

**CENTRO PAULA SOUZA
ETEC DE ITAQUAQUECETUBA
ETIM de Informática**

**Aláiny Alves de Andrade
Danilo Almeida R. de Amarante
Samuel Pereira dos Santos
Vanessa Pereira dos Anjos**

MASTERSCORE

**Itaquaquecetuba
2015**

Alaíny Alves de Andrade
Danilo Almeida R. de Amarante
Samuel Pereira dos Santos
Vanessa Pereira dos Anjos

MASTERSCORE

Trabalho de Conclusão de Curso apresentado ao Ensino Médio Integrado ao Técnico em Informática da Etec de Itaquaquecetuba, orientado pelos Professores Alexandro Tadeu e Mônica Oliveira, como requisito parcial para obtenção do título do técnico em Informática.

Itaquaquecetuba

2015

Aláiny Alves de Andrade
Danilo Almeida R. de Amarante
Samuel Pereira dos Santos
Vanessa Pereira dos Anjos

MASTER SCORE

ETEC DE ITAQUAQUECETUBA, 16/11/2015

BANCA EXAMINADORA

Eduardo Chagas Ferreira
Especialista em engenharia de sistema em tecnologia Java
ETEC de Itaquaquecetuba

Kelly Cristiane de Oliveira Dalpozzo
Bacharel em sistema da informação
ETEC de Itaquaquecetuba

Mônica Oliveira Raimundo
Licenciada em tecnologia em informática
ETEC de Itaquaquecetuba

Dedicamos a todos os estudiosos musicistas que, graças a eles, foi possível fundamentar todas as teorias e requisitos da música utilizados para concretização do trabalho.

Dedicamos também aos nossos professores que, com muito empenho, tem nos orientado nas nossas dificuldades e assim, feito com que obtivéssemos louvor em nosso trabalho.

Agradecemos, a priori, a Deus que nos tem sustentado de pé nos momentos mais inoportunos, dando força, saúde, incentivo e ajuda.

À instituição, ETEC de Itaquaquecetuba, e todo o seu corpo docente que oportunizaram uma janela que hoje vislumbrou um horizonte superior, eivado pela acendrada confiança no mérito e ética aqui presentes.

Aos nossos orientadores Mônica e Alexandro, pelo suporte que sempre esteve à disposição nos poucos tempos que lhes couberam, pelas suas correções e incentivos.

E aos nossos pais, pelo amor, incentivo e apoio incondicional.

“Música é a arte de manifestar os diversos afetos da nossa alma mediante ao som.”

PASCHOAL BONA

RESUMO

O objetivo do presente trabalho é desenvolver um sistema, Master Score (Mestre da Partitura ou Partitura Mestre em português), cuja finalidade é oferecer uma ferramenta de aprendizagem intuitiva e prática aos estudantes da teoria musical. Visto que existe uma defasagem de recursos informatizados para o aprendizado da música no mercado. O desenvolvimento do trabalho foi elaborado em três capítulos, sendo eles: Análise de Dados, Banco de Dados e Desenvolvimento do Sistema. E foi feito através de pesquisas bibliográficas e pesquisa de campo, seguindo os parâmetros do método científico que busca capturar e analisar a realidade do ambiente pesquisado.

Palavras-chaves: Aprendizagem, intuitiva, prática, música, teoria.

ABSTRACT

The objective of work presentisto develop a software, Master Score (Mestre da Partitura or Partitura Mestrein portuguese), whos epurposeisto provide a tool learning intuitive and practical for the musical theory students. Since thereis a lag of computerized resources for the music's learning at the market. The work develop ment was done in three chapters, which are: Data Analysis, Database and System Development. And it was do ne through library research and fiel dresearch, following the parameters of the scientific method that seeks to capture and analyze the reality of there search edenvironment.

Key- word: Learning, intuitive, practical, music ,theory,.

LISTA DE ILUSTRAÇÕES

Figura 1 – Parte da figura musical.....	15
Figura 2 – Figuras musicais	15
Figura 3 – Pentagrama.....	16
Figura 4 – Caso de uso Acesso ao sistema	19
Figura 5 – Casos de Uso Educação Musical.....	19
Figura 6 – Casos de Uso Treinamento.....	20
Figura 7 – Ciclo de Vida Clássico ou cascata	20
Figura 8 – DER.....	24
Figura 9 – MER	25
Figura 10 – Diagrama de classes.....	30
Figura 11 – Tela de abertura com logo e mascote do sistema.....	31
Figura 12 – Tela inicial com a opção de criar uma conta e fazer login.....	31
Figura 13 – Form do cadastro	32
Figura 14 – Tela principal, apresentando as informações do usuário	33

LISTA DE TABELAS

Tabela 1 – Documentação de Caso de Uso.....	52
Tabela 2 – Cadastro.....	52
Tabela 3 – Progresso	53
Tabela 4 – Perfil	53
Tabela 5 – Teste	53

LISTA DE GRÁFICOS

Gráfico 1 – Idade (ALUNO)	38
Gráfico 2 – Sexo (ALUNO)	38
Gráfico 3 – Tempo de Estudo.....	39
Gráfico 4 – Métodos mais utilizados para aprender a teoria musical	39
Gráfico 5 – Maior dificuldade para aprender a teoria musical	40
Gráfico 6 – Técnicas para melhorar o aprendizado da música	40
Gráfico 7 – Conhecimento sobre a informatização na área musical	41
Gráfico 8 – Dificuldades nas ferramentas existentes	41
Gráfico 9 – Ferramentas ajudam na aprendizagem?	42
Gráfico 10 – Idade (PROFESSOR)	45
Gráfico 11 – Sexo (PROFESSOR).....	45
Gráfico 12 – Grau de formação musical	46
Gráfico 13 – Disciplina lecionada	46
Gráfico 14 – Dificuldade dos alunos.....	47
Gráfico 15 – Ferramentas usadas para ensinar a música	47
Gráfico 16 – Opinião sobre a integração da informática	48
Gráfico 17 – Opinião sobre a informatização da música	48

SUMÁRIO

1 INTRODUÇÃO	13
2 ANÁLISE DE DADOS.....	15
2.2 AnálisedeDados	16
2.2.1 Levantamento de Requisitos	17
2.2.2. Especificação de Requisitos	18
2.2.3 Metodologia de Desenvolvimento de Software	20
3 BANCO DE DADOS	21
3.1 LINGUAGEM SQL.....	23
3.2 BrModelo	23
3.3Diagramas de Banco de Dados	24
4 DESENVOLVIMENTO DO SISTEMA.....	25
4.1 Lógica de Programação	25
4.2 Linguagem de Programação.....	27
4.2.1 Csharp.....	27
4.2.2 Ferramentas de Apoio	28
4.3 Diagrama de Classe.....	29
4.4 Apresentação do Sistema	30
5 CONSIDERAÇÕES FINAIS	34
REFERÊNCIAS.....	35
6 APÊNDICE	36
APÊNDICE A – Pesquisa de campo	36
APÊNDICE B – DOCUMENTAÇÃO DE CASO DE USO.....	49
APÊNDICE C – DICIONÁRIOS DE DADOS	52

1 INTRODUÇÃO

É de fundamental importância o uso de métodos que norteiem o aprendizado para obter um ensino de qualidade e na área musical estes métodos, disponíveis hoje, são ligeiramente limitados. O aluno, interessado em aprender acaba cedendo a métodos de aprendizagem pouco instrutivos, tais como livros, por exemplo.

A música é uma arte complexa e que exige muita dedicação dos estudantes. Além da enorme quantidade de teoria ela demanda um esforço muito grande na parte prática, visto que cada instrumento musical possui suas técnicas. O aluno, para se tornar apto a tocar uma música em um determinado instrumento musical tem que, antes, praticar inúmeras vezes.

Existe uma grande dificuldade no aprendizado da teoria musical, o que faz com que os estudantes de música fiquem desanimados, pois a música é uma arte que exige muito treino para tornar-se apto, sendo necessário muita repetição e insistência, o que torna cansativo e pouco intuitivo, a longo prazo, sendo essencial o uso de métodos variados.

A música não é somente uma arte que escutamos e achamos bonito ou feio, através dela é possível perceber os sentimentos ao qual o compositor quis transmitir, é possível integrar “a vida humana em seus aspectos físicos e metafísicos” (SILVEIRA; RIBEIRO, 2012, p. 2).

O estudo e aprendizagem da música é fundamental para obter-se a educação, pois como disse Platão “a educação musical é a parte principal da educação, porque o ritmo e a harmonia têm o poder de penetrar na alma e tocá-la fortemente” (1999 apud SILVEIRA; RIBEIRO, 2012, p. 13).

Tendo em vista que o método mais utilizado para o ensino da música são os livros didáticos dessa área, o aluno apresenta grande dificuldade na compreensão da teoria musical, considerado como um conceito bastante abstrato, pois requer do estudante a identificação do som físico dentro de uma linguagem simbólica (a partitura, por exemplo).

Sendo assim, tanto os alunos iniciantes na educação da música, quanto os já praticantes, compartilham do obstáculo de encontrar uma ferramenta que traga uma abordagem mais prática e didática do conceito musical em seu fundamento.

Em meio ao mercado enorme de estudantes de música, a necessidade de ferramentas intuitivas e práticas para facilitar e nortear o aprendizado do aluno é fundamental. A pesquisa de campo realizada pelo grupo demonstra que mesmo com a existência de aplicativos e sites de aprendizagem musical, existe ainda a dificuldade de fixar o conteúdo – pelo fato destes mesmos sistemas exigirem um pré-conhecimento sobre o assunto e apresentarem um conteúdo muito mais prático a teórico – limitando as habilidades do aluno que acaba, em sua maioria, não compreendendo o que está sendo estudado.

O trabalho tem como objetivo geral realizar um levantamento de informações que apresente metodologias que proporcionem aos estudantes de música, um melhor aprendizado e fixação do conteúdo inicial de forma mais prática. E como objetivo específico desenvolver um sistema que possibilite instruir os alunos que estudam música, desde sua parte teórica até a parte prática, usando conceitos e definições sobre o assunto de forma lúdica e interativa.

A metodologia utilizada para elaboração do trabalho será baseada em pesquisas bibliográficas e pesquisa de campo, seguindo os parâmetros do método científico que busca capturar e analisar a realidade do ambiente pesquisado.

A monografia está estruturada em três capítulos. O primeiro capítulo aborda a respeito da análise de dados, onde é feito o levantamento de requisitos, suas especificações e a metodologia de desenvolvimento de software que é onde falamos sobre UML e ciclo de vida.

O segundo capítulo retrata sobre o banco de dados, pois fala qual foi a linguagem escolhida para guardar todos os dados e também a plataforma que preferimos trabalhar. Está especificado, também, o DER, o MER e os diagramas de banco de dados do software em questão.

Já o terceiro capítulo explana o desenvolvimento do sistema em si, onde fala sobre a lógica de programação, a linguagem utilizada, as ferramentas de apoio, os diagramas de classe e põe fim a apresentação do sistema.

2 ANÁLISE DE DADOS

A música é a arte dos sons, a qual se manifesta os afetos da alma, mediante ao som, sendo um arranjo de combinações melódicas que formam uma sinfonia harmônica rítmica formando assim a música.

O Som define-se por toda e qualquer vibração percebida pelo ouvido humano, havendo duas espécies que nosso ouvido consegue perceber: o natural e o produzido. Considera-se o som natural, aquele que é emitido por fontes naturais (trovões, ventos, etc.), sendo o som produzido, aquele emitido por voz e instrumentos.

Na concepção de som natural e produzido, há ainda outros dois tipos de som, sendo: o musical e o não musical, denominado também por ruído. O som musical é resultado de vibrações sonoras regulares, sendo uniforme e podendo ser grafado (como por exemplo, um acorde de um violão). Já o som não musical, ou som indeterminado, é o ruído, resultado de vibrações sonoras irregulares, que não se podem grafá-lo, como por exemplo, a buzina de um carro, grito, apito, etc.

