

**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Colegiado de Engenharia da Computação**



UFOP
Universidade Federal
de Ouro Preto

**Musicália - Concepção e
desenvolvimento de um jogo para
iniciação musical**

Wemerson Geisler de Lima

**TRABALHO DE
CONCLUSÃO DE CURSO**

ORIENTAÇÃO:
Tiago França Melo de Lima

**Julho, 2018
João Monlevade/MG**

Wemerson Geisler de Lima

**Musicália - Concepção e desenvolvimento de
um jogo para iniciação musical**

Orientador: Tiago França Melo de Lima

Monografia apresentada ao curso de Engenharia da
Computação do Departamento de Computação e Sis-
temas da Universidade Federal de Ouro Preto como
requisito parcial para obtenção do grau de Bacharel
em Engenharia da Computação

Universidade Federal de Ouro Preto

João Monlevade

Julho de 2018

L732m Lima, Wemerson Geisler de.
Musicália [manuscrito]: concepção e desenvolvimento de um jogo para
iniciação musical / Wemerson Geisler de Lima. - 2018.

68f.: il.: color; tabs.

Orientador: Prof. MSc. Tiago França Melo de Lima.

Monografia (Graduação). Universidade Federal de Ouro Preto. Instituto de
Ciências Exatas e Aplicadas. Departamento de Computação e Sistemas de
Informação.

1. Jogos de aprender. 2. Jogos educativos. 3. Música - Instrução e estudo. 4.
Teoria musical. 5. Design de jogos. I. Lima, Tiago França Melo de. II.
Universidade Federal de Ouro Preto. III. Título.

CDU: 394.3:780.7

Catálogo: ficha.sisbin@ufop.edu.br



UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS
COLEGIADO DO CURSO DE ENGENHARIA DE COMPUTAÇÃO



ANEXO IV – Folha de Aprovação
Curso de Engenharia de Computação

FOLHA DE APROVAÇÃO DA BANCA EXAMINADORA

Musicália - Concepção e desenvolvimento de um jogo para iniciação musical

Wemerson Geisler de Lima

Monografia apresentada ao Instituto de Ciências Exatas e Aplicadas da Universidade Federal de Ouro Preto como requisito parcial da disciplina CSI496 – Trabalho de Conclusão de Curso II do curso de Bacharelado em Engenharia de Computação e aprovada pela Banca Examinadora abaixo assinada:

Prof. [Me. Tiago França Melo de Lima]
[DECSI] - [UFOP]
Professor Orientador

Prof. [Dra. Gilda Aparecida de Assis]
[DECSI] - [UFOP]
Professor Convidado

Prof. [Dr. George Henrique Godim da Fonseca]
[DECSI] - [UFOP]
Professor Convidado

João Monlevade, 12 de Julho de 2018



ATA DE DEFESA

Aos 12 dias do mês de Julho do ano de 2018, às 19h00 na sala H102 do Instituto de Ciências Exatas e Aplicadas, foi realizada a defesa de Monografia pelo aluno **WEMERSON GEISLER DE LIMA**, sendo a Comissão Examinadora constituída pelos professores: Me. Tiago França de Melo Lima, Dra. Gilda Aparecida de Assis e Dr. George Henrique Godim da Fonseca.

O candidato apresentou a monografia intitulada: "*Musicália - Concepção e desenvolvimento de um jogo para iniciação musical*". A Comissão Examinadora deliberou, por unanimidade, pela aprovação do candidato com nota 9.5 (NOVE PONTOS E MEIO), concedendo-lhe o prazo de 15 dias para incorporações das alterações sugeridas no texto final.

Na forma regulamentar, foi lavrada a presente ata, que é assinada pelos membros da Comissão Examinadora e pelo graduando.

João Monlevade, 12 de Julho de 2018.

Prof. [Me. Tiago França Melo de Lima]
[DECSI] - [UFOP]
Professor Orientador

Prof. [Dra. Gilda Aparecida de Assis]
[DECSI] - [UFOP]
Professor Convidado

Prof. [Dr. George Henrique Godim da Fonseca]
[DECSI] - [UFOP]
Professor Convidado

[Wemerson Geisler de Lima]
Graduando

*Este trabalho é dedicado à Deus, minha família e aos meus amigos,
por sempre acreditarem e me darem a coragem necessária
para enfrentar os desafios diários e nunca desistir.*

Agradecimentos

A **Deus**, toda honra, toda glória e todo louvor.

Agradeço aos meus pais amados, **Agamenon e Eliene**, vocês são os melhores pais que eu poderia ter na vida. Aos meus irmãos, **Anderson, Walison** e a **Jéssica**, por ser uma companheira de todas as horas, por sempre me motivar nos momentos mais difíceis pelo incentivo de cada dia e por ser a melhor irmã do mundo. e à toda a minha família por serem a melhor base de caráter que eu poderia ter durante a minha trajetória, por terem me ajudado a me tornar o homem que sou, pelo carinho e o enorme amor que nos une todos os dias, por toda força que me deram para chegar até aqui.

Agradeço aos meus amigos queridos, que sempre motivaram a continuar seguindo para o alto e avante! Mesmo naqueles momentos que eu pensei que não daria certo, vocês foram fundamentais na minha formação, não só acadêmica, mas de vida. Obrigado.

Agradeço ao Fabrício e Leonardo que abriram as portas para que eu pudesse trabalhar em um período de horários complicadíssimos na Universidade, mas nunca deixaram de me apoiar e de me dar a oportunidade no mercado de trabalho durante todos esses anos, minha gratidão a vocês.

Agradeço ao meu orientador, Prof. Tiago França Melo de Lima, por me motivar a concluir este trabalho, desde que cursei IHC sempre soube que seria um grande orientador, obrigado pela paciência, pelos questionamentos e ideias que me fizeram crescer e evoluir, por agir sempre como um membro da equipe, e por sempre acreditar no potencial dos projetos.

Agradeço aos meus amigos do JFC, todos vocês são indispensáveis em minha vida.

Agradeço aos meus amigos do LEDES por terem sido amizades incríveis, pessoas que me ajudaram muito a evoluir e a ter maior conhecimento e ser um cara apaixonado pela criação de Game, desde o início eu sabia que era o melhor lugar de extensão da UFOP, obrigado por todo o apoio e companheirismo durante esses últimos semestres.

Agradeço à todos e a cada um dos que conviveram comigo durante todos esses anos de graduação, meus colegas de turma que passaram por tudo isso junto comigo, aos amigos que fiz, pelas madrugadas de estudo, pelo incentivo que era recíproco a todo instante e companheirismo.

Por fim, agradeço à Universidade Federal de Ouro Preto e aos professores que fizeram parte dessa jornada, pelo ensino gratuito e de qualidade, ser parte disso foi mais um sonho realizado.

*“Tudo o que é verdadeiro, tudo o que é honesto, tudo o que é justo,
tudo o que é puro, tudo o que é amável, tudo o que é de boa fama,
se há alguma virtude, e se há algum louvor, nisso pensai.”*

(Filipenses 4:8)

Resumo

A música está presente em diversos estágios da nossa vida, desde a infância à vida adulta. Ela faz parte do nosso dia-a-dia e está presente nas diversas culturas. No Brasil, a educação musical é conteúdo obrigatório do componente curricular Arte. Mas sua efetiva adoção se torna um desafio diante da escassez de recursos básicos tais como instrumentos musicais e profissionais especializados. As tecnologias de informação e comunicação podem ser utilizadas para minimizar esse problema, como por exemplo, o uso de jogos sérios voltados para apoiar atividades de ensino-aprendizagem. Motivação e engajamento são alguns dos benefícios geralmente esperados pelo uso de jogos na educação. E eles também podem ser empregados no ensino de música. No entanto, há uma carência de jogos nessa área. Esse trabalho propôs a concepção e desenvolvimento de um jogo sério voltado para iniciação musical, visando introduzir, de forma lúdica, conceitos básicos de teoria musical. No desenvolvimento foram utilizados princípios e práticas de metodologias ágeis, com ciclos curtos de concepção, design, construção (prototipação / codificação), avaliação e implantação (lançamento de versão). Os resultados incluem documentos de *design* do jogo e uma versão de demonstração que pode ser utilizada em processos de avaliação. Os trabalhos futuros incluem implementar novas funcionalidades e melhorias de usabilidade e jogabilidade, avaliar o jogo como ferramenta pedagógica e disponibilizar uma versão completa em lojas de aplicativos.

Palavras-chaves: jogos sérios, jogo educativo, projeto de jogo, desenvolvimento de jogo, educação musical, música, teoria musical, Musicália.

Abstract

Music is present in various stages of our lives, from childhood to adulthood. It is part of our daily lives and is present in many cultures. In Brazil, music education is an obligatory content of the Art curriculum component. However, its effective adoption becomes a challenge due to the scarcity of essential resources such as musical instruments and specialized professionals. Information and communication technologies can be used to mitigate this problem; for example, the use of serious games aimed at supporting teaching-learning activities. Motivation and engagement are some of the benefits generally expected from the use of games in education. Thereby, they can also be employed to support teaching music. However, there is a shortage of games in this area. This work proposed the design and development of a serious game aimed at musical initiation, to introduce, in a playful way, some basic concepts of musical theory. In the development, principles and practices of agile methodologies were used, with short cycles of design, design, construction (prototyping/coding), evaluation and deployment (release version). The results include game design documents and a demo version which can be evaluated in educational settings. Future work includes the development of new features, improvements on usability and gameplay, assessment of the game as a pedagogical tool, and the release of a full version in application stores.

Key-words: serious games, educacional game, game design, game development, music education, music, music theory, Musicália.

Lista de ilustrações

Figura 1 – Interface do jogo <i>MusicBlox</i>	21
Figura 2 – Interface do jogo <i>Mission to Magmanon</i>	22
Figura 3 – Exemplo de utilização do jogo <i>Simply Piano</i>	23
Figura 4 – Exemplo de protótipo de baixa fidelidade criado para especificação e discussão de ideias	25
Figura 5 – Utilização da ferramenta <i>Mockup.io</i> para projetar e analisar o fluxo de interação entre as interfaces do aplicativo	26
Figura 6 – Interface gráfica do Unity (Versão: 2018.1.6f1) (UNITY, 2018a)	27
Figura 7 – Trecho de arquivo <i>JSON</i> para representação da música "Brilha, brilha estrelinha"(título original: <i>Twinkle Twinkle Little Star</i> , autor: Wolfgang Amadeus Mozart).	28
Figura 8 – Utilização do <i>plugin ZenHub</i> integrado ao <i>GitHub</i> , no <i>Google Chrome</i>	29
Figura 9 – Esboço dos cenários do Reino da Melodia (imagem superior) e Reino da Harmonia (imagem inferior). Arte: Danilo Caldeira	33
Figura 10 – Cenários do Reino da Melodia (imagem superior) e Reino da Harmonia (imagem inferior). Arte: Danilo Caldeira	33
Figura 11 – Esboço dos personagens Batuque (Reino do Ritmo) e Lili (Reino da Melodia). Arte: Dasayeve Xavier	34
Figura 12 – Personagens Batuque (Reino do Ritmo) e Lili (Reino da Melodia). Arte: Dasayeve Xavier	34
Figura 13 – Protótipo de baixa fidelidade ilustrando a ideia inicial do fluxo de interação entre as interfaces do jogo. Arte: Danilo Caldeira	35
Figura 14 – Protótipo que ilustra o fluxo de interação entre interfaces do jogo. Artes: Danilo Caldeira e Dasayeve Xavier	36
Figura 15 – Criação da interface principal do jogo, de forma evolutiva, a partir de protótipos de baixa fidelidade. Arte: Dasayeve Xavier	36
Figura 16 – Protótipo de baixa fidelidade da fase: Ritmo no teclado. Arte: Wemerson Geisler	39
Figura 17 – Protótipo de baixa fidelidade da fase: Notas no teclado. Arte: Wemerson Geisler	42
Figura 18 – Protótipo de baixa fidelidade da fase: Melodia no teclado. Arte: Wemerson Geisler	45
Figura 19 – Protótipo de baixa fidelidade da fase: Acordes no teclado. Arte: Wemerson Geisler	48
Figura 20 – Protótipo de baixa fidelidade da fase: Notas na pauta. Arte: Wemerson Geisler	51

Figura 21 – Protótipo de baixa fidelidade da fase: Figuras de tempo. Arte: Ana Carolina Brandão	53
Figura 22 – Implementação do efeito <i>parallax scrolling</i> no Reino da Melodia, envolvendo atividades de programação e arte. Arte: Danilo Caldeira	54
Figura 23 – a) Tela inicial do jogo (Arte: Dasayeve Xavier). b) Mapa do mundo mágico de Musicália (Arte: Danilo Caldeira)	54
Figura 24 – a) Tela Missão onde o jogador recebe a missão atual e é redirecionado para o reino da próxima fase. - Arte: Dasayeve Xavier	55
Figura 25 – Cadeado indica fase bloqueada e o numero de estrelas no cadeado indica a quantidade necessária para desbloquear a fase. - Arte: Danilo Caldeira	55
Figura 26 – Após acessar o reino e abrir uma fase, o jogador recebe instruções sobre seu objetivo. - Cenário: Danilo Caldeira - Personagem: Dasayeve Xavier - ícones: Wemerson Geisler	56
Figura 27 – Missão do reino do ritmo, onde o jogador deve pressionar a tecla correta do teclado no momento certo para acompanhar o tempo da música a ser tocada - Arte: Danilo Caldeira	56
Figura 28 – Missão do reino das notas, onde o jogador deve pressionar a tecla correspondente à nota exibida na tela e tocar a música corretamente - Arte: Danilo Caldeira	57
Figura 29 – Missão do reino da Pauta, onde o jogador deve pressionar a tecla correspondente à nota exibida na figura da partitura e tocar a música corretamente. - Arte: Danilo Caldeira	57
Figura 30 – Missão do reino do ritmo, onde o jogador deve associar o tempo correto de acordo com a figura de tempo apresentada na tela - Arte: Danilo Caldeira	58
Figura 31 – Missão do reino da harmonia, onde o jogador deve compor os acordes corretamente de acordo com a notação apresentada na tela. - Arte: Danilo Caldeira	58
Figura 32 – Missão do reino da melodia, onde o jogador deve seguir as instruções das setas e tocar a melodia da musica indicada. - Arte: Danilo Caldeira	59

Lista de abreviaturas e siglas

GDD	<i>Game Design Document</i>
ICEA	Instituto de Ciências Exatas e Aplicadas
LEDS	Laboratório de Engenharia e Desenvolvimento de Sistemas
UFOP	Universidade Federal de Ouro Preto
UFSCar	Universidade Federal de São Carlos

Sumário

1	INTRODUÇÃO	15
1.1	Objetivos	17
1.2	Organização do trabalho	17
2	CONCEITOS BÁSICOS E TRABALHOS RELACIONADOS	18
2.1	Educação musical	18
2.2	Jogos sérios	19
2.3	Educação musical através de jogos e aplicativos	21
3	METODOLOGIA	24
3.1	Métodos	24
3.2	Ferramentas	25
3.2.1	Especificação, projeto e documentação	25
3.2.2	Desenvolvimento	26
3.2.3	Gerenciamento	28
4	RESULTADOS	30
4.1	<i>Game Design</i>	30
4.1.1	Estória do jogo	30
4.1.2	Reinos de Musicália (cenários)	31
4.1.3	Personagens	32
4.1.4	Projeto de interface e interação	35
4.1.5	Minijogos	35
4.1.5.1	Ritmo no teclado	37
4.1.5.2	Notas no teclado	39
4.1.5.3	Melodia no teclado	42
4.1.5.4	Acordes no teclado	45
4.1.5.5	Notas na pauta	48
4.1.5.6	Figuras de tempo	51
4.2	Musicália: versão demo	54
5	CONSIDERAÇÕES FINAIS	60
	REFERÊNCIAS	61

ANEXOS	66
ANEXO A – CONCEITOS MUSICAIS	67

1 Introdução

No Brasil, conforme a Lei de Diretrizes e Bases da Educação e as Diretrizes Curriculares Nacionais para Educação Básica, a música é componente curricular obrigatória (BRASIL, 2008; BRASIL, 2013). No entanto, a educação musical ainda é pouco discutida no ensino básico e um dos motivos é a carência de recursos essenciais como instrumentos musicais e professores especializados (QUEIROZ, 2013; PENNA, 2014; QUEIROZ; MARINHO, 2014). A educação musical possui um importante papel na educação formal do indivíduo, e pode ser analisada sobre diferentes valores: estético, social, psicológico, tradicional (COUTO; SANTOS, 2009).

“Reconhecendo música como uma linguagem e com possibilidades variadas de exploração no contexto criativo, faz-se ela, de suma importância no currículo escolar. Música como prática de linguagem proporciona um ambiente criativo para que os alunos aprendam, explorando suas variadas possibilidades de experiências afetivas e sociais, além de desenvolver a sensibilidade musical .” (BRÉSCIA, 2003, p.18)

As tecnologias de informação e comunicação podem ser utilizadas como ferramenta de apoio na educação musical. Algumas iniciativas têm sido desenvolvidas nesse sentido, e incluem o desenvolvimento de softwares teóricos-práticos tanto para o público infantil quanto adulto, e voltados para diferentes atividades e instrumentos (KRÜGER, 2014). São exemplos softwares para criação e edição de partituras como o *MuseScore* (MUSESCORE, 2018), *Guitar Pro* (GUITARPRO, 2018) e *Encore* (ENCORE, 2018). Algumas iniciativas são desenvolvidas através de projetos de pesquisa, tais como o *Sistema de Treinamento Rítmico* (STR) (FRITSCH; VICCARI; MORAES, 1998), *Portal EduMusical* (FICHEMAN et al., 2004) e *Editor Musical* (FICHEMAN et al., 2003). Outras envolvem também o desenvolvimento de hardware específico, como por exemplo o *Ambiente Sonoro Ubíquo*, que é composto por hardware e software para permitir a produção sonora de forma simples, possibilitando que atividades educacionais sejam realizadas pelos estudantes a partir de toques no próprio corpo (SANTOS; FILIPPO; PIMENTEL, 2015).

