



UNIVERSIDADE FEDERAL RURAL DO SEMIÁRIDO
DEPARTAMENTO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

YAGO RAVELL GONÇALVES SOUSA

**MELODIZ: UMA SOLUÇÃO EM SOFTWARE PARA AUXÍLIO
NO DESENVOLVIMENTO DA PERCEPÇÃO MUSICAL**

MOSSORÓ/RN
2017

YAGO RAVELL GONÇALVES SOUSA

**MELODIZ: UM SOLUÇÃO EM SOFTWARE PARA AUXÍLIO
NO DESENVOLVIMENTO DA PERCEPÇÃO MUSICAL**

Monografia apresentada ao Departamento de Ciências Exatas e Naturais como parte dos requisitos para obtenção do grau de Bacharel em Ciência da Computação, no curso de Bacharel em Ciência da Computação da UFERSA.

Orientador: Prof. Dr. Leandro Carlos de Souza
Coorientador: Prof. Dr. Bruno de Sousa Monteiro

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

S725m Sousa, Yago Ravell Gonçalves.
Melodiz: uma solução em software para o auxílio
no desenvolvimento da percepção musical / Yago
Ravell Gonçalves Sousa. - 2017.
34 f. : il.

Orientador: Leandro Carlos de Souza.
Coorientador: Bruno de Sousa Monteiro.
Monografia (graduação) - Universidade Federal
Rural do Semi-árido, Curso de Ciência da
Computação, 2017.

1. Percepção musical. 2. Aplicativo android. 3.
Intervalos musicais. I. Souza, Leandro Carlos de,
orient. II. Monteiro, Bruno de Sousa, co-orient.
III. Título.

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

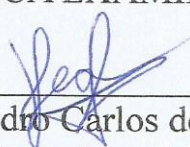
YAGO RAVELL GONÇALVES SOUSA

**MELODIZ: DESENVOLVIMENTO DE UM APLICATIVO
PARA AUXÍLIO NO DESENVOLVIMENTO DA PERCEPÇÃO
MUSICAL**

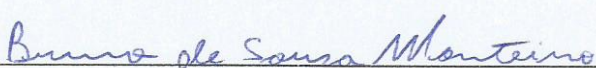
Monografia apresentada ao Departamento de Ciências Exatas e Naturais como parte dos requisitos para obtenção do grau de Bacharel em Ciência da Computação, no curso de Bacharel em Ciência da Computação da UFERSA.

APROVADA EM: 25 / 09 / 2017

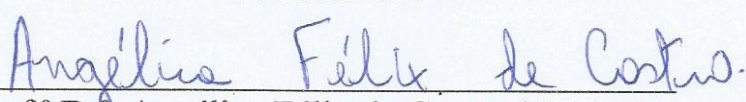
BANCA EXAMINADORA



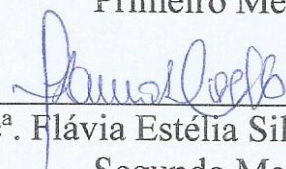
Prof. Dr. Leandro Carlos de Souza (UFERSA)
Orientador e Presidente



Prof. Dr. Bruno de Sousa Monteiro (UFERSA)
Coorientador



Prof.ª Dr.ª Angélica Félix de Castro (UFERSA)
Primeiro Membro



Prof.ª MSc.ª Flávia Estéla Silva Coelho (UFERSA)
Segundo Membro

MOSSORÓ/RN 2017

RESUMO

Este trabalho tem como objetivo propor e avaliar uma solução em software, para dispositivos moveis na plataforma Android, para auxiliar o desenvolvimento da percepção musical. Este aplicativo disponibiliza aos usuários exercícios com base na identificação de notas musicais e intervalos melódicos. Para incentivar os usuários, ele atribui pontos e medalhas, conforme os objetivos estipulados vão sendo alcançados. No decorrer do jogo, o aplicativo apresenta conceitos da teoria musical, de modo a fornecer para os usuários um primeiro contato com estes. Ao fim do jogo, o usuário pode consultar sua evolução por meio de suas medalhas, bem como comparar sua performance com a de outros usuários através do Ranking. Este ainda pode, caso deseje, recomeçar o jogo, visando praticar a identificação dos sons e/ou obter uma pontuação final mais elevada. O trabalho descreve ainda toda a teoria necessária para a produção do aplicativo, bem como a teoria musical, o processo desenvolvimento de softwares para o ambiente Android e o protocolo MIDI para execução das notas. Por fim é apresentado seu processo de validação, que se dá por meio de um questionário para avaliar o grau de satisfação do usuário em relação ao aplicativo proposto.

LISTA DE FIGURAS

Figura 1 – Notas musicais da escala ascendente de dó e seus acidentes	10
Figura 2 – Intervalos ascendentes dentro da escala	11
Figura 3 – IDE <i>Android Studio</i>	14
Figura 4 – Diagrama de casos de uso do aplicativo Melodiz	16
Figura 5 – Diagrama de classes do aplicativo Melodiz	18
Figura 6 – Modelo do Banco de Dados da Aplicação	21
Figura 7 – Tela de jogo do aplicativo Melodiz	22
Figura 8 – Tela de jogo do aplicativo Melodiz	23
Figura 9 – Galeria de medalhas do aplicativo Melodiz	24
Figura 10 – <i>Ranking</i> do aplicativo Melodiz	25
Figura 11 – Nível de interesse dos entrevistados no estudo musical	26
Figura 12 – Nível de experiência dos entrevistados com ferramentas digitais de ensino	26
Figura 13 – Avaliação da capacidade intuitiva do aplicativo Melodiz	27
Figura 14 – Nível de satisfação quanto a curva de dificuldade do Melodiz	27
Figura 15 – Grau de facilidade da teoria musical apresentada no Melodiz	28
Figura 16 – Avaliação do potencial de recomendação do Melodiz.....	28

SUMÁRIO

1. INTRODUÇÃO	6
1.1. Descrição do problema abordado	7
1.2. Objetivos	7
1.3. Justificativa	7
1.4. Organização	8
2. REFERENCIAL TEÓRICO	9
2.1. Trabalhos Relacionados	9
2.2. Conceitos básicos de teoria musical	9
2.3. <i>Android</i>	12
2.3.1. <i>Android Studio</i>	13
2.3.2. API <i>Android.media.MIDI</i>	14
2.3.3. Biblioteca <i>mididriver</i>	15
3. DESENVOLVIMENTO DO MELODIZ	17
3.1. Diagrama de Casos de Uso	17
3.2. Diagrama de Classes	18
3.3. Banco de Dados	20
3.4. Interface Gráfica	21
3.5. Análise de Satisfação de Uso	25
4. CONCLUSÃO E TRABALHOS FUTUROS	30
BIBLIOGRAFIA	31
APÊNDICE A	33

1. INTRODUÇÃO

A música pode ser definida como a arte dos sons, combinados de acordo com as variações da altura, proporcionados segundo sua duração e ordenados segundo a lei da estética. Ela é composta de três elementos principais: a melodia, o ritmo e a harmonia. No entanto, a combinação dos dois primeiros já encerra um sentido expressivo musical (Priolli, 2006).

A música é um movimento cultural que acompanha a humanidade desde os seus primórdios. Ela possui diferentes vertentes, às quais se tornaram elementos característicos de diferentes tribos. Devido a isso, o envolvimento do homem com ela está em um constante processo de evolução. Desta forma, é crescente o número de pessoas que são incentivadas a começarem seus estudos nessa área.

Aos novos adeptos dessa vertente artística, o desenvolvimento inicial tende a não ser muito satisfatório. Melo (2009) já citava esse problema, ao mencionar que o ensino da música oferece um certo grau de dificuldade. Fator esse que explica a baixa taxa de estudantes de música que seguem seus estudos.

Visando um crescimento no número de adeptos à educação musical tem se apoiado em ferramentas que buscam um aperfeiçoamento na execução mecânica dos mais variados instrumentos. Essa abordagem tem o intuito de prover um estímulo inicial para o estudante, que em um intervalo menor de tempo será capaz de executar as primeiras peças. Estas ferramentas tendem a proporcionar resultados rápidos, algo que pode elevar o entusiasmo dos estudantes.