Dentro da escrita da música, cada figura de som tem sua respectiva notação, como demonstra a figura 1 e 2:



Figura 1 – Parte da figura musical

Fonte: Método de Teoria e Solfejo. 1 ed. São Paulo. Congregação Cristã no Brasil, 2009, p.14.

As figuras de som e de silêncio mais usadas atualmente são:

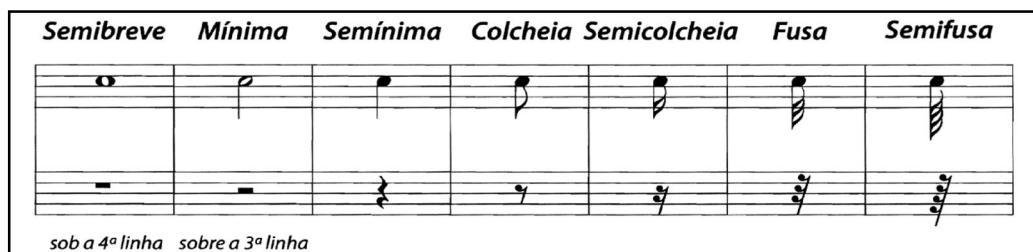


Figura 2 – Figuras musicais

Fonte: Método de Teoria e Solfejo. 1 ed. São Paulo. Congregação Cristã no Brasil, 2009, p.14.

Para podermos representar o som através das figuras (notação musical), utiliza-se o “pentagrama”. Do grego, penta = 5; grama = linha. Um conjunto de 5 linhas e 4 espaços, conforme demonstrado na figura 3.

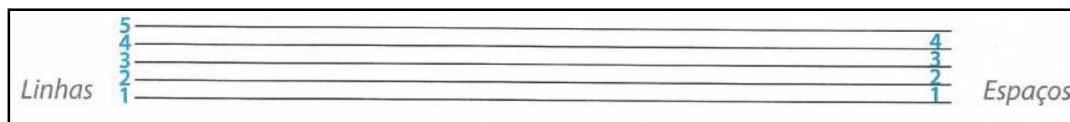


Figura 3 – Pentagrama

Fonte: Método de Teoria e Solfejo: Com aplicação ao hinário. 2ª ed, 2014, p.12.

A música não é apenas uma arte que analisamos positivamente ou negativamente, por ela é possível sentir o que o compositor quis transmitir, é integrar “a vida humana em seus aspectos físicos e metafísicos” (SILVEIRA; RIBEIRO, 2012, p. 2).

A música e a educação estão interligadas, pois conforme Platão “a educação musical é a parte principal da educação, porque o ritmo e a harmonia têm o poder de penetrar na alma e tocá-la fortemente” (1999 apud SILVEIRA; RIBEIRO, 2012, p. 13), logo a música é fundamental na aprendizagem, pois seus aspectos e conceitos interagem com a parte emocional do ser humano relacionando o aprendizado com o cotidiano fortemente.

Em virtude das contribuições da música para com o aprendizado, a necessidade de desenvolver um software com a finalidade de nortear e facilitar o aprendizado da música foi visto como uma maneira eficaz de ajudar os estudantes de música e todos os interessados em aprendê-la.

2.2 Análise de Dados

Devido à complexidade do processo de desenvolvimento de software, foram normalizadas etapas para auxiliar na busca por informações. Visando compreender o domínio do negócio, que nada mais é que “[...] a parte real que é relevante ao desenvolvimento de um sistema” (BEZERRA, 2006, p.20).

É nesta fase também, que são levantadas todas as necessidades do usuário, denominadas de requisitos, dentro dos princípios de análise de sistemas da *Unified Modeling Language*¹(UML).

Segundo a pesquisa de campo² quantitativa, realizada com os estudantes e professores da área de aprendizado musical, que o aluno, principalmente o iniciante,

¹Em português: Linguagem de Modelagem Unificada.

demonstra um embaraço em entender a natureza do som dentro das convenções da linguagem musical, ou seja, sendo a música uma arte sonora, o estudante demonstra dificuldade na identificação da percepção física do som, com a linguagem simbólica da teoria musical.

Além disso, há um déficit de concentração e raciocínio nos momentos de estudos, complicando ainda mais a interpretação da teoria e em seguida sua aplicação e treino.

2.2.1 Levantamento de Requisitos

Pesquisando por meio de observação do ambiente do usuário, leitura de bibliografias do assunto e ouvindo profissionais e estudantes da área da música, foram levantadas as necessidades e o público-alvo que o sistema pretende atingir.

Segundo as teorias de música estabelecidas por Bohumil Med³ e Paschoal Bona⁴, o estudante de música deve aprender como princípios básicos e fundamentais: a notação musical, pentagrama/pauta, simbologia da música, as figuras e valores de som e silêncio da música, as propriedades primas do som e da música, fórmulas de compasso, entre outros conceitos avançados e derivados destes, a fim de aprender a linguagem musical representada nas partituras.

Portanto, teremos como requisitos funcionais, os quais definem as funcionalidades do sistema:

- Apresentar noções básicas das propriedades do som, com as figuras de som e silêncio próprias da música.
- Estimular noção de linguagem rítmica do usuário.
- Apresentar a notação musical e sua simbologia.
- Estimular o reconhecimento audiovisual da notação musical.
- Estimular a prática dos conhecimentos adquiridos pelo usuário.

²A pesquisa de campo está exemplificada no Apêndice A.

³Bohumil Med nasceu em 21 de dezembro de 1939 na então Tchecoslováquia, graduou-se em Música, com habilitação em Trompa, pelo Conservatório de Praga. Publicou quatro livros, entre os quais destacam-se Teoria da Música, utilizado em escolas de todo o Brasil.

⁴Paschoal Bona (Cerignola, 3 de novembro de 1808 — Milão, 2 de dezembro de 1878) foi um compositor e teórico musical italiano.

- Conter níveis de aprendizagem.
- Realizar teste avaliativo do usuário, para poder fornecer as atividades de estudos adequadas ao seu nível.
- Avaliar e retornar a avaliação do desempenho do usuário.
- Apresentar os conceitos da música de forma lúdica e interativa, instigando o treinamento contínuo do usuário com as atividades.
- Apresentar uma explicação do tema para o usuário.
- Permitir o cadastro do usuário e o seu acesso ao sistema.

2.2.2. Especificação de Requisitos

Dentro dos princípios de análise de sistemas da UML, a análise de requisitos, também chamada de especificação de requisitos, abrange a etapa que será feito o detalhamento das necessidades do usuário, de modo que, se encontrem soluções para os problemas da fase anterior.

Nessa etapa, a modelagem do sistema, feito por meios dos modelos de representação (diagramas UML), com seu detalhamento correspondente. Sendo assim, foram criadas estratégias de solução com base nas necessidades do usuário.

No primeiro momento, para avaliar e informar por onde o usuário deve começar, o sistema fornecerá um conjunto de perguntas, para obter o conhecimento prévio do iniciante, indicando o nível certo pelo qual deve iniciar. Posteriormente, para atender aos tópicos sobre aprender os conceitos musicais, o sistema irá treinar o usuário, através de atividades interativas e audiovisuais, buscando continuamente estimular o mesmo a continuar o treino. Também, a fim de avaliar o nível de aprendizado, o sistema deve quantificar e exibir ao usuário os erros e acertos nas atividades, conforme os caso de uso, representado pelas figuras 4, 5 e 6, que estão devidamente explicados na documentação de caso de uso⁵ dos mesmos.

⁵A documentação dos casos de uso estão presentes no Apêndice B.