Os jogos eletrônicos também podem ser usados como recursos pedagógicos. Eles podem ser empregados para tornar o processo de ensino-aprendizagem mais interativo, criando experiências capazes de promover uma aprendizagem mais efetiva e duradoura (ARANHA, 2006). Quando possuem um objetivo além do entretenimento, eles são denominados jogos sérios (MICHAEL; CHEN, 2005a). Quando voltados para a educação, mesmo que o entretenimento não seja o propósito primário, é necessário oferecer ao jogador uma experiência rica e interessante - caso contrário, não haverá estímulo à sua utilização e não serão obtidos os resultados esperados na aprendizagem. No projeto de jogos educativos,

além de aspectos como o design do jogo em si, outro aspecto importante é a definição do conteúdo que será abordado e a forma isso será feito.

Pode-se encontrar exemplos de jogos sérios voltados para diferentes finalidades (DONDLINGER, 2007), tais como aprendizagem de idiomas, conscientização sobre a dengue (LIMA et al., 2017), alfabetização de crianças com síndrome de Down (FARIAS et al., 2013). Quando empregados como recurso pedagógico, para apoiar o ensino-aprendizagem, eles podem abranger diversas áreas do conhecimento e ser destinados para diferentes públicos-alvo – desde crianças à idosos. São exemplos os jogos *Castelo dos enigmas*, voltado para o ensino de programação para alunos do ensino médio (SCAICO et al., 2012); *Tabuada Legal*, para ensino de multiplicações (CARDOSO; GIRALDELLO; M, 2013); e o *ABCLingo*, para alfabetização de crianças (KARLINI; RIGO, 2014)

Alguns exemplos de jogos voltados para o ensino de música também podem ser encontrados. No jogo *Uma Odisseia de Acordes* (SANTIAGO et al., 2015), o usuário deve identificar as notas que compõem os acordes, e desse modo, espera-se auxiliar um iniciante no estudo de música a compreender o nome e a representação das notas. O *Musikinésia* (BORDINI et al., 2016) tem como objetivo introduzir os conceitos da partitura musical e melodia utilizando-se de um teclado virtual. O *Incrível Músico das Neves* (SANTIAGO et al., 2014) aborda alguns conceitos básicos de teoria musical, como por exemplo intervalos harmônicos (que geram acordes que produzem sensação agradável ao ouvido). O *Musical Ear* é um jogo desenvolvido para treinar o “ouvido musical” e preparar o jogador para identificar melhor alguns dos principais acordes e escalas usados na música (MUSICALEAR, 2016). O *Simply Piano* (JOYTUNES, 2018), por sua vez, possui como objetivo ensinar teclado ou piano através da interação com instrumentos reais - um piano ou teclado físico pode ser utilizado como controle para a leitura e execução de partituras. No entanto, não conseguimos encontrar nenhum exemplo no formato de um jogo sério, que fosse voltado para iniciação musical e que permitisse, de forma lúdica, a aprendizagem dos conceitos básicos de teoria musical. Uma proposta nesse sentido foi elaborada no jogo *Musical Instructor* (BRANDÃO, 2016), que tinha como objetivo apoiar o ensino-aprendizagem de conceitos sobre música e teoria musical. No entanto, os resultados obtidos não foram avaliados ou validados em um contexto de ensino-aprendizagem, e a versão desenvolvida apresenta diversas limitações em termos de *game design* (história, personagens, sistema de recompensas, *design* de níveis, mecânica do jogo), jogabilidade, usabilidade e experiência de uso.

Nesse contexto, esse trabalho propõe a concepção, projeto e desenvolvimento de um jogo sério voltado para crianças e pré-adolescentes, visando apoiar a iniciação musical, abordando conceitos como ritmo, notas musicais, escala e partitura. É um trabalho interdisciplinar que envolve diferentes áreas: jogos sérios, teoria musical, ensino de música, *design* e avaliação de usabilidade e de experiência de uso.

1.1 Objetivos

O trabalho tem como objetivo geral conceber, projetar e desenvolver um jogo sério visando apoiar a iniciação musical. Dessa forma, fazem parte do escopo a elaboração e revisão de documentos relacionados ao desenvolvimento de jogos (ex. *game design document*), construção e avaliação de protótipos, e a implementação de uma versão de demonstração do jogo. O trabalho possui os seguintes objetivos específicos:

- elaboração do documento de *design* do jogo, que inclui informações tais como história, roteiro, personagens, regras, fluxos e estilos de interação, mecânica do jogo;
- concepção, *design*, desenvolvimento e avaliação de protótipos;
- disponibilização e publicação de uma versão de demonstração no formato de aplicativo para dispositivos móveis.

1.2 Organização do trabalho

O trabalho está organizado em cinco capítulos. O segundo capítulo apresenta conceitos básicos necessários ao entendimento do texto e trabalhos relacionados. No terceiro capítulo são descritas as etapas que foram utilizadas para a realização do trabalho. No quarto capítulo são apresentados e discutidos os resultados do trabalho. Por fim, no último capítulo são feitas as considerações finais e mencionados possíveis trabalhos futuros.

2 Conceitos básicos e trabalhos relacionados

Nesse capítulo são apresentados conceitos básicos sobre educação musical e jogos sérios, e exemplos de aplicativos e jogos sobre música.

2.1 Educação musical

A música faz parte de diversas fases da vida humana e desempenha um importante papel em nosso desenvolvimento (CORREA, 2013; GOHN; STAVRACAS, 2010; CHIARELLI; BARRETO, 2005). No Brasil, a Lei 11.769/2008 alterou a Lei de Diretrizes e Bases da Educação para dispor sobre a obrigatoriedade do ensino da música na educação básica (BRASIL, 2008). E conforme as Diretrizes Curriculares Nacionais para Educação Básica (BRASIL, 2013), a educação musical é conteúdo obrigatório como parte do componente curricular Arte.

A educação musical tem como objetivo a musicalização - tornar o indivíduo sensível e receptivo ao fenômeno sonoro (GAINZA, 1988). E além de contribuir para a formação musical dos alunos, o ensino de música para crianças e jovens pode ser uma ferramenta eficiente de transformação social, na medida em que o ambiente de ensino e aprendizagem pode proporcionar o respeito, a amizade, a cooperação e a reflexão, importantes e necessários para a formação humana (SOUZA; JOLY, 2010). É importante ressaltar que o ensino de música em escolas regulares não tem a intenção de formar músicos profissionais (PROCOPIO, 2015).

"O objetivo primeiro da educação musical é facilitar o acesso à multiplicidade de manifestações musicais da nossa cultura, bem como possibilitar a compreensão de manifestações musicais de culturas mais distantes. Além disso, o trabalho com música envolve a construção de identidades culturais de nossas crianças, adolescentes e jovens e o desenvolvimento de habilidades interpessoais. (LIANE; BEN, 2003, p.179)

São vários os elementos que podem ser abordados no contexto da educação musical, como por exemplo o som, a harmonia, a melodia, o ritmo, o timbre e a letra.

No entanto, a falta de recursos essenciais, como instrumentos musicais e professores qualificados na área, faz com que a educação musical seja pouco discutida no ensino básico (FICHEMAN et al., 2004). Apesar disso, é gradativo o aumento do número de trabalhos brasileiros que manifestam interesse e indicam o uso de tecnologias digitais como ferramenta de apoio às atividades de ensino de música (CARVALHO, 2015).

2.2 Jogos sérios

É possível encontrar na literatura diferentes definições sobre o que é um jogo. Por exemplo, de acordo com [Rollings e Adams \(2010\)](#), um jogo é um tipo de atividade lúdica, conduzida no contexto de uma realidade imaginada, fictícia, na qual o(s) participante(s) tenta(m) atingir pelo menos um objetivo arbitrário e não trivial, agindo de acordo com as regras. [CRAWFORD \(1982\)](#) define um jogo como sendo um sistema fechado, formal, que subjetivamente representa um subconjunto de realidade. Segundo ([ROGERS, 2014](#)), um jogo é uma atividade que requer pelo menos um jogador, possui regras e possui condição de vitória.

Há diferentes tipos e classificações que podem ajudar a categorizar os jogos. Por exemplo, podem ser físicos ou digitais. Os jogos cujo propósito primário é algo que não é o entretenimento, são denominados jogos sérios ([MICHAEL; CHEN, 2005b](#); [DJAOUTI et al., 2011](#)). Eles podem ser empregados na educação, indo além das vídeo-aulas e livros, possibilitando aos estudantes não apenas o aprendizado, mas a experiência e aplicação dos conteúdos absorvidos ([MICHAEL; CHEN, 2005b](#)). Os jogos sérios podem ser utilizados como recurso didático ou complementar, paradidático, a depender da forma como serão inseridos no planejamento pedagógico e atividades de ensino. E os educadores possuem um papel fundamental, pois é através do contexto, reflexão crítica e intervenções que os jogos poderão ou não contribuir para o desenvolvimento dos jogadores e a construção da aprendizagem ([GRÜBEL; BEZ, 2006](#)). A aprendizagem baseada em jogos visa o engajamento do estudante na atividade de tal modo que ele alcance os objetivos de aprendizagem, e os jogos sérios favorecem, ainda, a aprendizagem fora do ambiente escolar e permitem ao estudante aprender com os erros sem os custos negativos que teriam no mundo real ([BACHEN; RAPHAEL, 2011](#)). O design de jogos sérios voltados para educação tem sido amplamente adotado, abordando diferentes temas e áreas do conhecimento (alfabetização, conceitos físicos, saúde pública) e voltado para diferentes públicos-alvo (crianças, jovens, adultos e idosos).

Projetar sistemas computacionais interativos, como jogos eletrônicos, não é uma tarefa trivial - é fundamental que se leve em conta de que forma, onde e por quem serão utilizados ([ROGERS; SHARP; PREECE, 2013](#)). Desenvolver um jogo digital envolve atividades de várias disciplinas, e vai além da 'simples' programação do software. Inicialmente, os jogos eram concebidos por um pequeno grupo, compostos basicamente de programadores e eram concluídos de forma rápida, mas a evolução tecnológica trouxe novas possibilidades e desafios ([RODRIGUES; MACHADO; VALENÇA, 2010](#)). Em geral, projetos de jogos envolvem equipes de várias áreas desempenhando diferentes funções (ex. *designer*, programador, artista, produtor, testador, compositor, redator). O tamanho da equipe e da duração do projeto varia de acordo com a complexidade do jogo e recursos disponíveis. Vários processos e metodologias têm sido propostas para apoiar o

desenvolvimento de jogos (RODRIGUES; MACHADO; VALENÇA, 2010).

Um artefato de referência comumente utilizado é o *Game Design Document* (GDD), ou Documento de Projeto do Jogo. Ele documenta o jogo e contém diversas informações que irão auxiliar a equipe durante o desenvolvimento, como por exemplo, história do jogo, personagens, regras, mecânica, interação. Geralmente é desenvolvido de forma contínua e incremental ao longo do projeto. Alguns desafios no design e desenvolvimento de jogos para a aprendizagem, elencados por Protosaltis et al. (2011), são: atender as expectativas dos usuários e instrutores; encontrar o balanço entre aprendizagem e diversão / engajamento; como oferecer uma experiência significativa; como envolver a parte emocional do jogador; aparência visual. Um aspecto importante é a necessidade de conhecer os potenciais usuários / jogadores, para que seja possível identificar suas expectativas, definir e atender os requisitos em termos de usabilidade, interação, e jogabilidade, por exemplo (BARBOSA; SILVA, 2010).

A prototipação é uma boa técnica para apoiar o *design* e avaliação de propostas de *design* no processo de implementação de sistemas interativos, como jogos digitais. As etapas do *framework* proposto por Falcão e Gomes (2006), inspirado no design interativo e centrado no usuário e voltado para o desenvolvimento de interfaces tangíveis educacionais, podem também ser empregadas no desenvolvimento de jogos como aquele proposto por esse projeto:

- **Fase 1 - Análise de Competidores:** é fundamental pesquisar e conhecer o campo a ser desenvolvido, reavaliar propostas com base em estudos em uma mesma área de pesquisa.
- **Fase 2 - Compreensão do Contexto:** consiste em conhecer e compreender o contexto de uso do futuro artefato, extraindo elementos de contexto e requisitos gerais, através dos métodos de observação e entrevistas semi-estruturadas.
- **Fase 3 - Criação:** o foco dessa fase está nos conceitos que o produto pretende ensinar, e é realizada através da observação de situações de ensino induzidas, gerando *insights* para a concepção do produto.
- **Fase 4 - Prototipação:** nessa fase, técnicas de prototipação são utilizadas para concepção e análise do produto, como por exemplo cenários (forma de prototipação rápida e de baixo custo que podem evidenciar falhas e problemas em potencial), prototipação em vídeo (encenação com usuários utilizando protótipos simples) e prototipação de alta fidelidade (implementação parcial do produto).

2.3 Educação musical através de jogos e aplicativos

De forma semelhante a outras áreas, o uso de jogos para o ensino de música é objeto de estudo de pesquisadores brasileiros e estrangeiros. [Carvalho \(2015\)](#) fez análise de alguns estudos realizados no contexto do Brasil. [Aliel e Gohn \(2012\)](#) apontam os avanços tecnológicos para o desenvolvimento de jogos que possibilitaram o uso de instrumentos musicais reais como controladores (*joysticks*). De acordo com [Nogueira \(2010\)](#), os jogos *Guitar Hero* e *Rock Band* foram influenciadores para indústria de jogos musicais. Eles contribuíram para a propagação da cultura auditiva, mas não são classificados como jogos de educação musical. Alguns exemplos são apresentados nos parágrafos seguintes.

O jogo *Let's Tap* ([SEGA, 2009](#)), disponível para *Nintendo Wii*, não faz uso de controles como *joysticks*. Utilizado-se de uma superfície plana, ele faz a captação das vibrações dos toques do jogador através sensores. Ele aborda conceitos como intensidade e ritmo ([FARIA; OLIVEIRA, 2011](#)). [Correa et al. \(2016\)](#) apresenta um aplicativo de realidade aumentada para o ensino de notas musicais. O *T-Acordes* ([UFSCAR, 2018a](#)) é um aplicativo voltado para a construção de acordes, que permite ao usuário indicar o acorde raiz, criar variações e obter a cifra resultante. O *T-Notas* ([UFSCAR, 2018b](#)) permite exercitar a memorização das notas em uma pauta musical. Ambos foram desenvolvidos para o curso de educação musical da Universidade Federal de São Carlos (UFSCar).

O *MusicBlox* ([LESCARGOT, 2012](#)) é uma aplicação que visa permitir a composição de melodias de forma lúdica. O jogador seleciona blocos em uma matriz para criar uma sequência de sons, que são executados de forma sequencial, da esquerda para a direita (Figura 1). A altura das colunas permite representar sons graves e agudos. A interface é simples e lúdica, mas alguns conceitos necessários para educação musical não são abordados.



Figura 1 – Interface do jogo *MusicBlox*

O *Mission to Magmanon* (ECYBERMISSON, 2008) é um jogo que tem como objetivo ensinar notas na pauta juntamente com tempos musicais. No jogo, o usuário deve pressionar as teclas correspondentes às notas que vão aparecendo na tela, da direita para a esquerda. Além disso, o tempo de cada nota deve ser respeitado, em conformidade com a figura que é exibida. Para isso, o jogador deve manter pressionada a tecla pelo tempo adequado, conforme a figura apresentada. A figura 2 mostra a tela inicial da aplicação. Dentre os exemplos analisados, esse é um dos que mais se destaca em relação à abordagem de conteúdos de teoria musical.

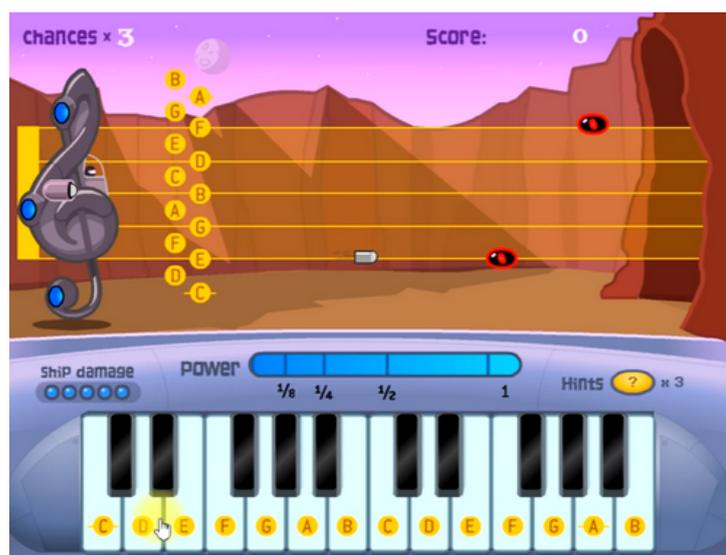


Figura 2 – Interface do jogo *Mission to Magmanon*

O *Simply Piano* (JOYTUNES, 2018) é um aplicativo para dispositivos móveis, voltado para a aprendizagem de piano e partitura. A interação é feita através do uso de instrumentos musicais reais, piano ou teclado. Ele é voltado tanto para usuários iniciantes quanto para avançados. A proposta é muito interessante, mas ela é essencialmente prática, baseada na observação do que é exibido na tela do dispositivo móvel e repetição no piano ou teclado, como ilustrado na Figura 3. O aplicativo não aborda vários conceitos teóricos necessários à educação musical.

Em um projeto de pesquisa, *Adessi e Pachet* (2006) realizaram experimentos com o uso de sistemas musicais interativos. Em um deles, os autores utilizaram o *Continuator* (um sistema capaz de produzir música de maneira parecida com a de um ser humano que toca um teclado) (SONYCSL, 2016) com o objetivo de ensinar métodos de harmonização e improvisação. Segundo os autores, os resultados sugerem que o sistema pode despertar o comportamento musical em crianças e desenvolver interações criança-máquina. Esse estudo de caso reforça a necessidade de alternativas voltadas para o público infanto-juvenil visando a aprendizagem de conceitos básicos de música.