Estudar música é buscar entender a música, pura e simplesmente. Enquanto sua teoria facilita o reconhecimento de certos padrões, características de gêneros específicos. Apenas o desenvolvimento da percepção musical garante o real entendimento de uma ideia musical. A principal ferramenta para se estudar música deve ser a percepção e a assimilação de sons através da audição. Para tal, deve-se treinar a capacidade de ouvir, buscando o chamado ouvido absoluto. Este possibilita uma rápida e precisa identificação sonora e o desenvolvimento da percepção musical.

No contexto de ensino musical este trabalho propõe o desenvolvimento de um aplicativo para a plataforma *Android* que auxilia estudantes na evolução da percepção musical. O aplicativo se caracteriza por um jogo e se baseará em processos repetitivos de identificação de sons, isolados e combinados. Serão apresentados, ao usuário, desafios com

um grau crescente de dificuldade, baseados em uma lógica do conteúdo e na sua curva de aprendizado.

1.1. Descrição do problema abordado

Um estudante que não desenvolve um bom ouvido musical não tem sequer a capacidade de afinar seu próprio instrumento sem o auxílio de ferramentas especializadas. É importante que um instrumento esteja bem ajustado e com afinação precisa, pois assim, poderá produzir sons agradáveis aos ouvidos. Esse deve ser o principal objetivo dos aspirantes a músicos, pois apenas assim podem produzir música (Losch e Fortes, 2014).

Outra barreira enfrentada por adeptos das ferramentas de aprimoramento mecânico no seu processo de estudo é aprender novas músicas sozinho. Visto que, como estes não possuem a capacidades de identificar as notas ouvidas, não conseguem encontrá-las em seu instrumento. Isso os leva a ter um repertório reduzido, adquirido de terceiros, sem uma garantia da precisão desse material.

1.2. Objetivos

O principal objetivo deste trabalho é descrever uma solução de *software* para o desenvolvimento da percepção musical. Mais especificamente, a educação do ouvido absoluto. Como artefatos que contribuem para alcançar o objetivo geral, podemos citar:

- Identificação de notas isoladas;
- Identificação de duas notas tocadas isoladamente que compõem intervalos, separados em módulos específicos;
- Identificação de intervalos apresentados em módulos anteriores;
- Identificação isolada de duas notas apresentadas simultaneamente, dentre todos os intervalos apresentados nos módulos anteriores.

1.3. Justificativa

Os *softwares* presentes na literatura que abordam o problema se mostram eficientes em garantir uma introdução ao estudo musical. No entanto, foram desenvolvidas visando abordar o ensino da teoria musical (Branco e Matos, 2009; Yatsuda et al., 2011; Bordini,

2015), fazendo alusão a notações musicais e instrumentos específicos. A solução proposta aborda apenas o reconhecimento de sons, de uma forma lúdica, visando não alienar usuários não familiarizados com o universo musical.

1.4. Organização

Este trabalho é composto de quatro capítulos, este capítulo introdutório e os descritos a seguir.

Capítulo 2 – Referencial Teórico

Este capítulo tem como objetivo apresentar uma noção básica do conhecimento nas áreas necessárias para a execução deste trabalho. Entre essas informações estão: conceitos musicais, como as definições de notas e melodia, que facilitam o entendimento do problema. Além disso, o capítulo fornece uma breve explicação sobre o desenvolvimento de aplicativos para a plataforma *Android*.

Capítulo 3 – Desenvolvimento do Melodiz

A finalidade deste capítulo é descrever a solução proposta, destacando os passos para sua implementação. Isto é, apresenta uma solução lógica no âmbito do ensino musical e os meios para desenvolvê-la digitalmente.

Capítulo 4 – Conclusão e Trabalhos Futuros

Neste capítulo são expressas as opiniões sobre o trabalho realizado, em relação a ideia proposta inicialmente. Além disso, são abordados possíveis direcionamentos com o intuito de estender a área do conhecimento musical alcançada pelo aplicativo.

2. REFERENCIAL TEÓRICO

Para uma compreensão do problema abordado, são necessários certos conhecimentos teóricos. Visando reafirmar a viabilidade do projeto, serão apresentados alguns trabalhos similares, que mostram que o problema dá margem para diferentes soluções. Por conseguinte, será apresentada a teoria referente às tecnologias utilizadas na elaboração deste trabalho.

2.1. Trabalhos Relacionados

Esta seção relaciona alguns trabalhos correlatos. A maioria deles investiga e cria soluções para a questão do ensino musical baseado em ferramentas digitais. As soluções descritas apresentam propósitos similares, mas exemplificam como o mesmo problema pode ser abordado de diferentes maneiras.

Branco e Matos (2009) dinamizaram o desenvolvimento da percepção musical através de um sistema de apoio à realização de exercícios. O sistema é voltado ao desenvolvimento da percepção musical, com base na identificação de intervalos musicais. Ele se caracteriza como uma ferramenta de auxílio à aprendizagem a ser utilizado por alunos, professores, profissionais e por qualquer pessoa que tenha interesse em música.

Yatsuda et al. (2011) propuseram o Treinamento Auditivo Pela Web (TAW). Este trabalho possibilita o ensino musical à distância com treinamento e o acompanhamento da aprendizagem e do progresso na percepção musical. Tudo isto pode ser feito com o uso de um computador que tenha recursos de áudio e conexão com a *Internet*.

Bordini (2015) propõe a formação de professores em tecnologia digital através de um estudo sobre a utilização do jogo MUSIKINÉSIA na educação musical. O trabalho investiga de que modo os professores de Música do ensino superior de uma instituição pública federal fazem uso de um jogo educacional eletrônico como artefato capaz de contribuir ao ensino de um conteúdo musical específico.

2.2. Conceitos básicos de teoria musical

Uma música pode ser decomposta em 3 elementos principais: melodia, harmonia e ritmo. A melodia é formada pelos sons distribuídos sequencialmente. A harmonia pela execução simultânea de dois ou mais sons que apresentam uma coerência musical. O ritmo caracteriza o intervalo de tempo na duração dos sons e o espaço entre eles (Priolli, 2006).

Os sons são caracterizados, em relação a sua frequência, como notas musicais. Na música ocidental existem sete notas: Dó, Ré, Mi, Fá, Sol, Lá e Si. A faixa de frequência sonora entre elas é o que chamamos de intervalo. Esses intervalos são medidos em semitons, a menor distância usada musicalmente entre duas notas (Priolli, 2006). Os intervalos usado entre as sete notas variam entre um (Mi a Fá, Si a Dó) e dois semitons (Dó a Ré, Ré a Mi, Fá a Sol, Sol a Lá e Lá a Si).

O termo “escala” é atribuído à sequência ordenada de notas, que obedecem um certo padrão de distribuição. Para que uma escala seja dita completa é adicionada ao fim da mesma uma oitava nota, igual à primeira, porém numa frequência mais alta (Priolli, 2006). a execução crescente (em frequência) das notas executadas é dominada ascendente (Dó a Si), do contrário (Si a Dó) é dita descendente, esta nomenclatura é válida também para intervalos.

Devido aos intervalos de dois semitons entre a maioria das notas de uma escala, esta possui os chamados acidentes. Esses são notas intermediárias, que não fazem parte da escala natural. Elas são descritas pela nota da escala que as precedem e o sinal de sustenido (#) ou a nota que as procedem com o sinal de bemol (*b*). A escala musical é apresentada na Figura 1.

Figura 1 – Notas musicais da escala ascendente de dó e seus acidentes

Dó	Dó#Réb	Ré	Ré#Mib	Mi	Fá	Fá#Solb	Sol	Sol#Láb	Lá	Lá#Sib	Si	Dó
----	--------	----	--------	----	----	---------	-----	---------	----	--------	----	----

Fonte: Autoria própria

Devemos observar, por exemplo, que embora Dó# e Réb sejam notações aplicadas de forma diferente em um contexto musical representam o mesmo som (Priolli, 2006). Na Tabela 1, tem-se as 7 notas naturais e os 5 acidentes criados pelos intervalos de dois semitons, formando um total de 12 notas.

