo aprender com cada reconhecimento que executa.

Sendo assim, o *JavaScript*, linguagem de programação interpretada e estruturada de script, multiparadigma de alto nível, foi escolhida para a implementação do software. A escolha dessa linguagem se deu pela integração que possui com o ambiente server-side *Node.js*, que possibilita o processamento de uma grande quantidade de dados, sem perder performance no processo.

Posteriormente, foi realizada uma pesquisa das principais linguagens e bibliotecas de programação que facilitam a organização dos dados obtidos no reconhecimento que o *Audiveris* realizava. Assim, com o auxílio do *npm* (um gerenciador de pacotes para a linguagem *JavaScript*) foram utilizadas as bibliotecas *MusicXML-Interfaces* (Biblioteca de utilitários JavaScript de baixo nível para analisar, serializar, criar e corrigir músicas do *MusicXML*), *File-System* (Biblioteca responsável por carregar arquivos para serem utilizados no programa), *JsZip* (Biblioteca responsável por criar, ler e editar arquivos *.zip*), *Retro-Audio-Js* (Biblioteca que realiza a execução de partituras no formato *JSON* pela *API Web-Audio* do navegadores de internet), a fim de realizar o processamento dos dados obtidos do reconhecimento da partitura e sua conversão para o padrão de arquivos *JSON*.

Como visto na Figura 2, os dados retirados da partitura, após o processamento, são o título da partitura (*title*), o tempo de execução (*tempo*), o compasso (*time-signature*) e as notas (*sheets* contidos dentro do *score*). Cada nota é representada por seu nome (C, D e E), sua oitava (4 e 5) e a sua duração (.16 e .32). As pausas, neste formato de arquivo, são representadas por um traço (-), seguidas da sua duração.

Figura 2 – Exemplo de arquivo *JSON*

```
1 {
2   "title":"(Happy Birthday)",
3   "tempo":"60",
4   "time_signature":"3/4",
5   "score":[
6     {
7       "instrument":"oscillator-square",
8       "volume":0.5,
9       "sheet":[
10        "-.8","g4.32","g4.32","g4.32","g4.32","a4.16","g4.16","c5.16",
11        "b4.16","-.16","g4.32","g4.32","g4.32","g4.32","a4.16","g4.16",
12        "d5.16","d5.16","c5.16","c5.16","e5.32","e5.32","g5.16","e5.16",
13        "c5.16","c5.16","c5.16","c5.16","b4.16","a4.16","f5.32","f5.32",
14        "e5.16","c5.16","d5.16","d5.16","c5.16","c5.16","g4.32","g4.32",
15        "g4.32","g4.32","c5.16","c5.16","-.8","-.8"
16      ]
17    },
18    {
19      "instrument":"oscillator-square",
20      "volume":0.5,
21      "sheet":[
22        "-.8","-.8","-.8","-.8","g4.32","g4.32","g4.32","g4.32","e4.16","e4
```

Fonte: Elaboração dos autores, 2020.

² <https://github.com/Audiveris/audiveris.git>

Após o estágio de organização dos dados, foram realizados testes de execução para averiguar a qualidade do reconhecimento e organização das partituras. Para isso, foi utilizada a biblioteca *Retro-Audio-JS*, que se baseia na *Web Audio API*, padrão utilizado por navegadores de internet para a execução de áudios. Essa biblioteca realiza a leitura de arquivos que contenham informações referentes a uma partitura no formato *JSON*, assim como apresentado na Figura 2.

4. Resultados e Discussões

O software consiste em 3 fases principais e essenciais para seu funcionamento. A **primeira fase** consiste na conversão de partituras pelo programa *Audiveris*, que realizará o reconhecimento de notação musical em arquivos no formato *PDF*, convertendo e exportando elas para o formato *MXL*. Esse processo é realizado de forma automática pelo *Audiveris*, sendo necessário somente que o usuário escolha o arquivo e selecione a opção *Transcribe Book* para realizar o reconhecimento. O arquivo *MXL* é alocado em uma pasta reconhecida pelo software em desenvolvimento, o que facilita seu acesso.

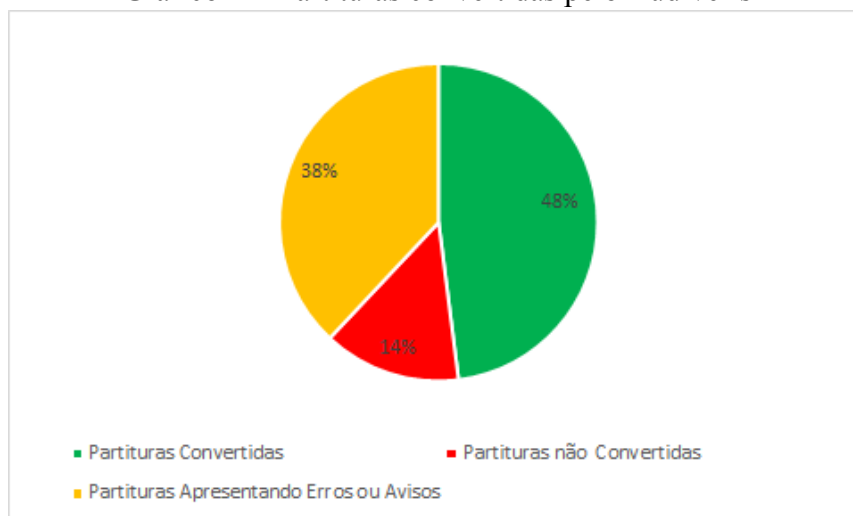
A **segunda fase** é realizada pelo *software*, que se preocupa com a organização dos dados do arquivo *MXL* e a criação do arquivo *JSON*. Por fim, na última fase, o software realiza a execução do arquivo *JSON* através do navegador do *Google Chrome*, com o auxílio da biblioteca *Retro-Audio-Js*.