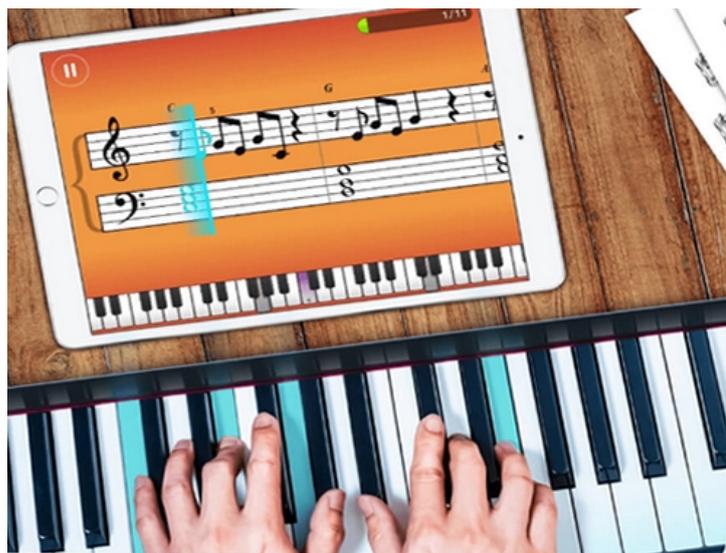


Figura 3 – Exemplo de utilização do jogo *Simply Piano*.

Segundo [Carvalho \(2015\)](#), pesquisadores brasileiros que se dedicam ao estudo de jogos digitais na educação possuem uma percepção que os jogos podem ser aliados dos professores, mas a falta de compreensão do jogo como ferramenta educacional ainda é um obstáculo. Ainda segundo o autor, há uma carência de informação sobre os jogos brasileiros voltados para a educação musical, e isso pode resultar na adoção de jogos estrangeiros, não adaptados à realidade e necessidades da educação brasileira ([CARVALHO, 2015](#)). Por isso, o autor sugere que seja realizado um levantamento junto às empresas brasileiras dos títulos existentes, e que eles sejam objetos de avaliação sobre sua efetividade para o aprendizado de música. De acordo com [Veber e Rosa \(2017\)](#), os jogos digitais podem ser aliados e contribuir significativamente para a formação crítica musical dos usuários, e são ótimas ferramentas de aprendizagem.

O Musicália é um jogo sério, voltado para crianças e jovens, que tem como objetivo apoiar a iniciação musical, introduzindo de forma lúdica conceitos e elementos de teoria musical tais como harmonia, melodia, ritmo, tempo, notas musicais, escala e partitura. É um grande desafio combinar aspectos didáticos e educacionais fundamentais ao ensino com aspectos essenciais do *design* de jogos, para que os jogadores consigam relacionar os conteúdos desejados e se sintam entretidos ao mesmo tempo. O *game design* proposto busca promover a aprendizagem de conceitos básicos (teóricos e práticos) de teoria musical, de forma lúdica e interativa, oferecendo aos usuários uma experiência de uso agradável e divertida.

3 Metodologia

Nesse capítulo são descritas as etapas realizadas e ferramentas utilizadas para o desenvolvimento do jogo proposto. As atividades do projeto envolveram levantamento bibliográfico e apropriação do conhecimento técnico necessário, concepção, projeto, codificação e teste de versões do jogo. Algumas dessas atividades foram desenvolvidas por membros da equipe do projeto¹, e a contribuição será indicada sempre que isso ocorrer.

3.1 Métodos

A realização do projeto envolveu atividades de capacitação, pesquisa e desenvolvimento do jogo, que foram executadas de forma iterativa e incremental, com base em princípios ágeis² e no *framework Scrum* (SABBAGH, 2014).

Inicialmente foi realizada a capacitação necessária sobre a linguagem de programação e as ferramentas utilizadas no projeto, tanto para desenvolvimento quanto para gestão das atividades (seção 3.2). Dentre as referências utilizadas nessa etapa pode-se destacar Hejlsberg, Wiltamuth e Golde (2006), Norton (2013) e a documentação do *Unity 3D* (UNITY, 2018a).

Um levantamento bibliográfico sobre jogos e educação musical foi realizado, e foram analisados exemplos de jogos e aplicativos sobre música (Capítulo 2). Com isso foi possível identificar exemplos alternativos, analisar aspectos positivos e limitações, e perceber a ausência de um jogo voltado para o ensino de teoria musical para crianças e jovens, motivando a continuidade do trabalho. O levantamento foi feito a partir de publicações acadêmicas-científicas e de jogos disponíveis em lojas de aplicativos (*Play Store* e *Apple Store*).

O desenvolvimento do jogo propriamente dito foi feito através de ciclos curtos envolvendo atividades de concepção, design, construção, testes e avaliação. A concepção inclui atividades de *brainstorming*, geração e documentação das ideias, por exemplo, utilizando protótipos de baixa fidelidade como esboços em papel feitos à mão. Na etapa de *(re)design*, as ideias aprovadas são aprimoradas e documentadas em artefatos, através de descrição textual, modelos, diagramas e desenhos. A construção / prototipação envolve atividades de codificação e testes, para produzir protótipos interativos (funcionais) visando a realização de testes e avaliação para obtenção de *feedback* sobre melhorias e correções

¹ Além do autor da monografia e do orientador, também fizeram parte da equipe do projeto Danilo Martins Caldeira e Dasayeve Kaique Souza de Oliveira Xavier.

² Manifesto ágil: <<http://www.manifestoagil.com.br/>>

necessárias. Por fim, a fase de implantação envolve aprovar e disponibilizar a nova versão dos artefatos produzidos ao longo do ciclo.

Durante o projeto, as seguintes etapas foram realizadas:

- Concepção e elaboração do documento de *design* do jogo, que inclui informações como história, roteiro, personagens, especificações de fases, fluxos, estilos de interação, mecânica do jogo etc.
- Desenvolvimento de protótipos de baixa e alta fidelidade, sendo atualizados constantemente em um *mockup* de fluxo, para avaliação de interação;
- Desenvolvimento de uma versão de demonstração do jogo Musicália.
- Avaliação da versão de demonstração.

3.2 Ferramentas

3.2.1 Especificação, projeto e documentação

Protótipos de baixa fidelidade foram elaborados através de desenhos feitos à mão e digitalizados, e também utilizando o software *Microsoft Paint*. A construção desses protótipos contribuiu para apresentação rápida de ideias visando a discussão, concepção e projeto do jogo. A prototipação de baixa fidelidade foi utilizada ao longo de todo o desenvolvimento do projeto, como por exemplo, para projetar novas fases e apoiar a tomada de decisões de design, como ilustrado na Figura 4.

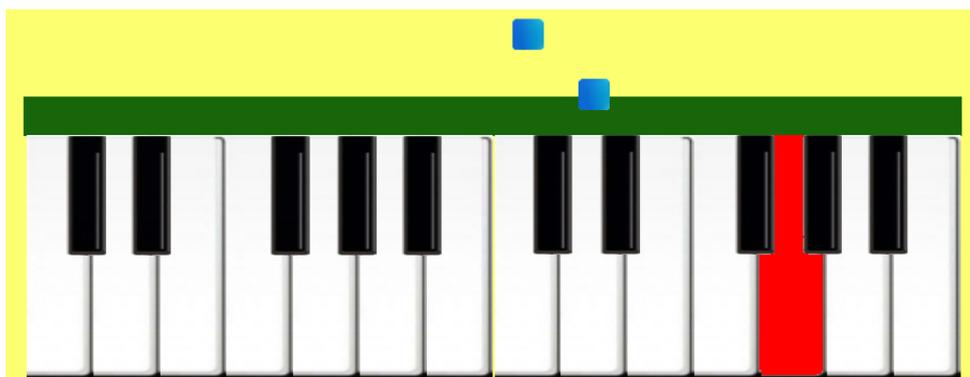


Figura 4 – Exemplo de protótipo de baixa fidelidade criado para especificação e discussão de ideias

Outra ferramenta utilizada foi o *Mockup.io* ([MOCKUP.IO](https://www.mockup.io/), 2018) - um serviço *web* que oferece recursos para criação de protótipos de baixa fidelidade interativos. Ele permite enviar imagens que correspondem às telas (interfaces) do sistema e simular a interação do

usuário através da transição entre elas (Figura 5). A ferramenta foi utilizada para auxiliar no design do fluxo de interação entre as interfaces do jogo. Dessa forma, foi possível de forma rápida e fácil projetar e testar fluxos alternativos para avaliar as melhores opções de transição e interação entre as interfaces do jogo.

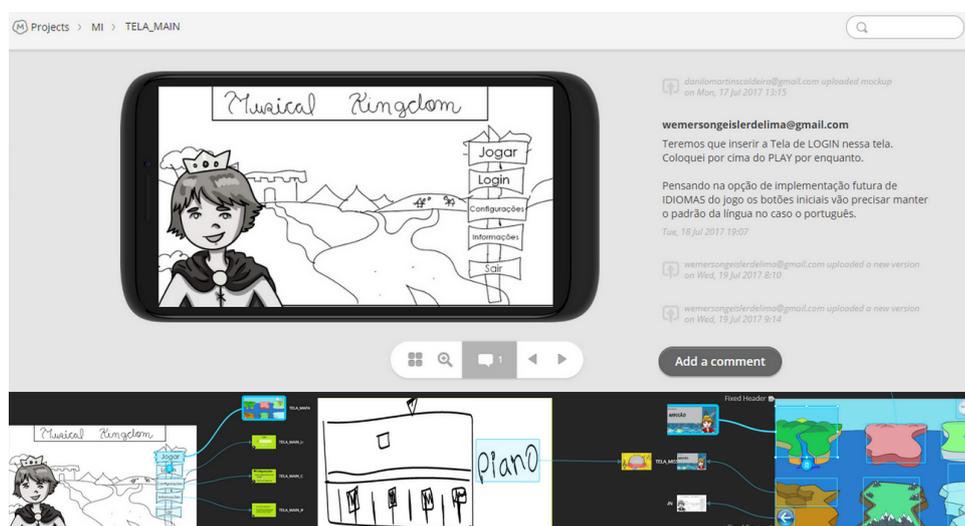


Figura 5 – Utilização da ferramenta *Mockup.io* para projetar e analisar o fluxo de interação entre as interfaces do aplicativo

O documento de design do jogo (GDD) foi elaborado de forma evolutiva ao longo do projeto, usando o *Google Docs* - ferramenta que oferece recursos para escrita colaborativa. As especificações das funcionalidades foram feitas utilizando notação textual, histórias de usuário, diagramas de casos de uso e a ferramenta *issues* do *GitHub*.

3.2.2 Desenvolvimento

Desenvolver jogos é uma atividade desafiadora, que requer habilidades de diferentes áreas do conhecimento, e portanto, necessita de ferramentas adequadas para auxiliar nas tarefas de desenvolvimento (PASSOS et al., 2009). O *Unity 3D* (UNITY, 2018a) é uma *game engine* voltada para a criação de jogos 2D e 3D. Ela oferece muitos recursos para auxiliar nas atividades de desenvolvimento, e permite gerar o jogo para diversas plataformas: dispositivo móvel (*Android*, *IOS*, *Windows Phone*), computador, *console*, web e *smart TV*. Outras vantagens são a disponibilidade de uma boa documentação e a ampla comunidade de usuários, que contribuem para a aprendizagem inicial da ferramenta e também para a busca de soluções para problemas mais específicos. Além disso, a ferramenta oferece uma licença gratuita para fins acadêmicos e não-comerciais (UNITY, 2018b).

A interface gráfica do *unity* (Figura 6) permite acessar facilmente os principais recursos para o desenvolvimento do jogo. De forma geral, ela apresenta os seguintes componentes:

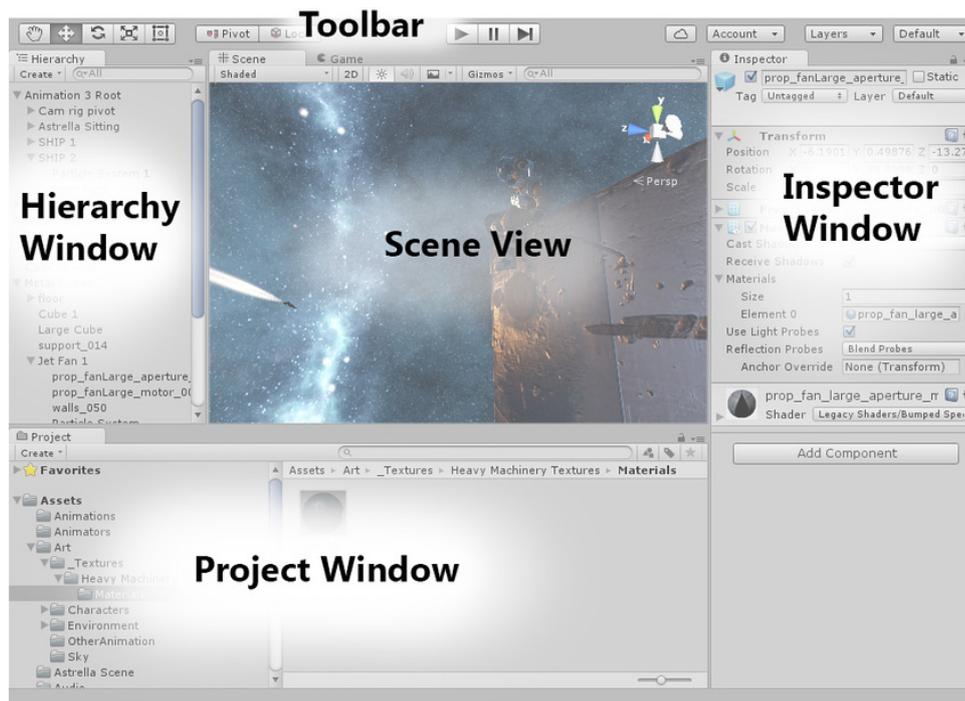


Figura 6 – Interface gráfica do Unity (Versão: 2018.1.6f1) (UNITY, 2018a)

- **Toolbar:** barra de ferramentas, fornece acesso rápido a recursos essenciais, como por exemplo para manipular a visão da cena e os objetos dentro dela, e controles de reprodução do jogo.
- **Hierarchy Window:** exibe uma representação hierárquica (árvore) dos objetos que compõe a cena.
- **Scene View:** representação visual da cena, algo mais próximo do que será visto pelo usuário. Permite visualizar (perspectiva 2D ou 3D), navegar e editar a cena.
- **Inspector Window:** exibição e edição das propriedades do objeto atualmente selecionado. Dessa forma, seu layout e conteúdo irá variar de acordo com o objeto selecionado.
- **Project Window:** janela para importação, exibição, edição e exclusão dos arquivos do projeto.

A linguagem de programação C# foi utilizada para codificação do jogo. O formato *JSON*, que permite a serialização de dados estruturados, foi utilizado para representar as músicas disponíveis no jogo. A Figura 7 apresenta um trecho do arquivo correspondente à música "Brilha, brilha estrelinha" (título original: *Twinkle Twinkle Little Star*, autor: Wolfgang Amadeus Mozart) no formato *JSON*. As mensagens de texto apresentadas nas interfaces do jogo também foram definidas em arquivos *JSON*, para permitir a fácil atualização, revisão e futura tradução para outros idiomas.

```
1 {"musicas":[
2     {
3         "id": 1,
4         "notas":[
5             {
6                 "tempo": 0.5,
7                 "nota": "g",
8                 "oitava": 1,
9                 "nome": "SOL"
10            },
11            {
12                "tempo": 1,
13                "nota": "g",
14                "oitava": 1,
15                "nome": "SOL"
16            },
17            {
18                "tempo": 1,
19                "nota": "d",
20                "oitava": 2,
21                "nome": "RE "
22            },
23        ],
24    },
25 ]}
```

Figura 7 – Trecho de arquivo *JSON* para representação da música "Brilha, brilha estrelinha" (título original: *Twinkle Twinkle Little Star*, autor: Wolfgang Amadeus Mozart).

3.2.3 Gerenciamento

O *GitHub* ([GITHUB, 2018](#)) é um serviço web que oferece recursos para controlar versões e mudanças em projetos de software, baseado no *Git* (sistema distribuído de controle de versões). O *GitHub* oferece, de forma gratuita, diversas funcionalidades para apoiar o desenvolvimento de projetos de software, como revisão de código, controle de mudanças, gerenciamento de projeto, e hospedagem do código-fonte do projeto. Uma interface *web* amigável permite realizar diversas tarefas. Além disso, ele permite a integração com várias outras ferramentas que podem ser usadas de forma complementar.

O *ZenHub* ([ZENHUB, 2018](#)) é uma ferramenta voltada para o gerenciamento ágil de projetos que, integrada ao *GitHub*, oferece serviços adicionais para facilitar a gestão de projetos. São exemplos de recursos a possibilidade de criar e visualizar *issues* ("tarefas") do projeto na forma de um quadro *kanban* (Figura 8), e a geração automática de relatórios sobre o status do projeto. A ferramenta é distribuída como uma extensão para os navegadores *Google Chrome* e *Mozilla Firefox*.