As notas da Tabela 1 estão contidas dentro da mesma oitava, que é o intervalo entre duas notas, no qual a primeira possui o dobro ou metade da frequência da segunda. Desta forma, as duas representam a mesma nota, uma soando mais aguda (de maior frequência) e outra mais grave (de menor frequência). No caso descrito na tabela 1, o segundo Dó está uma oitava acima do primeiro.

Assim como no intervalo de oitava, as outras distâncias (intervalos) entre as notas da escala também apresentam nomenclatura particular. Ela baseia-se no número de notas na

escala entre a primeira nota do intervalo e a segunda. Ao analisarmos a escala, tomando como primeira nota do intervalo o dó, obtemos os intervalos ascendentes que compõem a Figura 2.

Figura 2 – Intervalos ascendentes dentro da escala

INÍCIO DO INTERVALO	FIM DO INTERVALO	TIPO DO INTERVALO	DISTÂNCIA (SEMITONS)
DÓ	RÉ	SEGUNDA	2
DÓ	MI	TERÇA	4
DÓ	FÁ	QUARTA	5
DÓ	SOL	QUINTA	7
DÓ	LÁ	SEXTA	9
DÓ	SI	SÉTIMA	11
DÓ	DÓ	OITAVA	12

Fonte: Autoria própria

A posição final do intervalo na escala dita o seu tipo, Ré é segunda nota da escala, portanto Dó a Ré é chamado de intervalo de segunda. Analogamente, como Mi é a terceira nota da escala, Dó a Mi caracteriza um intervalo de terça, e assim sucessivamente. A principal informação a ser extraída da Figura 2 é a distância entre as notas de cada intervalo, pois, conhecendo as sete notas musicais e seus cinco acidentes, pode-se encontrar o fim do intervalo a partir de qualquer uma das doze notas.

A teoria apresentada é de extrema importância para se iniciar no estudo da música, pois, a porta de entrada para conteúdos mais complexos abordados posteriormente. No entanto, é inegável que para qualquer iniciante o essencial é conseguir perceber música.

Através do treinamento da audição ele poderá compreender sons das mais variadas alturas e frequências, intervalos e padrões rítmicos. Eles devem ser o foco principal nessa fase introdutória. Em geral, para o desenvolvimento da percepção musical, os estudantes da área realizam uma quantidade exaustiva de exercícios repetitivos. Desta forma, aos poucos, eles tornam-se capazes de associar o que está sendo ouvido a algum elemento da teoria musical (Goulart, 2002).

No contexto de aprendizagem de música, é necessária a construção de ferramentas que auxiliem no processo de aprendizagem, a fim de se evitar o cansaço das repetições e uma evolução mais acelerada. Além disso, a ferramenta pode interagir com o usuário e lhe proporcionar espontaneidade. Isto pode se dar através de atividades como jogos e brincadeiras, por exemplo. Essas atividades tornam a aprendizagem mais dinâmica e divertida, evitando assim, falta de estímulo para o usuário.

2.3. *Android*

Android é um sistema operacional para dispositivos móveis desenvolvido pela *Google*. Ele se destaca por ser a principal escolha para *smartphones*, *tablets*, entre outros. Fator esse que o torna mais importante, dada sua crescente popularidade desses dispositivos (*android*, 2017).

A portabilidade dos dispositivos móveis tem causado migração dos usuários. Estes têm dado preferência aos computadores de mesa e notebooks na realização de tarefas mais extensas, atribuindo às mais corriqueiras às unidades portáteis. Isto se dá devido à facilidade de estar com a ferramenta em mãos sempre que necessária.

Tendo em vista o crescente uso desses dispositivos, a criação de um sistema estável para gerenciá-los era de suma importância. Este é um dos fatores que garantiu a popularização do *Android*, sua confiabilidade e desempenho perante outros sistemas semelhantes. Que atribui uma rápida popularidade e predileção perante os fabricantes e consumidores (*android*, 2017).

Além dessa crescente popularidade devido ao seu desempenho, a plataforma também se difundiu por ser *open-source*. Este modelo de sistema permite que os usuários possam desenvolver para a plataforma e distribuir as extensões desenvolvidas. Este desenvolvimento é facilitado pelo uso de interfaces de programação de aplicativos (*Application Programming Interface* – API). Essas interfaces permitem ao usuário a programação mantendo conexões com sistemas e aplicativos externos, sendo de responsabilidade da API gerenciar essa conexão.

O projeto faz uso, em especial, da API *Android.media.MIDI*, responsável pelo gerenciamento do protocolo MIDI (Musical Instrument Digital Interface). Este protocolo descreve uma comunicação de mensagens geradas por eventos, como um toque na tela do dispositivo. Por conseguinte as mensagens ativam eventos específicos, como o disparo de

sons pré-gravados. Desta forma, não à necessidade de armazenar sons na aplicação, apenas solicitar ao player MIDI a execução de sons gerados por este.

Outra grande vantagem do desenvolvimento para o ambiente *Android* é a presença de um banco de dados integrado, o SQLite. Este permite aos desenvolvedores persistir dados de seus projetos sem a necessidade de instalação de um banco de dados externo. O que gera uma maior portabilidade para o aplicativo desenvolvido. O SQLite funciona através de comandos SQL (Structured Query Language), linguagem, mais utilizada para o desenvolvimento de bancos de dados.

A facilidade na programação gerada por essas funcionalidades do *Android*, embora indiretamente, estimulam o crescimento no número de seus usuários. Pois assim mais aplicativos são desenvolvidos e distribuídos, possibilitando a estes usuários o acesso a um vasto catálogo de aplicativos para usufruir. Esses aplicativos podem ser instalados diretamente no aparelho ou então baixados através do serviço de distribuição da *Google*, a *Play Store*, diretamente do aparelho, quando conectado a *Internet*.

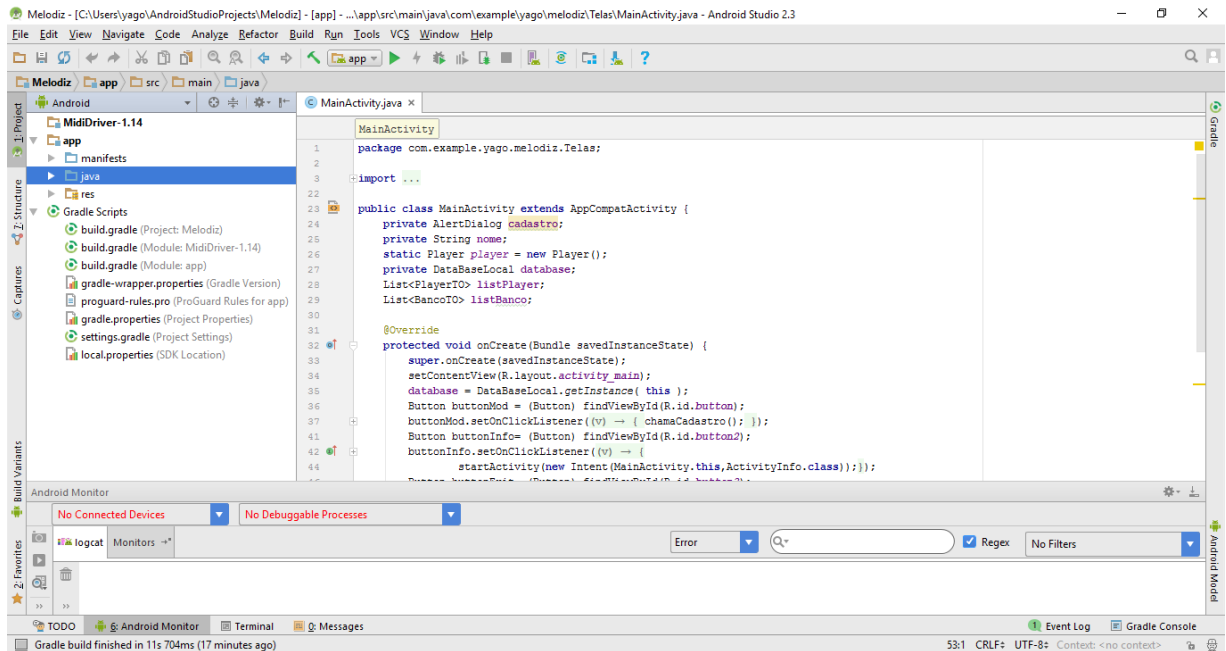
Através do ambiente de desenvolvimento Android SDK pode-se desenvolver para as mais variadas versões do Android. Este ambiente faz uso da linguagem de programação Java e a maioria de suas ferramentas para a criação dos aplicativos. Além disso, o ambiente ainda dispõe de um emulador de dispositivos para testes, para usuários que não desejam dispor de um aparelho durante o desenvolvimento.