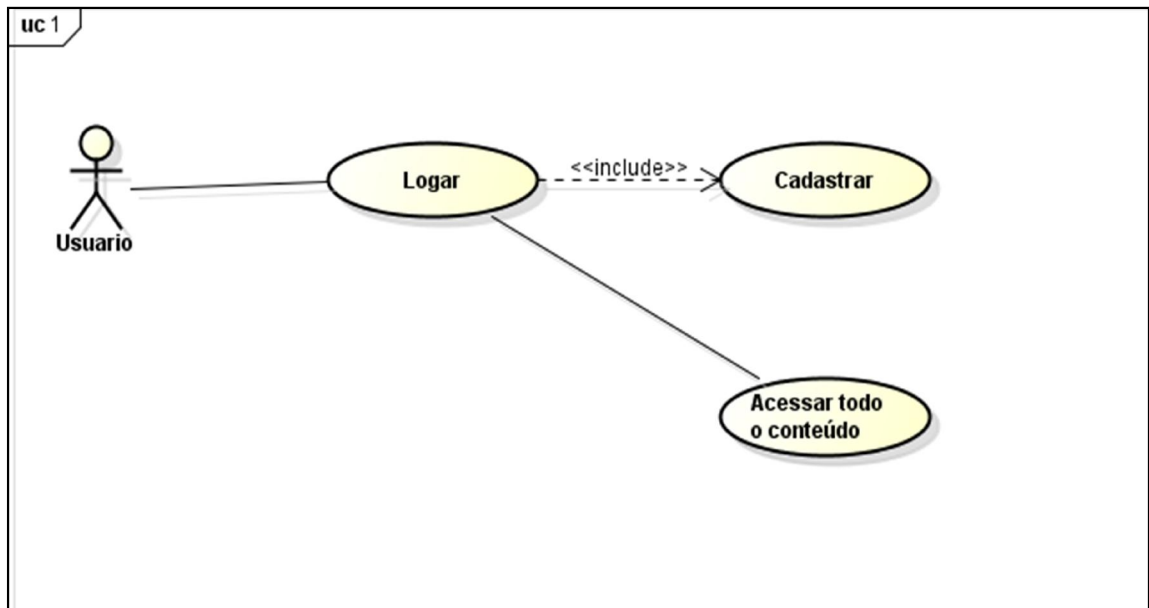


Figura 4 – Caso de uso Acesso ao sistema
Fonte: Autoria própria

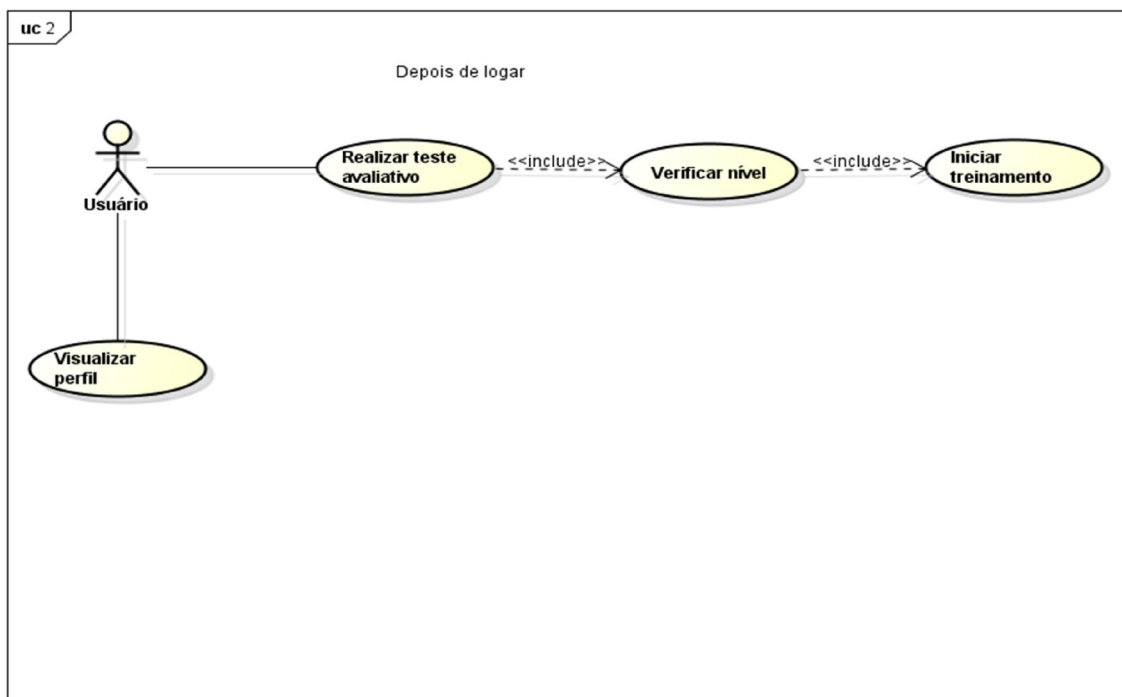


Figura 5 – Casos de Uso Educação Musical
Fonte: Autoria própria

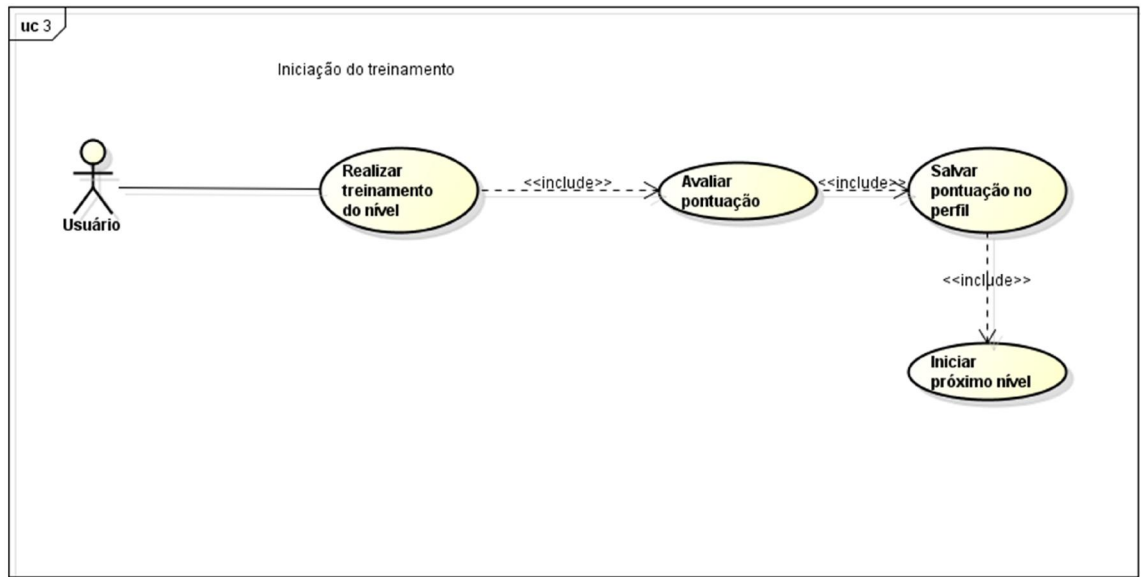


Figura 6 – Casos de Uso Treinamento
Fonte: Autoria própria

2.2.3 Metodologia de Desenvolvimento de Software

A metodologia adotada para o desenvolvimento foi o modelo de ciclo de vida clássico ou cascata, que é um modelo de processo sequencial que permite demarcar pontos de controle, com o objetivo de facilitar a gestão de projetos (veja figura 7).

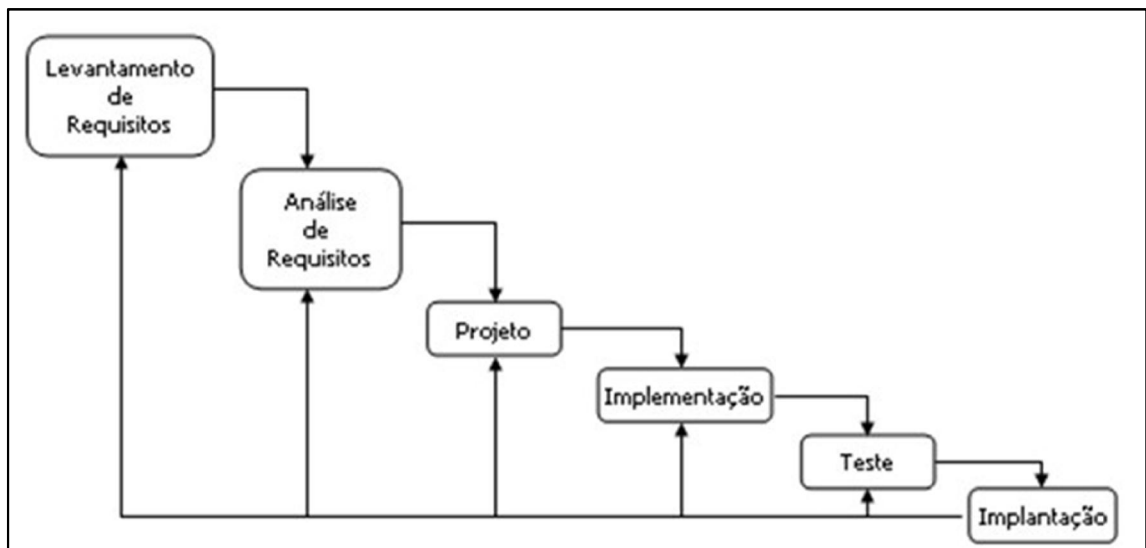


Figura 7 – Ciclo de Vida Clássico ou cascata
Fonte: BEZERRA, 2006, p.32.

Para o levantamento de requisitos foi utilizado o *brainstorming*⁶, pesquisas bibliográficas sobre o tema, pesquisa de campo qualitativa e quantitativa, com a aplicação de questionário, dentro de um método científico para capturar e analisar a realidade da área.

A análise ou especificação de requisitos, trata-se da etapa em que se dará a validação e verificação das necessidades do usuário, além de elaborar as soluções dessas mesmas necessidades, e é nesta fase do projeto que são construídos modelos para representar e facilitar o entendimento do sistema a ser desenvolvido.

Sendo assim, para a especificação e análise dos requisitos levantados, foi utilizado primeiramente o diagrama de caso de uso da UML, idealizado pelo engenheiro de software sueco Ivar Jacobson em 1970. Este diagrama “[...] é uma representação das funcionalidades externamente observáveis do sistema e dos elementos externos ao sistema que interagem com ele” (BEZERRA, 2006, p.46).

Na fase de projeto de sistema, “[...] determina-se *“como”*⁷ o sistema funcionará para atender aos requisitos”(BEZERRA, 2006, p.25), levando em consideração as ferramentas físicas, como a arquitetura do sistema.

3 BANCO DE DADOS

“Um banco de dados é uma coleção de dados operacionais armazenados, sendo usados pelos sistemas de aplicação de uma determinada organização” (1985 APUD Date; Bittencourt, 2004. P.1). Ainda de acordo com o autor, um banco de dados é dividido em partes separadas, na qual cada uma delas possui uma responsabilidade dentro do sistema.

Dentro do banco de dados temos os componentes de gerenciador de arquivos, responsável pela alocação do espaço para o armazenamento e das estruturas de dados usadas para conceber a informação contida no disco; o gerenciador de banco de dados, que emprega a interface entre os dados de nível baixo guardados em disco e os programas de consulta, disponibilizados ao sistema.

⁶ Em português: “tempestade cerebral”, significa criatividade em equipe.

⁷ Grifo do autor

Além desses componentes, ainda possuem o processador de consultas, que tende a traduzir as consultas registradas em uma linguagem de nível alto para algoritmos de nível baixo que o gerenciador do banco de dados envolve o pré-compilador DML, que converte comandos DML impingidos em um aplicativo para chamadas de expressão normal na linguagem instalada, os arquivos de dados, que como o próprio nome diz, armazena o banco de dados por si mesmos e o dicionário de dados⁸, que é o artefato responsável pelo armazenamento dos metadados sobre a estrutura do banco de dados.

Já o Sistema Gerenciador de Banco de Dados (SGBD) é o conjunto de softwares que se responsabilizam pelo gerenciamento dos dados armazenados. Um SGBD facilita a eliminação de redundância e inconsistência de dados, garantindo o acesso a vários usuários ao mesmo tempo, cobrindo a integridade dos dados, entre outros.

Os SGBD surgiram no início da década de setenta e tinham como objetivo facilitar a programação de aplicações para banco de dados, sendo que os primeiros sistemas criados eram demasiado caros e difíceis de utilizar e requeriam especialistas habilitados para empregar o sistema específico. Este tipo de SGBD passou a dominar o mercado a partir da década de oitenta, devido à diminuição de preço das plataformas de hardware/software, se convertendo ao padrão internacional.

As pesquisas na área de banco de dados também resultaram em um conjunto de processos, técnicas e notações para o projeto de banco de dados, em que primeiramente foi realizado com técnicas sem caráter científico por poucos especialistas no SGBD, e hoje é executado com o amparo de técnicas padronizadas e suportadas por ferramentas CASE.

O projeto de um banco de dados, frequentemente, incide em três etapas, sendo a primeira etapa a modelagem conceitual, que compreende em capturar formalmente os requisitos de informação do banco de dados, a segunda etapa, que seria o projeto lógico, no qual são definidas as estruturas de dados que programarão os requisitos calhados na modelagem conceitual e a terceira o projeto físico, que determina os parâmetros físicos de acesso ao banco de dados, e procura apurar o desempenho do sistema.

⁸O dicionário de dados deste sistema está presente no Apêndice C.

3.1 LINGUAGEM SQL

SQL significa “*Structured Query Language*” (Linguagem Estruturada de Pesquisa), que teve seus embasamentos no modelo relacional de Codd, em 1970. Graças a esta nova maneira de consulta e manipulação de dados, o uso da linguagem SQL foi se tornando cada vez maior, e com isso uma ampla quantidade de SGBD's foram criados contendo a linguagem básica da SQL, como por exemplo, o SQL/DS e DB2 da IBM, RDB da Digital, ORACLE da Oracle Corporation, SYBASE da Sybase INC, Microsoft SQL Server, entre outros).

“A linguagem SQL é um padrão de linguagem de consulta comercial que usa uma combinação de construtores em Álgebra e Cálculo Relacional” (Takaietal, 2005, p. 67). O programa permite que uma relação tenha uma ou mais tuplas idênticas em todos os seus atributos e valores, ou seja, o SQL é um multi-conjunto de tuplas, e é dividido em partes, sendo elas: Linguagem de definição de dados (DDL), Linguagem interativa de manipulação de dados (DML), Incorporação DML (*EmbeddedSQL*), Definição de visões, Autorização, Integridade e Controle de Transações.

A linguagem SQL tem um papel importante nos SGBD, pois possui muitos enfoques, dentre eles: linguagem interativa de consulta, linguagem de programação para acesso a banco de dados, linguagem de administração de banco de dados, linguagem cliente/servidor, linguagem para banco de dados distribuído, e o caminho de acesso a outros bancos de dados em diferentes máquinas, pois a SQL auxilia na conversão de diferentes produtos, desde micro até mainframe, podendo manipular objetos de diferentes classes entre as funções de um SGBD.

O software que será utilizado para a armazenagem e manipulação de dados do sistema criado será o SQL Server, pelo fato de obter facilidade em relação à manipulação e entendimento e que foi desenvolvido especialmente para o ambiente relacional, podendo também ser calhada a qualquer ambiente não relacional.

3.2 BrModelo

A ferramenta BrModelo foi criada pelo analista Carlos Henrique Cândido (curso pós-graduação em banco de dados do Centro Universitário UNIVAG), além de ser um programa gratuito, facilita o aprendizado para a modelagem de dados. O programa funciona como uma espécie de editor, e suas funcionalidades são construção de

modelo entidade relacionamento e mapeamento para o modelo relacional de banco de dados. O sistema é deveras funcional e facilita em relação ao entendimento mais claro a respeito da modelagem de dados, tendo em consideração que os modelos possuem uma facilidade em serem criados e alterados.

3.3 Diagramas de Banco de Dados

Um diagrama é um modelo gráfico utilizado dentro da modelagem de software para simbolizar os artefatos dos componentes de software utilizados e seus inter-relacionamentos.

