



**UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
CURSO DE BACHARELADO EM ENGENHARIA DA
COMPUTAÇÃO**



Mário Raymundo Freire Wessen Neto

**Sistema Web para segmentação automática de áudio e
mapeamento de mensagens MIDI**

**RECIFE
2021**

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

Mário Raymundo Freire Wessen Neto

Sistema Web para segmentação automática de áudio e mapeamento de mensagens MIDI

Monografia apresentada ao Centro de Informática (CIN) da Universidade Federal de Pernambuco (UFPE), como requisito parcial para conclusão do Curso de Engenharia da Computação, orientada pelo professor Filipe Carlos de Albuquerque Calegario.

RECIFE

2021

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

Mário Raymundo Freire Wessen Neto

Sistema Web para segmentação automática de áudio e mapeamento de mensagens MIDI

Monografia submetida ao corpo docente da Universidade Federal de Pernambuco, defendida e aprovada em 03 de Agosto de 2021.

Banca Examinadora:

Filipe Carlos de Albuquerque Calegario.

Doutor

Orientador

Giordano Ribeiro Eulalio Cabral

Doutor

Examinador

AGRADECIMENTOS

Agradeço primeiramente à minha Mãe e Irmã que me ajudaram com todo e qualquer desafio que eu tenha enfrentado.

Ao meu Pai que mesmo tendo partido tão cedo, me deixou tanto conhecimento, amor e coragem.

À minha esposa por todo apoio e parceria em todos os momentos.

E por fim a todos os professores que me ajudaram nessa jornada.

“Se você admite que não há esperança, então você garante que não há esperança. Se você admite que existe um instinto pela liberdade, que existem oportunidades de mudar as coisas, então há chance de contribuir para um mundo melhor.”

Noam Chomsky.

RESUMO

Com a evolução tecnológica, vieram novas formas de produção de conteúdo audiovisual, demandando a necessidade da agilidade na produção desse conteúdo, tal como a distribuição e construção do processo criativo. Olhando especificamente a indústria musical, existem diversas ferramentas de criação de conteúdo voltadas para a experimentação com samples, como a Superpads, Sequence64, iZotope RX8 ou Moises AI, todas trazendo excelentes resultados. Contudo, essas ferramentas possuem algumas limitações como a não realização da segmentação automática, a possibilidade de mapeamento de mensagens MIDI e a impossibilidade de upload de uma música desejada pelo usuário. Por isso, propomos uma ferramenta que traz recursos relevantes como uma segmentação automática de músicas escolhidas pelo usuário, fornecendo também a possibilidade de interação utilizando dispositivos que implementem o protocolo MIDI.

Palavras-chave: Mensagens MIDI, Segmentação de áudio, Produção musical, Sistema Web.

ABSTRACT

With technological evolution, new forms of audiovisual content production have appeared, demanding the need for agility in the production of this content, such as the distribution and construction of the creative process. Looking specifically at the music industry, there are several content creation tools aimed at experimenting with samples, such as Superpads, Sequence64, iZotope RX8 or Moises AI, all bringing excellent results. However, these tools have some limitations, such as not performing automatic segmentation, the possibility of mapping MIDI messages and the impossibility of uploading a song desired by the user. Therefore, we propose a tool that brings relevant features such as an automatic segmentation of songs chosen by the user, also providing the possibility of interaction using devices that implement the MIDI protocol.

Keywords: MIDI Messages, Audio Segmentation, Music Production, Web System.

LISTA DE FIGURAS

Figura 1 - Etapas da produção musical.....	18
Figura 2 - MCP60 da Akai.....	21
Figura 3 - MCP Lile II da Akai.....	22
Figura 4 - Tela da aplicação Super Pad.....	24
Figura 5 - Interface Sequence64.....	25
Figura 6 - Interface da aplicação IZOTOPE RX8.....	26
Figura 7 - Interface da aplicação MOISES IA.....	27
Figura 8 - Instrumentos identificados pela Spleeter.....	30
Figura 9 - Página da aplicação.....	34
Figura 10 - Virtual MIDI Piano Keyboard.....	34
Figura 11 - Fluxo da aplicação.....	35
Figura 12 - API Insomnia.....	36
Figura 13 - Áudio sem encaixe de repetição.....	39
Figura 14 - Áudio com ruído.....	39

LISTA DE GRÁFICOS

Gráfico 1 - Análise da separação de fonte.	38
Gráfico 2 - Análise da geração de sample corretos.....	40
Gráfico 3 - Análise de assertividade entre as músicas.....	41
Gráfico 4 - Análise de assertividade por stems.	42

LISTA DE TABELAS

Tabela 1 - Comparação de funcionalidades entres aplicações de produção musical.	28
Tabela 2 - Lista de Música utilizada no experimento.....	37

TABELA DE SIGLAS

Sigla	Significado	Página
MIDI	Musical Instrument Digital Interface	15
MPC	Music Production Controller	21
DAW	Digital Audio Workstation	22
API	Application Programming Interface	30
FFMPEG	Fast Forward MPEG	31

SUMÁRIO

1.	INTRODUÇÃO	14
1.1.	MOTIVAÇÃO	14
1.2.	OBJETIVOS	15
2.	CONCEITOS BÁSICOS	16
2.1.	MÚSICA.....	16
2.1.1.	Definição	16
2.1.2.	Produção Musical.....	17
2.2.	MIDI	18
2.3.	SAMPLE	20
2.3.1.	MPC	21
3.	APLICAÇÕES RELACIONADAS	23
3.1.	SUPER PAD.....	23
3.2.	SEQUENCE64	24
3.3.	IZOTOPE RX8.....	25
3.4.	MOISES IA	26
3.5.	COMPARAÇÃO.....	27
4	ALGORITMOS UTILIZADOS.....	29
4.1	SPLEETER.....	29
4.2	LIBROSA	30
4.3	FFMPEG.....	31
4.4	WEB MIDI API	32
5	SOLUÇÃO PROPOSTA.....	33
5.1	RECURSOS	33
5.2	TECNOLOGIAS ESCOLHIDAS.....	33
5.3	ARQUITETURAS.....	33
5.3.1	Página Web.....	34
5.3.2	API	35
6	EXPERIMENTO E ANÁLISE	37
6.1	ANÁLISE DA SEPARAÇÃO DE FONTE	38
7	CONCLUSÃO E TRABALHOS FUTUROS	43
7.1	CONTRIBUIÇÕES.....	43

7.2	TRABALHOS FUTUROS.....	43
8	BIBLIOGRAFIA	45

1. INTRODUÇÃO

Alguns autores como BRÉSCIA (2003) defendem que a música está presente no nosso meio desde a formação da humanidade, sendo utilizada pelas primeiras civilizações em diversos momentos da sociedade. Hoje em dia, vivemos em uma era onde a evolução tecnológica mudou totalmente a relação das pessoas com a música, tanto na forma de consumo quanto no âmbito da produção musical.

Qualquer indivíduo que tenha um mínimo contato com a música, vai estar diretamente envolvido com tecnologia, indo desde a nomenclatura como *play* e *stop* que estão presentes em aparelhos de som a até o uso de processadores digitais de som (GOHN, 2001).

Com o passar do tempo, a tecnologia permitiu o desenvolvimento de diversas plataformas de *streaming*¹ de áudio e com isso a flexibilidade de poder escutar música em qualquer dispositivo com acesso a essas plataformas. Segundo Scatamburlo e Campos (2020): “O consumo de *streaming* de áudio alcança 30% do total da população de internet no Brasil (aproximadamente 120MM), em torno de 36.6 Milhões de usuários únicos multiplataforma (*Desktop + Mobile*)”.

O aumento desse consumo trouxe uma realidade na qual diariamente somos bombardeados com uma diversidade de músicas. Para alcançar a criação e desenvolvimento de toda essa diversidade musical, a forma de fazer música também mudou com a utilização de programas de produção musical. “Todo o processo se tornou mais simples, surgiram novas possibilidades (antes impossíveis), ficou mais fácil até mesmo para se produzir música de qualidade em casa.” (SPARFLEX, 2020).

1.1. MOTIVAÇÃO

Tendo em vista o cenário atual, as ferramentas de produção musical têm por finalidade trazer novas possibilidades no processo de criação das construções musicais, otimizando o trabalho tanto do produtor musical experiente, quanto do

¹ Tecnologia de transmissão de dados pela internet que permite o acesso a conteúdo de qualquer dispositivo com conexão e em tempo real sem a necessidade de download.

iniciante. Tornando assim, a utilização desses softwares de produção, uma forma de unificação entre a música e a ciência. (LIANG, 2021).

Contudo, é comum termos a sensação de excesso de parametrização como visto por ASSIS (2016):

Como ferramenta de suporte à criação, o computador transformou a complexa arte do estúdio analógico em selecionar sonoridades com cliques. Essas facilidades, porém, não vêm sem um preço. Se antes os produtores e músicos tinham que se esforçar e criar soluções dentro do estúdio, hoje tudo vem pré-programado, parametrado para funcionar – o que significa que as soluções já estão prontas, sem tanto espaço para a invenção.

Analisando ferramentas de produção musical como Superpad, Sequence64, iZotope RX8 e Moises IA, vemos que de fato existem limitações para uma maior liberdade expressiva.

1.2. OBJETIVOS

Dado o contexto, propomos uma solução para trazer um maior dinamismo no processo criativo, fornecendo a possibilidade de o músico efetuar um upload das músicas desejadas, um sistema que realiza uma separação por faixa de instrumentos e ser gerado samples desse áudio de forma automática. Além disso, a ferramenta também possibilita que o músico interaja com os samples por mensagens *Musical Instrument Digital Interface* (MIDI).