O ambiente *Android* SDK disponibiliza as ferramentas essenciais para programação para Android, mas ele requer um ambiente de desenvolvimento Integrado (Integrated Development Environment – IDE), que gerenciar as ferramentas do *Android* SDK e fornece o editor para a programação. Para esta função foi utilizada a ferramenta oficial, a IDE *Android Studio*.

2.3.1. *Android Studio*

O *Android Studio* é uma IDE desenvolvida pela *Google* visando atender os programadores para ambiente *Android*. Conforme apresentado na Figura 3, essa ferramenta proporciona ao usuário uma forma de integrar e ter acesso a todos os elementos necessário para a programação. Pois através do mesmo pode-se acessar o Android SDK, realizar a escrita do código, fazer a depuração, fazer o *link* de bibliotecas, bancos de dados e do emulador (*android*, 2017).

Figura 3 – IDE *Android Studio*



Fonte: Imagem do projeto MELODIZ (captura de tela)

A IDE fornece *templates* para a criação de projetos e seus componentes, como classes java, *layouts* das telas e arquivos de configuração, como o *AndroidManifest*, que contém informações essenciais do aplicativo. Com isso, usuários menos experientes com a codificação para *Android* podem fazer uso desse modelos para agilizar o processo. Além de garantir um padrão para as aplicações desenvolvidas no *Android Studio* (android, 2017).

2.3.2. API *Android.media.MIDI*

Esta API desenvolvida especificamente para aplicações *Android* contém classes responsáveis pelo gerenciamento do protocolo de eventos MIDI. Os eventos citados podem ser gerados por periféricos de entrada, como dispositivos USB, *Bluetooth* ou mesmo internamente por dispositivos virtuais. Além disso, a API permite ao programador acessar informações importantes referentes aos dispositivos MIDI em uso (android, 2017).

O protocolo MIDI descreve um padrão de interconexão física com ambientes digitais, que foi desenvolvido por uma empresa de sintetizadores. Desde a sua criação é o padrão adotado por instrumentos digitais, controladores que ativam sons gerados digitalmente (Silva, 2016). Este é o principal motivo para adoção do protocolo para o gerenciamento de

sons do aplicativo. Ele é capaz de criar notas musicais, individualmente ou em conjunto, através da captação de um evento, como um toque na tela do dispositivo.

Para a utilização da API *Android.media.MIDI* é necessária uma série de passos para a configuração dos dispositivos, visando o do protocolo MIDI, especificamente, ele declara a aplicação MIDI no manifesto *android*, checa o suporte para MIDI da versão *android* instalado no dispositivo, instância o gerenciador MIDI, entre outras (*android*, 2017). Por isso, visando facilitar o uso da API, essas configurações ficam ao encargo de uma biblioteca, a *mididriver*.

2.3.3. Biblioteca *mididriver*

A biblioteca *mididriver*, criada pelo usuário do *GitHub* Bill Farmer, se encarrega desse gerenciamento básico do protocolo. Funções como as configurações iniciais da interface, até a tradução das mensagens MIDI geradas pelos eventos para o dispositivo virtual. Para que o programador foque sua atenção na lógica de sua aplicação, em vez de configurações básicas do sistema (*GitHub*, 2017).

3. DESENVOLVIMENTO DO MELODIZ

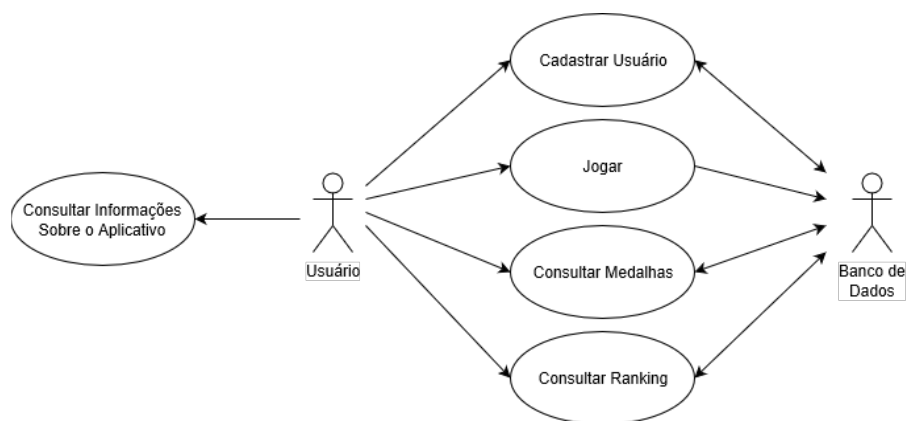
Este capítulo apresenta o desenvolvimento do aplicativo Melodiz. Aqui são verificadas como as tecnologias exploradas no capítulo 2 são combinadas para gerá-lo. O processo de criação do Melodiz é dividido em subtópicos para descrever melhor a fase de desenvolvimento lógico e implementação. Assim, serão citados documentos do projeto como casos de uso, diagramas de classes, além de partes fundamentais da codificação.

3.1. Diagrama de Casos de Uso

O diagrama de caso de uso é uma ferramenta de auxílio no projeto de sistemas digitais, nele são descritos os requisitos funcionais do sistema. Assim, descrevem as ações executadas pelo programa, conforme entradas fornecidas por elementos externos (atores). Esses elementos podem ser de usuários do sistema ou sistemas externos que trabalham em conjunto com o software projetado.

A Figura 4 descreve os casos de uso do aplicativo MELODIZ, seus atores e as ações que dependem deles. Os cinco casos apresentados serão descritos, visando identificar a relação particular entre os atores em cada caso particular. Esses atores são o usuário do aplicativo e o banco de dados da aplicação, que fornece informações do usuário com o intuito de garantir um aumento dos desafios, baseando-se na sua curva de aprendizagem.