O Diagrama Entidade e Relacionamento apresentado na figura 8 demonstra a forma que as entidades se relacionarão no sistema, na qual a entidade Cadastro recebe os dados introduzidos pelo o usuário, a entidade Perfil recebe os dados do Cadastro e do Progresso do usuário, e Teste (resultado do teste avaliativo concretizado pelo usuário) envia o resultado final para Progresso.

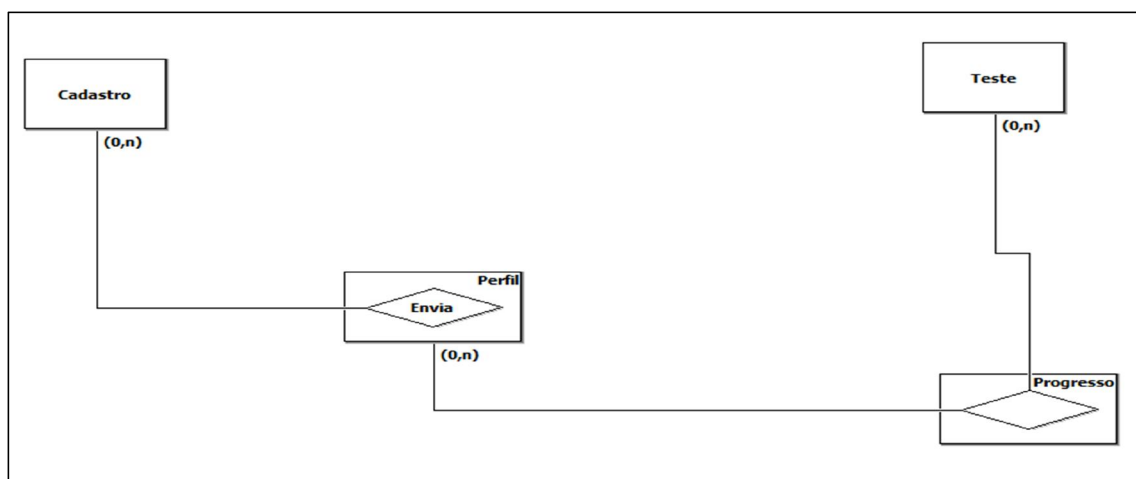


Figura 8 – DER
Fonte: Autoria própria

Já no Modelo Entidade Relacionamento é apontado, detalhadamente, os objetos das tabelas, os seus atributos e também seus relacionamentos, tendo como exemplo a Tabela Cadastro, que possui como objeto a chave primária Cod_Cadastro, que tem como atributo Número, o objeto Usuário que tem como atributo Texto, e também a Tabela Perfil, que seria uma Tabela associativa, na qual possui o objeto Cod_Progresso e Cod_Cadastro como chaves estrangeiras, que pertencem às tabelas Progresso e Cadastro, respectivamente (confira na Figura 9).

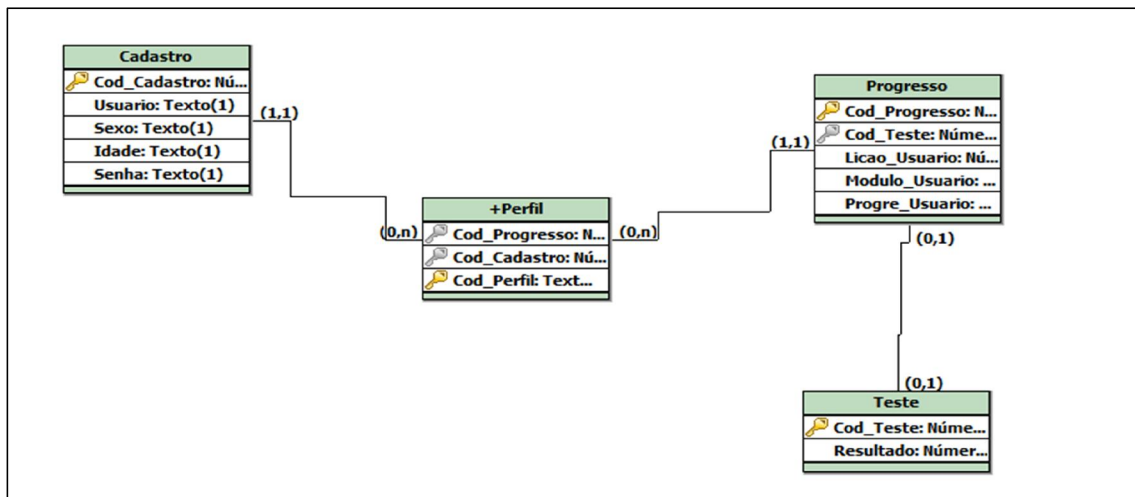


Figura 9 – MER
Fonte: Autoria própria

4 DESENVOLVIMENTO DO SISTEMA

No desenvolvimento de projeto do sistema, “[...] determina-se *“como”*⁹ o sistema funcionará para atender aos requisitos.”(BEZERRA, 2006, p.25), levando em consideração as ferramentas físicas, como a arquitetura do sistema, em que foi adotado o software voltado à plataforma desktop, construído na linguagem de programação CSharp, utilizando como padrão de interface gráfica os formulários do Microsoft Visual Studio e gerenciador de banco de dados Microsoft SQL Server, além da utilização de ferramentas de apoio como o Photoshop e Illustrator para manipulação de imagens.

Sendo todas as ferramentas utilizadas escolhidas de modo a melhor atender o objetivo de criação do sistema, como também, pela facilidade que fornecem aos elaboradores do mesmo, possibilitando assim um maior desempenho das atividades propostas.

4.1 Lógica de Programação

A lógica de programação não é nada mais do que a coordenação coesiva das instruções do programa para que o seu objetivo seja arranjado. Para designar essa coordenação, instruções simples do programa, como desenhar uma imagem na tela

⁹ Grifo do autor

do computador ou mudar o valor de uma variável, é conectado, por assim dizer, á estruturas lógicas que conduzem o fluxo da execução do programa.

Através dessas instruções, podem ser definidos os passos para que ela seja feita com sucesso, mas quando esse raciocínio não é suficientemente preciso, tem-se grandes chances de não alcançar os objetivos propostos. E a mesma coisa acontece com os programas de computador.

A estrutura e a forma de como o programa serão feitas, depende exclusivamente da pessoa que está programando. Alguns são práticos e outros são mais detalhistas, e esse detalhamento tem-se pelo preparo das instruções do programa utilizando estruturas de controle que irão conduzir o andamento de execução, que são determinadas como estruturas de controle (condicionais ou repetições).

As estruturas condicionais permitem efetivar desvios de fluxo com base na estimativa de condições. Pode-se pegar como exemplo a realização da troca de um pneu. a frase “o estepe está ok” é a condição que permitirá a efetivação das demais atividades relacionadas à troca. Se a condição não for verdadeira, ou melhor, “se o estepe não está cheio o suficiente”, os seguintes passos seriam: “ligar para o seguro”, “aguardar socorro”, etc.

Já as estruturas de repetição ou loops são indicativas de que uma ou mais tarefas necessitam de ser executadas mais de uma vez. Por exemplo, novamente no caso do pneu, ao surgir à frase “parafusar todos os parafusos” está indicando que se devem identificar todos os parafusos soltos e parafusá-los. A função do loop em si, é consentir a repetição de instruções sem ser necessário reescrevê-las, até que uma das condições permaneça completamente atendida. Nesse exemplo, a condição é que todos os parafusos fiquem parafusados. Ao ter que escrever, parafuse o primeiro parafuso, parafuse o segundo parafuso, parafuse o terceiro parafuso..., na qual significaria uma repetição dispensável de instruções.

A lógica de um programa é preparada por meio da combinação de várias instruções (de manipulação, entrada ou saída de dados), condições e laços. Por isso é que a lógica de programação é a condição principal para o para quando for desenvolver um software.

4.2 Linguagem de Programação

Linguagem de programação é um método convencional para difundir instruções para um computador, ou seja, um conjunto de regras semânticas e sintáticas utilizadas para definir um programa. A linguagem de programação permite que seja especificado precisamente sobre quais dados o computador vai atuar, como eles serão armazenados ou transmitidos e quais ações devem ser tomadas sob diversas conjunturas.

Com isso, linguagens de programação são projetadas para amparar uma sintaxe de nível maior, que pode ser de melhor compreensão por programadores, elas também são ferramentas de extrema importância para que se possa escrever programas com maior rapidez e mais organizados, deixando os programas com menor dependência de ambientes computacionais específicos.

4.2.1 Csharp

A linguagem de programação utilizada, o CSharp (C#), é uma linguagem orientada a objetos tencionada a oferecer a melhor combinação de expressividade, simplicidade e desempenho. Também consagra os conceitos de muitas outras linguagens, mas especialmente de Java e C++.

A Plataforma .NET é centrada ao redor de uma Common Language Runtime (CLR, conceito similar ao da Java Virtual Machine, JVM) e um conjunto de bibliotecas que pode ser empregado em uma grande variedade de linguagens, as quais podem trabalhar juntas, já que todas são compiladas para uma mesma linguagem intermediária, a Microsoft Intermediate Language (MSIL). (Escola Alcides Maya Linguagem de Programação I (Linguagem C#), pg. 2.).

Ou seja, é possível desenvolver aplicativos misturando C# e Visual Basic ou qualquer outra linguagem tolerada. Em comparação às linguagens de script, o C# possui uma execução mais rápida, pelo fato de todos os programas desenvolvidos em sua plataforma serem compilados, gerando com isso uma extensão exe ou dll.

As principais características do C# são o suporte para definir e trabalhar com classes, pois as classes produzem novos tipos, fazendo assim com que se expande a linguagem e consiga manipular melhor o problema que está para ser resolvido, contendo palavras-chave para declaração de novas classes e de suas prioridades e métodos, como também, a implementação de encapsulamento.

A linguagem ainda permite interfaces, um meio de construir um contrato com uma classe para os artifícios que a interface produz dentro do C#, na qual uma classe pode herdar de apenas um objeto, porém uma classe pode programar diversas interfaces. Dentro da plataforma, tudo que está relacionado a uma declaração de classe se localiza na própria declaração, e definições de classe não precisam de arquivos de cabeçalho ou de arquivos separados.

Além disso, ela proporciona um suporte para as *structs* (estruturas), que seria um supérfluo limitado que quando é instanciado, faz muito menos pedidos ao sistema operacional e também à memória, ao contrário do que uma classe convencional faria, sem contar ainda que um *struct* pode herdar de uma classe ou ser herdada por uma classe, porém pode programar uma interface, adequando características orientadas a componentes, que seriam os eventos, propriedades e atributos. O C# oferece suporte para, palavras-chave para abarcar operações como inseguras e acesso direto à memória empregando ponteiros do estilo C++.

4.2.2 Ferramentas de Apoio

Além da linguagem de programação, foi utilizado o Adobe Photoshop, software responsável pelo tratamento das imagens, na qual ele não possui muitos recursos de criação, que pode transformar, editar e corrigir erros nos pixels, nas cores e em efeitos de luz e sombreamento. Esse software possui uma variedade maior de cores, e opções de efeitos para texto, por isso foi utilizado para a criação do nosso logo. Ele também possui uma ferramenta de criação de efeitos como, por exemplo, botões, rodapés, painéis e etc.

Outra ferramenta de criação de gráficos e desenhos utilizados para desenvolvimento do sistema foi o Adobe Illustrator, que diferente do Photoshop, não trata a imagem, ela modela, cria ilustrando de uma forma rápida e fácil. Por possuir recursos de criação de imagem, o Illustrator foi utilizado para a criação do mascote¹⁰ do sistema, além de ser usado para correção de contorno de algumas imagens. Esse software também possui efeitos em 3D, e possui uma ferramenta de curvatura, que passa um aspecto melhor a imagem, pois não deixa a figura quadrilátera ou pixeladas¹¹.