O fluxo de trabalho de codificação foi baseado no *Git Feature Branch Workflow* ([ATLASSIAN, 2018](#)). A partir do *branch* principal, para cada nova *issue*, um novo *branch* é criado. Nesse *branch* eram então realizadas operações para modificação do código-fonte, visando implementar as mudanças especificadas na *issue*, e enviadas para o repositório local e remoto através de comandos *git* (*update*, *add*, *commit*, *push*). Após finalizar as

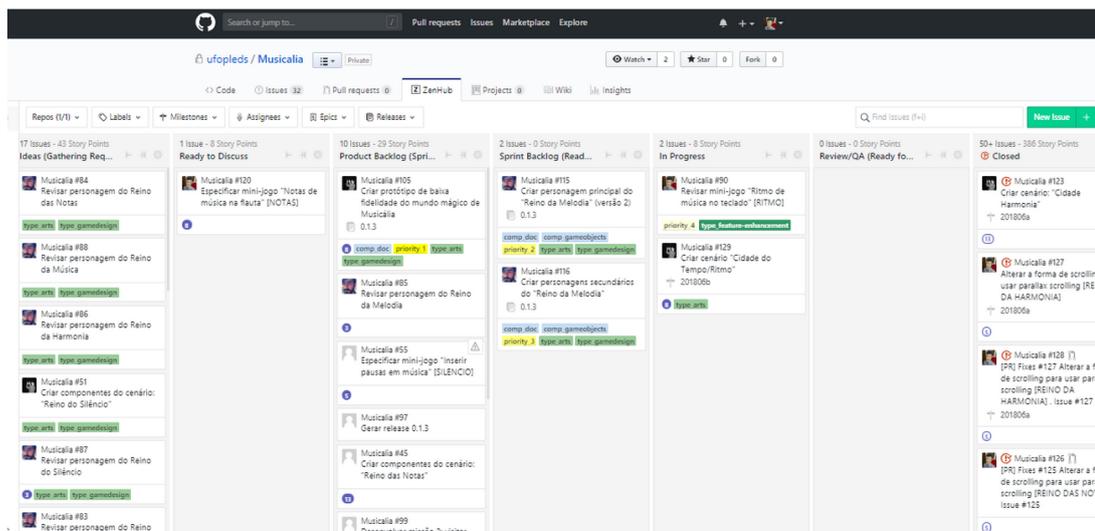


Figura 8 – Utilização do *plugin ZenHub* integrado ao *GitHub*, no *Google Chrome*

atividades de codificação e testes, uma *pull request* era criada para permitir a revisão do código e aceitação ou não das mudanças realizadas. Sempre que possível, essa revisão foi feita por uma outra pessoa da equipe, visando melhorar a qualidade do código-fonte e do produto sendo desenvolvido. Quando aprovada a *pull request*, uma operação *merge* era então realizada e as modificações eram adicionadas no *branch* principal do projeto, gerando uma nova versão do software.

4 Resultados

Nesse capítulo são apresentados os resultados do trabalho, que incluem o *design* do jogo, a elaboração de documentos, o desenvolvimento de protótipos e de uma versão de demonstração do jogo.

4.1 *Game Design*

Musicália é um jogo sério que está sendo desenvolvido para apoiar o ensino de teoria musical, de forma lúdica e interativa. A proposta do jogo surgiu a partir do jogo *Musical Instructor* (BRANDÃO, 2016). Embora tenham sido aproveitadas algumas ideias, o jogo foi completamente reformulado - desde aspectos básicos do *game design* como estória, personagens e cenários à aspectos de implementação (arte e software). Por exemplo, o *Musical Instructor* era baseado na ideia de salas de aula e a interlocução com o jogador era feita por uma maestra, enquanto no *Musicália* temos um mundo mágico com personagens não-humanos e o enredo nada lembra o paradigma tradicional de sala de aulas.

O *game design* foi elaborado de forma iterativa e incremental, a partir de várias sessões de *brainstorming* (tempestade de ideias), com a participação de toda a equipe do projeto. Inicialmente foram definidas a estória do jogo (enredo), os personagens principais, os cenários, e o conteúdo teórico a ser abordado dentro do jogo foi organizado em diferentes reinos. Maestros e professores de música foram consultados sobre essa organização do conteúdo. Todas as atividades foram realizadas com base no *workflow* utilizado no laboratório LED^S¹, partindo da especificação inicial de uma ideia até a sua implementação e aprovação. Parte desses resultados são apresentados ao longo desse capítulo.

4.1.1 Estória do jogo

Musicália é um mundo mágico que possui a música como energia vital. As músicas são armazenadas em amuletos, que funcionam como “cápsulas de energia”. Antigamente, *Musicália* era formada por um único reino, o Reino da Música. Os reis do Reino da música tiveram uma filha e um filho. A princesa havia sido escolhida para suceder o trono. Ressentido por não ter sido escolhido, o príncipe acabou fugindo do reino.

Sillentium é um ser mágico que vivia escondido em uma floresta pantanosa e sempre quis conquistar o Reino da Música. Ele sabia que a união entre o príncipe e a princesa era essencial para manter a harmonia no mundo. Ao saber da fuga do príncipe, foi ao seu encontro e começou a exercer sua influência sobre ele - e criaram um novo

¹ <<http://www.leds.ufop.br>>

reino, que viria a ser chamado de Reino do Silêncio. Durante muito tempo, *Sillentium* foi estimulando no príncipe sentimentos como angústia, tristeza e mágoa, que acabaram por contaminar também o Reino da Música. Além disso, o *Sillentium* havia conseguido “converter” para seu reino os *Waimugs*, uma raça de "monstrinhos" muito alegres e do bem quando influenciados pela energia vital da música, mas que se tornam extremamente desordeiros sem essa influência. Para isso, *Sillentium* roubou o baú de amuletos musicais e quebrou cada um deles em vários pedaços, que se espalharam pelos reinos de Musicália. Sem música no mundo, ele dominou os *Waimugs*, que passaram a vagar por todos os cantos de Musicália causando bagunça e desordem.

Diante do caos, em busca de retomar a ordem do mundo e impedir a completa destruição de Musicália, a princesa saiu em busca de aventureiros que poderiam se voluntariar para ajudar. Quando encontrou um aventureiro (papel assumido pelo jogador) disposto a enfrentar o desafio, passaram a percorrer o mundo de Musicália com o objetivo de recuperar os pedaços de amuletos, organizar a bagunça deixada pelos *Waimugs* e resgatar o príncipe da influência do *Sillentium*.

Para ajudar a princesa nesse desafio, o jogador, no papel de aventureiro, deverá percorrer cada um dos reinos de Musicália para completar diferentes missões. Cada reino possui uma temática própria, abordando diferentes tipos de conteúdo de música (ex. ritmo, tempo, notas, melodia, harmonia e pauta). Dessa forma, ao longo do jogo, o jogador terá que realizar diferentes tarefas (ex. recuperar um pedaço de amuleto, concertar as notas de um piano), as quais envolvem algo relacionado à música (ex. ordenar corretamente as notas / teclas de um piano e tocar uma música). Além da princesa, outros personagens irão acompanhar o jogador durante esse desafio, apresentando orientações e informações importantes sempre que necessário.

4.1.2 Reinos de Musicália (cenários)

A definição dos reinos que fariam parte do mundo mágico de Musicália foi pensada sempre de forma associada aos conteúdos teóricos de música que fariam parte de cada reino. Dessa forma, o primeiro passo foi identificar e agrupar os conteúdos básicos que seriam abordados no jogo visando a iniciação musical e introdução à teoria musical. Em seguida, especialistas (maestros e professores de música) foram consultados sobre o conteúdo e seu agrupamento. Definida essa questão, foi realizado um levantamento de características (adjetivos) que representariam cada um desses grupos de conteúdo. E a partir dessas características, teve início o processo de criação dos cenários e personagens para cada reino. Essas informações são apresentadas na Tabela 4.1.2. A definição de cada conceito relacionado à música é apresentada no Anexo A. Os nomes dos reinos ainda são provisórios

Um roteiro de atividades e uma ficha (*template*) de cenários foram utilizados para a criação dos reinos, com base nas características levantadas. As seguintes atividades do

Reino	Conceitos	Características
Ritmo	- ritmo, compasso, intervalos e figuras de tempo	- metódico, organizado, disciplinado, preciso, rústico
Notas	- notas naturais, tons, sustenidos e bemóis.	- alegre, colorido, descontraído, radiante, divertido
Melodia	- melodias, oitavas, composições e instrumentos musicais.	- flexível, romântico, juvenil, bonito, ousado
Harmonia	- harmonias, acordes e cifras.	- sereno, calmo, relaxante, pacífico, elegante
Pauta	- pautas, ligaduras e claves.	- rigoroso, formal, ordenado, exigente, sábio, líder, conciliador
Silêncio	- pausas, e simbologias.	- silencioso, frio, observador simples, místico

roteiro foram realizadas de forma colaborativa pela equipe do projeto: (1) definir escopo básico do cenário; (2) elaborar esboço visual do cenário; (3) preencher ficha do cenário; (4) criar arte conceitual do cenário.

A definição de um escopo básico para os cenários ajudou a tomar decisões sobre a arte, como por exemplo cores, estilos e objetos que iriam compor o reino. A ficha de cenário contém as seguintes informações: identificação; características físicas; história de vida; características emocionais e psicológicas; arte: esboço visual; arte conceitual. Dessa forma, dentre outras, é possível encontrar na ficha de cenário croquis, características de paisagem, estilos de objetos (ex. árvores, construções), personagens secundários, cores. A figura 9 ilustra exemplos de esboços para criação dos cenários do Reino da Melodia e Reino da Harmonia, e a figura 10 apresenta a arte final dos respectivos cenários.

4.1.3 Personagens

O jogo contém personagens primários, que irão fazer interlocução com o jogador, e secundários, utilizados somente para compor os cenários dando mais vida aos reinos. No design do jogo, os personagens primários irão interagir com o jogador, tanto para passar orientações necessárias quanto para aumentar a motivação e engajamento.

De forma semelhante à criação dos reinos e respectivos cenários, algumas atividades foram realizadas para construção desses personagens: (1) definir escopo básico do personagem; (2) elaborar esboço visual do personagem; (3) criar ficha do personagem; (4) criar arte conceitual (*concept art*) do personagem.

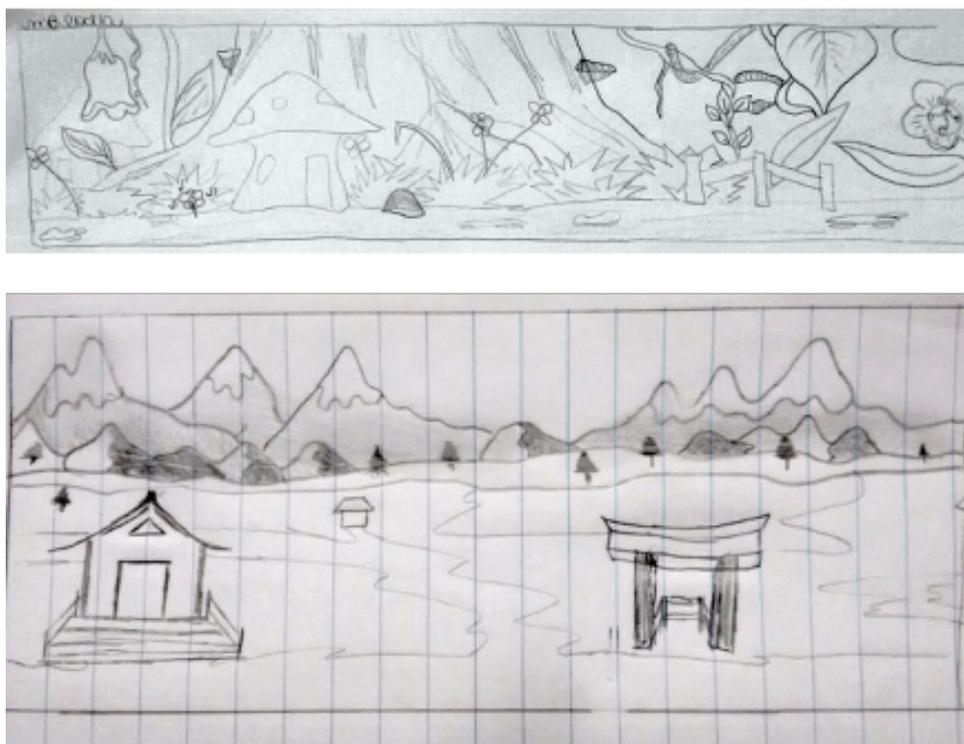


Figura 9 – Esboço dos cenários do Reino da Melodia (imagem superior) e Reino da Harmonia (imagem inferior). Arte: Danilo Caldeira



Figura 10 – Cenários do Reino da Melodia (imagem superior) e Reino da Harmonia (imagem inferior). Arte: Danilo Caldeira

A ficha de personagem é composta pelas seguintes informações: identificação; características físicas; história de vida; características emocionais e psicológicas; arte - esboço visual; arte conceitual. A figura 11 ilustra exemplos de esboços para criação dos personagens do Reino da Melodia e Reino da Harmonia, e a figura 12 apresenta a arte final dos respectivos personagens. De forma semelhante, as atividades de criação dos personagens foram realizadas de forma colaborativa pela equipe do projeto.

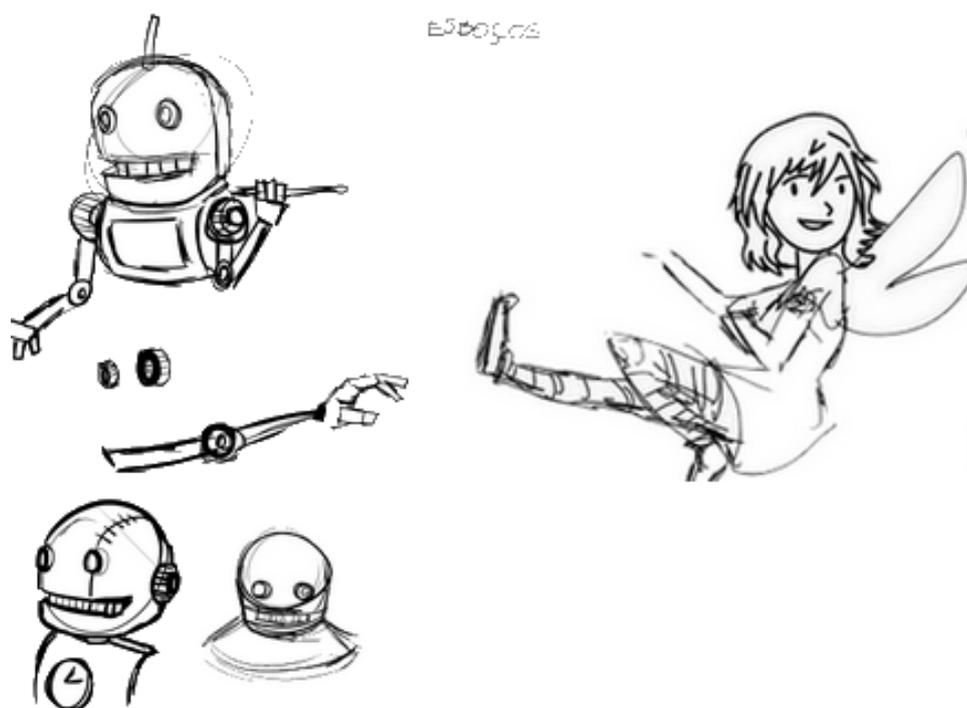


Figura 11 – Esboço dos personagens Bataque (Reino do Ritmo) e Lili (Reino da Melodia).
Arte: Dasayeve Xavier



Figura 12 – Personagens Bataque (Reino do Ritmo) e Lili (Reino da Melodia). Arte:
Dasayeve Xavier

4.1.4 Projeto de interface e interação

Tendo como objetivo salvar o mundo mágico de Musicália, o jogador deverá percorrer os diferentes reinos e realizar variadas missões. Dessa forma, em paralelo à criação dos cenários e personagens, foram elaborados projetos de interface e da interação com o jogador. De forma semelhante, foram realizadas seções de *brainstorming* e construídos protótipos de baixa fidelidade (Figura 13). Um protótipo interativo de baixa fidelidade (Figura 5) foi construído, utilizando a ferramenta *Mokcup.io* (MOCKUP.IO, 2018) para analisar o fluxo de interação entre as interfaces do jogo.

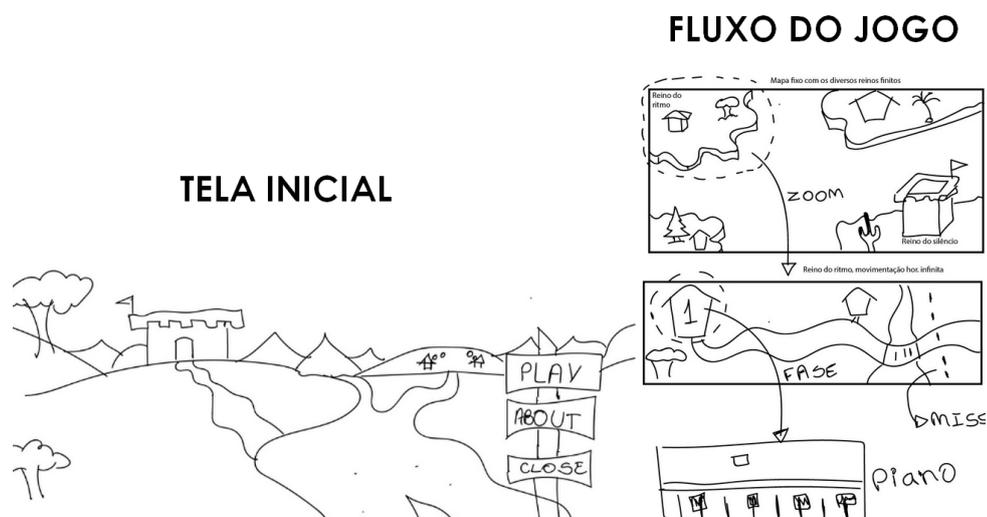


Figura 13 – Protótipo de baixa fidelidade ilustrando a ideia inicial do fluxo de interação entre as interfaces do jogo. Arte: Danilo Caldeira

A utilização de protótipos de baixa fidelidade durante todo o processo de desenvolvimento contribuiu para antecipar a tomada de decisões e reduzir o esforço e tempo de retrabalho. A figura 14 apresenta uma evolução do protótipo que ilustra o fluxo de interação entre as interfaces do jogo.

De modo semelhante, as interfaces do jogo também foram construídas de forma evolutiva, sempre iniciadas a partir de protótipos de baixa fidelidade. A figura 15 ilustra esse processo para a interface inicial do jogo.