2. CONCEITOS BÁSICOS

O desenvolvimento de trabalhos na área de computação musical é interdisciplinar, que relacionam conceitos não tecnológicos como música e produção musical à utilização de aprendizagem de máquina. Nesta seção, serão apresentados conceitos utilizados neste projeto.

2.1. MÚSICA

Com o avanço da tecnologia, a facilidade de criação e propagação da música se deu de uma forma nunca imaginada. Cada vez mais se tem visto a criação de novos estilos musicais, com uma gama de efeitos sonoros e tendo como uma das principais ferramentas a computação. Porém, para entender este processo, precisamos de definições do que é música e como é esse processo de produção musical.

2.1.1. Definição

Alguns autores como Clifton (1983, apud FREITAS, 1997, p. 24) define a música da seguinte forma:

Música é um arranjo ordenado de sons e silêncios cujo sentido é representativo ao invés de denotativo. (...) música é a realização da possibilidade de qualquer som apresentar a algum ser humano um sentido que ele experimenta em seu corpo.

Já segundo Moraes (1991), a música é uma maneira peculiar de sentir e pensar, que propõe novas maneiras de se fazê-lo.

É por isso que se pode perceber a música não apenas naquilo que o hábito convencionou chamar de música, mas – e sobretudo – onde existe (...) a invenção de linguagens: formas de ver, representar, transfigurar e de transformar o mundo.

Percebemos que esses são conceitos abstratos e que podem dar espaço para reflexões e outras formas de definição. Já no lado da computação, conseguimos ser mais específicos, pois trabalhamos com arquivos de áudio que são formados por uma sequência de bits, podendo ou não conter algum sentido, onde para essa área musical são analisadas as suas métricas e formatos.

Para situar nosso trabalho, é importante entender também o processo de produção musical.

2.1.2. Produção Musical

A produção musical possui uma definição bem abrangente, mas tem como sua base o processo de criação e refinamento de uma música gravada até estar pronta para veiculação. Ela pode se referir a todo processo de criação de uma música, desde a escrita e composição, passando pela fase de gravação e design de som até a mixagem e masterização (GABRIEL, 2020).

Uma característica importante da produção musical moderna é a utilização de ferramentas digitais, tornando a realização desse processo mais fácil e acessível. Essa facilidade permitiu a criação do *Home Studios*, que é um estúdio montado em casa composto por um bom computador, um software de produção musical e um equipamento de som, montando assim uma estrutura caseira.

O processo de produção musical pode ser dividido em cinco etapas (Figura 1) e em cada uma é utilizado um tipo de software diferente que faz parte desse contexto (CHAGAS,2020):

- Pré produção: Início do processo em que é realizada uma gravação guia para testar o andamento da música, tonalidade, melodias, ritmos, alcance da voz e todos os arranjos musicais.
- Gravação: Fase em que a gravação da música é feita e onde vários takes de pequenos trechos ou tracks inteiras da música.
- Edição: Etapa na qual é feita a edição de tudo o que foi gravado. São realizadas ações de manipulação dos takes em busca de montar a track final. Também são realizados ajustes para limpeza de ruído que possam ser captados durante a gravação e realizar ajustes na voz e na melodia dos equipamentos.
- Mixagem: Neste ponto, são realizados ajustes nos sons da música em busca do melhor equilíbrio. Processos como equalização, ajuste de volume, adição de efeitos e alterações podem ser realizados.
- Masterização: É a fase final, na qual é realizada a transformação da mídia de gravação em mídia de consumo.

Figura 1- Etapas da produção musical.



Fonte: Autoria própria.

Este trabalho tem como foco auxiliar na produção tanto na fase de Edição permitindo a manipulação do que foi gravado, quanto na de Mixagem ao desejar trabalhar com diversos sons sendo ou não de instrumentos musicais, facilitando assim, o processo de produção musical.

2.2. MIDI

O protocolo MIDI surge de uma necessidade comum a diversos outros tipos de protocolos, que é a padronização entre a comunicação de instrumentos musicais com os computadores.

Segundo Silva (2007), na década de 80, dois fabricantes (Dave Smith – Sequential in the USA e Ikutaro Kakehashi – Roland Corporation) iniciaram um trabalho em conjunto para desenvolvimento do padrão do MIDI, e posteriormente em 1983 outras empresas como Yamaha, Oberheim, Arp, Moog, também optaram por utilizá-lo.

Uma definição muito concisa desse padrão é:

MIDI (*Musical Instrument Digital Interface*) especifica um esquema de interconexão física e um método de comunicação lógica que possibilitam o controle de instrumentos musicais em tempo real. Especifica também uma sintaxe para a codificação de informações de performance composta por uma sequência de mensagens e dados de comunicação em formato binário. (SILVIA, 2017, p12).

Em seu processo de transmissão, o protocolo MIDI utiliza o sistema serial, na qual a troca de informações é realizada de forma sequencial. Durante esse processo, é feita a transferência bit a bit, onde a cada 10 bits uma palavra é formada, servindo para descrever um evento da performance musical (MARTINS, 2016).

O envio desses eventos é classificado como mensagens MIDI. Essas mensagens não possuem áudio, mas contém informações como: notação musical, altura (pitch), sinais de controle para parâmetros como vibrato, volume e panorama, pistas e sinais para definir o tempo.

Pelo fato dessas mensagens serem orientadas a eventos, elas foram projetadas para serem utilizadas em instrumentos baseados em teclado, pois ao realizar a ação de apertar uma tecla e soltar, mensagens são enviadas (CASTRO, 2016).

As mensagens MIDI possuem vários tipos e alguns deles são:

- *Note On*: Essa mensagem é disparada quando a tecla é pressionada contendo a informação de *note* que indica qual tecla foi pressionada e a informação de *velocity* indicando a força que ela foi pressionada podendo variar de 0 a 127.
- *Aftersustain*: Quando a tecla permanece pressionada, essa mensagem é enviada informando a pressão que essa tecla está recebendo e que pode variar de 0 a 127. Esta informação pode influenciar no volume, timbre e vibrato da nota.
- *Note Off*: No momento em que a tecla é solta a mensagem é enviada também contendo a informação de qual tecla foi solta (*note*) e a sua velocidade (*velocity*).

Uma das outras características do MIDI é a possibilidade de enviar mensagens sobre até 16 canais independentes, ou seja, 16 instrumentos ou sons simultâneos.

A criação desse protocolo foi fundamental para a evolução da produção, composição e finalização dos projetos musicais. Tornando possível a experimentação de sons, facilidade de interconexão entre os equipamentos, automação e armazenamento dessas informações. Uma outra grande vantagem foi o ganho de tempo de uma produção, causando a diminuição do custo total de um projeto musical (SILVA, 2017).

2.3. SAMPLE

Assim como a tecnologia MIDI, o *sample* revolucionou a maneira como as pessoas faziam música antigamente. Sendo não só essencial à música eletrônica (gênero que surgiu e cresceu com o avanço da computação), o *sample* é usado em muitos trabalhos de produção musical independente do gênero, além de ser uma ferramenta versátil e de fácil utilização em qualquer estúdio (COLODRO, 2017).

Sample é uma palavra oriunda do inglês que significa “amostra”, que também é utilizada em outras esferas além do musical. No ponto de vista da produção musical, o *sample* musical pode ser definido como uma amostra ou trecho sonoro (TAMMARO, 2020).

Esses sons podem ser caracterizados com trechos (ou partes inteiras) de músicas já existentes, instrumentos de forma isolada ou até sons do cotidiano, como o trem passando nos trilhos, uma buzina ou a chuva no telhado. Esse conceito surgiu na década de 40, onde músicas foram feitas através de pequenas amostras de sons já gravados e era conhecido como *Musique Concrète*.² (ROCHA, 2018).

A Música Concreta consistia em reunir diferentes amostras de som formando uma nova composição. As colagens dos conteúdos sonoros reuniam não apenas sons de instrumentos, mas possuía caráter experimental utilizando sons de trens e barulhos mecânicos gravados previamente. (COLODRO,2017).

Nos anos 70 e 80, o *sample* cresceu no meio musical devido ao hip-hop e ao rap, tendo a sua consolidação na década de 90. Os DJs utilizavam recortes das gravações de vinis e tocavam essas amostras em festas, contribuindo assim para esse crescimento.

Com o passar do tempo mais dois termos surgiram devido ao *sample*:

- *Sampling*: é a ação de produzir ou coletar amostras de músicas.
- *Sampler*: Na área da música, ele pode ser definido como um hardware ou software específico, utilizado para armazenar samples e manipulá-los de forma individual ou conjunta.

O Metrollon foi o primeiro *sampler* a se tornar conhecido, sendo fabricado na Inglaterra na década de 60. Ele era um teclado, onde cada tecla possuía uma fita

² Termo francês que traduzido para português é chamado de Música Concreta.

magnética que tinha capacidade de armazenar sons com duração de até 8 segundos. (CRAB, 2014).

2.3.1. MPC

O *Music Production Center* (MPC) surgiu com o objetivo de trazer facilidade ao utilizar esses recursos. O primeiro modelo MPC60 (MIDI Production Center) da Akai (Figura 2), foi lançado em 1988 e era a junção de uma bateria eletrônica com um *sampler* que também possuía funções de edição.

Com interface diferente dos equipamentos anteriores, o MPC possuía botões sensíveis à pressão semelhante ao tocar um teclado, onde cada botão pode armazenar um sample de até 13 segundos. Além disso, o MPC tinha a capacidade de armazenar até 99 *tracks*, permitindo que os usuários adicionassem sons de algum instrumento em *loop*, realizando uma criação de forma contínua. (EITC, 2020).