¹⁰Apresentamos no sistema um passáro como nosso mascote denominado Master.

¹¹ As imagens pixeladas são aquelas de baixa qualidade que aparecem granuladas e de forma não nítida.

4.3 Diagrama de Classe

Nos diagramas de classe são descritas as classes que comportam os atributos e métodos, assim como as interações das classes, que posteriormente dentro do sistema são chamadas por meio das ações conjunto com a interface, definindo um determinado comportamento da aplicação.

Foi utilizado dentro do sistema desenvolvido o diagrama de classes devido à estrutura do mesmo, visto que o sistema é comporta tanto o modelo de programação orientado à objetos quanto orientado a eventos. Sendo assim, quando uma determinada ação (denominadas métodos) precisa acontecer várias vezes durante a programação é viável a criação de uma classe que possa ser instanciada¹² para um melhor desempenho do desenvolvimento do sistema.

Dentro deste sistema foram criadas as classes 'Cadastro' e 'PerfilUsuario' contendo atributos que são chamados durante o decorrer da programação. Já as classes 'UsuarioNegocio' e 'AcessoBd' são responsáveis pela conexão com o banco de dados. E as classes 'ContarPontos' e 'ResultadoTeste' são as classes com métodos de contagem de variável, utilizadas para a contagem de pontos do teste avaliativo realizado pelo usuário, assim como, a pontuação que o usuário realizar durante as atividades apresentadas dentro do sistema. Além disso, há também a classe principal que é uma classe gerada pelo C# para o *debugging* do sistema, todas essas estão inseridas e são chamadas dentro do espaço da interface e – denominada nesse caso de 'Apresentação'. A seguir é exibido o diagrama de classes no modelo da UML deste sistema. Veja na figura 10.

¹²Conceito dado quando uma classe ou objeto é chamada numa determinada ação.

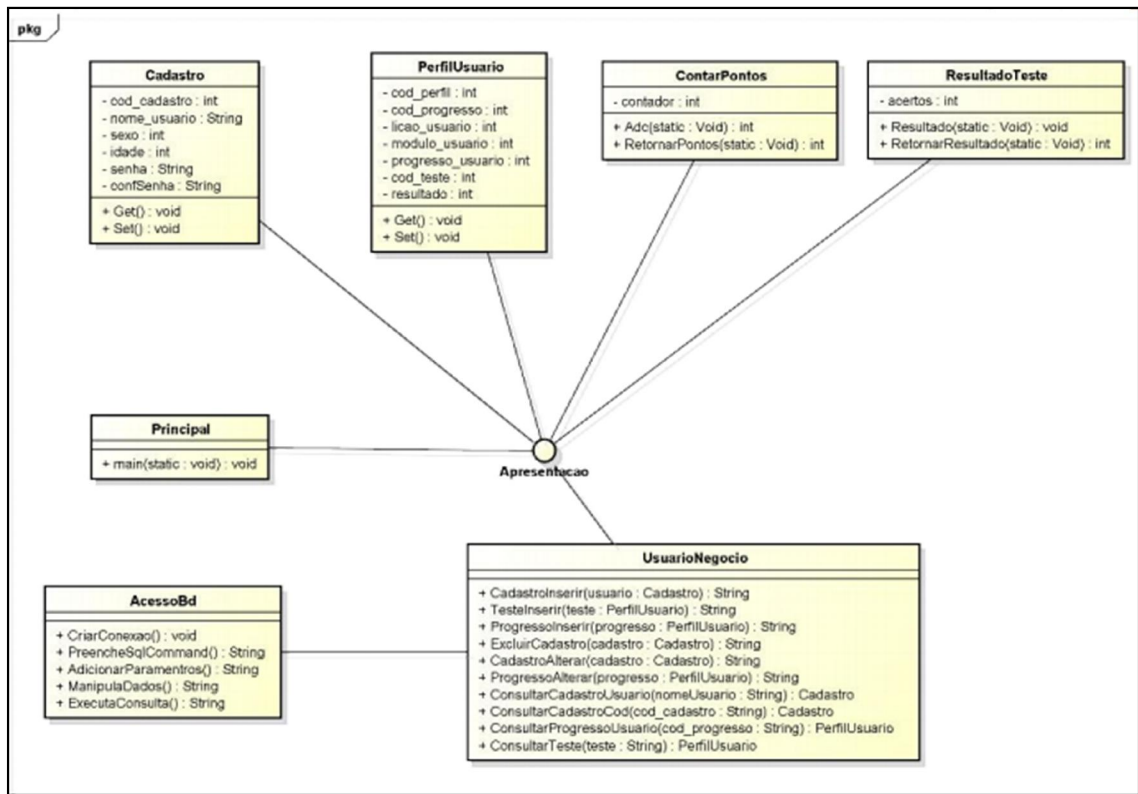


Figura 10 – Diagrama de classes
Fonte: Autoria própria

4.4 Apresentação do Sistema

O logo do sistema é composto pela utilização de elementos naturais, que servem para representar a música, já que a natureza muitas vezes se relaciona com a mesma por seus inúmeros sons marcante. O pássaro foi utilizado como mascote pelo seu canto, que pode ser um dos sons mais distintos dentro da natureza.

As cores principais do Master Score são o azul e laranja. O azul foi escolhido por ser uma cor clara e fria, assim sendo causa pouca fadiga visual ao usuário. Já a cor laranja dentro da roda de cores, é contrária ao azul possuindo assim uma combinação entre elas. Por ser uma cor quente e mais viva é utilizada em alguns detalhes no sistema, sendo também a cor do mascote. Na figura 11 abaixo está exibido à tela inicial com o mascote e logo do sistema, denominado Master Score.



Figura 11 – Tela de abertura com logo e mascote do sistema
Fonte: Autoria própria

A primeira tela é a de entrada, na qual se o usuário já tiver o cadastro pode logar no sistema e se não tiver pode criar no botão 'CRIAR CONTA', conforme a figura abaixo:

Figura 12 – Tela inicial com a opção de criar uma conta e fazer login.
Fonte: Autoria própria

Neste form o usuário poderá realizar o cadastro inserindo todos os dados exigidos como nome, idade, sexo e senha; conforme a figura abaixo:



O formulário de criação de conta do Master Score possui um cabeçalho azul com o logo 'Master Score' e o título 'Criar Conta'. Abaixo, há uma seção cinza com o título 'Insira os campos abaixo:'. Os campos de entrada são: 'Nome' (campo de texto), 'Sexo' (com opções 'Feminino' e 'Masculino' selecionáveis por rádio), 'Idade' (campo de texto), 'Senha' (campo de texto com a dica 'Insira até 5 caracteres*') e 'Confirmar Senha' (campo de texto com a dica 'Confirme a senha acima *'). Um botão laranja 'CADASTRAR' está posicionado no final do formulário.

Figura 13 – Form do cadastro
Fonte: Autoria própria

No perfil do usuário será apresentado o nome, etapa atual e o seu progresso. A etapa atual apresentará o nível da atividade que o usuário está. O progresso irá mostrar o avanço do usuário dentro de um determinado nível. Exemplo: Etapa Atual: 2 Progresso (será representado por uma barra laranja), conforme a figura abaixo:

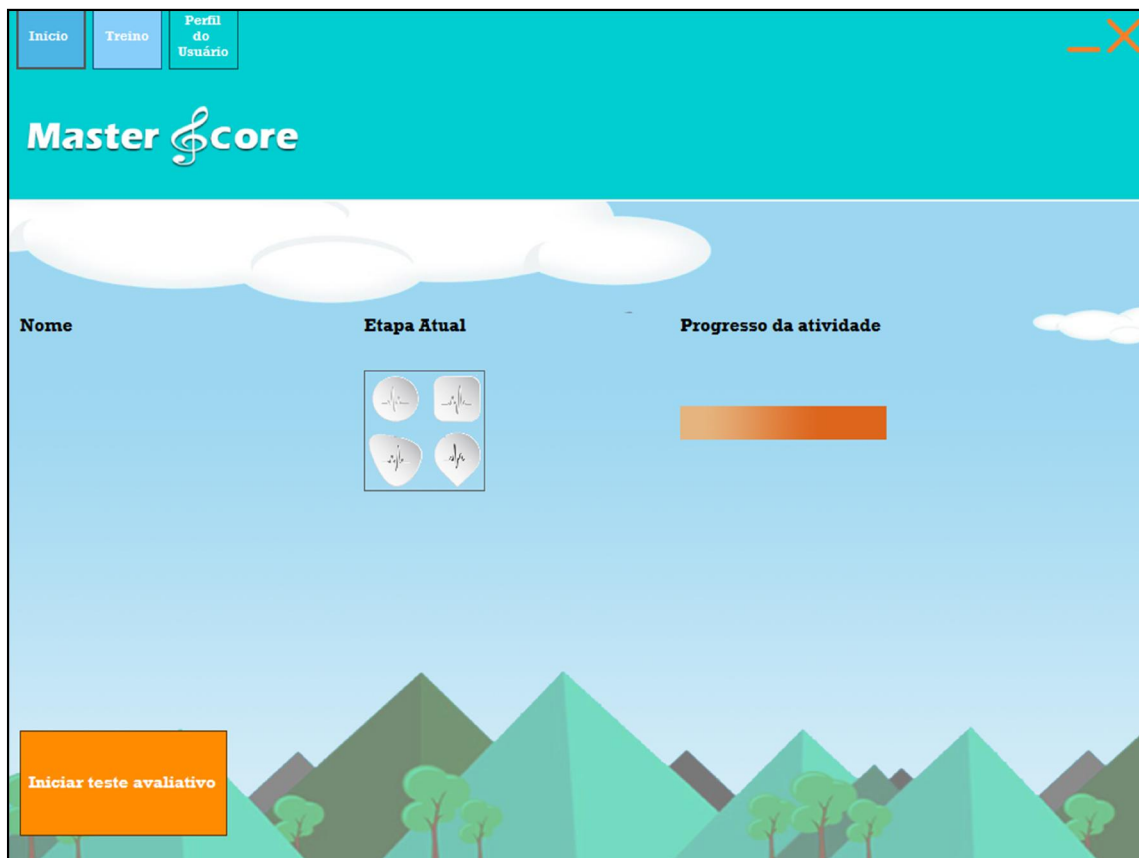


Figura 14 – Tela principal, apresentando as informações do usuário
Fonte: Autoria própria

5 CONSIDERAÇÕES FINAIS

Com o decorrer das pesquisas deste projeto, concluímos que existe uma grande defasagem no aprendizado musical para os estudantes que estão iniciando. E visto que, a inovação da informática auxilia diversas áreas, o Master Score vem com a proposta de auxiliar os educandos, tanto os autodidatas, quanto os que estão inseridos em escolas.

O nosso sistema visa uma forma de ensino interativo, através de jogos e explicações que integram as duas maneiras de ensino – auditiva e visual, para que assim o usuário possua uma melhor assimilação dos conteúdos teóricos básicos da música.

REFERÊNCIAS

ABREU Felipe Machado, Mauricio de. **Projeto de banco de dados: Uma Visão Prática**. Editora Érica Ltda, 11ª edição, 1996.

ADOBE Illustrator CC, Gráficos e ilustrações vetoriais. Software para desktop. Disponibilizado via web.

ADOBE Photoshop CC, Edição e composição de imagens. Software para desktop. Disponibilizado via web.

ANTUNES, Elsaby. **Apostila de Iniciação Musical**. 1 ed. 2008.

BITTENCOURT, Rogério G. **Aspectos Básicos de Banco de Dados**. 2004.

BrModelo, Software free para modelagem de dados. Software para desktop. Disponibilizado via web. Disponível em: www.devmedia.com.br/brmodeloumsoftwarefreeparamodelagemdedados/2192.

BEZERRA, Eduardo. **Princípios de Análise e Projeto de Sistemas UML**. Elsevier editora, 2006, x p.