Figura 4 – Diagrama de casos de uso do aplicativo Melodiz



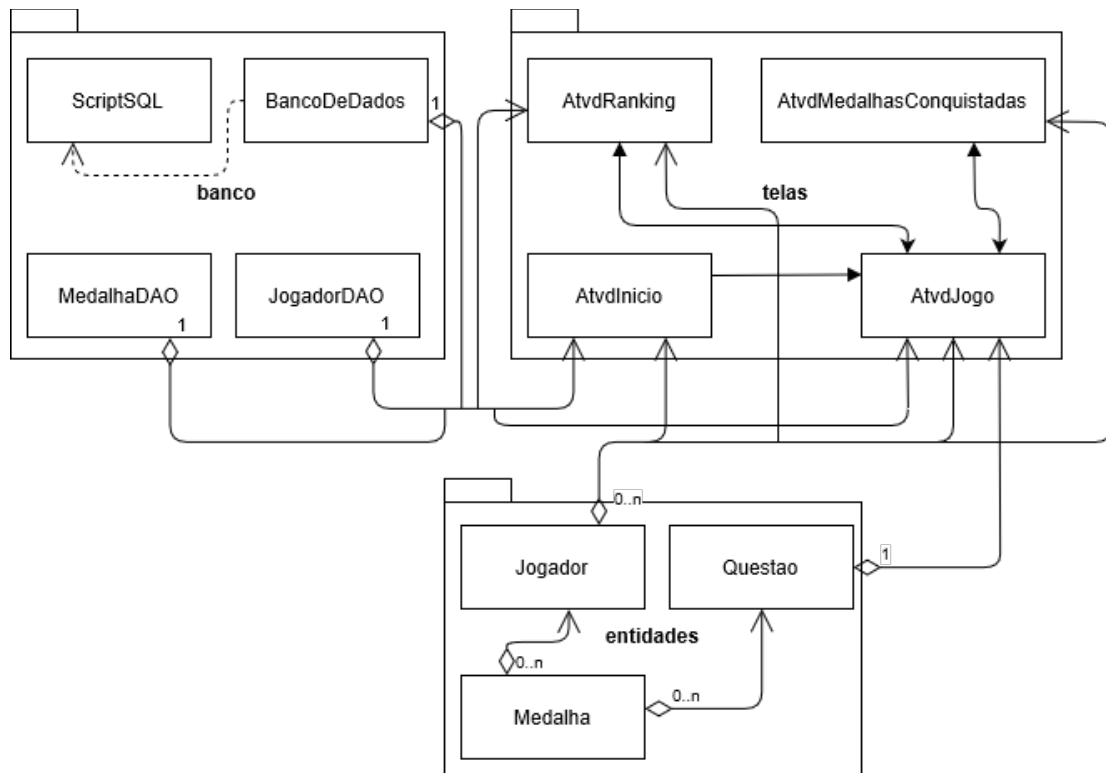
Fonte: Imagem criada através da plataforma draw.io

- Cadastrar Usuário: após consulta no banco para garantir que nenhum usuário cadastrado, o aplicativo solicita o cadastro do usuário. Esse cadastro é feito através do nome, com um histórico de partidas zerado;
- Jogar: módulo que incentiva um desenvolvimento inicial do usuário, focando no reconhecimento de notas isoladas. No qual, o mesmo pode treinar, o quanto achar necessário, antes de tentar identificar os intervalos;
- Consultar Medalhas: ao clicar na opção Medalhas o usuário terá acesso à sua galeria de medalhas. Nesta galeria, estão disponíveis para consulta, as notas e intervalos musicais referentes as medalhas já conquistadas. Esta tela tem como intuito demonstrar ao usuário quantos desafios ele deverá enfrentar para concluir o jogo;
- Consultar *Ranking*: ao clicar na opção *Ranking*, o usuário terá acesso a uma tela composta pela lista de usuários e suas devidas pontuação. Estes estão ordenados em ordem decrescente de pontuação. Assim, o usuário poderá comparar seu desempenho com o dos outros jogadores;
- Consultar Informações Sobre o Aplicativo: área dedicada para que o usuário possa ter acesso a instruções sobre o processo de execução, bem como informações sobre o aplicativo em geral.

3.2. Diagrama de Classes

Outro documento de extrema importância para o desenvolvimento de um software é o diagrama de classes. Ele oferece uma base de desenvolvimento para as classes a serem construídas, bem como a comunicação entre elas. A Figura 5 representa o diagrama de classes do MELODIZ, que contém os principais elementos para o projeto, como variáveis e funções que compõem esses objetos.

Figura 5 – Diagrama de classes do aplicativo Melodiz



Fonte: Imagem criada através da plataforma draw.io

O pacote ENTIDADES apresenta os objetos e as funções responsáveis pelas regras de negócio do *software*. Essas regras determinam o funcionamento do jogo, garantindo que alcance às expectativas de execução determinadas no projeto do *software*. Cada uma das classes do pacote implementa regras específicas de elementos distintos, que trabalham em conjunto para garantir o fluxo correto de execução. As classes citadas são:

- **Jogador:** A classe que aborda os componentes e métodos que representam o usuário do software. Este usuário é composto por um nome, módulo do jogo no qual o usuário se encontra, sua pontuação atual, uma lista de suas medalhas e um identificador para o jogador em questão. Ainda nesta classe, estão implementados os métodos para consulta e edição dos componentes deste usuário, bem como métodos para envio dos dados do usuário para o banco de dados;
- **Medalha:** A classe Medalha descreve os componentes das medalhas do jogador. Estas medalhas são compostas pelo número de partidas necessárias para conquistá-la, a quantidade de consultas para a identificação da resposta, um marcador que informa que a mesma foi ou não conquistada e identificadores para a medalha e o jogador ao

qual pertence. Além dos métodos necessários para consulta e edição dos seus elementos, a classe implementa métodos que facilitam a gravação dos dados desta medalha no banco de dados;

- **Questão:** A classe *Questão* é responsável por gerar, a cada nova partida, a nota ou intervalo que deve ser encontrado. Para isso, busca informações do jogador e suas medalhas, tomando como base seu módulo atual e as medalhas deste conquistadas. Esta classe possui como componentes um identificador do retorno, um elemento *Random* para sua criação e uma cópia das medalhas do jogador, facilitando assim as consultas. Os métodos dessa classe possuem como retorno um identificador que é tratado pela classe de jogo, para obtenção da nota ou intervalo correspondente.

O pacote telas é o responsável por gerenciar os componentes dos *layouts*, bem como instanciar as entidades e garantir a comunicação entre os mesmos. Este pacote é composto de quatro classes, cada uma associada a um layout específico, sendo de responsabilidade dessas o mantimento do fluxo de execução. Como instanciar as classes responsáveis por gerenciar as telas seguintes e repassar às mesmas todas as informações necessárias para sua execução. As classes citadas são:

- **AtvdInicio:** A classe *Atvdinicio* é responsável por realizar cadastro de novos jogadores, além de informar os já cadastrados. Para estes o login é feito apenas clicando-se no nome do jogador em questão. Após o cadastro ou seleção do jogador, a classe encapsula os componentes do mesmo, passando-os a *AtvdJogo*;
- **AtvdJogo:** Principal classe do projeto, a classe *AtvdJogo* implanta as regras de negócio. Ela recebe as informações do jogador, solicita novos desafios da classe *Questoes*, analisando o retorno desta e externando o problema associado. Após o envio da resposta, a classe a avalia, comparando-a com a opção correta. Esta classe oferece ainda informações sobre o nível, progresso e pontuação do jogador. Além disso, é responsável pela atualização dos dados do mesmo após cada partida;
- **AtvdMedalhasConquistadas:** A classe *AtvdMedalhasConquistadas* detecta os toque nas medalhas, informando ao usuário, caso o mesmo já tenha obtido está medalha, a resposta para o desafio referente a mesma;
- **AtvdRanking:** A classe *AtvdRanking* realiza uma consulta para detectar todos os jogadores, organizando-os em ordem decrescente de pontuação e apresentando-os.

O último pacote a ser abordado é o pacote banco, responsável por gerenciar toda a comunicação com o banco de dados SQLite. Neste pacote estão contidas classes para abertura da conexão com o banco, criação das tabelas e classes para garantir a conexão entre o projeto e esses elementos de gerenciamento do banco de dados. As classes citadas são:

- BancoDeDados: A classe BancoDeDados é responsável conexão com o Banco, criando-o e com os atributos nela contidos, como nome e número de versão;
- ScriptSQL: A classe ScriptSQL possui os comandos para criação das tabelas no banco de dados SQLite.
- JogadorDao: A classe Jogador Dao possui os métodos necessários para gravar e obter dados da tabela Jogadores. A mesma garante a persistência dos dados possa ser executada pelas classes do pacote telas. A ação é possível através dos seus métodos, que possuem como retorno e/ou atributos os elementos pretendidos, como instâncias da classe Jogador;
- MedalhaDao: Analogamente a classe anterior, a classe MedalhasDao Torna possui a persistência de dados na tabela Medalhas. Ela contém métodos bastante semelhantes ao da classe JogadorDao, quanto a sua execução. Sendo que, a diferença primordial entre as duas é o tipo de dado persistido, e a tabela para sua persistência.