Depois que essas habilidades foram dominadas, os usuários descobriram que o MPC tornou a produção mais fácil e eficiente. Oferecer gravação e edição em tempo real com alta qualidade de som permite efetivamente ao usuário produzir música ao vivo, conforme ela é criada. O MPC simplificou ou eliminou muitas das tarefas ocupadas e demoradas associadas à produção musical, deixando aos músicos bastante tempo para criar e se concentrar em seu trabalho, em vez de suar por uma caixa de ressonância enorme para obter o mesmo resultado. (ACIMAN,2018).

Figura 2 - MPC60 da Akai.



Fonte: Vintage Synth.³

³ Disponível em: <http://www.vintagesynth.com/akai/mpc60.php>. Acesso em: julho/2021.

Do primeiro até os atuais, os MPCs continuam tendo como principal recurso a sua capacidade de armazenamento e reprodução de samples gravados em uma sequência gravada pelo usuário (MARC, 2019). (Figura 3).

Figura 3 - MPC Live II da Akai.



Fonte: Akai Professional.⁴

Uma das principais evoluções desse equipamento, é a sua integração com computadores. Os MPCs atuais trazem a possibilidade de trabalhar integrado com *Digital Audio Workstation* (DAWs), que são softwares usados para compor, produzir, gravar, mixar e editar áudio (BRANDÃO, 2017).

Outra forma de poder utilizá-lo é como software, pois é possível emular o MPC e utilizar suas funcionalidades em um programa.

⁴ Disponível em: <https://www.akaipro.com/mpcliveii>. Acesso em: julho/2021.

3. APLICAÇÕES RELACIONADAS

Nesta seção, apresentamos algumas aplicações relacionadas com uma descrição inicial seguida por uma análise comparativa.

As aplicações do meio de produção musical são as mais diversas possíveis, porém para a análise olharemos para algumas que auxiliam no processo de criação de samples, entregando recursos semelhantes ao proposto por esse trabalho, sendo eles a separação de fonte, entrega de *beat* do segmento, realização de forma automática dessas segmentações, segmentar um áudio escolhido pelo usuário e leitura de mensagens MIDI.

3.1. SUPER PAD

O Super Pad é um emulador de controlador de produção musical (MPC), que se tornou um equipamento essencial para os DJs e foi criado pelo engenheiro curitibano Arthur Motelevicz. Lançada há quatro anos, a ferramenta já atingiu os 50 milhões de downloads e viralizou na China, que é um dos maiores mercados do mundo. (ADRION, 2021).

Como pode ser visto na Figura 4, a aplicação tem como interface algo semelhante a um MPC. Nela é possível selecionar músicas prefixadas, para então ter acesso aos samples já cadastrados para essa melodia. A aplicação nos fornece recursos de gravação dos remix e é muito simples de ser utilizada.

Figura 4 - Tela da aplicação Super Pad.



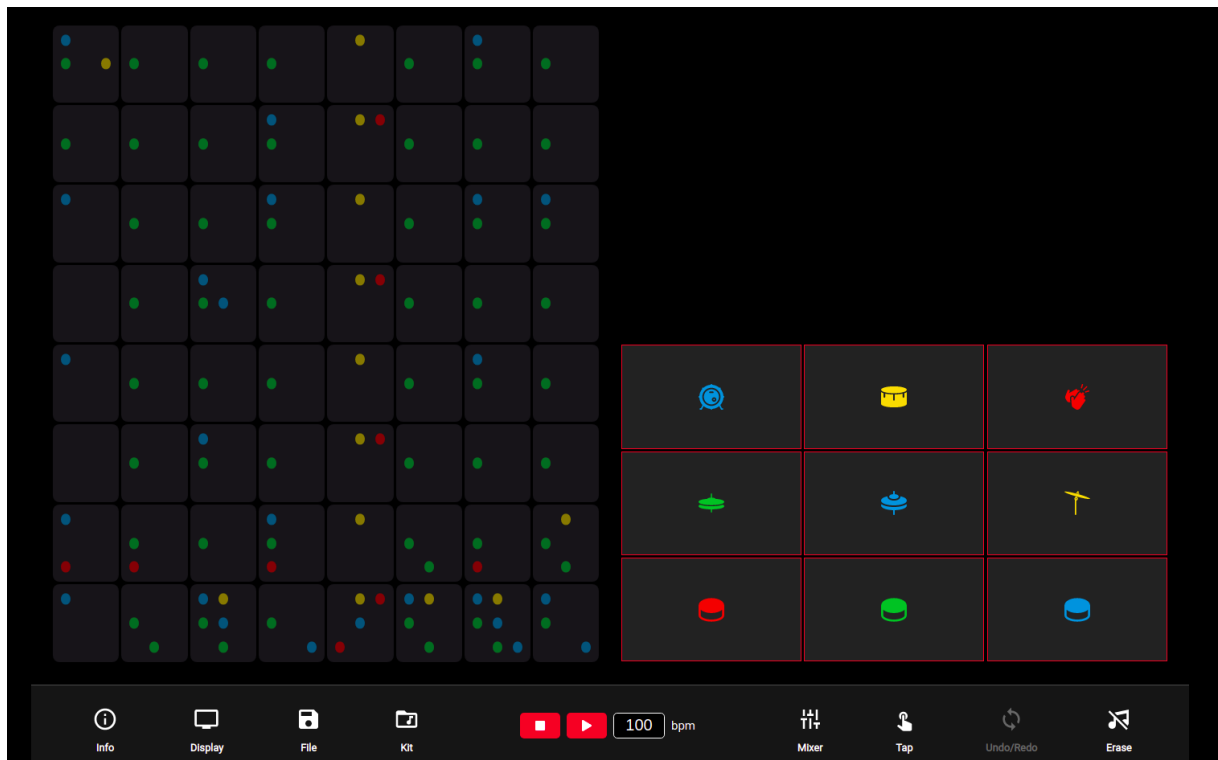
Fonte: Super Pads.⁵

3.2. SEQUENCE64

Pela definição do seu criador Nick Carbone, o Sequencer64 é uma aplicação web que permite sequenciar rapidamente um padrão de 64 passos para um *sampler* de 9 sons. Porém vemos uma ferramenta bem mais interessante onde através de diversos sons e opções de remix cadastrados, é possível gerar batidas bem interessantes, realizar modificações nesses elementos sonoros e no final baixar os resultados. Vemos uma interface bastante amigável e de simples utilização, como pode ser visto na Figura 5.

⁵ Disponível em: <http://superpadsapp.com/>. Acesso em: junho/2021.

Figura 5 - Interface Sequence64



Fonte: Sequence64⁶

3.3. IZOTOPE RX8

Entre as diversas ferramentas desenvolvidas pela empresa IZotope, está a RX8, não será possível descrever neste trabalho todos os recursos dessa ferramenta, pois são vastos, porém cabe mencionarmos a possibilidade de separação de fonte com 5 *stems*. Esse processo é feito com uma incrível qualidade de áudio, sendo possível diversas modificações nesses segmentos, além de todos os demais recursos como a possibilidade da criação de uma cadeia de módulos, onde é possível realizar processamento de diversos áudios simultaneamente. Na Figura 6 vemos a interface do RX8 com o recurso de separação de fontes.

⁶ Disponível em: <https://www.sequencer64.com/sequencer/session>. Acesso em: junho/2021.

Figura 6 - Interface da aplicação IZOTOPE RX8



Fonte: IZOTOPE.⁷

3.4. MOISES IA

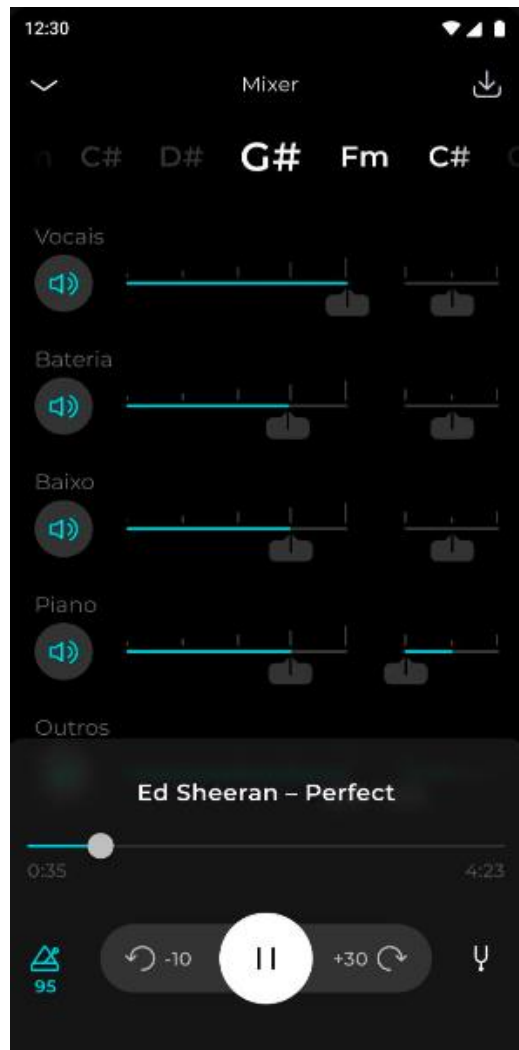
A plataforma Moises foi criada pelo desenvolvedor brasileiro Geraldo Ramos, sendo capaz de separar as trilhas de instrumentos e vocais de um arquivo de áudio através da inteligência artificial. Além disso, ainda utilizando a IA, é possível masterizar faixas de música, gerando trilhas com qualidade de estúdio (RAMOS, 2020). A aplicação permite escolher músicas no formato mp3 ou de links do YouTube.

Depois que a música é escolhida, ela é enviada para um servidor na nuvem no qual o processo de separação é realizado e posteriormente é enviada de volta ao usuário onde ele pode ouvir e realizar o download (ALMEIDA, 2019).