A fim de verificar a eficácia do software em desenvolvimento, foram escolhidas 50 partituras, retiradas dos sites *IMSLP* e *Superpartituras* a fim de realizar os testes de conversão pelo *Audiveris* e organização de dados pelo *software*. Assim, através da utilização de uma interface, que permite a escolha do arquivo correspondente à música desejada, elaborada através da criação de um servidor baseado em *Node.js*, utilizando os conceitos de desenvolvimento Web para controlá-la juntamente com a biblioteca *Retro-Audio-Js*, foi possível interpretar e reproduzir os arquivos através da *API* do navegador, permitindo os testes de execução. Inicialmente, para o levantamento dos dados analisados, as partituras passaram pelos testes da primeira fase, preocupando-se em averiguar a eficácia com a qual o *Audiveris* realizava a tradução e conversão destas partituras para o formato de arquivo *MXL*, como podemos analisar no Gráfico 1. Os resultados completos dos testes estão disponíveis para download³.

A partir do Gráfico 1, podemos observar que, das 50 partituras utilizadas, apenas 14% delas, o que representa um todo de 7 partituras, não puderam ser convertidas pelo *Audiveris*. O programa apresentou um erro (*sheet music almost blank*) que indicava que a partitura utilizada não podia ser visualizada pelo *Audiveris*, o que impossibilitou a tradução e conversão do *PDF* para o arquivo *MXL*. Além disso, 38% das partituras, que representam 19 delas, apresentaram alguns avisos em relação à conversão do programa. Dentre eles, podemos citar: *Staff No Header Clef*, *No Clef Found* e *Error in performing*, que indicavam que o programa não reconhecia alguns elementos presente na partitura, seja pelo fato de serem diferentes, devido ao formato *PDF*, que muitas vezes vinham de *scanners* das partituras, seja pelo fato de que o *Audiveris* não ter tido contato com tal notação, sendo necessário um treinamento para que pudesse reconhecê-las. Mesmo assim, essas partituras tiveram o seu arquivo *MXL* elaborado.

³ Testes de verificação da eficácia do software: <https://cutt.ly/ZvurF3R>

Gráfico 1 - Partituras convertidas pelo Audiveris



Fonte: Elaboração dos autores, 2020

Outrossim, 48% das partituras, que representam 24 delas, foram convertidas pelo Audiveris para o padrão *MXL* com sucesso, sem apresentar erros ou avisos em sua conversão, o que provou a eficácia do software. Assim, estas partituras, bem como as 19 que apresentaram avisos, foram utilizadas para os testes da segunda fase, dando um total de 43 partituras utilizadas. Através das análises realizadas nos arquivos *MXL* nesta fase, foram definidas as *tags* (marcações do padrão *XML*) importantes para a elaboração do arquivo *JSON*. No Quadro 2 são apresentadas as principais *tags* presentes nos arquivos MusicXML compactados e utilizadas para a organização dos dados. Essas *tags* foram utilizadas para a elaboração dos arquivos *JSON* pelo *software*, iniciando a organização dos dados presentes dentro dos arquivos *MusicXML* compactados, como notas, tempo e compasso, respectivas as *Tags* apresentadas.

Quadro 2 - Principais *Tags* dos arquivos *MusicXML* utilizadas pelo *software*

<i>Tags</i>	Definição
<i>Measure</i>	Indica o compasso lido
<i>Times</i>	Indica o tempo da partitura
<i>Clef</i>	Indica a clave da partitura
<i>Tempo</i>	Indica as notas por minuto que devem ser executadas
<i>Duration</i>	Indica a duração da nota
<i>Pitch</i>	Representa uma nota
<i>Step</i>	Indica o nome da nota
<i>Rest</i>	Indica uma pausa
<i>Chord</i>	Representa um acorde
<i>Octave</i>	Indica a oitava da nota ou do acorde

Fonte: Elaboração dos autores, 2020

4. Considerações finais e trabalhos futuros

Este artigo teve como problemática o ensino de música em escolas brasileiras de educação básica, amparado pela Lei 11.769/2008. Compreende-se ao longo da pesquisa que poucas escolas têm capacidade de atender à lei, uma vez que o ensino desta matéria não é tratado com seriedade por grande parte da comunidade, além da dificuldade de encontrar profissionais e um ambiente próprio para ministrar aulas de música.