4.1.5 Minijogos

Os protótipos de baixa fidelidade (esboços feitos a mão ou digitalmente) ajudaram a elaborar e avaliar versões alternativas de design. Usar desenhos manuais (esboços) para prototipação de baixa fidelidade, mesmo em estágios iniciais de desenvolvimento, é útil pois permite rapidamente identificar possíveis problemas, testar novas ideias e tomar decisões de design (ex. tamanho e posicionamento de objetos). Eles foram úteis para a criação dos cenários e personagens, mas também para a criação dos minijogos que fazem parte do jogo

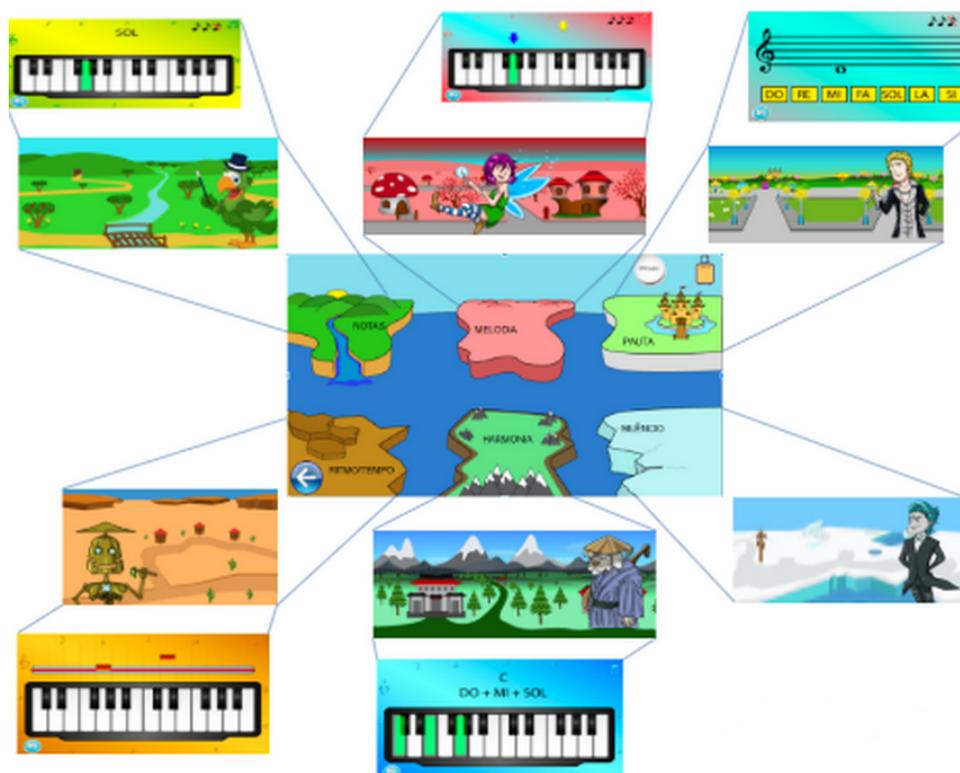


Figura 14 – Protótipo que ilustra o fluxo de interação entre interfaces do jogo. Artes: Danilo Caldeira e Dasayeve Xavier



Figura 15 – Criação da interface principal do jogo, de forma evolutiva, a partir de protótipos de baixa fidelidade. Arte: Dasayeve Xavier

Musicália. Definimos um minijogo como uma fase, que possui objetivos, regras e mecânica próprios. De forma simplificada, o processo de desenvolvimento dos minijogos seguiu os seguintes passos: (1) proposta de minijogo - concepção inicial através de esboços (desenhos manuais ou digitais); (2) discussão e aprovação / reprovação da proposta; e se aprovada, (3) especificação formal por meio de descrição textual e protótipos de baixa fidelidade para ilustrar o projeto de interface e interação (ex. mecânica do jogo); (4) construção de uma versão interativa para teste em dispositivos móveis (celular ou tablet); (5) avaliação informal feita por outro membro da equipe; quando necessário, (6) implementação das correções imediatamente após o feedback recebido; e, por fim, (7) inclusão do minijogo na versão atual do jogo. Os passos 1, 2, 3 e 5 eram realizados de forma colaborativa, por toda a equipe do projeto, enquanto os passos 4, 6 e 7 eram de responsabilidade exclusiva do

autor desse trabalho. A seguir, serão apresentados os minijogos que foram implementados e estão disponíveis na versão demo do jogo.

4.1.5.1 Ritmo no teclado

Objetivo: o jogador deve pressionar a tecla correta do teclado no momento certo para acompanhar o tempo da música a ser tocada. Uma orientação visual é apresentada para orientar o jogador - objetos (ex. bolinhas) "caindo" sobre o teclado indicam a tecla correta e o momento certo é indicado pela sobreposição da bolinha em uma área (ex. retângulo).

Elementos de música: teclado com duas oitavas a ser utilizado pelo jogador, metrônomo musical a ser usado para dar feedback visual e sonoro sobre o tempo/ritmo.

Descrição: o jogador poderá ganhar pontos conforme pressiona as teclas no ritmo da música. O jogador deve pressionar a tecla correta quando o objeto (ex. bolinha) estiver sobreposto no centro da área (ex. retângulo) acima do teclado. A quantidade de pontos depende da distância da bolinha a esse centro. O jogador NÃO irá ganhar pontos se: pressionar tecla correta quando a bolinha estiver fora do retângulo ou pressionar tecla errada quando a bolinha estiver dentro do retângulo. Variações no nível de dificuldade podem ser feitas pela música a ser tocada, pela velocidade em que a música é tocada, e pelo nível de precisão exigido.

Regras: Pré-condição: o jogador possui uma quantidade P de pontos e E de estrelas. Pós-condição: o jogador irá acumular a quantidade máxima de pontos e estrelas que recebeu ao completar a tarefa (mini-jogo), ou seja, se o jogador realizar a tarefa duas ou mais vezes, somente a pontuação máxima deve ser registrada.

Especificação:

```
qtd_notas_musica = N (ex. musica brilha brilha estrelinha)
// quantidade de notas que a musica possui. Esse numero eh
// conhecido a priori e define o maximo possivel de acertos
pontos = 0; qtd_notas_jogador = 0;
qtd_acertos = 0; qtd_erros = 0;
```

```
WHILE (qtd_notas_musica > 0) {
    // acerto implica usuario pressionar tecla correta
    // no momento certo
    IF acerto == "perfect" { // jogador pressionou tecla
        // quando a "bolinha "
```

```

                                //estava sobreposta a
                                //regiao mais central
    pontos += 2
    qtd_notas_jogador++; qtd_acertos++
    exibe_feedback("perfect");
} ELSE IF acerto == "good" { // jogador pressionou tecla
                                //quando a "bolinha"
                                //estava sobreposta a regiao
                                //menos central

    pontos += 1
    qtd_notas_jogador++; qtd_acertos++
    exibe_feedback("good");
} ELSE { // jogador errou, pois pressionou tecla incorreta
    //ou em momento errado
    //(quando nenhuma bolinha estava sobreposta a regiao)
    qtd_notas_jogador++; qtd_errores++
    exibe_feedback("erro");
}
    qtd_notas_musica—
}

//e possivel calcular a eficacia do jogador com base na
//quantidade de notas da musica e
//na quantidade de erros/acertos

max_pontos = 2 * qtd_notas_musica
desempenho = pontos / max_pontos

IF (9 <= desempenho <= 10)
    qtd_estrelas = 3
ELSE IF (8 <= desempenho < 9)
    qtd_estrelas = 2
ELSE IF (7 <= desempenho < 8)
    qtd_estrelas = 1
ELSE
    qtd_estrelas = 0

```

Orientações de GUI e usabilidade: Apresentar feedback visual e sonoro para o usuário:

- ao pressionar tecla: mudar cor da tecla (verde em caso de acerto perfeito, amarelo em caso de acerto, vermelho em caso de erro) + tocar som (em caso de acerto, tocar som correspondente à nota da tecla pressionada; em caso de erro, tocar "som de erro") + "explosão" da bolinha quando jogador acertar
- ver seção "Elementos de música": o teclado deve ter 2 oitavas, e um metrônomo deve ser incluído na interface para marcar o tempo/ritmo (visual e sonoro)

Protótipo de baixa fidelidade: Figura: 16

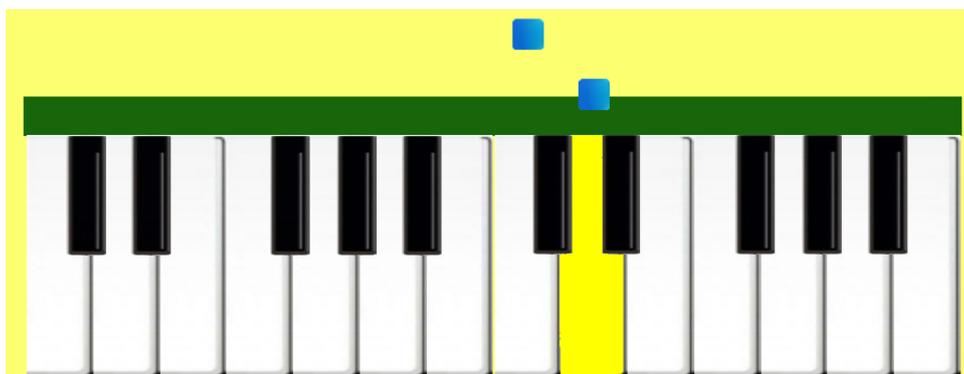


Figura 16 – Protótipo de baixa fidelidade da fase: Ritmo no teclado. Arte: Wemerson Geisler

4.1.5.2 Notas no teclado

Objetivo: o jogador deve pressionar a tecla correta do teclado de acordo com a nota que é exibida, da música a ser tocada.

Elementos de música: teclado com duas oitavas a ser utilizado pelo jogador. O teclado terá duas versões: uma com marcação das notas (completa ou parcial) no teclado e outra sem.

Descrição: o jogador irá ganhar pontos na medida em que pressionar no teclado a tecla correspondente à nota exibida na parte superior da tela. A quantidade de pontos obtida dependerá unicamente de acertar ou não a tecla correspondente à nota que foi exibida. O jogador não irá ganhar e nem perder pontos se pressionar uma tecla que não corresponde à nota exibida. No entanto, os erros serão contabilizados ao final, para medir seu desempenho - cada erro corresponde à redução de 10% do valor máximo de pontos possíveis, podendo o valor de pontos ficar negativo ao longo da fase, porém se ao final da fase o jogador obter pontuação negativa, a pontuação final será considerada 0. O jogador terá um número limitado de chances (vidas), que irão limitar a quantidade máxima de erros. Na versão inicial não há tempo para realizar a tarefa, mas uma variação seria incluir um *timer* com tempo limite para que o jogador pressione a tecla correspondente à nota

exibida. Variações no nível de dificuldade podem ser feitas por 1) exibição ou não de informações das notas nas teclas (completa, parcial, nenhuma), 2) *timer* com tempo limite para pressionar a tecla uma vez que a nota foi exibida.

Regras: Pré-condição: o jogador possui uma quantidade P de pontos e E de estrelas; o jogador possui quantidade E de estrelas suficiente para desbloquear essa fase (tarefa). Pós-condição: o jogador irá acumular a quantidade máxima de pontos e estrelas que recebeu ao completar a tarefa (mini-jogo), ou seja, se o jogador realizar a tarefa duas ou mais vezes, somente a pontuação máxima deve ser registrada.

Especificação:

```

qtd_notas_musica = N (ex. musica brilha brilha estrelinha)

//quantidade de notas que a musica possui -
//esse numero e conhecido a priori e define
//o maximo possivel de acertos

pontos = 0; qtd_notas_jogador = 0; qtd_acertos =
0; qtd_erros = 0; qtd_vidas = 4;

WHILE (qtd_notas_musica > 0) {
    // acerto implica usuario pressionar tecla
    //correta correspondente a nota que foi exibida

    IF acerto == "correct" { // jogador pressionou
        //tecla correta correspondente a nota exibida

            pontos += 1
            qtd_notas_jogador++; qtd_acertos++
            exhibe_feedback("correct");

        } ELSE { // jogador errou, pois pressionou
            //tecla incorreta
                qtd_vidas--; qtd_erros++

                //atualiza pontos conforme a quantidade de vidas
                //perdidas (-10% do maximo de pontos para cada vida perdida)

                pontos = FLOOR( pontos - qtd_erros * (max_pontos/10) )

```

```

                // arredondar para baixo , para obter
                menor inteiro

                exibe_feedback("erro");
            }
            qtd_notas_musica—
        }
IF qtd_notas_jogador = qtd_notas_musica || qtd_vidas == 0
    fim de jogo
    IF pontos < 0
        pontos=0;

//e possivel calcular a eficacia do jogador com base na
//quantidade de notas da musica e na
//quantidade de erros/acertos

max_pontos = qtd_notas_musica

desempenho = pontos / max_pontos
IF (9 <= desempenho <= 10)
    qtd_estrelas = 3
ELSE IF (8 <= desempenho < 9)
    qtd_estrelas = 2
ELSE IF (7 <= desempenho < 8)
    qtd_estrelas = 1
ELSE
    qtd_estrelas = 0

```

Orientações de GUI e usabilidade: apresentar *feedback* visual e sonoro para o usuário:

- ao pressionar tecla: mudar cor da tecla (vermelho em caso de erro) + tocar som (em caso de acerto, tocar som correspondente à nota da tecla pressionada; em caso de erro, tocar "som de erro")
- ver seção "Elementos de música": o teclado deve ter 2 oitavas.
- quando for exigido do jogador "adivinhar/completar" a próxima nota da sequência, deve-se exibir um *feedback* visual para informar ao jogador que ele deve pressionar uma tecla - mas sem indicar qual é a tecla.

Protótipo de baixa fidelidade: Figura: 17

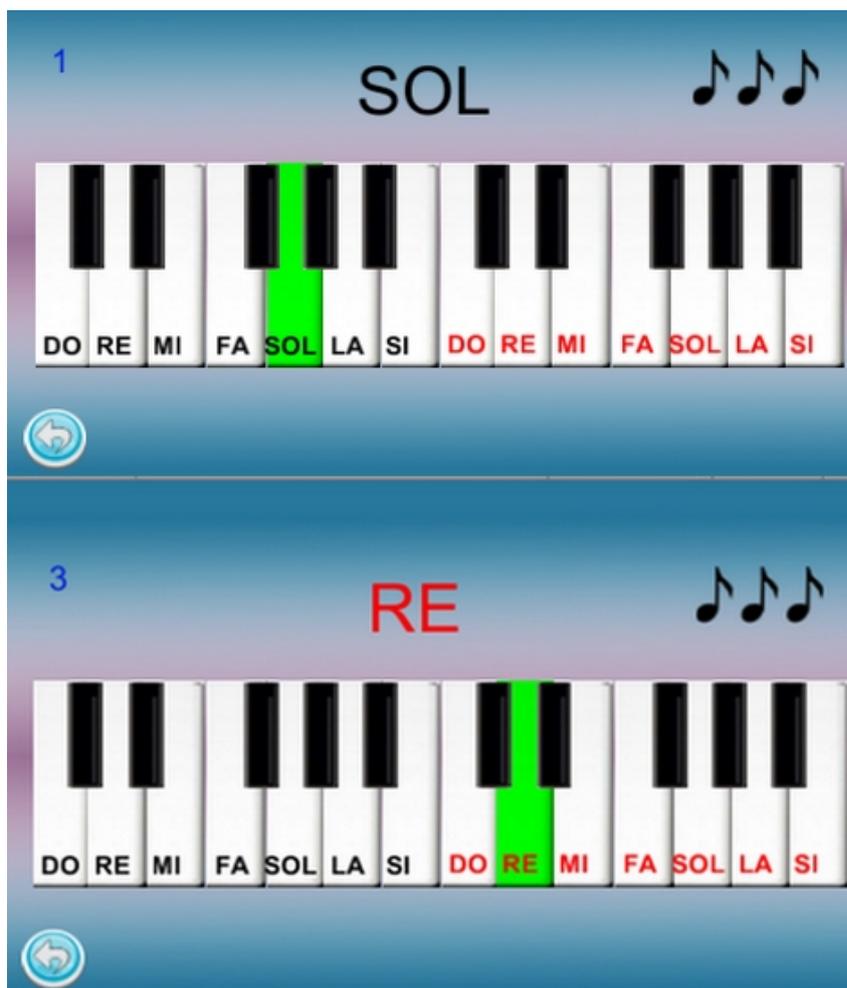


Figura 17 – Protótipo de baixa fidelidade da fase: Notas no teclado. Arte: Wemerson Geisler

4.1.5.3 Melodia no teclado

Objetivo: o jogador deve pressionar corretamente a sequência de teclas para tocar a melodia da música. Uma orientação visual é apresentada ao jogador - ex. uma tecla colorida no teclado indicando qual é a tecla atual que o jogador deve pressionar e uma "bolinha" indicando próxima nota que deve ser pressionada.

Elementos de música: teclado com duas oitavas a ser utilizado pelo jogador.

Descrição: o jogador irá ganhar pontos na medida em que pressionar no teclado a tecla correspondente à nota indicada (tecla colorida no teclado). A quantidade de pontos obtida dependerá unicamente de acertar ou não a tecla correspondente à nota que foi exibida. O jogador não irá ganhar e nem perder pontos se pressionar uma tecla que não corresponde à nota exibida. No entanto, os erros serão contabilizados ao final, para medir

seu desempenho - cada erro corresponde à redução de 10% do valor máximo de pontos possíveis, podendo o valor de pontos ficar negativo ao longo da fase, porém se ao final da fase o jogador obter pontuação negativa, a pontuação final será considerada 0. O jogador terá um número limitado de chances (vidas), que irão limitar a quantidade máxima de erros. Na versão inicial não há tempo para realizar a tarefa, mas uma variação seria incluir um *timer* com tempo limite para que o jogador pressione a tecla correspondente à nota exibida. Variações no nível de dificuldade podem ser feitas por 1) *timer* com tempo limite para pressionar a tecla uma vez que a nota foi indicada no teclado., 2) deixar o jogador "adivinhar/completar" a próxima nota da sequência, ou seja, o teclado não apresenta *feedback* da tecla atual que precisa ser pressionada.