O grande diferencial da ferramenta é o seu recurso de realizar a separação de fontes utilizando a biblioteca Spleeter, criada pela Deezer. Além disso, recursos como a identificação de acordes, metrônimos, a possibilidade de carregar uma música a sua escolha e possibilidade de download do conteúdo gerado. A interface é bastante simples de utilizar e pode ser vista a Figura 7.

⁷ Disponível em: <https://www.izotope.com/en/shop/rx-8-standard.html>. Acesso em: junho/2021.

Figura 7 - Interface da aplicação MOISES IA



Fonte: MOISES.⁸

3.5. COMPARAÇÃO

Olhando para o Super Pads, vemos a ausência de alguns recursos. Um dos pontos críticos é a impossibilidade de escolher qualquer música, pois como os *samples* são fixos e cadastrados para aquela playlist preexistente, o usuário não tem essa liberdade de fazer um upload ou apontar para um link e utilizar os *samples* da música escolhida.

O sequence64 nos entrega *samples* prontos de alguns instrumentos. Essa forma é interessante, mas é um empecilho se o usuário quiser um elemento de som externo, pois não é possível carregar uma música.

⁸ Disponível em: <https://moises.ai/pt-br/>. Acesso em: junho/2021.

Ao testar o IZotope RX8, vimos a ferramenta mais completa entre as analisadas. A separação de áudio realizada por ela, mostra um nível de detalhe nos instrumentos até melhor do que será visto em seguida na aplicação Moises. Além de diversas formas de processamento de áudio que a ferramenta nos traz, um ponto de atenção é que ela tem como propósito o processo de limpeza de áudio. Sendo assim, então recursos como a geração de beat são possíveis por conter a possibilidade de recorte de elementos, porém não temos uma detecção automática.

Na aplicação da Moises, vemos que utiliza MP3 para o processamento, enquanto a RX8 pode-se utilizar de wav. Isso mostra que já pelo formato, temos uma perda de informação. Além desse ponto, vemos que não é possível recortar ou gerar samples nos segmentos gerados.

Nenhuma das aplicações vistas, tem interação com MIDI, o que seria uma forma bem interessante de inserção e criação de novos elementos musicais junto aos seus recursos de processamento.

Para termos uma visão mais clara desses pontos, é possível ver na Tabela 1 ocorrência desses recursos ou não:

Tabela 1 - Comparação de funcionalidades entres aplicações de produção musical.

	SUPER PAD	SEQUENCE 64	IZOTOPE RX8	MOISES IA	APLICAÇÃO PROPOSTA
Separação de Fonte	Não	Não	Sim	Sim	Sim
Gerar Beat	Sim	Sim	Sim	Não	Sim
Segmentação automática	Não	Não	Sim	Sim	Sim
Inserção de áudio	Não	Não	Sim	Sim	Sim
Mapeamento MIDI	Não	Não	Não	Não	Sim

Fonte: Autoria própria.

4 ALGORITMOS UTILIZADOS

Nesta seção, iremos apresentar algumas bibliotecas que foram utilizadas no desenvolvimento deste trabalho, onde os conceitos apresentados anteriormente são bases para o perfeito entendimento do funcionamento desses recursos.

4.1 SPLEETER

Partindo da problemática bastante conhecida na Computação Musical que é a dificuldade de separar o áudio dos instrumentos musicais de uma música em trilhas independentes, ferramentas que realizam essa separação começaram a ser desenvolvidas, dentre elas a Spleeter.

Como definido pelos seus desenvolvedores Hennequin et al. (2020), Spleeter é uma biblioteca de separação de áudio desenvolvida pela Deezer com modelos pré-treinados escritos em Python. Para a criação desses modelos, ela utiliza a biblioteca de código aberto para *machine learning* chamada Tensorflow, que torna mais fácil treinar o modelo de separação de origem. Como resultado, foram obtidos modelos de última geração já treinados para realizar vários tipos de separação.

Os modelos treinados conseguem realizar a separação nas seguintes partes:

- 2 *stems*: vocais e sons de acompanhamento.
- 4 *stems*: vocais, bateria, baixo e outros sons.
- 5 *stems*: vocais, bateria, baixo, pianos e outros sons.

Trabalhar com trilhas independentes traz a possibilidade de realizar várias ações como: remix, upmixing, escuta ativa, propósitos educacionais, mas também pré-processamento para outras tarefas, como transcrição. (HENNEQUIN et al, 2020).

Figura 8 - Instrumentos identificados pela Spleeter.



Fonte: Deezer.⁹

Para utilizá-la, basta instalar a *Application Programming Interface* (API) Python chamada *Separator* e realizar a configuração em seu programa. A Deezer também disponibiliza uma documentação completa informando como realizar a instalação e o modo de uso.

4.2 LIBROSA

A Librosa é uma biblioteca que tem como foco realizar o processamento de áudios e conteúdos musicais. Em um alto nível, a biblioteca fornece implementações de uma variedade de funções comuns usadas em todo o campo de recuperação de informação musical. (MCFEE et al, 2015).

O foco dessa biblioteca é mais voltado a retirar características do áudio que se relacionam com análises de músicas, contudo dentro das comunidades de análise de áudio utilizando Python esta biblioteca é amplamente usada para os mais variados propósitos. Esta biblioteca também tem o código aberto e sua base de cálculos funciona usando SciPy e NumPy como base. (KNEIPP, 2019, P.24).

Suas funções tentam estabelecer a normalização de variáveis, valores padrão e parâmetros para esses recursos sonoros. (KNEIPP, 2019).

⁹ Disponível em: <https://deezer.io/releasing-spleeter-deezer-r-d-source-separation-engine-2b88985e797e>. Acesso em: junho / 2021.

Dentre elas temos a *beat time* que é o recurso que possibilita a extração do tempo do beat de um áudio que é passado como parâmetro para essa biblioteca, tornando possível assim sabermos o tempo de ocorrência.

4.3 FFMPEG

O *Fast Forward MPEG* (FFMPEG) é o *framework* multimídia que é capaz de realizar as seguintes ações: decodificar, codificar, transcodificar, ser utilizado como mux e demux, além de transmitir, filtrar e reproduzir praticamente qualquer coisa que humanos e máquinas tenham criado. Essa ferramenta tem suporte desde os formatos mais antigos até os atuais. (FFMPEG, 2010).

Não importando se eles foram projetados por algum comitê de padrões, uma comunidade ou empresa. Outra característica importante é que ele também é altamente portátil: o FFMPEG compila, executa e passa em nossa infraestrutura de teste FATE em Linux, Mac OS X, Microsoft Windows, BSDs, Solaris etc. (ABOUT..., 2010).

O FFMPEG traz diversos recursos, além de também estar disponível em forma de biblioteca para que possa ser utilizado pelos desenvolvedores. Sendo elas:

- *Libavutil*: Essa biblioteca tem como foco simplificar a programação, incluindo geradores de números aleatórios, estruturas de dados, rotinas matemáticas, utilitários de multimídia centrais e muito mais
- *Libavcodec*: Possui decodificadores e codificadores para codecs de áudio / vídeo.
- *Libavformat*: Permite realizar a ação de mux e demux para formatos de contêineres de multimídia.
- *Libavdevice*: Ela possui dispositivos de entrada e saída que realizam a captura e renderização em muitas estruturas de software de entrada / saída de multimídia comuns, incluindo Video4Linux, Video4Linux2, Vfw e ALSA.
- *Libavfilter*: Possui dispositivos de filtros de mídia.
- *Libswscale*: Essa biblioteca executa operações de escala de imagem altamente otimizada e também realiza a conversão de espaço de cores em formato de pixel.

- *Libswresample*: Ela tem a responsabilidade de realizar a otimização do áudio, trazendo uma nova amostra, além de realizar novamente a masterização e por fim e converter em outros formatos.

4.4 WEB MIDI API

A Web MIDI API tem como objetivo permitir o acesso direto a dispositivos que respondem a mensagens MIDI como sintetizadores externos, sistemas de iluminação ou até mesmo os sintetizadores de software que são integrados a muitos sistemas operacionais comuns. (API..., 2015).

Essa API também é explicitamente projetada para permitir que aplicações web possam estabelecer comunicação com hardware externos, permitindo assim receber e entender todos os comandos.

A API Web MIDI também deve ser usada em conjunto com outras APIs e elementos da plataforma da web, principalmente a API de áudio da web. Esta API também se destina a ser familiar aos usuários de APIs MIDI em outros sistemas, como CoreMIDI da Apple e API MIDI do Windows da Microsoft (API..., 2015, tradução nossa).

5 SOLUÇÃO PROPOSTA

Após ter a visão dos recursos entregues pelas ferramentas já existentes, vimos que a possibilidade de escolher as suas próprias músicas e aplicar todo o processo de segmentação seria um diferencial interessante, além disso sentimos a necessidade de se poder experimentar tocando esses elementos sonoros.

5.1 RECURSOS

A aplicação conta com a possibilidade de realizar um *upload* de um arquivo de áudio, e então realizar a sua segmentação, entregando a separação por fontes e através da identificação dos *beats* da música, conseguimos gerar *samples* referentes a cada trilha identificada. Além disso, o músico tem a possibilidade de utilizar algum instrumento ou simulador de mensagens MIDI, atrelando a nota correspondente ao *sample* desejado e assim interagir tocando os trechos relacionados.

5.2 TECNOLOGIAS ESCOLHIDAS

Além de um domínio das tecnologias *javascript* como Node.js e React, vemos que há com essas ferramentas um processo de fácil entendimento e escalabilidade, em conjunto para o processamento de áudio na API. Optamos por utilizar Python pela facilidade de bibliotecas já existentes e o forte desempenho da linguagem para processamento de arquivos. Escolhemos o banco MongoDB pela facilidade de estruturação dos dados e o alto desempenho na resposta.