Escola Alcides Maya Linguagem de Programação I (Linguagem C#)

FARIA, João Pascoal. **Introdução à Linguagem de Programação C#**. FEUP, Setembro de 2001.

HEUSER, Carlos A. **Projeto de banco de dado**. Editora Sagra, 4ª edição, Instituto de Informática da UFRGS, 1998.

Método de Teoria e Solfejo. 1 ed. São Paulo. Congregação Cristã no Brasil, 2009.

MTS. 2 ed. São Paulo. Congregação Cristã no Brasil, 2014.

O.K. Takai; I.C. Italiano; J.E. Ferreira. **Introdução a Banco de Dados**. DCC-IME-USP, 2005.

SILVEIRA, Carlos; RIBEIRO Alan. **O pensamento filosófico de Schopenhauer sobre a música e suas possíveis contribuições para a educação musical brasileira**. Disponível em: http://www.theoria.com.br/edicao0212/schopenhauer_e_a_musica.pdf. Acesso em: 06/05/2015.

6 APÊNDICE

APÊNDICE A – Pesquisa de campo

PESQUISA DE CAMPO

PÚBLICO-ALVO 1: Estudantes de linguagem musical

MEIO/TÉCNICA DE APLICAÇÃO: Entrevista pessoal

TAMANHO DA AMOSTRA:31

OBJETIVO PRINCIPAL:

- Conhecer o perfil do estudante que virá a ser um possível usuário.
- Reconhecer as dificuldades do público-alvo para com o tema.

OBJETIVO SECUNDÁRIO:

- Elaborar os requisitos de negocio do projeto de sistema
- Identificar as defasagens do ensino da música na teoria e prática.

PERGUNTAS

1. Qual sua Idade?
 - 8 a 10 anos.
 - 11 a 15 anos.
 - 16 a 20 anos.
 - Acima de 21 anos.
2. Qual seu sexo?
 - Masculino
 - Feminino
3. Há quanto tempo estuda música?
 - 1 a 3 anos
 - 4 a 6 anos

- 7 a 9 anos
 - Acima de 10 anos
4. Quais os métodos mais utilizados por você para aprender teoria musical?
- Livros
 - Internet
 - Livros e internet
5. Qual a maior dificuldade que você tem para aprender teoria musical?
- Contagem rítmica
 - Solfejo
 - Identificação da notação musical
 - Tonalidade (bemol sustenido...)
 - Unidade de tempo
 - Outros:
6. Na sua concepção o que facilitaria o aprendizado da dificuldade citada acima?

Resposta pessoal

7. Você conhece ferramentas informatizadas na área do aprendizado musical?
- Sim
 - Não
8. Essas mesmas ferramentas auxiliaram na sua aprendizagem?
- Sim, bastante
 - Sim, um pouco.
 - Não
9. Quais as dificuldades encontradas dentro das ferramentas anteriores em relação com sua aprendizagem?

Resposta pessoal

RESULTADO

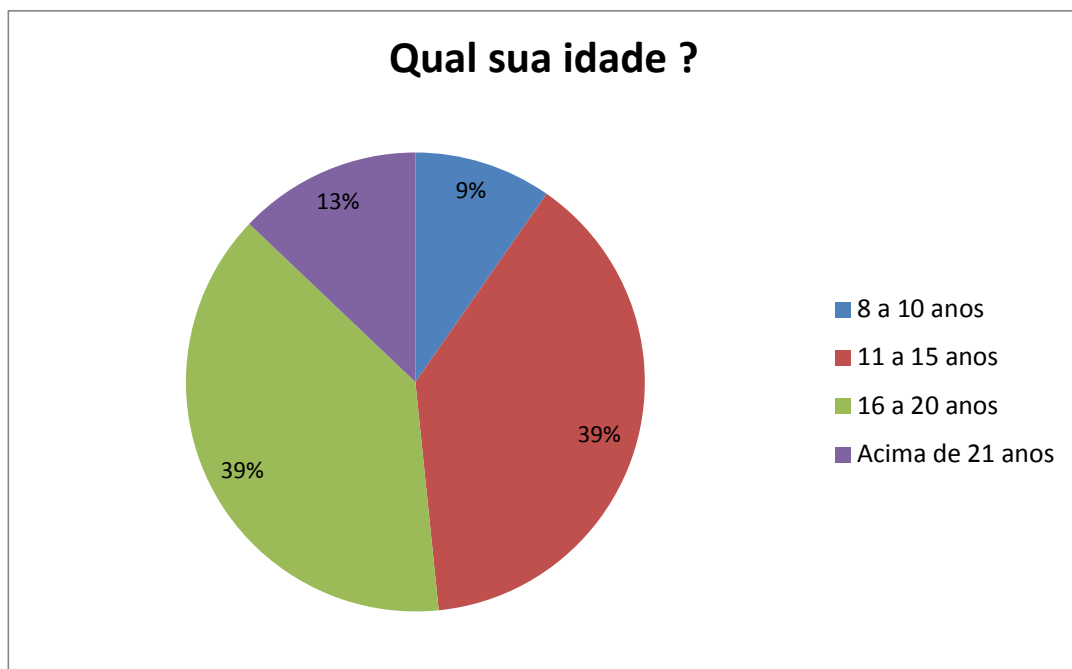


Gráfico 1 – Idade (ALUNO)

Fonte: Autoria Própria



Gráfico 2 – Sexo (ALUNO)