Regras: Pré-condição: o jogador possui uma quantidade P de pontos e E de estrelas; o jogador possui quantidade E de estrelas suficiente para desbloquear essa fase (tarefa). Pós-condição: o jogador irá acumular a quantidade máxima de pontos e estrelas que recebeu ao completar a tarefa (mini-jogo), ou seja, se o jogador realizar a tarefa duas ou mais vezes, somente a pontuação máxima deve ser registrada.

Especificação:

```

qtd_notas_musica = N (ex. musica brilha brilha estrelinha)
//quantidade de notas que a musica possui -
//esse numero e conhecido a priori e define
//o maximo possivel de acertos

pontos = 0; qtd_notas_jogador = 0;
qtd_acertos = 0; qtd_errores = 0;

ENQUANTO (qtd_notas_musica > 0) {
    //acerto implica usuario pressionar tecla
    //correta correspondente a nota que foi exibida

    IF acerto == "correct" { //jogador pressionou tecla
        //correta correspondente a nota exibida
        pontos += 1
        qtd_notas_jogador++; qtd_acertos++
        exhibe_feedback("correct");
    } ELSE { //jogador errou, pois pressionou tecla incorreta
        qtd_notas_jogador++; qtd_errores++ qtd_vidas—
        //atualiza pontos conforme a quantidade de

```

```

        //vidas perdidas (-10% do maximo de pontos para cada
        vida perdida)

        pontos = FLOOR( pontos - qtd_erro * (max_pontos/10) )
        // arredondar para baixo, para obter
        menor inteiro
        exibe_feedback("erro");
    }
    qtd_notas_musica—
}

IF qtd_notas_jogador = qtd_notas_musica || qtd_vidas == 0
    fim de jogo
    IF pontos < 0
        pontos=0;

//e possivel calcular a eficacia do jogador com base
//na quantidade de notas da musica e na
//quantidade de erros/acertos

max_pontos = qtd_notas_musica

desempenho = pontos / max_pontos
IF (9 <= desempenho <= 10)
    qtd_estrelas = 3
ELSE IF (8 <= desempenho < 9)
    qtd_estrelas = 2
ELSE IF (7 <= desempenho < 8)
    qtd_estrelas = 1
ELSE
    qtd_estrelas = 0

```

Orientações de GUI e usabilidade: apresentar *feedback* visual e sonoro para o usuário:

- ao pressionar tecla: mudar cor da tecla (vermelho em caso de erro) + tocar som (em caso de acerto, tocar som correspondente à nota da tecla pressionada; em caso de erro, tocar "som de erro").

- ver seção "Elementos de música": o teclado deve ter 2 oitavas.
- quando for exigido do jogador "adivinhar/completar" a próxima nota da sequência, deve-se exibir um *feedback* visual para informar ao jogador que ele deve pressionar uma tecla - mas sem indicar qual é a tecla.

Protótipo de baixa fidelidade: Figura: 18

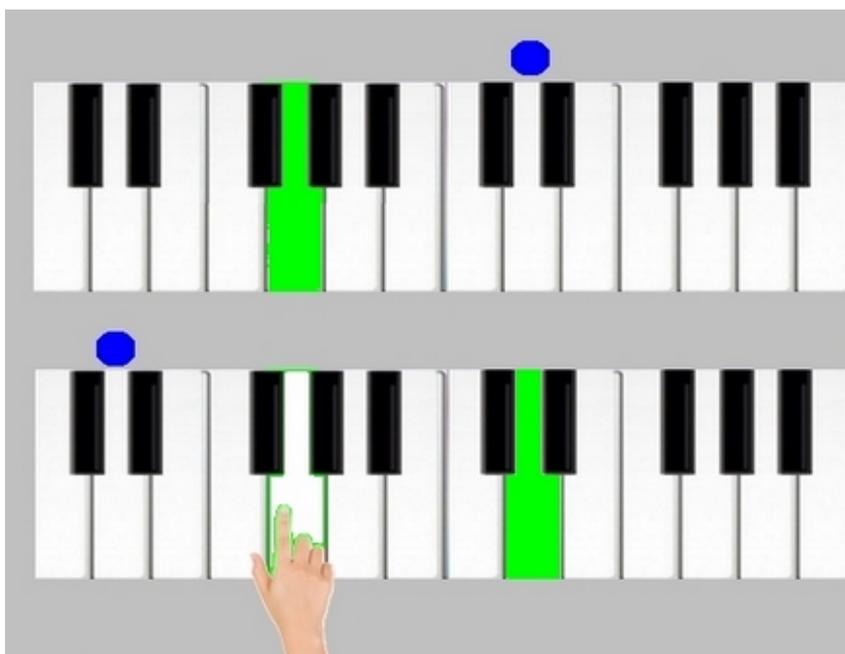


Figura 18 – Protótipo de baixa fidelidade da fase: Melodia no teclado. Arte: Wemerson Geisler

4.1.5.4 Acordes no teclado

Objetivo: o jogador deve pressionar corretamente o conjunto de teclas que compõe cada um dos acordes da música. Uma orientação visual é apresentada ao jogador - ex. o acorde desejado e quais as notas que o jogador deve pressionar para compor o acorde corretamente.

Elementos de música: teclado com duas oitavas a ser utilizado pelo jogador, acorde e notas que formam o acorde.

Descrição: o jogador irá ganhar pontos na medida em que pressionar no teclado a composição de teclas correspondentes ao acorde indicado (na parte superior da tela). A quantidade de pontos obtida dependerá do jogador acertar ou não a composição de notas correspondente ao acorde que foi exibido. O jogador não irá ganhar pontos se errar a composição do acorde, ou seja, pressionar uma ou mais notas que não fazem parte do acorde exibido. No entanto, os erros serão contabilizados ao final, para medir seu desempenho -

cada erro corresponde à redução de 10% do valor máximo de pontos possíveis. O valor de pontos pode ficar negativo ao longo da fase, porém, se ao final da fase o jogador obter pontuação negativa, a pontuação final será considerada zero. O jogador terá um número limitado de chances (vidas), que irão limitar a quantidade máxima de erros. Na versão inicial não há tempo para realizar a tarefa, mas uma variação seria incluir um *timer* com tempo limite para que o jogador pressione o conjunto de notas correspondente ao acorde exibido. Variações no nível de dificuldade podem ser feitas por 1) *timer* com tempo limite para compor o acorde uma vez que as notas foram indicada na parte superior., 2) deixar o jogador "adivinhar/completar" alguma nota da composição do acorde, ou seja, nem todas as notas do acorde são exibidas.

Regras: Pré-condição: o jogador possui uma quantidade P de pontos e E de estrelas; o jogador possui quantidade E de estrelas suficiente para desbloquear essa fase (tarefa). Pós-condição: o jogador irá acumular a quantidade máxima de pontos e estrelas que recebeu ao completar a tarefa (mini-jogo), ou seja, se o jogador realizar a tarefa duas ou mais vezes, somente a pontuação máxima deve ser registrada.

Especificação:

```

qtd_acordes_musica = N (ex. musica brilha brilha estrelinha)
//quantidade de acordes que a musica possui —
//esse numero e conhecido a priori e define
//o maximo possivel de acertos

pontos = 0; qtd_acordes_jogador = 0;
qtd_acertos = 0; qtd_errores = 0;

ENQUANTO (qtd_acordes_musica > 0) {
    // acerto implica usuario pressionar a composicao
    //correta correspondente ao acorde que foi exibido

    IF acerto == "correct" { // jogador pressionou
        //corretamente todas as teclas que compoem o acorde exibido.
        pontos += 1
        qtd_acordes_jogador++; qtd_acertos++
        exhibe_feedback("correct");
    } ELSE { // jogador errou, nao pressionou corretamente
        //todas as notas que compoem o acorde
        qtd_acordes_jogador++; qtd_errores++ qtd_vidas—

```

```

        //atualiza pontos conforme a quantidade de vidas perdidas
        //(-10% do maximo de pontos para cada
        vida perdida)
        pontos = FLOOR( pontos - qtd_erros * (max_pontos/10) )
        // arredondar para baixo, para obter
        menor inteiro
        exibe_feedback("erro");
    }
    qtd_acordes_musica—
}

IF qtd_acordes_jogador = qtd_acordes_musica || qtd_vidas == 0
    fim de jogo
    IF pontos < 0
        pontos=0;

//e possivel calcular a eficacia do jogador
//com base na quantidade
//de acordes da musica e na quantidade de erros/acertos

max_pontos = qtd_acordes_musica

desempenho = pontos / max_pontos
IF (9 <= desempenho <= 10)
    qtd_estrelas = 3
ELSE IF (8 <= desempenho < 9)
    qtd_estrelas = 2
ELSE IF (7 <= desempenho < 8)
    qtd_estrelas = 1
ELSE
    qtd_estrelas = 0

```

Orientações de GUI e usabilidade: apresentar *feedback* visual e sonoro para o usuário:

- ao pressionar tecla: mudar cor da tecla (vermelho em caso de erro) + tocar som (em caso de acerto, tocar som correspondente à nota da tecla pressionada; em caso de erro, tocar "som de erro").
- ver seção "Elementos de música": o teclado deve ter 2 oitavas.

- quando for exigido do jogador "adivinhar/completar" alguma nota da composição do acorde, deve-se exibir um *feedback* visual para informar ao jogador que ele deve pressionar as teclas - mas sem indicar qual é a nota.

Protótipo de baixa fidelidade: Figura: 19

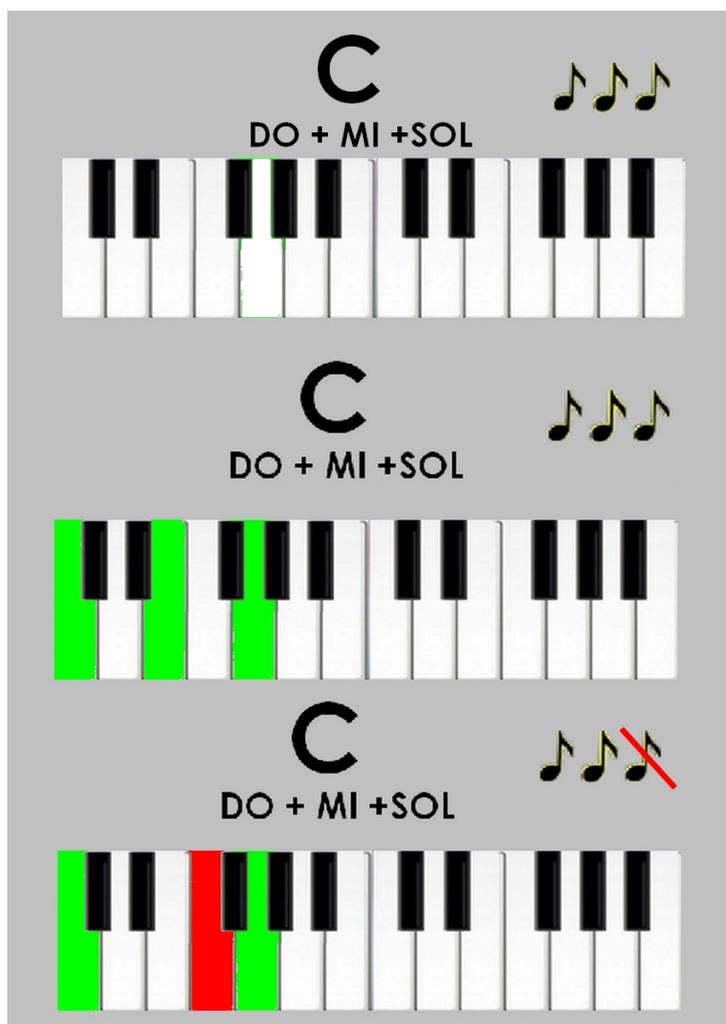


Figura 19 – Protótipo de baixa fidelidade da fase: Acordes no teclado. Arte: Wemerson Geisler

4.1.5.5 Notas na pauta

Objetivo: o jogador deve indicar corretamente a nota (botão) que corresponde à nota exibida na pauta musical. Uma orientação visual é apresentada ao jogador - pauta musical com a nota que deve ser identificada, notas musicais (botões) a serem indicadas pelo jogador.

Elementos de música: pauta musical.

Descrição: o jogador irá ganhar pontos se indicar a nota que correspondente à nota exibida na pauta musical. A quantidade de pontos obtida dependerá de acertar ou não qual foi a nota exibida na pauta. O jogador será penalizado (redução de 10% do valor máximo de pontos possíveis) se indicar uma nota que não corresponde àquela exibida na pauta. Se ao final da fase o jogador estiver com pontuação negativa, a pontuação final será considerada zero. O jogador possui um número limitado de chances (vidas), que irão limitar a quantidade máxima de erros. Na versão inicial não há tempo para realizar a tarefa, mas uma variação seria incluir um *timer* com tempo limite para que o jogador indique a nota correspondente àquela exibida na pauta. Variações no nível de dificuldade pode ser dadas por 1) *timer* com tempo limite para indicar a nota, 2) exibir uma sequencia de notas na pauta.

Regras: Pré-condição: o jogador possui uma quantidade P de pontos e E de estrelas; o jogador possui quantidade E de estrelas suficiente para desbloquear essa fase (tarefa). Pós-condição: o jogador irá acumular a quantidade máxima de pontos e estrelas que recebeu ao completar a tarefa (mini-jogo), ou seja, se o jogador realizar a tarefa duas ou mais vezes, somente a pontuação máxima deve ser registrada.

Especificação:

```

qtd_notas_pauta = N (ex. musica brilha brilha estrelinha)
//quantidade de notas que a musica possui -
//esse numero e conhecido a priori e define
//o maximo possivel de acertos
pontos = 0; qtd_notas_jogador = 0;
qtd_acertos = 0; qtd_erros = 0;

ENQUANTO (qtd_notas_pauta > 0) {
    // acerto implica usuario pressionar tecla
    //correta correspondente a nota que foi exibida na pauta

    IF acerto == "correct" { // jogador pressionou o botao
        //coreto correspondente a nota exibida na pauta
        pontos += 1
        qtd_notas_jogador++; qtd_acertos++
        exhibe_feedback("correct");
    } ELSE { //jogador errou, pois pressionou o botao incorreto
        qtd_notas_jogador++; qtd_erros++ qtd_vidas—
        // atualiza pontos conforme a quantidade de vidas perdidas

```

```

        //(-10% do maximo de pontos para cada
        vida perdida)
        pontos = FLOOR( pontos - qtd_erro * (max_pontos/10) )
        //arredondar para baixo, para obter
        menor inteiro
        exibe_feedback("erro");
    }
    qtd_notas_pauta --
}

IF qtd_notas_jogador = qtd_notas_pauta || qtd_vidas == 0
    fim de jogo
    IF pontos < 0
        pontos=0;

//e possivel calcular a eficacia do jogador com base
//na quantidade de notas da musica e na quantidade de erros/acertos

max_pontos = qtd_notas_pauta

desempenho = pontos / max_pontos
IF (9 <= desempenho <= 10)
    qtd_estrelas = 3
ELSE IF (8 <= desempenho < 9)
    qtd_estrelas = 2
ELSE IF (7 <= desempenho < 8)
    qtd_estrelas = 1
ELSE
    qtd_estrelas = 0

```

Orientações de GUI e usabilidade: apresentar *feedback* visual e sonoro para o usuário:

- a pauta deve permitir exibir as notas de duas oitavas.
- ao pressionar a nota, mudar sua cor (verde em caso de acerto / vermelho em caso de erro) + tocar som (em caso de acerto, tocar som correspondente à nota exibida na partitura; em caso de erro, tocar "som de erro").

Protótipo de baixa fidelidade: Figura: 20

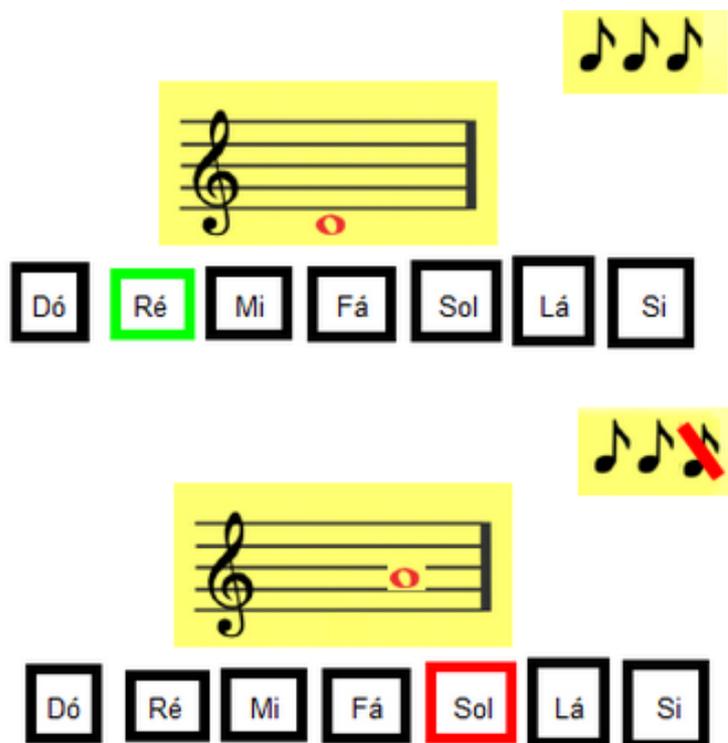


Figura 20 – Protótipo de baixa fidelidade da fase: Notas na pauta. Arte: Wemerson Geisler

4.1.5.6 Figuras de tempo

Objetivo: o jogador deve indicar corretamente a nota (botão) que corresponde à figura de tempo exibida na tela. Uma orientação visual é apresentada ao jogador - figura musical que deve ser identificada, tempos musicais (botões) a serem indicadas pelo jogador.