5.3 ARQUITETURAS

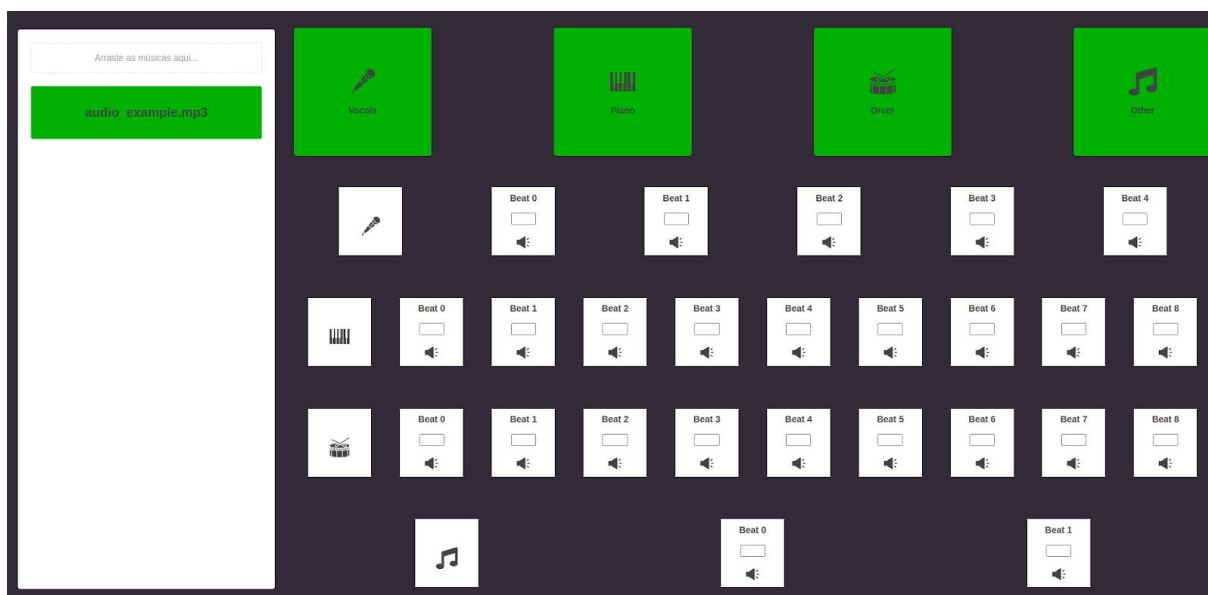
A arquitetura da ferramenta conta com uma estrutura de *FrontEnd* e *BackEnd*, onde esses dois módulos conversam através de requisições Rest. Na Figura 11 conseguimos ter uma visão dessa arquitetura. A escolha dessa arquitetura se deu além do domínio das tecnologias escolhidas, pelo fato de ser uma estrutura escalável e de fácil análise.

5.3.1 Página Web

A página web foi desenvolvida em React e conta com dois grandes módulos, sendo o primeiro composto pela lista e carregamento de arquivos, e o segundo pela estrutura de leitura das mensagens MIDI e a reprodução desses áudios.

Podemos ver na Figura 9 a estrutura da página, onde temos a esquerda um campo para upload e a direita os segmentos a serem escolhidos. Ao selecionar a lista de samples, é apresentado um campo de entrada de texto onde será possível inserir a nota desejada para a leitura da mensagem MIDI.

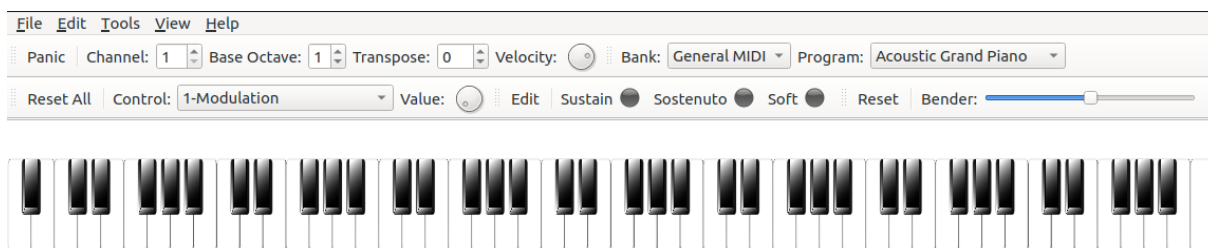
Figura 9 - Página da aplicação.



Fonte: Autoria própria.

Para a simulação das mensagens MIDI, utilizamos a aplicação Virtual Midi Piano Keyboard (Figura 10).

Figura 10 - Virtual MIDI Piano Keyboard.



Fonte: Autoria própria.

5.3.2 API

A API é onde está contida a maior complexidade da aplicação. Nela, temos o primeiro módulo que é o responsável por receber o *request* com o *upload* e listagem das músicas já inseridas.

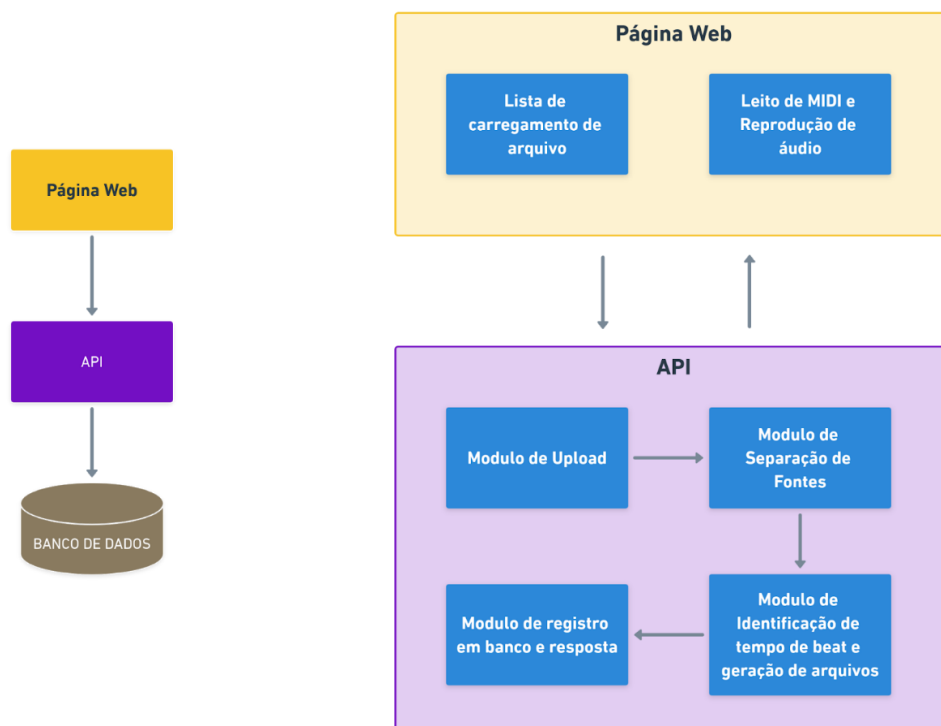
Para a etapa de *upload*, temos um módulo de separação de fontes. Nesse código é chamado um *script* em Python que utiliza a biblioteca Spleeter, passando para ele o caminho e a quantidade de *stems* a ser processado. Nesta fase, definimos o modelo de 5 *stems*.

Feito o processamento pelo Spleeter, temos como próximo passo, ler cada fonte detectada e executar um *script* Python. Esse *script* é responsável por detectar o tempo de ocorrência dos *beats* e gerar os arquivos de *samples*. Para realizar essa detecção, utilizamos a biblioteca Librosa, que conta com o recurso chamado *beat time*.

Ele nos entrega um *array* com os tempos, permitindo utilizar o FFMPEG para ler o arquivo original e gerar os arquivos de *beat* a partir dele.

Por fim é feito um registro no MongoDB, que é o banco de dados responsável por guardar informações dos arquivos que foram enviados. É possível identificar esse fluxo na Figura 11:

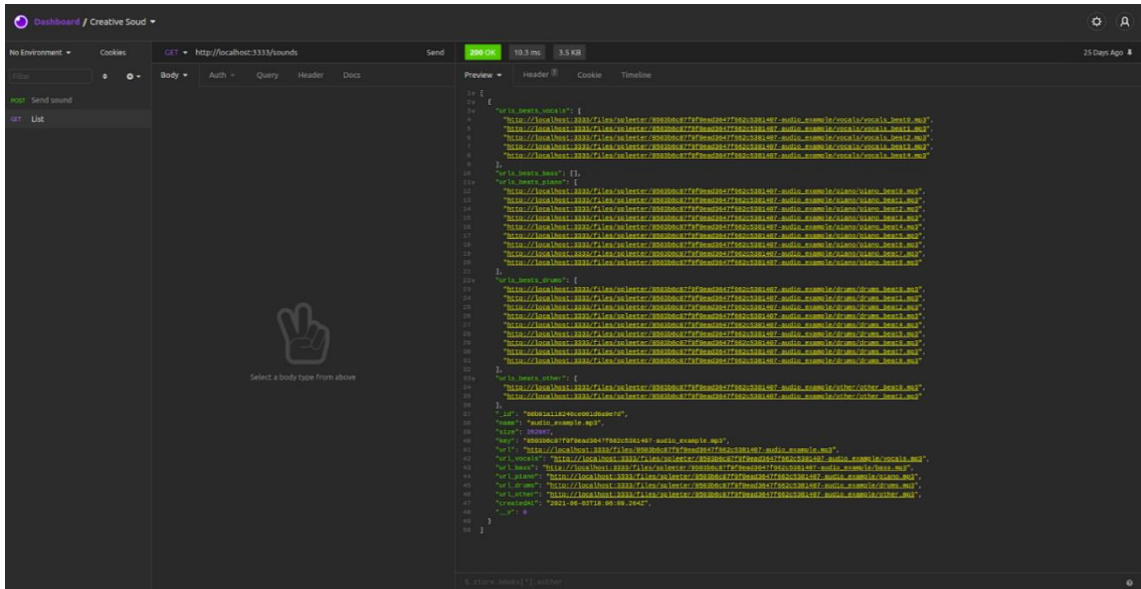
Figura 11 - Fluxo da aplicação.