Fonte: Autoria Própria

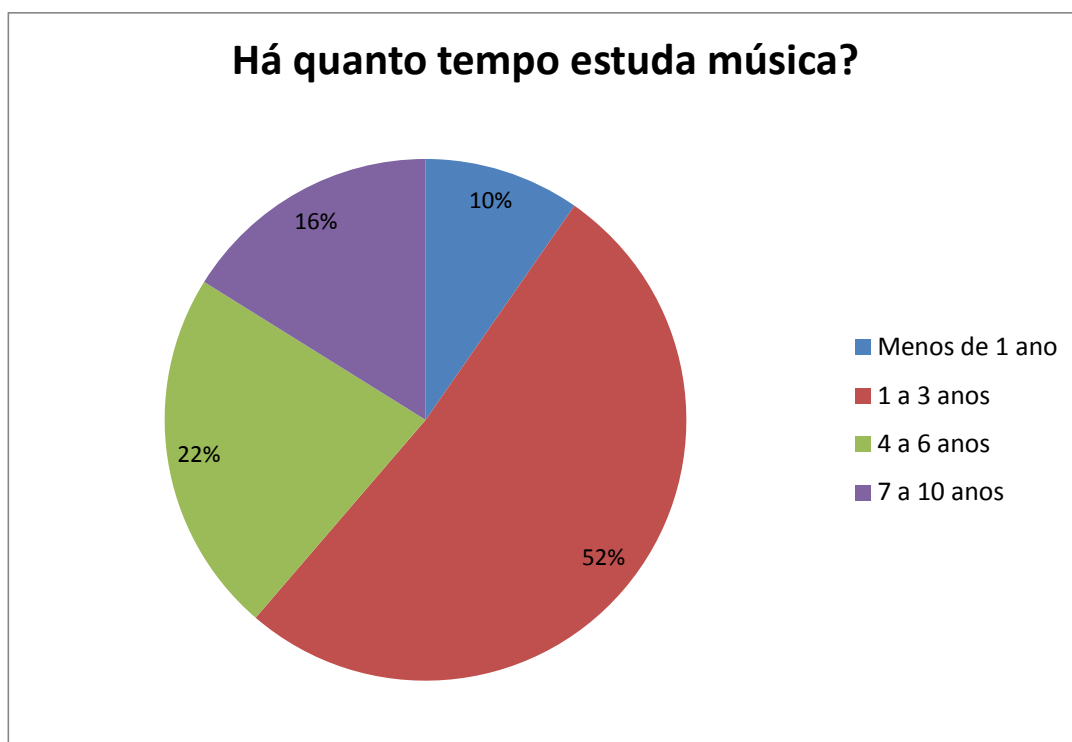


Gráfico 3 – Tempo de Estudo

Fonte: Autoria Própria

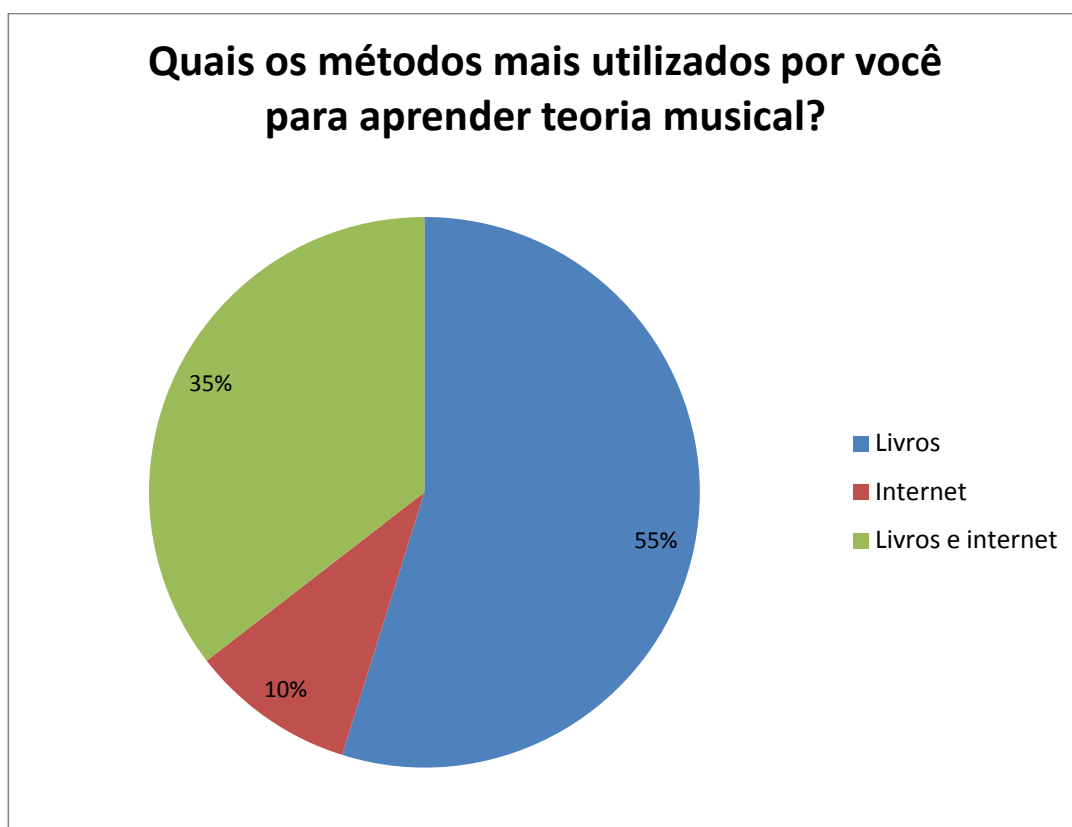


Gráfico 4 – Métodos mais utilizados para aprender a teoria musical

Fonte: Autoria Própria

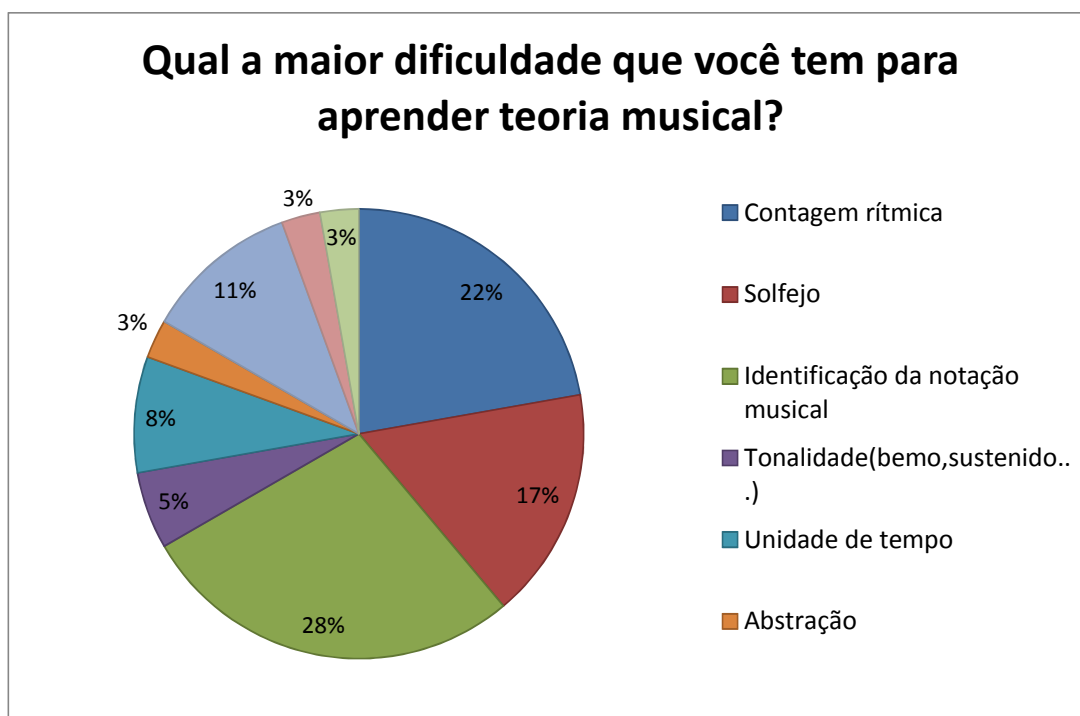


Gráfico 5 – Maior dificuldade para aprender a teoria musical

Fonte: Autoria Própria



Gráfico 6 – Técnicas para melhorar o aprendizado da música

Fonte: Autoria Própria

Você conhece ferramentas informatizadas na área do aprendizado musical?

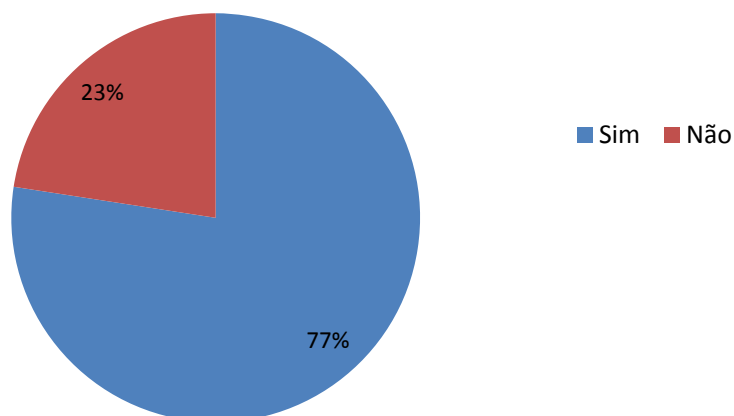


Gráfico 7 – Conhecimento sobre a informatização na área musical

Fonte: Autoria Própria

Quais as dificuldades encontradas dentro das ferramentas anteriores em relação com sua aprendizagem?

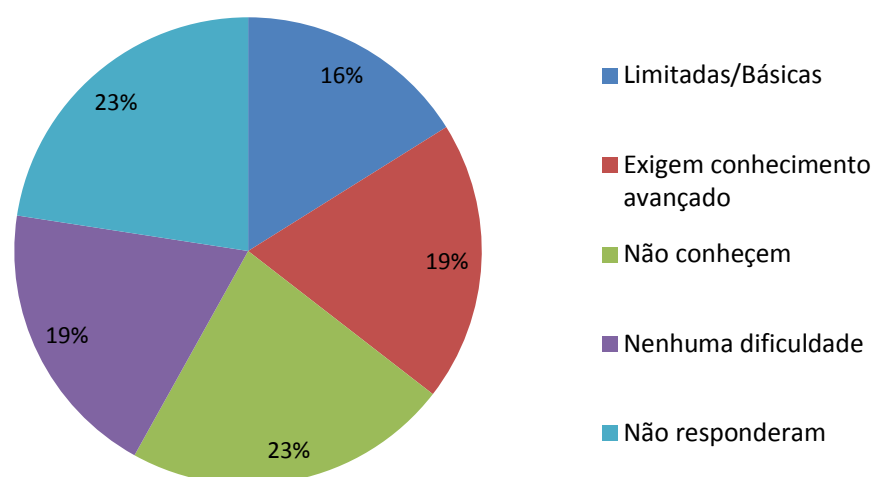


Gráfico 8 – Dificuldades nas ferramentas existentes

Fonte: Autoria Própria



Gráfico 9 – Ferramentas ajudam na aprendizagem?

Fonte: Autoria Própria

PÚBLICO-ALVO 2: Professores de teoria e prática musical.

MEIO/TÉCNICA DE APLICAÇÃO: Entrevista pessoal

TAMANHO DA AMOSTRA: 9

OBJETIVO PRINCIPAL:

- Conhecer o estudante de música segundo a ótica do professor.

OBJETIVO SECUNDÁRIO:

- Identificar as dificuldades do ensino da teoria e prática da linguagem musical.

PERGUNTAS

1. Qual sua idade?
 - 18 a 20 anos
 - 21 a 25 anos
 - 26 a 30 anos
 - 31 a 35 anos
 - Acima de 36 anos
2. Qual seu sexo?
 - Feminino
 - Masculino
3. Qual seu grau de formação dentro da área de música?
 - Não possui
 - Graduação
4. Qual a disciplina que você ensina dentro da área de música?
 - Teoria musical
 - Prática (instrumentos)
 - Canto
 - Geral
5. Qual a maior dificuldade que os alunos têm em relação a aprender linguagem musical?
 - Contagem rítmica
 - Definição dos conceitos

- Solfejo
 - Identificação da notação musical
 - Tonalidade (bemol sustenido...)
 - Unidade de tempo
 - Outros:
6. Qual a principal metodologia que você usa para o ensino da música?
- Livros
 - Aplicativos
 - Ensino prático
 - Todos
 - Outros:
7. Em sua opinião a integração da informática com esses métodos facilitaria o aprendizado do aluno iniciante?
- Sim
 - Não
 - Talvez
8. Por que da resposta anterior?

Resposta pessoal

RESULTADO

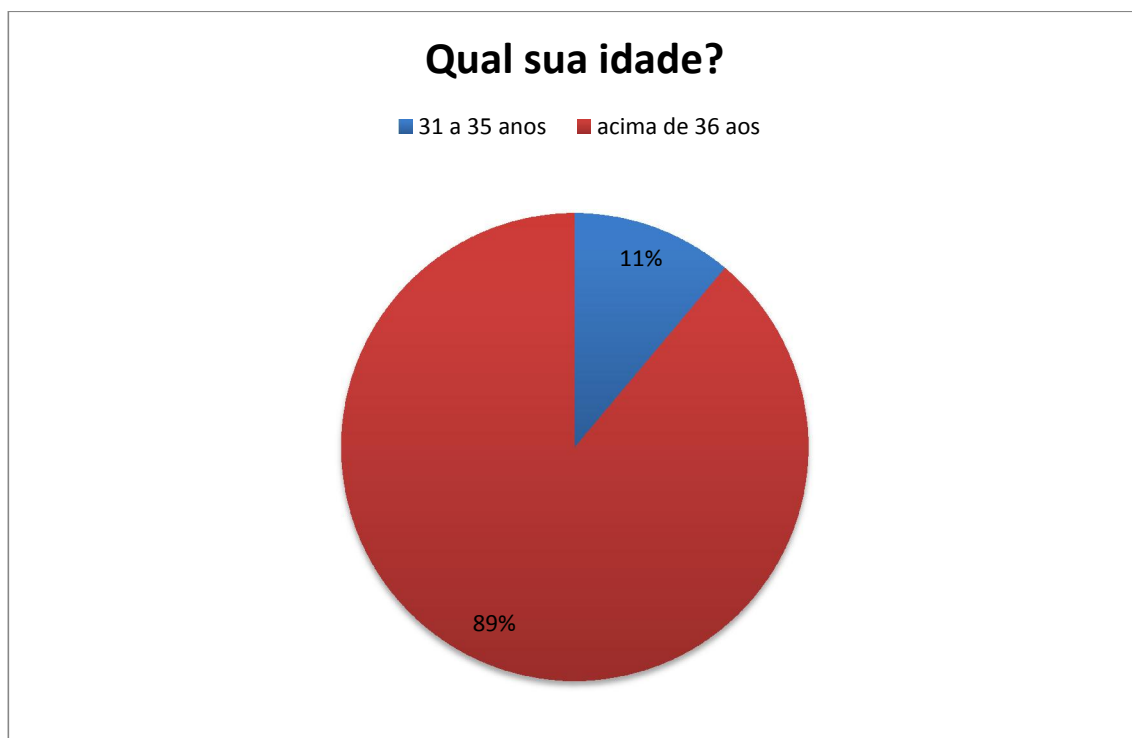


Gráfico 10 – Idade (PROFESSOR)

Fonte: Autoria Própria



Gráfico 11 – Sexo (PROFESSOR)

Fonte: Autoria Própria

Qual seu grau de formação dentro da área de música?

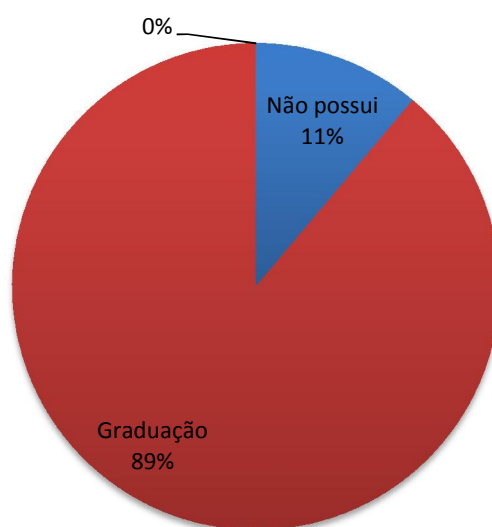


Gráfico 12 – Grau de formação musical

Fonte: Autoria Própria

Qual a disciplina que você ensina dentro da área de música?

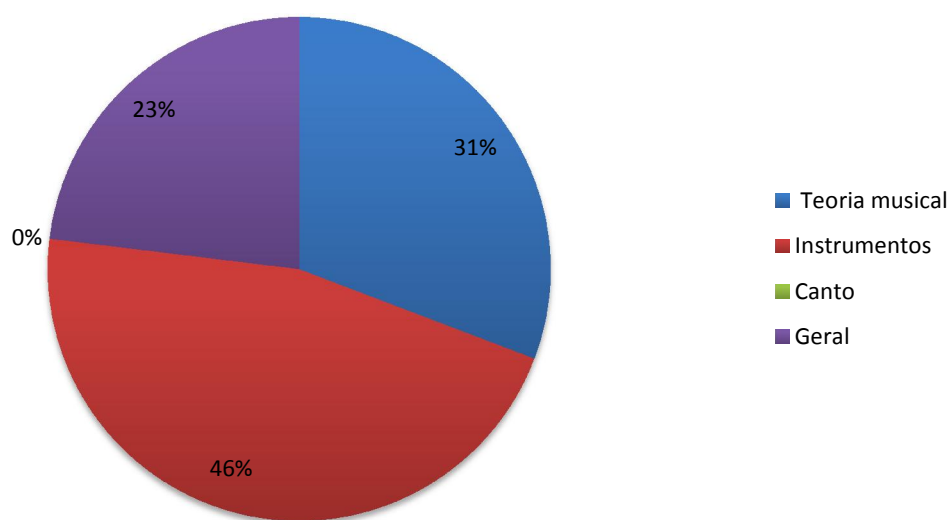


Gráfico 13 – Disciplina lecionada

Fonte: Autoria Própria

Qual a maior dificuldade que os alunos têm em relação a aprender linguagem musical?