Elementos de música: figuras de tempo.

Descrição: o jogador irá ganhar pontos se indicar o valor de tempo correspondente à figura exibida na tela. A quantidade de pontos obtida dependerá de acertar ou não qual foi a nota exibida na pauta. O jogador será penalizado (redução de 10% do valor máximo de pontos possíveis) se indicar um tempo que não corresponde àquela representada na figura. Se ao final da fase o jogador estiver com pontuação negativa, a pontuação final será considerada zero. O jogador possui um número limitado de chances (vidas), que irão limitar a quantidade máxima de erros. Na versão inicial não há tempo para realizar a tarefa, mas uma variação seria incluir um *timer* com tempo limite para que o jogador indique a nota correspondente àquela exibida na figura. Variações no nível de dificuldade pode ser dadas por *timer* com tempo limite para indicar o tempo correspondente.

Regras: Pré-condição: o jogador possui uma quantidade P de pontos e E de estrelas; o jogador possui quantidade E de estrelas suficiente para desbloquear essa fase

(tarefa). Pós-condição: o jogador irá acumular a quantidade máxima de pontos e estrelas que recebeu ao completar a tarefa (mini-jogo), ou seja, se o jogador realizar a tarefa duas ou mais vezes, somente a pontuação máxima deve ser registrada.

Especificação:

```

qtd_figuras_tempo = N //quantidade de figuras que possui —
//esse numero e conhecido a priori e define
//o maximo possivel de acertos

pontos = 0; qtd_notas_jogador = 0;
qtd_acertos = 0; qtd_eros = 0;

ENQUANTO (qtd_figuras_tempo > 0) {
    //acerto implica usuario pressionar botao
    //correspondente a figura que foi exibida na tela

    IF acerto == "correct" { // jogador pressionou o
        //botao coreto correspondente a figura exibida na tela
        pontos += 1
        qtd_figuras_jogador++; qtd_acertos++
        exhibe_feedback("correct");

    } ELSE { // jogador errou, pois pressionou o botao incorreto
        qtd_figuras_jogador++; qtd_eros++ qtd_vidas—
        // atualiza pontos conforme a quantidade de vidas
        //perdidias (-10% do maximo de pontos para cada
        vida perdida)
        pontos = FLOOR( pontos - qtd_eros * (max_pontos/10) )
        //arredondar para baixo, para obter
        menor inteiro
        exhibe_feedback("erro");
    }
    qtd_figuras_tempo—
}

IF qtd_figuras_jogador= qtd_figuras_tempo—|| qtd_vidas == 0
fim de jogo
IF pontos < 0

```

```
pontos=0;

//e possivel calcular a eficacia do jogador com base na
//quantidade de figuras da tempo e na quantidade de erros/acertos

max_pontos = qtd_figuras_tempo

desempenho = pontos / max_pontos
IF (9 <= desempenho <= 10)
    qtd_estrelas = 3
ELSE IF (8 <= desempenho < 9)
    qtd_estrelas = 2
ELSE IF (7 <= desempenho < 8)
    qtd_estrelas = 1
ELSE
    qtd_estrelas = 0
```

Orientações de GUI e usabilidade: apresentar *feedback* visual e sonoro:

- ao pressionar o tempo, mudar sua cor (verde em caso de acerto / vermelho em caso de erro) + tocar som (em caso de acerto, tocar “som de acerto”; em caso de erro, tocar "som de erro")

Protótipo de baixa fidelidade: Figura: 21

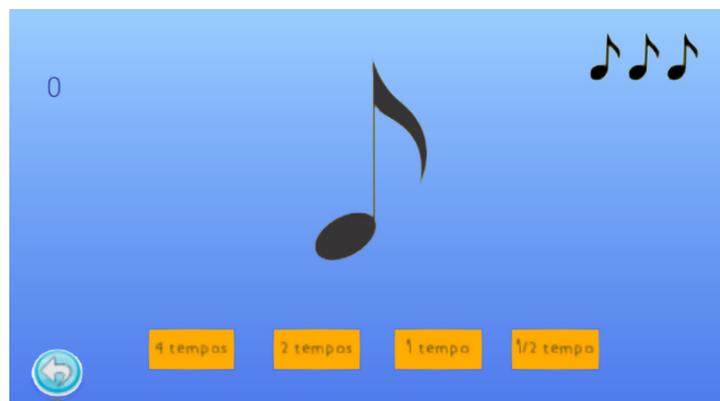


Figura 21 – Protótipo de baixa fidelidade da fase: Figuras de tempo. Arte: Ana Carolina Brandão

4.2 Musicália: versão demo

Uma versão de demonstração foi desenvolvida para ajudar na realização de testes e avaliação da proposta do jogo. Nessa versão foram implementados os minijogos apresentados na seção 4.1.5. Ela foi disponibilizada somente para dispositivos com sistema operacional *Android*, mas poderá ser futuramente exportada também para outras plataformas.

O desenvolvimento de jogos é uma atividade interdisciplinar, que requer diferentes habilidades (ex. desenvolvimento de software, design de artes, design de sons, design do jogo). Pode-se utilizar como exemplo para ilustrar essa necessidade de diferentes papéis e habilidades a funcionalidade ou efeito de *parallax scrolling*, disponível na versão demo do jogo. Esse efeito consiste em criar a ilusão de profundidade e movimento em planos 2D, e é obtido pela movimentação das imagens do fundo em velocidade mais lenta que as imagens que estão no primeiro plano. A implementação desse efeito na versão demo envolveu atividades de programação e de arte. Se fosse incluída uma trilha sonora, por exemplo, teríamos mais uma habilidade adicional necessária.



Figura 22 – Implementação do efeito *parallax scrolling* no Reino da Melodia, envolvendo atividades de programação e arte. Arte: Danilo Caldeira

A partir da tela inicial do jogo (Figura 23a), o jogador será direcionado para um mapa (Figura 23b) que corresponde ao mundo mágico de Musicália. Cada ilha corresponde a um dos reinos de Musicália: Ritmo, Notas, Melodia, Harmonia, Pauta e Silêncio.



Figura 23 – a) Tela inicial do jogo (Arte: Dasayeve Xavier). b) Mapa do mundo mágico de Musicália (Arte: Danilo Caldeira)

O jogador deverá realizar diferentes missões, em cada um dos reinos. Cada missão corresponde a um conjunto de tarefas que ele precisará completar naquele reino. As tarefas estão associadas a passar pelos minijogos. Ou seja, para completar uma missão em um reino, o jogador deverá passar pelos minijogos disponíveis no reino e que correspondem à essa missão específica. A Figura 24 ilustra a apresentação da missão para o jogador, que é feita com a interlocução de um personagem primário do jogo.



Figura 24 – a) Tela Missão onde o jogador recebe a missão atual e é redirecionado para o reino da próxima fase. - Arte: Dasayeve Xavier

Caso nunca tenha acessado o jogo, apenas a primeira missão estará liberada. Para liberar acesso as fases seguintes o jogador durante as missões terá que conquistar no mínimo duas estrelas, as estrelas indicam o desempenho do jogador naquela missão. As fases bloqueadas estarão com um cadeado, indicando que o desempenho mínimo ainda não foi alcançado. (Figura 25). Além das estrelas, o *game design* agrega outros componentes de premiação e indicadores de desempenho, tais como pontos, artefatos (pedaços de amuletos, instrumentos) e "vidas".



Figura 25 – Cadeado indica fase bloqueada e o numero de estrelas no cadeado indica a quantidade necessária para desbloquear a fase. - Arte: Danilo Caldeira

Ao acessar cada reino e acessar uma fase, o jogador receberá instruções com objetivos específicos de cada desafio (Figura 26), terá ainda que completar uma missão que aborda de forma lúdica algum tipo de conhecimento sobre música. Após o fim de cada minijogo, o jogador receber *feedback* de seu desempenho, resultado de sua pontuação e habilita a próxima missão.



Figura 26 – Após acessar o reino e abrir uma fase, o jogador recebe instruções sobre seu objetivo. - Cenário: Danilo Caldeira - Personagem: Dasayeve Xavier - ícones: Wemerson Geisler

Por exemplo, uma das missões do reino do ritmo (Figura 27) envolve pressionar a tecla correta do teclado no momento certo para acompanhar o tempo da música a ser tocada.

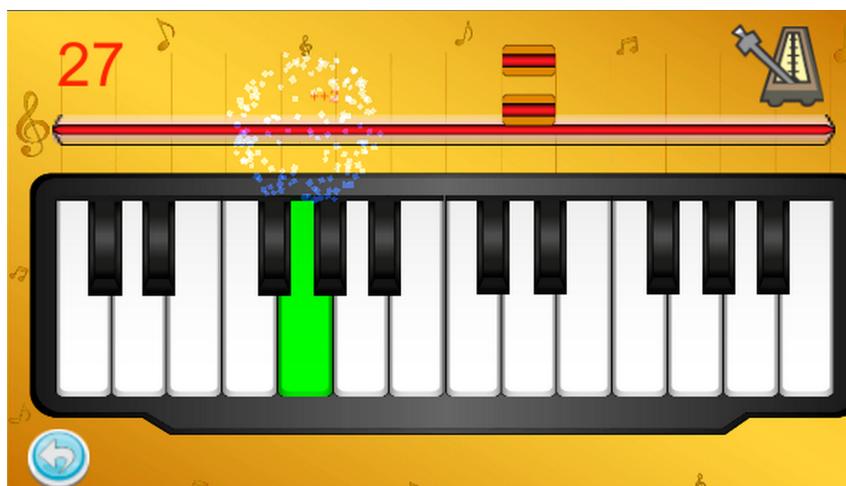


Figura 27 – Missão do reino do ritmo, onde o jogador deve pressionar a tecla correta do teclado no momento certo para acompanhar o tempo da música a ser tocada - Arte: Danilo Caldeira

No reino das notas, uma das missões especificadas no *game design*, baseia-se em associar o nome das notas com a tecla correspondente em um teclado (Figura 28).

No reino da Pauta, para se obter sucesso na missão, o jogador deverá obter conhecimento e correlacionar corretamente as notas exibidas na figura da pauta tocando a música correspondente (Figura 29).



Figura 28 – Missão do reino das notas, onde o jogador deve pressionar a tecla correspondente à nota exibida na tela e tocar a música corretamente - Arte: Danilo Caldeira

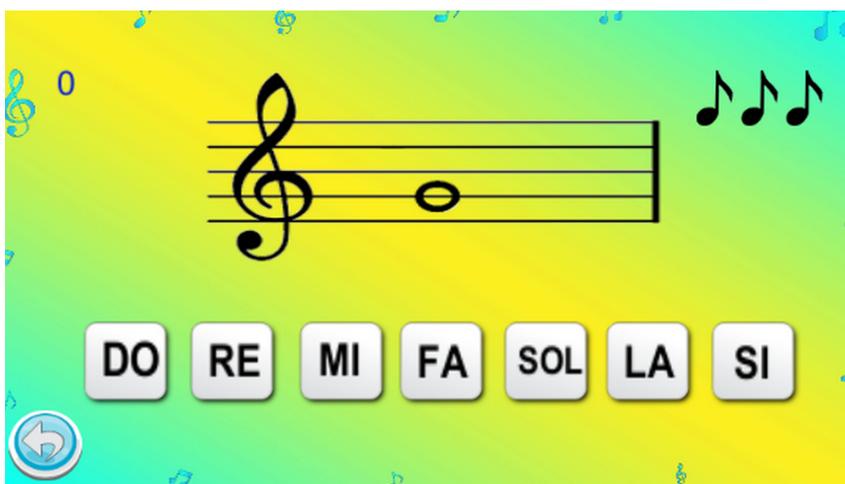


Figura 29 – Missão do reino da Pauta, onde o jogador deve pressionar a tecla correspondente à nota exibida na figura da partitura e tocar a música corretamente. - Arte: Danilo Caldeira

A versão de demonstração possui outros exemplos que foram implementados, seus protótipos são apresentados nas figuras 30, 31, 32. Uma visão geral sobre cada um deles pode ser encontrada na versão do *Game Design Document*. Informações sobre o projeto podem ser obtidas no link <www.leds.ufop.br/musicalia>. Todos os minijogos implementados estão descritos na seção 4.1.5

Além de testes funcionais realizados pela equipe durante o processo de desenvolvimento, foram realizadas avaliações informais em testes elaborados na disciplina de Avaliação de Sistemas interativos, ofertada na UFOP. Nesse processo, voluntários e alunos se dispuseram para testar o jogo, e apontar sugestões de melhorias, dentre eles, poucos com



Figura 30 – Missão do reino do ritmo, onde o jogador deve associar o tempo correto de acordo com a figura de tempo apresentada na tela - Arte: Danilo Caldeira

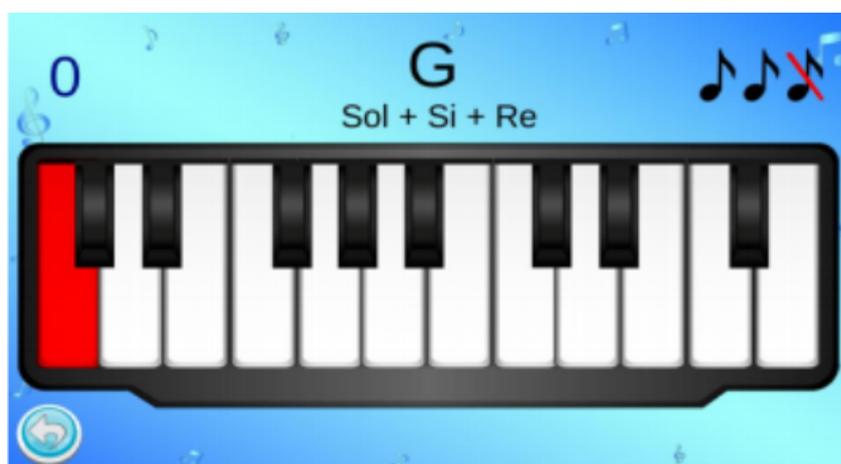


Figura 31 – Missão do reino da harmonia, onde o jogador deve compor os acordes corretamente de acordo com a notação apresentada na tela. - Arte: Danilo Caldeira

conhecimentos prévios sobre teoria musical, o que ajudou a formar opiniões de públicos-alvo diferentes. Foram realizadas avaliações de observações e sugestões sucintas porém, de grande ajuda para nós. Todas as sugestões e pontos importantes observados pela equipe eram documentados em forma de anotações. Caso quisessem, os voluntários podiam opinar sobre a experiência de interação. Toda essa etapa de avaliação, destinou-se a testes de usabilidade e contexto de uso e foi de grande ajuda na continuidade da concepção e desenvolvimento do jogo.

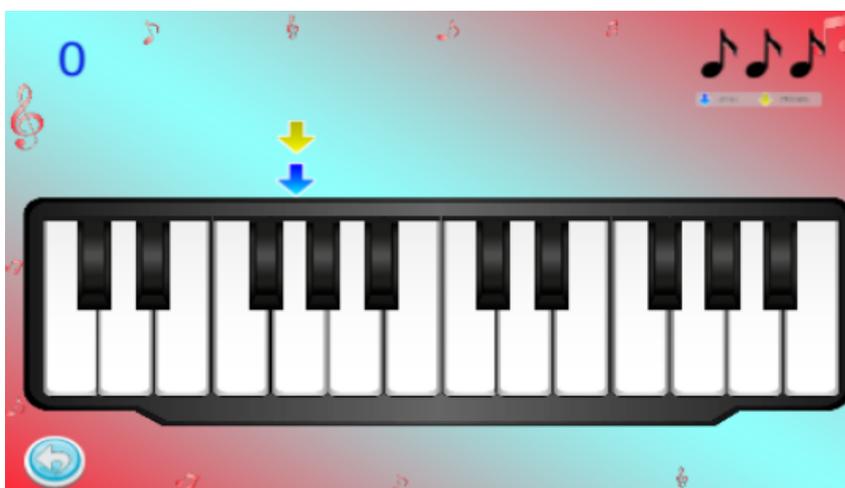


Figura 32 – Missão do reino da melodia, onde o jogador deve seguir as instruções das setas e tocar a melodia da musica indicada. - Arte: Danilo Caldeira

5 Considerações finais

No Brasil, a educação musical é conteúdo obrigatório do componente curricular Arte. Mas sua efetiva adoção se torna um desafio diante da escassez de recursos básicos tais como instrumentos musicais e profissionais especializados.

A utilização de jogos sérios como ferramenta na educação é uma maneira de incentivar e motivar o interesse dos jogadores que desejam obter conhecimento, que de uma forma lúdica e divertida, vão exercitar e definir meios de superação durante as etapas do jogo. Entretanto o trabalho mostrou que há uma carência de projetos sendo desenvolvidos para essa finalidade.

Nesse contexto, foi proposta a concepção, projeto e desenvolvimento de um jogo sério para iniciação musical voltado para crianças e jovens, o Musicália. Assim, o trabalho multidisciplinar envolveu diferentes áreas: jogos sérios, teoria musical, ensino de música, criação de documentos, desenvolvimento de protótipos de baixa e alta fidelidade, desenvolvimento de uma versão de demonstração, *design* e avaliação de usabilidade e de experiência de uso.

O desenvolvimento de um jogo sério vai muito além de códigos de programação, envolve diversas atividades que precisam ser desenvolvidas por uma equipe, tais como elaborar história do jogo, projetar o fluxo de interação, definir regras e mecânica do jogo, criar especificações, projetar, desenvolver e testar o software. Foram vários os desafios enfrentados no desenvolvimento desse trabalho.

Os trabalhos futuros contemplam projetar e implementar melhorias na atual versão do jogo, como por exemplo especificar e desenvolver novas missões (fases); melhorias de usabilidade e jogabilidade; inclusão de novos idiomas, exportar o jogo para iOS e desenvolver uma versão de demonstração mais completa que possa ser disponibilizada em lojas de aplicativos.