Fonte: Autoria própria.

Para realizar testes na API é utilizado o software Insomnia (Figura 12), onde é possível visualizar toda estrutura da resposta.

Figura 12 - API Insomnia



Fonte: Autoria própria.

6 EXPERIMENTO E ANÁLISE

Seguindo referências de Djs e produtores musicais, selecionamos dois pontos para definir se temos ou não um bom *sample*, sendo eles o encaixem de repetição (*loop*) do *sample*, ou seja, a realização de uma transição harmônica do início e fim do mesmo *sample* e a ausência de ruídos e interferências (D, 2019). Neste experimento, verificamos se os *samples* criados automaticamente possuem esses critérios.

Para essa validação, realizamos um processo de experimentação e nele foram utilizadas as músicas especificadas na Tabela 2. Optamos por músicas que possuem a ocorrência de diversos acompanhamentos com o objetivo de ter uma segmentação mais igualitária.

Para executar todo o conjunto de aplicações, utilizamos uma estrutura *docker* responsável por deixar todos os serviços disponíveis.

Tabela 2 - Lista de Música utilizada no experimento.

AUTOR	MÚSICA	DURAÇÃO	TAMANHO
Vulfpeck	Animal Spirits	3m 10s	5,4 MB
Vulfpeck	Christmas In LA	2m 30s	3,3 MB
Vulfpeck	Daddy He Got A tesla	3m 25s	6,5 MB
Vulfpeck	Dean Town	3m 32s	6,1 MB
Ray Charles	Hit the road jack	2m	3,3 MB
Vulfpeck	It Gets Funkier	4m 51s	8,1 MB
Sven Karlsson	Standing Naked	3m 10s	5,3 MB

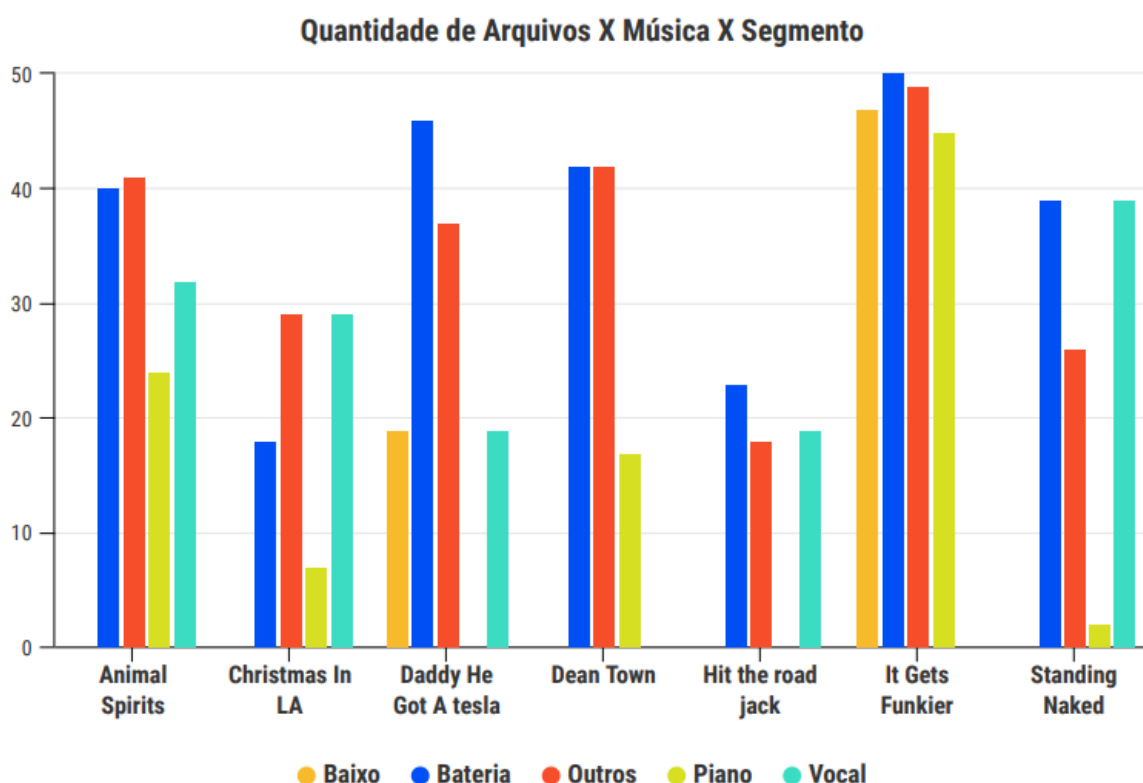
Fonte: Autoria própria.

Inicialmente, realizamos o *upload* de todas as músicas, e após a geração dos *samples*, avaliamos a sua qualidade. Em alguns casos, utilizamos a ferramenta de edição Audacity, para verificarmos falhas a nível do espectro do áudio.

6.1 ANÁLISE DA SEPARAÇÃO DE FONTE

Com objetivo de verificar a criação dos samples, que foram gerados após o uso da biblioteca Spleeter para realizar a detecção dos 5 *stems* e da Librosa para gerar os samples de cada *stem*, categorizamos cada sample criado em *stems* tendo como resultado a quantidade de arquivos gerados.

Gráfico 1 - Análise da separação de fonte.



Fonte: Autoria própria.

Podemos ver no Gráfico 1 que teremos em nossa base um maior número de samples de Bateria e Outros.

Com a informação da quantidade de arquivos gerados, poderemos utilizá-los como base para realizar a análise de qualidade desses samples que foram gerados.

6.2 ANÁLISE DE QUALIDADE DOS SAMPLES

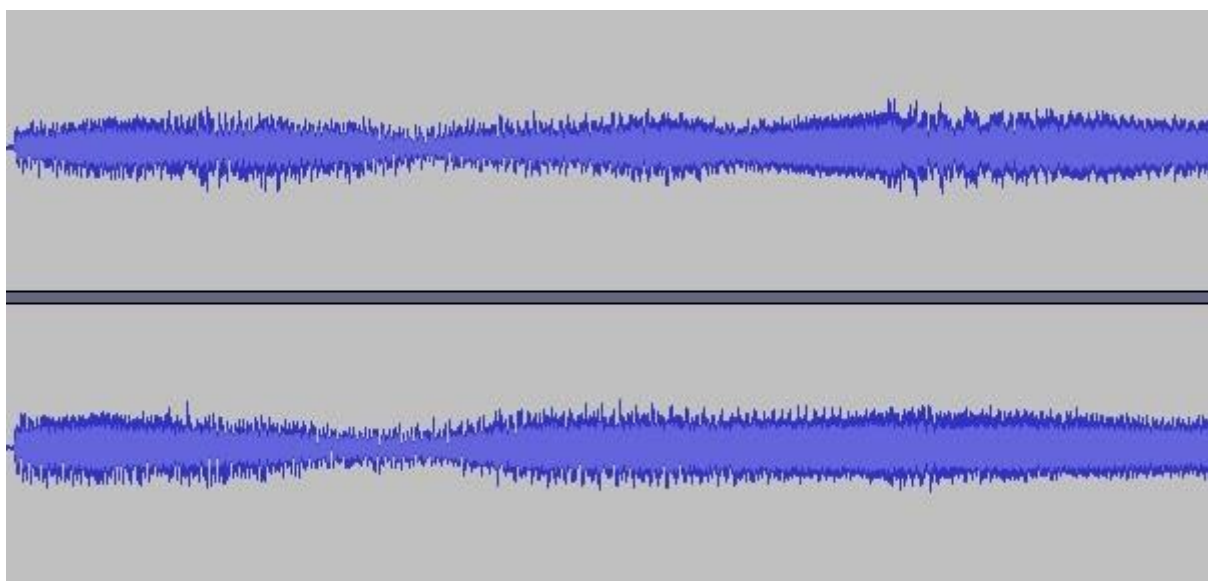
Para verificar a qualidade dos samples gerados, realizamos uma análise de forma manual em cada arquivo. Para auxiliar no entendimento desses parâmetros podemos fazer uma análise visual ao observar na Figura 13 o não encaixe na repetição desse áudio, causando assim uma ausência de harmonia ao escutá-lo em loop. Já na Figura 14 identificamos a presença ou não de ruídos ou interferências.

Figura 13 - Áudio sem encaixe de repetição



Fonte: Autoria própria.