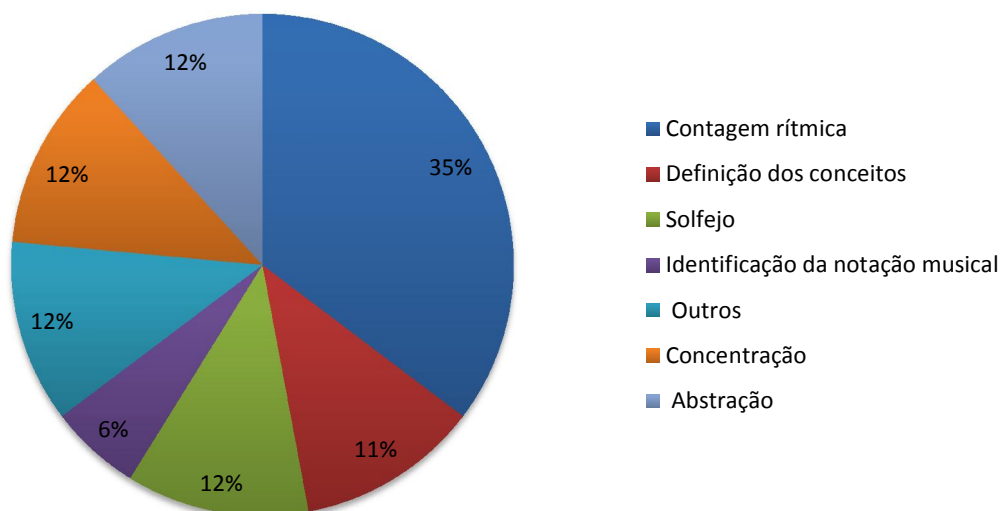


Gráfico 14 – Dificuldade dos alunos

Fonte: Autoria Própria

Qual a principal metodologia que você usa para o ensino da música?

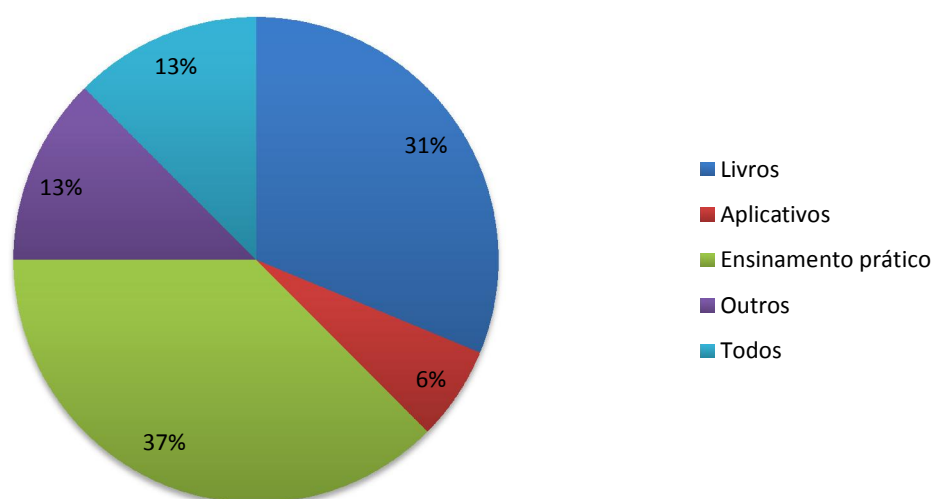


Gráfico 15 – Ferramentas usadas para ensinar a música

Fonte: Autoria Própria

Em sua opinião a integração da informática com esses métodos facilitaria o aprendizado do aluno iniciante?

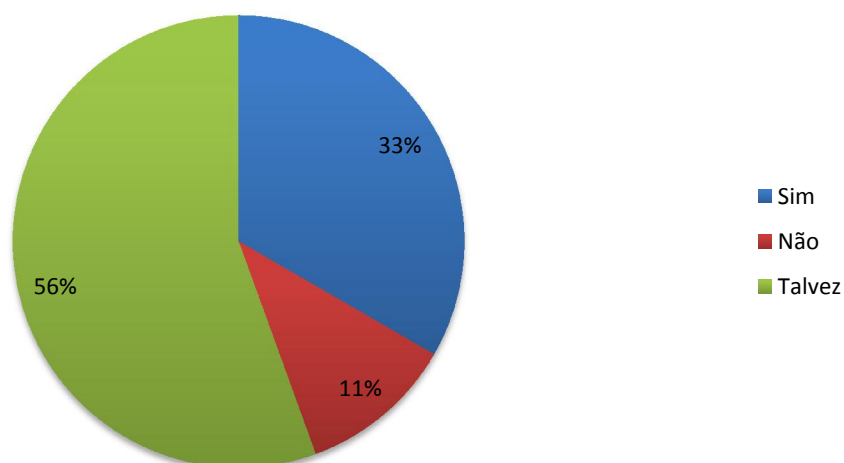


Gráfico 16 – Opinião sobre a integração da informática

Fonte: Autoria Própria

Por que da resposta anterior?

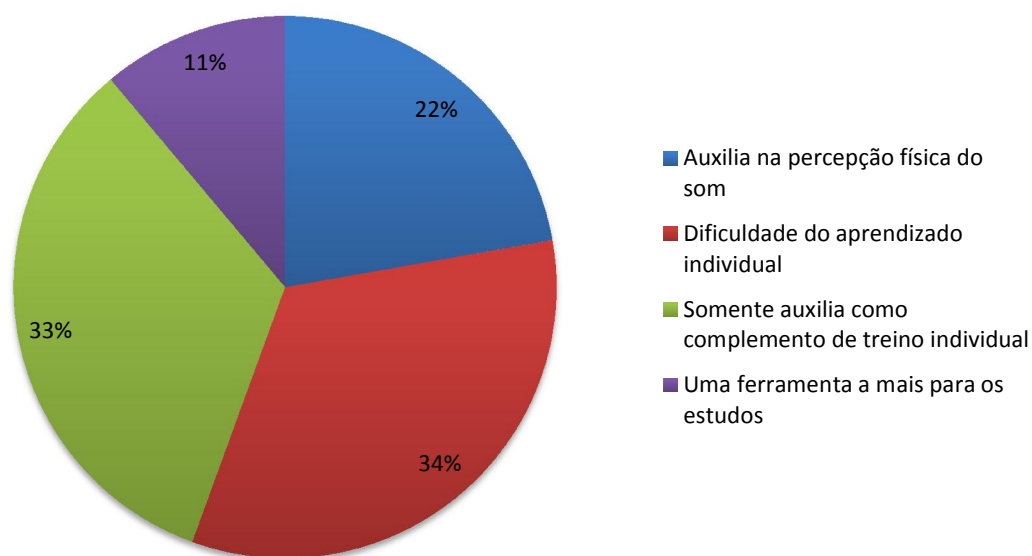


Gráfico 17 – Opinião sobre a informatização da música

Fonte: Autoria Própria

APÊNDICE B – DOCUMENTAÇÃO DE CASO DE USO

Nome do caso de uso	Requisitos comportamentais e desempenho
Caso de Uso Geral	Domínio de negócio
Ator Principal	Usuário
Atores Secundários	-
Resumo	Esse caso descreve como será exibido o sistema, em seu primeiro contato com o usuário se o mesmo ainda não possuir um cadastro.
Pré-condições	Se caso o usuário queira fazer um login, ele terá de possuir um cadastro.
Pós-condições	O usuário fornece informações pessoais, como data de nascimento, nome, senha.
Fluxo Principal	
Ações do Ator	Ações do sistema
1. Visitar página	
	2. Exibição das opções de CADASTRAR e LOGAR
3. Fazer cadastro	
	4. Armazenar dados
5. Realizar login	
	6. Acesso completo ao conteúdo do sistema.
Restrições/Validações	1. O usuário precisa fornecer todos os dados necessários para finalizar o cadastro.
	2. Se não possuir um cadastro, não se tem acesso completo ao

	conteúdo.
Fluxo Alternativo – Manutenção do Cadastro do Usuário	
Ações do Ator	Ações do Sistema
	1. Se necessário, gravar ou atualizar o cadastro e dados do usuário.

Nome do caso de uso	Avaliação do usuário
Caso de Uso Geral	Domínio de negócio
Ator Principal	Usuário
Atores Secundários	-
Resumo	O caso de uso demonstra o que terá de ser realizado após o login do Usuário, ou seja, demonstra o teste avaliativo.
Pré-condições	Só é possível visualizar o resultado se for feito o teste até o seu fim.
Pós-condições	O Usuário poderá ter acesso ao seu perfil independente de ter concluído ou não o teste avaliativo.
Fluxo Principal	
Ações do Ator	Ações do sistema
1. Iniciação do teste avaliativo.	
	2. Teste avaliativo do conteúdo geral de teoria musical.
3. Finalização do teste avaliativo.	
	4. Computar o resultado obtido.
	5. Exibir resultado.
	6. Avaliar nível adequado para

	usuário.
	7. Salvar no Perfil do usuário o resultado do teste avaliativo e o nível de início das atividades.
8. Visualizar Perfil do usuário.	
Restrições/Validações	1. Não se é possível para o usuário avançar no sistema sem a realização do teste.

Nome do caso de uso	Início das atividades do sistema
Caso de Uso Geral	Domínio de negócio
Ator Principal	Usuário
Atores Secundários	-
Resumo	No caso de uso mostra como o Usuário treina e pratica sua aprendizagem e é avaliado pelo o sistema.
Pré-condições	Antes de poder continuar ou começar seu treinamento o Usuário necessita realizar o <i>login</i> .
Pós-condições	Será salvo e atualizado automaticamente o nível em que o Usuário se encontra após finalizar suas atividades.
Fluxo Principal	
Ações do Ator	Ações do sistema
1. O Usuário inicia a fase liberada pelo sistema	
	2. O sistema exibe o conteúdo referente ao nível em que o Usuário se encontra. 3. O sistema exibi a atividade com

	dois passos: primeiro a breve explicação teórica correspondente ao tema, conjunto com a instrução de como realizar a atividade. Em seguida iniciará a atividade com interação do Usuário.
	4. Avaliação do treinamento.
	5. Ao finalizar, o sistema atualiza o nível do Usuário com a pontuação ganha.
6. O Usuário retorna a página de treinamento.	

Tabela 1 – Documentação de Caso de Uso

Fonte: Autoria Própria

APÊNDICE C – DICIONÁRIOS DE DADOS

Cadastro		
Nome do Campo	Tipo de Dados	Descrição
Cod_Cadastro	Numeração Automática	Chave Primária da Tabela.
Usuario	Texto	Nome do Usuário (Login).
Sexo	Texto	Sexo do Usuário.
Idade	Número	Idade do Usuário.
Senha	Texto	Senha criada pelo Usuário.

Tabela 2 – Cadastro

Fonte: Autoria Própria

Progresso		
Nome do Campo	Tipo de Dados	Descrição
Cod_Progresso	Numeração Automática	Chave Primária da Tabela.
Cod_Testes	Número	Chave Estrangeira da Tabela Testes.

Licao_Usuario	Número	Aula atual em que o usuário se encontra.
Modulo_Usuario	Número	Módulo/Nível atual do usuário.
Progre_Usuario	Número	Progresso atual do usuário

Tabela 3 – Progresso

Fonte: Autoria Própria

Perfil		
Nome do Campo	Tipo de Dados	Descrição
Cod_Profil	Numeração Automática	Chave Primária da Tabela.
Cod_Progresso	Número	Chave Estrangeira da Tabela Progresso.
Cod_Cadastro	Número	Chave Estrangeira da Tabela Cadastro.

Tabela 4 – Perfil

Fonte: Autoria Própria

Teste		
Nome do Campo	Tipo de Dados	Descrição
Cod_Testes	Numeração Automática	Chave Primária da Tabela.
Resultado	Número	Resultado do teste feito pelo usuário.

Tabela 5 – Teste

Fonte: Autoria Própria