Referências

- ADDESSI, A. R.; PACHET, F. Young children confronting the continuator, an interactive reflective musical system. *Musicae Scientiae*, SAGE Publications Sage UK: London, England, v. 10, n. 1_suppl, p. 13–39, 2006. Citado na página 22.
- ALIEL, L.; GOHN, D. Jogos eletrônicos musicais e EAD: contingência de ferramentas para aprendizagem instrumental. *SIED: EnPED-Simpósio Internacional de Educação a Distância e Encontro de Pesquisadores em Educação a Distância 2012*, 2012. Citado na página 21.
- ARANHA, G. Jogos eletrônicos como um conceito chave para o desenvolvimento de aplicações imersivas e interativas para o aprendizado. *Ciências & Cognição*, Instituto de Ciências Cognitivas, v. 7, n. 1, p. 105–110, 2006. Citado na página 15.
- ATLASSIAN. *Branch Workflow*. 2018. <<https://www.atlassian.com/git/tutorials/comparing-workflows#feature-branch-workflow>>. Acesso em: 10 mai. 2018. Citado na página 28.
- BACHEN, C. M.; RAPHAEL, C. Social flow and learning in digital games: A conceptual model and research agenda. In: *Serious games and edutainment applications*. [S.l.]: Springer, 2011. p. 61–84. Citado na página 19.
- BARBOSA, S.; SILVA, B. *Interação humano-computador*. [S.l.]: Elsevier Brasil, 2010. Citado na página 20.
- BORDINI, R. A. et al. Musikinésia: Jogo eletrônico gratuito para a aprendizagem de teclado e leitura musical. In: *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*. [S.l.: s.n.], 2016. v. 5, n. 1, p. 225. Citado na página 16.
- BRANDÃO, A. C. Musical Instructor: concepção e desenvolvimento de um jogo educativo para o ensino de música. 2016. Citado 2 vezes nas páginas 16 e 30.
- BRASIL. Lei nº 11.769, de 18 de Agosto de 2008. altera a ldb nº 9.394/96, para dispor sobre a obrigatoriedade do ensino da música na educação básica. *Diário Oficial da União*, 2008. Citado 2 vezes nas páginas 15 e 18.
- BRASIL. *Diretrizes curriculares nacionais gerais da educação básica*. [S.l.]: Brasília: MEC, SEB, DICEI, 2013. Citado 2 vezes nas páginas 15 e 18.
- BRÉSCIA, V. L. P. Educação musical: bases psicológicas e ação preventiva. *São Paulo: Átomo*, p. 18–9, 2003. Citado na página 15.
- CARDOSO, A.; GIRALDELLO, A. G.; M, B. N. A. Tabuada legal: um jogo sério para o ensino de multiplicações. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2013. v. 24, n. 1, p. 376. Citado na página 16.
- CARVALHO, A. Jogos digitais brasileiros para educação musical. *Augusto Guzzo Revista Acadêmica*, v. 1, n. 15, p. 74–91, 2015. Citado 3 vezes nas páginas 18, 21 e 23.

- CHIARELLI, L. K. M.; BARRETO, S. d. J. A importância da musicalização na educação infantil e no ensino fundamental: A música como meio de desenvolver a integração do ser. *Recreart, Santiago de Compostela*, 2005. Citado na página 18.
- CORREA, A. G. et al. AR Musical App for Children's Musical Education. In: *IEEE. 2016 IEEE International Symposium on Consumer Electronics (ISCE)*. [S.l.], 2016. p. 125–126. Citado na página 21.
- CORREA, A. N. Bebês produzem música? O brincar-musical de bebês em berçário. 2013. Citado na página 18.
- COUTO, A. C. N. do; SANTOS, I. R. S. Por que vamos ensinar música na escola? Reflexões sobre conceitos, funções e valores da educação musical escolar. *Opus*, v. 15, n. 1, p. 110–125, 2009. Citado na página 15.
- CRAWFORD, C. The art of digital game design. *Vancouver: Washington State University*, 1982. Citado na página 19.
- DJAOUTI, D. et al. Origins of serious games. In: *Serious games and edutainment applications*. [S.l.]: Springer, 2011. p. 25–43. Citado na página 19.
- DONDLINGER, M. J. Educational video game design: A review of the literature. *Journal of applied educational technology*, v. 4, n. 1, p. 21–31, 2007. Citado na página 16.
- ECYBERMISSON. *Mission to magmanon*. 2008. Disponível em: <<http://www.agame.com/game/mission-to-magmanon>>. Acesso em: 20 dez. 2018. Citado na página 22.
- ENCORE. *Encore*. 2018. Disponível em: <<https://www.encore.com>>. Acesso em: 20 mai. 2018. Citado na página 15.
- FALCÃO, T. P.; GOMES, A. S. Design de interfaces tangíveis educacionais: uma metodologia baseada em contexto. In: *ACM. Proceedings of VII Brazilian symposium on Human factors in computing systems*. [S.l.], 2006. p. 13–16. Citado na página 20.
- FARIA, L. C. F. de; OLIVEIRA, P. L. L. M. G. de. O jogo eletrônico como estímulo de aprendizagem musical. In: *Colloquium Humanarum*, v. 8, n. Especial, jul.-dez 2011, 2011. Disponível em: <<http://www.unoeste.br/site/pos/enapi/2011/suplementos/documentos/Humanarum-PDF/CDArtes.pdf>>. Acesso em: 22 dez. 2018. Citado na página 21.
- FARIAS, E. H. et al. Movabletrando: Jogo de movimentos para alfabetizar crianças com down. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2013. v. 24, n. 1, p. 316. Citado na página 16.
- FICHEMAN, I. K. et al. Editor musical: uma aplicação para a aprendizagem de música apoiada por meios eletrônicos interativos. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2003. v. 1, n. 1, p. 176–185. Citado na página 15.
- FICHEMAN, I. K. et al. Portal edumusical: Telemática aplicada à educação musical. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2004. v. 1, n. 1, p. 497–506. Citado 2 vezes nas páginas 15 e 18.

- FRITSCH, E. F.; VICCARI, R. M.; MORAES, Z. O. d. Desenvolvimento de software educacional para a música: Str-sistema de treinamento rítmico. *Simpósio Brasileiro de Computação e Música*, v. 5, p. 209–218, 1998. Citado na página 15.
- GAINZA, V. H. de. *Estudos de psicopedagogia musical*. [S.l.]: Grupo Editorial Summus, 1988. v. 31. Citado na página 18.
- GITHUB. *GitHub*. 2018. Disponível em: <<https://github.com/>>. Acesso em: 10 mai. 2018. Citado na página 28.
- GOHN, M. d. G.; STAVRACAS, I. O papel da música na educação infantil. *EccoS Revista Científica*, Universidade Nove de Julho, v. 12, n. 2, 2010. Citado na página 18.
- GRÜBEL, J. M.; BEZ, M. R. Jogos educativos. *RENOTE*, v. 4, n. 2, 2006. Citado na página 19.
- GUITARPRO. *Guitar Pro*. 2018. Disponível em: <<https://www.guitar-pro.com/en/index.php>>. Acesso em: 20 mai. 2018. Citado na página 15.
- HEJLSBERG, A.; WILTAMUTH, S.; GOLDE, P. *The C# programming language*. [S.l.]: Adobe Press, 2006. Citado na página 24.
- JOYTUNES. *Simply Piano*. 2018. Disponível em: <https://play.google.com/store/apps/details?id=com.joytunes.simplypiano&hl=pt_BR>. Acesso em: 20 jul. 2018. Citado 2 vezes nas páginas 16 e 22.
- KARLINI, D.; RIGO, S. J. Abclingo: Integrando jogos sérios e mineração de dados educacionais no apoio ao letramento. *Proceedings of SBGames*, p. 1149–1152, 2014. Citado na página 16.
- KRÜGER, S. E. Educação musical apoiada pelas novas tecnologias de informação e comunicação (TIC): pesquisas, práticas e formação de docentes. *Revista da ABEM*, v. 14, n. 14, 2014. Citado na página 15.
- LESCARGOT, H. *Music Blox*. 2012. Disponível em: <<http://www.agame.com/game/music-blox.html>>. Acesso em: 20 dez. 2018. Citado na página 21.
- LIANE, H.; BEN, L. D. Ensino de música: propostas para pensar e agir em sala de aula. *São Paulo: Moderna*, p. 113–125, 2003. Citado na página 18.
- LIMA, T. F. M. et al. Playing against dengue - Design and development of a serious game to help tackling dengue. In: IEEE. *Proceedings of the 5th International Conference on Serious Games and Applications for Health*. [S.l.], 2017. Citado na página 16.
- MICHAEL, D. R.; CHEN, S. L. *Serious games: Games that educate, train, and inform*. [S.l.]: Muska & Lipman/Premier-Trade, 2005. Citado na página 15.
- MICHAEL, D. R.; CHEN, S. L. *Serious games: Games that educate, train, and inform*. [S.l.]: Muska & Lipman/Premier-Trade, 2005. Citado na página 19.
- MOCKUP.IO. *Mockup.io*. 2018. Disponível em: <<https://mockup.io/about/>>. Acesso em: 29 mai. 2018. Citado 2 vezes nas páginas 25 e 35.
- MUSESCORE. *Muse Score*. 2018. Disponível em: <<https://musescore.com>>. Acesso em: 20 mai. 2018. Citado na página 15.

- MUSICALEAR. *MusicalEar*. 2016. Disponível em: <<https://www.microsoft.com/en-us/p/musicalear/9nblggh0d72v?activetab=pivot:overviewtab>>. Acesso em: 20 jul. 2018. Citado na página 16.
- NOGUEIRA, B. P. Guitar hero: novas práticas de consumo e cultura auditiva na música através dos videogames. *Contemporanea-Revista de Comunicação e Cultura*, v. 7, n. 2, 2010. Citado na página 21.
- NORTON, T. *Learning C# by developing games with unity 3D*. [S.l.]: Packt Publishing Ltd, 2013. Citado na página 24.
- PASSOS, E. B. et al. Tutorial: Desenvolvimento de jogos com unity 3d. In: *VIII Brazilian Symposium on Games and Digital Entertainment*. [S.l.: s.n.], 2009. p. 1–30. Citado na página 26.
- PENNA, M. Professores de música nas escolas públicas de ensino fundamental e médio: uma ausência significativa. *Revista da ABEM*, v. 10, n. 7, 2014. Citado na página 15.
- PROCOPIO, A. A importância da musicalidade na educação infantil. 2015. Citado na página 18.
- PROTOPSALTIS, A. et al. Serious games and formal and informal learning. *ELearning Papers*, n. 25, 2011. Citado na página 20.
- QUEIROZ, L. R. S. Música na escola: aspectos históricos da legislação nacional e perspectivas atuais a partir da lei 11.769/2008. *Revista da ABEM*, v. 20, n. 29, 2013. Citado na página 15.
- QUEIROZ, L. R. S.; MARINHO, V. M. Educação musical nas escolas de educação básica: caminhos possíveis para a atuação de professores não especialistas. *Revista da ABEM*, v. 15, n. 17, 2014. Citado na página 15.
- RODRIGUES, H. F.; MACHADO, L. d. S.; VALENÇA, A. M. G. Definição e aplicação de um modelo de processo para o desenvolvimento de serious games na área de saúde. In: *Proc. Congresso da Sociedade Brasileira de Computação-Workshop de Informática Médica*. [S.l.: s.n.], 2010. p. 1532–1541. Citado 2 vezes nas páginas 19 e 20.
- ROGERS, S. *Level Up! The guide to great video game design*. [S.l.]: John Wiley & Sons, 2014. Citado na página 19.
- ROGERS, Y.; SHARP, H.; PREECE, J. *Design de Interação*. [S.l.]: Bookman Editora, 2013. Citado na página 19.
- ROLLINGS, A.; ADAMS, E. *Fundamentals of Game Design. New Riders*. [S.l.]: Pearson Education: Berkeley, California, 2010. Citado na página 19.
- SABBAGH, R. *Scrum: Gestão ágil para projetos de sucesso*. [S.l.]: Editora Casa do Código, 2014. Citado na página 24.
- SANTIAGO, G. L. A. et al. O incrível músico das neves. 2014. Disponível em: <http://educacaomusical.sead.ufscar.br/jogos/OIMN/O_incrivel_musico_das_neves2/WebPlayer.html>. Citado na página 16.

- SANTIAGO, G. L. A. et al. Uma odisséia de acordes. 2015. Disponível em: <<http://audiovisual.uab.ufscar.br/em/glauber/epm2/odisseia/WebPlayer.html>>. Citado na página 16.
- SANTOS, T. M.; FILIPPO, D.; PIMENTEL, M. Computação ubíqua para apoiar a educação musical: explorações com o Makey Makey. In: *Anais do Workshop de Informática na Escola*. [S.l.: s.n.], 2015. v. 21, n. 1, p. 330. Citado na página 15.
- SCAICO, P. D. et al. Implementação de um jogo sério para o ensino de programação para alunos do ensino médio baseado em m-learning. In: *XX Workshop sobre Educação em Computação (WEI)*. [S.l.: s.n.], 2012. p. 3–43. Citado na página 16.
- SEGA. *Lets Tap*. 2009. Disponível em: <<https://www.nintendo.pt/Jogos/Wii/Let-s-Tap-281760.html#gameDetails>>. Acesso em: 20 mai. 2018. Citado na página 21.
- SONYCSL. *Continuator*. 2016. Disponível em: <<https://www.sonyosl.co.jp/paris/2807/>>. Acesso em: 28 mai. 2018. Citado na página 22.
- SOUZA, C. E. de; JOLY, M. C. L. A importância do ensino musical na educação infantil. 2010. Citado na página 18.
- UFSCAR. *T-Acordes*. 2018. Disponível em: <<http://educacaomusical.sead.ufscar.br/jogos/T-Acordes.php>>. Acesso em: 20 jul. 2018. Citado na página 21.
- UFSCAR. *T-Notas*. 2018. Disponível em: <<http://educacaomusical.sead.ufscar.br/jogos/T-Notas.php>>. Acesso em: 20 jul. 2018. Citado na página 21.
- UNITY. *Unity 3D*. 2018. Disponível em: <<https://docs.unity3d.com/Manual/index.html>>. Acesso em: 20 mai. 2018. Citado 4 vezes nas páginas 10, 24, 26 e 27.
- UNITY. *Unity 3D - Termos de uso - Get Started with Unity Persona*. 2018. Disponível em: <<https://store.unity.com/pt/download?ref=personal>>. Acesso em: 20 mai. 2018. Citado na página 26.
- VEBER, A.; ROSA, T. B. da. Jogos digitais online e ensino de música: propostas para a prática musical em grupo. *Música na Educação Básica*, v. 4, n. 4, 2017. Citado na página 23.
- ZENHUB. *ZenHub*. 2018. Disponível em: <<https://www.zenhub.com/>>. Acesso em: 16 abr. 2018. Citado na página 28.

Anexos

ANEXO A – Conceitos musicais

- **Acorde:** qualquer conjunto harmônico de três ou mais notas que se ouve como se estivessem soando simultaneamente.
- **Bemol:** Abaixa um semitom.
- **Cifra:** é um sistema de notação musical usado para indicar, por meio de símbolos gráficos ou letras, os acordes a serem executados.
- **Clave:** é o sinal colocado no início da pauta, sobre determinada linha, para dar nome às notas.
- **Compasso:** agrupamentos de tempo. A duração de tempo que ocupa todo o compasso denomina-se unidade de compasso. É unidade métrica constituída de tempos reunidos em porções iguais (divisão da pauta em traços). Tipos de compasso comuns: binário (2/4), ternário (3/4), quaternário (4/4).
- **Composições:** é uma criação original de música feita para repetidas execuções.
- **Figuras de tempo:** valores de pausas, ou duração das notas.
- **Harmonia:** a combinação de sons simultâneos.
- **Instrumento musical:** objeto construído com o propósito de produzir música. Nomes, conceitos, e timbre.
- **Intervalo:** espaço existente entre duas notas musicais. Intervalo simples x composto. Intervalo melódico x harmônico.
- **Ligadura:** faz-se a primeira nota e a segunda apenas soa até completar o tempo.
- **Melodia:** conjunto de notas dadas de forma sucessiva, e que fazem um sentido musical.
- **Notas musicais:** símbolos usados para representar os sons.
- **Notas naturais:** Dó, Ré, Mi, Fá, Sol, Lá, Si.
- **Oitava:** é o intervalo entre uma nota musical e outra com a metade ou o dobro de sua frequência.
- **Pauta:** conjunto de cinco linhas e quatro espaços que serve para escrever uma peça musical.

- **Ritmo:** a ideia de ritmo se baseia na divisão ordenada dos tempos e acompanhamento musical.
- **Semitom:** É o menor intervalo existente entre dois sons que o ouvido humano ocidental pode perceber e classificar.
- **Sustenido:** Eleva um semitom.
- **Tom:** É o intervalo existente entre dois sons, formado por dois semitons.



UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS
COLEGIADO DO CURSO DE ENGENHARIA DE COMPUTAÇÃO



ANEXO III – Termo de Responsabilidade

TERMO DE RESPONSABILIDADE

Eu, Womerson Queiroz de Lima,
declaro que o texto do trabalho de conclusão de curso intitulado
"Musicalia - Concepção e desenvolvimento de um jogo
para iniciação musical" é de
minha inteira responsabilidade e que não há utilização de texto, material fotográfico, código
fonte de programa ou qualquer outro material pertencente a terceiros sem as devidas
referências ou consentimento dos respectivos autores.

João Monlevade, 12 de Julho de 2018

Womerson Queiroz de Lima
Assinatura do aluno

Certifico que o aluno Wemerson Geisler de Lima, autor do trabalho de conclusão de curso intitulado “Musicália - Concepção e desenvolvimento de um jogo para iniciação musical” efetuou as correções sugeridas pela banca examinadora e que estou de acordo com a versão final do trabalho.



(Tiago França Melo de Lima)

Orientador

Ouro Preto, 05 de Junho de 2019.