3.3. Banco de dados

O banco da aplicação, ilustrado na Figura 6, dispõe de duas tabelas, Jogadores e Medalhas. A tabela jogador possui as informações referentes ao usuário, como nome e pontuação. Já a tabela Medalhas possui informações sobre as medalhas dos usuários, como o número de erros e consultas para conquistá-la. Esta possui ainda uma variável booleana que informa se a mesma foi ou não conquistada, além de um identificador para o jogador ao qual esta medalha está associada.

Figura 6 – Modelo do Banco de Dados da Aplicação

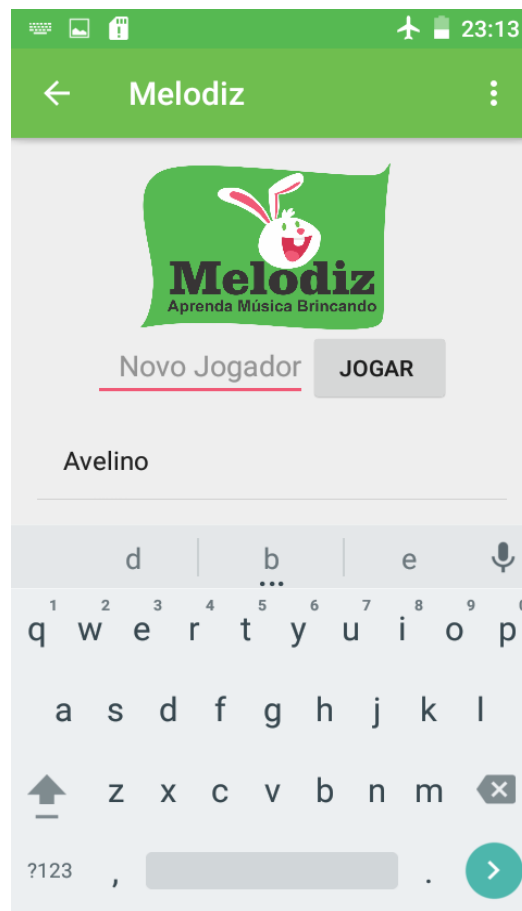
TABELAS			
JOGADORES		MEDALHAS	
VARIÁVEL	TIPO	VARIÁVEL	TIPO
id	long	idMedalha	long
nome	String	idJogador	long
modulo	int	conquistada	bool
pontuacao	int	tentativas	int
		consultas	int

Fonte: Autoria própria

3.4. Interface Gráfica

A interface gráfica do aplicativo foi desenvolvida em *eXtensible Markup Language* (XML), padrão para o *Android*. Cada uma das telas possui um arquivo de *layout* específico, além de arquivos especiais, como menus e listas. O aplicativo possui quatro telas principais: Tela de *Login*, de Jogo, Galeria de Medalhas e *Ranking*, exemplificadas nas Figuras 6, 7, 8 e 9, respectivamente.

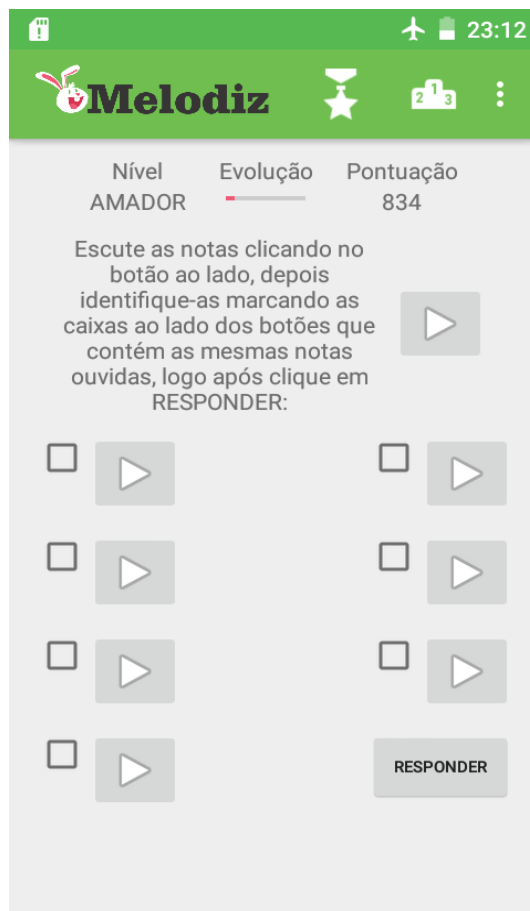
Figura 7 – Tela inicial do aplicativo Melodiz



Fonte: Autoria própria

A tela de *login* contém a logomarca do aplicativo, um campo de cadastro para novos usuário e uma lista de usuários já cadastrado. Esta tela possui um *layout* adicional para criação de uma barra superior customizada. Além do título do aplicativo, ela possui um campo superior para informações sobre o aplicativo.

Figura 8 – Tela de jogo do aplicativo Melodiz



Fonte: Autoria própria

A tela de jogo possui informações pertinentes ao jogador quanto a sua evolução. Abaixo destas temos o enunciado da questão atual e as teclas que representam as notas musicais. A tela contém ainda as caixas para a seleção da nota correspondente a resposta, e a tecla responder. A barra também é customizada, garantindo não apenas acesso às informações do aplicativo, mas também à tela de medalhas conquistadas e à de *ranking*.

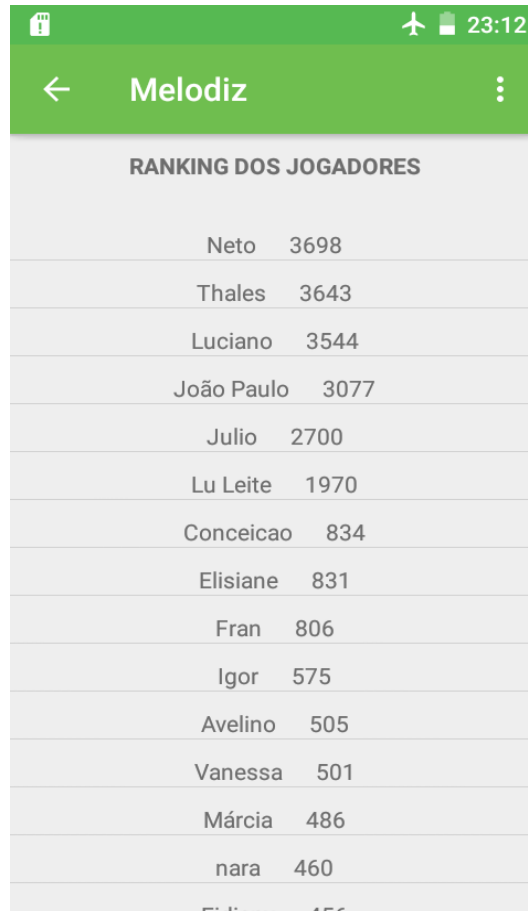
Figura 9 – Galeria de medalhas do aplicativo Melodiz



Fonte: Autoria própria

A tela de medalhas conquistadas apresenta um botão para cada uma das medalhas a serem conquistadas nos desafios. A barra superior, além de informações sobre o aplicativo, disponibiliza o retorno à tela de jogo.

Figura 10 – *Ranking* do aplicativo Melodiz



RANKING DOS JOGADORES	
Neto	3698
Thales	3643
Luciano	3544
João Paulo	3077
Julio	2700
Lu Leite	1970
Conceicao	834
Elisiane	831
Fran	806
Igor	575
Avelino	505
Vanessa	501
Márcia	486
nara	460
Felipe	456

Fonte: Autoria própria

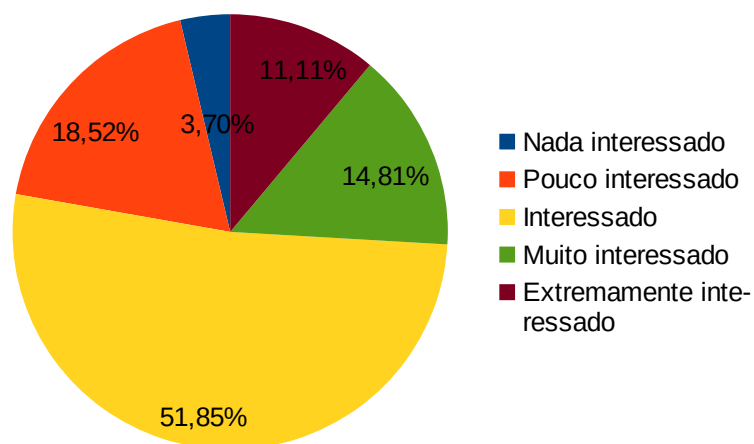
A tela de *ranking* apresenta uma lista dos jogadores cadastrados, organizados, de forma decrescente, por pontuação. Esta lista é composta por outro arquivo customizado de *layout*, para garantir a exibição apenas das informações desejadas de cada jogador. A barra superior também disponibiliza informações sobre o aplicativo e o retorno à tela de jogo.