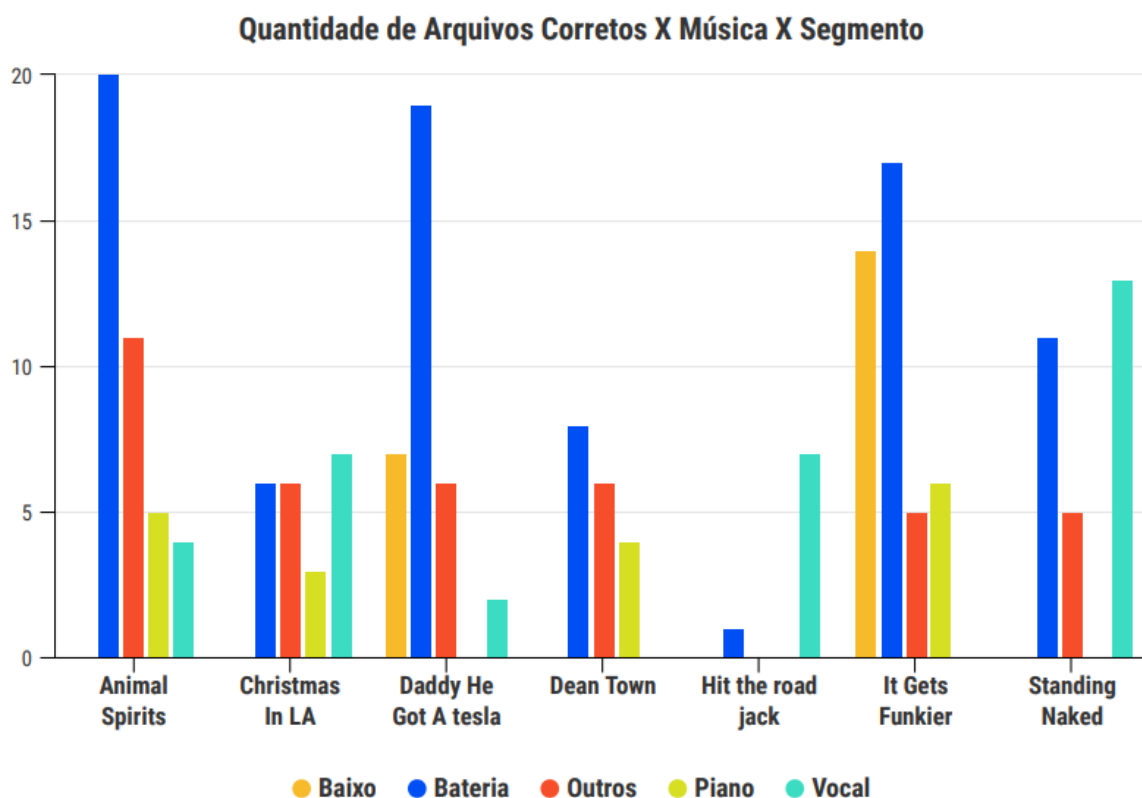
Figura 14 - Áudio com ruído.



Fonte: Autoria própria.

Feita essa classificação de cada sample com base nos parâmetros citados, realizamos a contagem dos considerados aceitos. Percebemos alguns pontos relevantes como: mesmo com a menor quantidade de números de arquivos do *stem* Baixo, o processo de geração de *sample* demonstrou um bom desempenho, além disso percebemos um considerável acerto na quantidade de *stems* da Bateria. Em contrapartida, vimos que essa assertividade varia muito com a música. Vemos no Gráfico 2 como está a distribuição dessa quantidade de acertos.

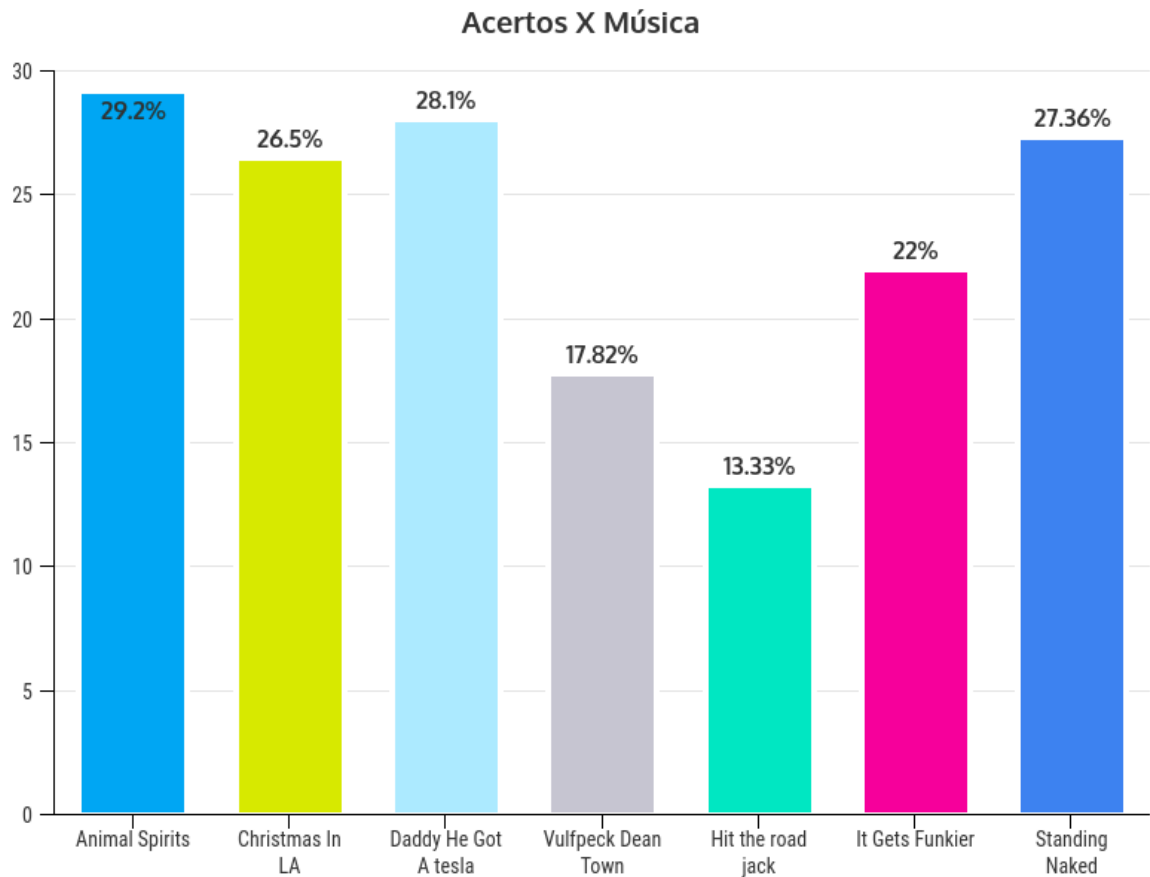
Gráfico 2 - Análise da geração de *sample* corretos.



Fonte: Autoria própria.

Olhando para os arquivos corretos gerados por *stem* em cada música, podemos observar que a quantidade de samples do *stem* “Bateria” continuou sendo o maior número. Já o *stem* “Outros” teve uma diminuição devido a grande quantidade de áudio agrupados em único *stem* causando assim a probabilidade de conter ruídos e interferências. No Gráfico 3, é vista essa comparação da assertividade entre as músicas.

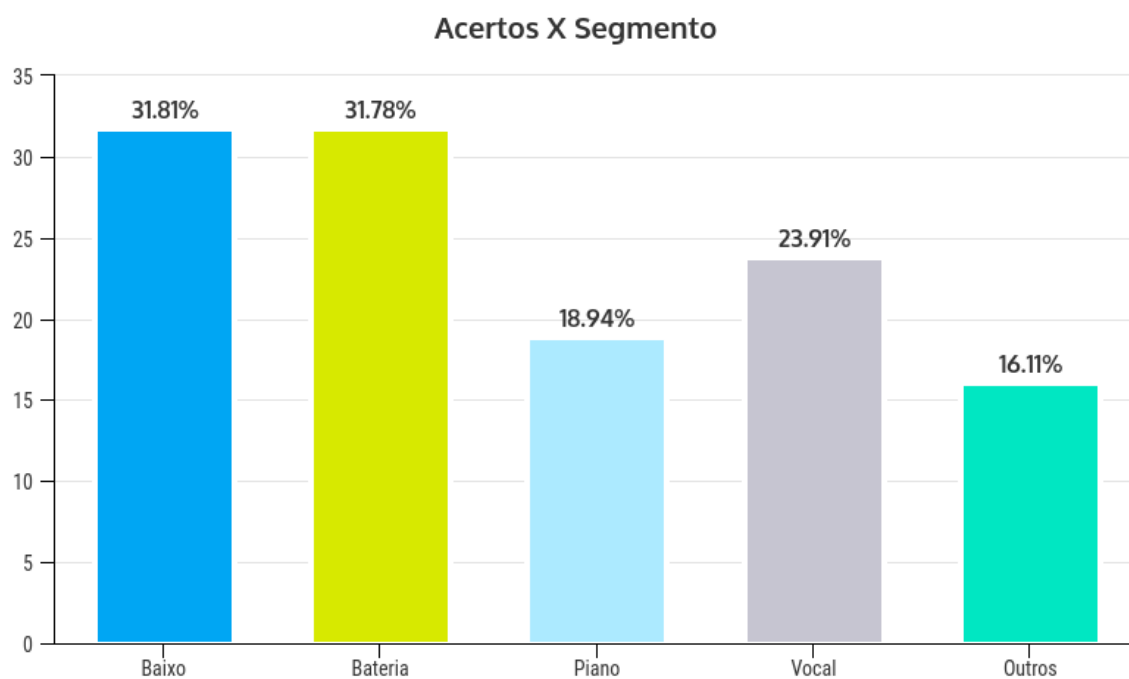
Gráfico 3 - Análise de assertividade entre as músicas.



Fonte: Autoria própria.

Por fim, olhamos para uma classificação por *stems*, onde dada a quantidade de elementos gerados, buscamos entender quais os *stems* obtiveram o melhor resultado. Vemos no Gráfico 4 que há uma proximidade no número de acertos dos segmentos de Bateria e Baixo. Como esperado, obtemos como menor acerto os *samples* de Outros, já que foi o segmento com maior dificuldade de interligação e maior número de falhas.

Gráfico 4 - Análise de assertividade por *stems*.



Fonte: Autoria própria.