Segundo os resultados teóricos obtidos é possível observar o impacto positivo causado pela música na educação quando inserida em seus primeiros anos da ensino básico, como o desenvolvimento de sistema do neurodesenvolvimento, a exemplo do sistema motor, a memória e controle de atenção, e contribuir para a formação de uma visão crítica e criativa de mundo por parte do estudante. Além disso, também foi possível visualizar a função da informática no processo de ensino, tendo o papel de auxiliar para os educadores, além de contribuir para uma melhora no desempenho escolar do estudante, dar motivação ao aluno para estudar e promover o acesso a temas que não são tratados normalmente.

Quanto aos resultados práticos, foi possível reconhecer e converter notação musical em arquivos de partituras em formato *PDF* utilizando o aplicativo Audiveris, exportando-os em arquivos *MXL*. Além disso, foi possível organizá-los em arquivos *JSON* através do software e reproduzi-los a fim de testar sua execução através da biblioteca *Retro-Audio-Js*. Os testes permitiram visualizar a capacidade que software teria de atender aos critérios estabelecidos, tais como compasso, duração de notas e andamento da música, apresentando resultados satisfatórios em 50% das partituras testadas

A discussão levantada quanto a esses resultados gira em torno dos benefícios e contribuições que o software em desenvolvimento daria no ensino-aprendizagem de música em escolas brasileiras de educação básica. Na visão de autores como Miletto et al. (2004) e Pereira (2007), foi possível observar que softwares como os descritos nesta pesquisa poderiam auxiliar positivamente no ensino de música, se tornando uma ferramenta que pudesse auxiliar o professor no momento em que fosse ministrar sua aula. Dessa forma, podemos observar que, o fato do *software* executar de forma correta elementos da partitura, como compasso e duração de notas, em 50% dos testes, ele pode contribuir para o aprendizado do estudante, além de contribuir para que o aluno compreenda e seja compreendido nos mais diversos ambientes com os quais entre em contato através da música.

Por fim, este artigo propõe no campo do ensino da música inovadores por apresentarem uma nova alternativa à problemática descrita anteriormente através do desenvolvimento do *software*, que pode auxiliar não apenas no ensino da música, mas na educação de modo geral. Como sugestão é possível fazer a construção de uma parte física para o projeto, possibilitando assim uma maior interatividade entre o educando e o programa, além de realizar um tratamento de erros e permitir configurações das partituras utilizadas através de uma interface gráfica. Ademais, também é possível realizar a implementação do *software* em escolas, com o objetivo de calcular o impacto que o programa pode causar nas aulas de música.

Referências

- Brasil, L. D. B. Lei 9394/96–Lei de Diretrizes e Bases da Educação Nacional. Disponível http://www.planalto.gov.br/ccivil_03/leis/19394.htm. Acesso em: 02/2020, v. 31, 2015.
- Brasil Lei nº 11.769, DE 18 DE AGOSTO DE 2008. Altera a Lei nº 9.394, de 20 de dezembro de 1996, Lei de Diretrizes e Bases da Educação, para dispor sobre a obrigatoriedade do ensino da música na educação básica. http://www.planalto.gov.br/ccivil_03/_ato2007-2010/2008/lei/111769.htm
- Falcao, E. F.,; Nascimento, M. I. M. Educação Musical nas Escolas Brasileiras: Breve Retrospectiva Histórica, algumas Tendências e a obrigatoriedade no Currículos atuais. GT - História e Sociedade nos Campos - PR, [s. l.], 2010.
- Flores, L. V.. Conceitos e tecnologias para educação musical baseada na web. Dissertação (mestrado) - Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, 79 f., 2002.
- Freitas, L. I. B. Música e Autômato: uma receita de nomeação de acordes. 2012.
- Ilari, B. A música e o cérebro: algumas implicações do neurodesenvolvimento para a educação musical. Revista da ABEM, v. 11, n. 9, 2014.
- Kater, C. O que podemos esperar da educação musical em projetos de ação social. Revista da ABEM, v. 12, n. 10, 2014.
- Kleber, M. A ABEM e a educação musical no século XXI: contextualizando o significado da dinâmica em rede. Revista da ABEM, v. 20, n. 28, 2013.
- Lacerda, O. Compendio de Teoria Elementar da Música. [S. l.]: RICORDI, 1961.
- Miletto, E. M., et al. Educação musical auxiliada por computador: algumas considerações e experiências. RENOUE: revista novas tecnologias na educação [recurso eletrônico]. Porto Alegre, RS, 2004.
- Pereira, É. Criando setups de softwares para educação musical. Revista Digital Art, p. 1-14, 2007.
- Queiroz, L. R. S. Música na escola: aspectos históricos da legislação nacional e perspectivas atuais a partir da Lei 11.769/2008. Revista da ABEM, v. 20, n. 29, 2013.
- SWANWICK, K. A Basis for Music Education . London: Routledge, 1979.