3.5. Análise de Satisfação de Uso

A satisfação de uso do projeto foi medida usando duas técnicas, um formulário baseado na escala *likert*, que organiza as respostas em um nível de concordância com a afirmação proposta, presente no APÊNDICE A e entrevista oral indutiva. Os usuários consultados possuíam diferentes níveis de envolvimento musical, buscando garantir perfis diferentes de possíveis usuários. Eles responderam perguntas objetivas sobre a execução do software e eram convidados a tecer comentários, apontando pontos positivos e negativos.

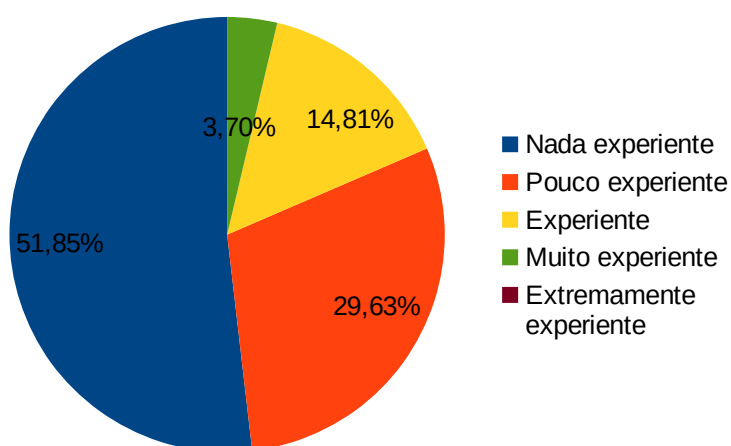
Analisando o contato dos entrevistados com o meio musical, entre os 27 participantes da pesquisa, 25,93% eram muito ou extremamente interessados no estudo da teoria musical, enquanto 18,52 % eram pouco, ou nada interessados, como apresentado na Figura 10. Na Figura 11 pode ser observada a porcentagem de entrevistados que tiveram pouca ou nenhuma experiência com aplicativos para o desenvolvimento musical. Os mesmos alcançaram a marca de 81,48 % dos entrevistados. Sendo que, deste grupo, 14,81 % são muito ou extremamente interessados no assunto, evidenciando uma baixa procura dessas ferramentas, devido, talvez, a proposta das mesmas.

Figura 11 – Nível de interesse dos entrevistados no estudo musical



Fonte: Imagem gerada através do *LibreOffice Calc*

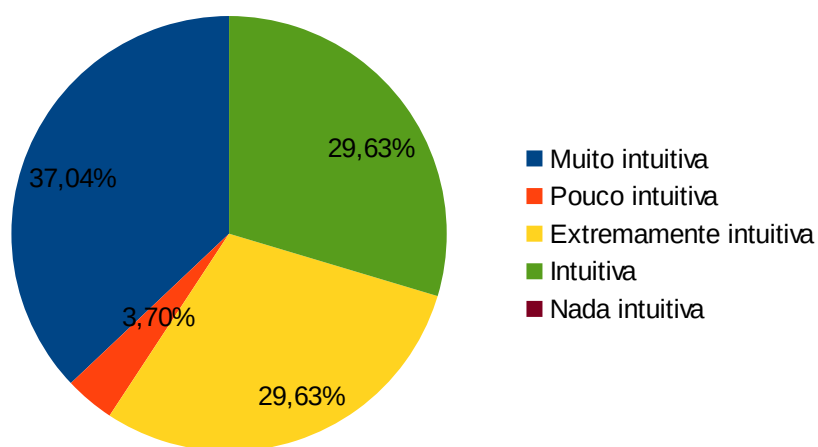
Figura 12 – Nível de experiência dos entrevistados com ferramentas digitais de ensino musical



Fonte: Imagem gerada através do *LibreOffice Calc*

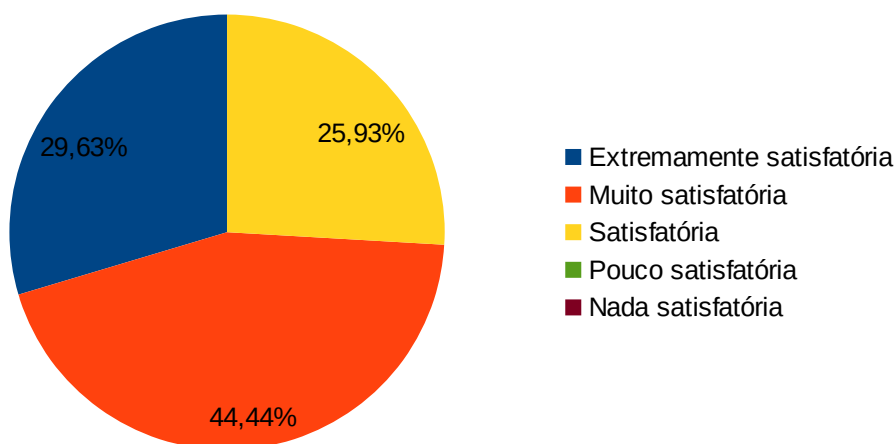
No que diz respeito a execução do aplicativo, 66,67 % dos entrevistados acharam a interface gráfica muito ou extremamente intuitiva, como demonstrado na Figura 12. Sobre a curva de dificuldade apresentada pelo jogo, como observado na Figura 13, os usuários que a classificaram como muito ou extremamente satisfatória alcançaram 70,31%. Já a classificação da teoria musical apresentada, como exemplificado na Figura 14, foi dita como muito ou extremamente fácil por 51,85 %, aonde, 15,38% destes se classificaram como pouco ou nada interessados no conteúdo.

Figura 13 – Avaliação da capacidade intuitiva do aplicativo Melodiz



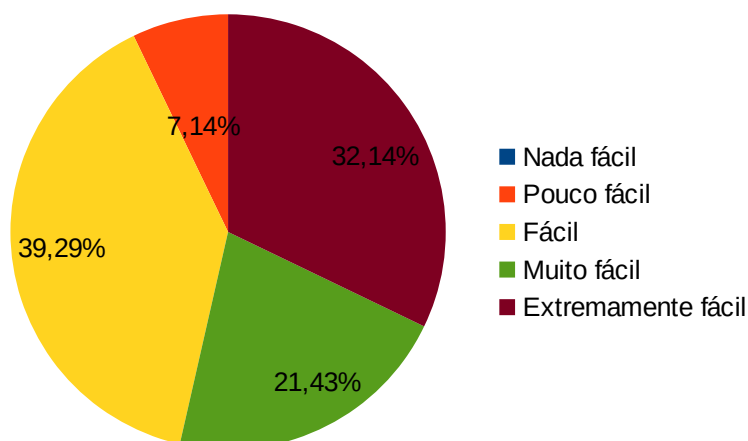
Fonte: Imagem gerada através do *LibreOffice Calc*