7 CONCLUSÃO E TRABALHOS FUTUROS

O trabalho trouxe uma visão de unificação de recursos de pós-produção e criação musical, que é utilizado como base para diversos aplicativos que auxiliam nesse processo. Podemos ver que a aplicação proposta traz uma nova funcionalidade que é a geração de samples de forma automática, recurso que não é encontrado atualmente nas aplicações apresentadas. Vimos que já temos resultados excelentes no mercado, mas que ainda há uma necessidade de flexibilizar o processo criativo dos músicos, entregando mais recursos e integrações.

7.1 CONTRIBUIÇÕES

Conseguimos com esse trabalho, mergulhar um pouco no processo de produção, mostrando conceitos importantes tanto para o entendimento, quanto para o desenvolvimento de novos recursos. Tivemos a oportunidade de relacionar funcionalidades de ferramentas já consolidadas no mercado.

Como contribuição principal, temos a proposta de um sistema web que permite a geração de *sample* de forma automática e realiza a reprodução desses *samples* através da leitura de mensagens MIDI pela interação do instrumento com o *browser*.

O uso do protocolo MIDI trouxe a facilidade de, ao utilizar um teclado como instrumento, cada sample gerado é relacionado a uma tecla e ao tocá-la uma mensagem MIDI é enviada contendo a informação de qual tecla foi acionada para identificar qual sample deve ser tocado. Essa integração pode trazer uma flexibilidade e praticidade de tocar diretamente do instrumento os *samples* gerados, tornando todo o processo automatizado.

Obtivemos como resultado dos experimentos realizados, taxas de acerto relevantes, mostrando que é possível melhorar e automatizar cada vez mais o processo de produção musical, permitindo já ter samples de qualidade ou até trabalhar na melhoria de samples da sua escolha.

7.2 TRABALHOS FUTUROS

Para trabalhos futuros, vemos a necessidade de um estudo mais próximo dos músicos. Tendo como foco entender com mais profundidade o processo criação tanto

a nível de interface e interação, quanto da opinião sobre os recursos entregues. Além disso, o aprimoramento na geração e classificação dos samples de forma automática, realizando a experimentação de outras bibliotecas ou técnicas para separação de fonte.

8 BIBLIOGRAFIA

ACIMAN, Alexander. **Meet the unassuming drum machine that changed music forever.** Vox, [S.I.], abril/2018. Disponível em: <https://www.vox.com/culture/2018/4/16/16615352/akai-mpc-music-history-impact>. Acesso em: julho/2021.

ADRION, Roseli. **Super Pads: como um app musical feito no Brasil se tornou sucesso na China.** Tilt Uol, [S.I.], maio/2021. Disponível em: <https://www.uol.com.br/tilt/noticias/redacao/2021/05/19/super-pads-como-um-app-musical-feito-no-brasil-se-tornou-sucesso-na-china.htm>. Acesso em: junho/2021.

ALMEIDA, Diogo. **Desenvolvedor brasileiro cria ferramenta gratuita que separa músicas em faixas de voz e instrumentos.** G1, [S.I.], novembro/2019. Disponível em: <https://g1.globo.com/pb/paraiba/noticia/2019/11/30/desenvolvedor-brasileiro-cria-ferramenta-gratuita-que-separa-musicas-em-faixas-de-voz-e-instrumentos.ghtml>. Acesso em: junho/2021.

ASSIS, Paulo. **Um Breve Panorama da Evolução da Tecnologia Musical: Promessas e Riscos para a Diversidade de Expressões Culturais.** Buenos Aires, 2016. Disponível em: <https://www.teseopress.com/diversidadedeexpressoesculturaisnaeradigital/chapter/um-breve-panorama-da-evolucao-da-tecnologia-musical-romessas-e-riscos-para-a-diversidade-de-expressoes-culturais>. Acesso em: junho/2021.

BRANDÃO, Monique. **Os 12 Melhores Aplicativos de DAW Do Mundo.** LANDR, [S.I.], março/2017. Disponível em: <https://blog.landr.com/pt-br/os-12-melhores-aplicativos-de-daw-mundo/>. Acesso em: julho/2021.

BRÉSCIA, V. L. P. **Educação Musical: bases psicológicas e ação preventiva.** São Paulo: Átomo, 2003.

CARBONE, Nick. **Sequencer64.** GitHub, [S.I.], março/2021. Disponível em: <https://github.com/drumnickydrum/sequencer64>. Acesso em: junho/2021.

CASTRO, Jônatas. **Tabela de controles MIDI em português**. Jônatas Castro, [S.I.], dezembro/2016. Disponível em: <http://jonatascastro.com/tabela-de-controles-midi-em-portugues/>. Acesso em: julho/2021.

CHAGAS, Sandro. **Veja quais são as 5 etapas da produção musical em home studio**. Áudio e produção, [S.I.], setembro/2020. Disponível em: <https://ideaudio.net.br/veja-quais-sao-as-5-etapas-da-producao-musical-em-home-studio/>. Acesso em: junho/2021.

COLODRO, Caue. **Você sabe o que é Sample?**. Planeta Música, [S.I.], janeiro/2017. Disponível em: <https://blog.planetamusica.net/voce-sabe-o-que-e-sample/>. Acesso em: julho/2021.

CRAB, Simon. **O 'Mellotron' e 'Novatron'. Leslie Bradley, Reino Unido, 1963**. 120 years, Londres, 2014. Disponível em: <http://120years.net/the-mellotron-chamber-linleslie-bradleyuk1963/>. Acesso em: julho/2021.

D, Navie. **Picking Better Samples (Sampling Tutorial)**. Youtube, 19 de setembro de 2019. Disponível em: <https://www.youtube.com/watch?v=i2RP-96F67Q>. Acesso em: julho/2021.

EITC. **Sample Science: Evolution of the AKAI MPC**. Eletronic Music Collective, [S.I.], fevereiro/2019. Disponível em: <https://electronicmusiccollective.com/akai-mpc-history/>. Acesso em: julho/2021.

FFMPEG. **ABOUT FFmpeg**. FFMPEG, [S.I.], março/2010. Disponível em: <https://ffmpeg.org/about.html>. Acesso em: junho/2021.

GABRIEL. **Produção Musical: Tudo Que Você Precisa Para Começar Hoje Mesmo**. Landr, [S.I.], outubro/2020. Disponível em: <https://blog.landr.com/pt-br/music-production/>. Acesso em: junho/2021.

GHON, Daniel M. **A Tecnologia na Música**. XXIV Congresso Brasileiro da Comunicação, Campo Grande /MS, p. 1-13. setembro/2001. Disponível em:

<http://www.portcom.intercom.org.br/pdfs/14980100322230945765212217541460451734.pdf>. Acesso em: junho/2021.

HENNEQUIN, Romain *et al.* **Spleeter: a fast and efficient music source separation tool with pre-trained models**. The Journal of Open Source Software,[S.I.], junho/2020. Disponível em: <https://joss.theoj.org/papers/10.21105/joss.02154>. Acesso em: julho/2021.

KNEIPP, Eduardo Campos. **Análise de áudios para Predição de No-Show**. Orientador: Eduardo Camponogara. 2019. Monografia (Graduação) - Curso de Graduação em Engenharia de Controle e Automação. Universidade Federal de Santa Catarina, Florianópolis, abril/2019. Disponível em: https://repositorio.ufsc.br/bitstream/handle/123456789/200059/PFC%20Eduardo%20Campos%20Kneipp_2019-1.pdf?sequence=1&isAllowed=y. Acesso em: junho/2021.

LITERATURE. SigSep, [S.I.], novembro/2018. Disponível em: <https://sigsep.github.io/literature/#references>. Acesso em: junho/2021.

MARC. **The Historical Timeline of the Akai MPC**. Millennial Mind, [S.I.], 2019. Disponível em: <https://www.millennialmind.co/the-historical-timeline-of-the-akai-mpc/>. Acesso em: julho/2021.

MARTINS, Marcelo. **O padrão MIDI: protocolo de tempo-real e formato de arquivo**, 2016. PDF. Disponível em: https://www.ime.usp.br/~kon/MAC5900/seminarios/seminario_Marcelo.pdf. Acesso em: julho/2021.

RAMOS, Gomes. **Moises lança Aplicativo de Música para 175 países**. MOISES, [S.I.], dezembro/2020. Disponível em: <https://moises.ai/pt-br/blog/lancamento-aplicativo-de-musica/>. Acesso em: julho/2021.

ROCHA, Guilherme Lucio da. **Explicando em detalhes: o que é Sample**. Kondzilla, [S.I.], maio/2018. Disponível em: <https://kondzilla.com/m/explicando-em-detalhes-o-que-e-sample>. Acesso em: julho/2021.

SCATAMBURLO, B; CAMPOS, N. **A Era do Áudio: Tendências do consumo de Streaming de Áudio no Brasil**. Comscore. [S.l], julho/2020. Disponível em: <https://www.comscore.com/por/Insights/Blog/Tendencias-do-consumo-de-Streaming-de-Audio-no-Brasil>. Acesso em: junho/2021.

SILVA, Marco Aurélio Botelho da. **Padrão Midi**. [S.l], 2007. Disponível em: <http://www.midiacom.uff.br/~debora/fsmm/trab-2007-1/midi.pdf>. Acesso em: junho/2021.

SPARFLEX. **Os melhores programas de produção musical**. Sparflexs fios e cabos especiais, [S.l], dezembro/2020. Disponível em: <https://www.sparflex.com.br/os-melhores-programas-de-producao-musical/>. Acesso em: junho/2021.

TAMMARO, Rodrigo. **Inspiração ou cópia: entenda o sample na produção musical**. Jornalismo Júnior, [S.l.], outubro/2020. Disponível em: <http://jornalismojunior.com.br/inspiracao-ou-copia-entenda-o-sample-na-producao-musical/>. Acesso em: julho/2021.

WEB MIDI API. W3C, [S.l], março/2015. Disponível em: <https://www.w3.org/TR/webmidi/>. Acesso em: junho/2021.