Figura 14 – Nível de satisfação quanto a curva de dificuldade do Melodiz



Fonte: Imagem gerada através do *LibreOffice Calc*

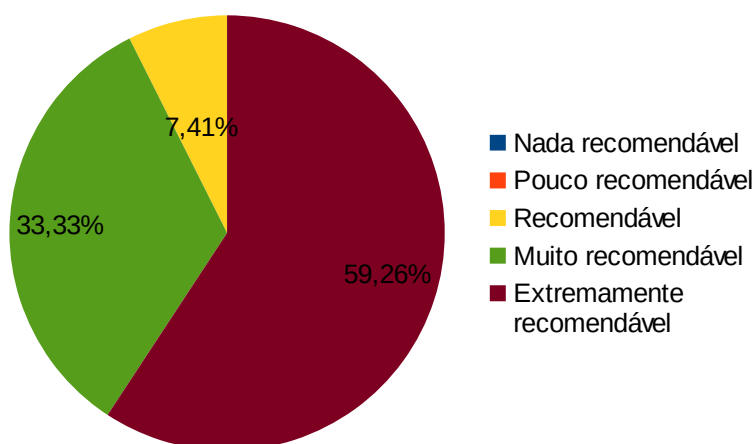
Figura 15 – Grau de facilidade da teoria musical apresentada no Melodiz



Fonte: Imagem gerada através do *LibreOffice Calc*

Ao serem questionados sobre o nível recomendação deste aplicativo para iniciantes no estudo da teoria musical, 92,59% dos entrevistados, como pode ser observado na Figura 15, o julgaram muito ou extremamente recomendável. Todos usuários que afirmaram ter um nível de experiência médio ou superior (experiente, muito experiente e extremamente experiente) indicaram esse grau elevado de recomendação. Fato que ajuda a validar a eficiência do trabalho desenvolvido.

Figura 16 – Avaliação do potencial de recomendação do Melodiz



Fonte: Imagem gerada através do *LibreOffice Calc*

Em relação os entrevistados oralmente, todos apontaram características similares do projeto, em relação a sua execução e área de atuação. Foi sugerida sua aplicação no ensino de jovens, visando aqueles menos familiarizados com o ambiente musical. Quanto a sua

abordagem gráfica, as opiniões se dividiram. Por um lado, alguns acreditam que a falta de elementos teóricos seja um ponto forte, não alienando novos estudantes. Enquanto outros apontaram que essa introdução de elementos, como notações de partituras e outros, possam aguçar a curiosidade destes jovens.

Um ponto em que todos os entrevistados presencialmente concordaram foi a capacidade de expansão do projeto. Pois assim, aproveitando-se da abordagem lúdica, poderiam ser introduzidos mais tópicos da teoria musical. Além de aderir um nível de dificuldade mais elevado, proporcionando ao usuário chances reduzidas de consulta e erro, para manter um o interesse do mesmo elevado, em relação ao jogo.

4. CONCLUSÃO E TRABALHOS FUTUROS

As soluções que tem por objetivo facilitar o ensino tendem a ter grande procura, principalmente quando abordam uma metodologia de ensino centrada nas experiências individuais de cada aluno. Estas tendem a garantir um processo mais agradável, pois apresentam aos usuários desafios coerentes com sua curva de aprendizado. Esse processo de aprendizado é principalmente indicado em áreas que podem exigir habilidades muito específicas, como é o caso do ensino musical.

Transformar o aprendizado, inclusive o musical, em um processo prazeroso é crucial para que os iniciantes não percam o interesse. As ferramentas que buscam esse propósito acabam não apenas por desenvolver os jovens estudioso, mas também por instigar um público menos interessado no assunto. Ao se portarem como ferramentas para o lazer, essas soluções digitais ajudam a levar para o meio musical para aqueles que ante a dificuldade do mesmo, não permitiram envolver-se.

O projeto proposto obteve êxito, principalmente ao provocar usuários leigos, levando-os, sem perceberem, a uma jornada de descoberta. Estes usuários, presos pela dinâmica do jogo, acabaram por desafiar suas capacidades perceptivas, encarando os erros não como fracassos, mas incentivo para buscarem mais acertos. A tela de *Ranking*, combinada com a competitividade natural do ser humano, levou os usuários a não só buscarem finalizar o jogo, mas arriscarem-se novamente, em busca de um resultado mais satisfatório.

A fase de validação ainda permitiu a identificação de novas funcionalidades a serem inseridas no projeto. Por exemplo, a implantação de um número fixo de tentativas por desafio, bem como um máximo de consultas, visando instigar ainda mais o jogador. O principal ponto abordado pelos usuários é a possibilidade de expansão do projeto. Pois, visto a extensão do campo abordado, seria extremamente proveitosa a inclusão de mais desafios.

BIBLIOGRAFIA

ANDROID. **Developer**. Disponível em: <<https://developer.android.com/develop/index.html>>. Acesso em: 09 Ago 2017.

BORDINI, Rogério Augusto. **Formação De Professores E Tecnologia Digital: Um Estudo Sobre A Utilização Do Jogo Musikinésia Na Educação Musical**, 2015.

BRANCO, Wilson Castello Neto; MATOS, Rodrigo de. **Dinamizando O Desenvolvimento Da Percepção Musical: Um Sistema De Apoio À Realização De Exercícios**, 2009.

GOULART, Diana. **Educação Musical**. 2002. Disponível em: <<http://www.dianagoulart.pro.br/faqs/sobredutec.html>>. Acesso: 01 Jun 2017.

LOSCH, Robson; FORTES, Reinaldo; **Afinador eletrônico**. Disponível em: <<http://www.unipac.br/site/bb/tcc/tcc1c2f77d281853937824f108f3b1b31fb.pdf>>. Acesso em: 25 Jun 2017.

MELO, Fábio Delicato Feijó. **Ferramenta de Apoio ao Ensino de Instrumento Musical**. 2009. 51 p. Monografia (Graduação em Engenharia da Computação) - Universidade de Pernambuco, Pernambuco, 2009.

GITHUB. **mididriver**. Disponível em: <<https://github.com/billthefarmer/mididriver>>. Acesso em: 03 Jul 2017.

PRIOLLI, Maria Luisa de Mattos. **Principios basicos da música para a juventude - vol 1** – 2006. 48ª Edição revista atualizada. Casa Oliveira de Musica Ltda, 2006.

SILVA, Marco Aurélio Botelho da. **Padrão Midi**. 2007. Universidade Federal Fluminense (UFF). Departamento de Engenharia de Telecomunicações da UFF.

YATSUDA, Glauber Aparecido; SCHIAVONI, Flávio Luiz; FILHO, Dante Alves Medeiros; TOFFOLO, Rael Bertarelli Gimenes. **TAW -Treinamento Auditivo pela WEB: Ensino**

Musical a Distância. Departamento de Informática e Departamento de Música – Universidade Estadual de Maringá (UEM), 2011.

APÊNDICE A – FORMULÁRIO DE SATISFAÇÃO DE USO

Nome: _____

1. Qual seu grau de interesse no estudo da música?

- ☐ Extremamente interessado
- ☐ Muito interessado
- ☐ Interessado
- ☐ Pouco interessado
- ☐ Nada interessado

Comentário: _____

2. Você tem experiência com outros aplicativos de ensino musical?

- ☐ Extremamente experiente
- ☐ Muito experiente
- ☐ Experiente
- ☐ Pouco experiente
- ☐ Nada experiente

Comentário: _____

3. O aplicativo possui interface intuitiva?

- ☐ Extremamente intuitiva
- ☐ Muito intuitiva
- ☐ Intuitiva
- ☐ Pouco intuitiva
- ☐ Nada Intuitiva

Comentário: _____

4. Os desafios apresentados tem uma curva de dificuldade satisfatória?

- ☐ Extremamente satisfatória
- ☐ Muito satisfatória
- ☐ Satisfatória
- ☐ Pouco satisfatória

☐ Nada satisfatória

Comentário: _____

5. A teoria (musical) apresentada é de fácil entendimento?

☐ Extremamente fácil

☐ Muito fácil

☐ Fácil

☐ Pouco fácil

☐ Nada fácil

Comentário: _____

6. Considera o aplicativo Melodiz recomendável para os interessados no estudo da teoria musical?

☐ Extremamente recomendável

☐ Muito recomendável

☐ Recomendável

☐ Pouco recomendável

☐ Nada recomendável

Comentário: _____