



**UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE EDUCAÇÃO
CURSO DE LICENCIATURA EM PEDAGOGIA**

JÉSSIKA LIMA TAVARES

**MODELOS, TÉCNICAS E INSTRUMENTOS DE ANÁLISE DE
SOFTWARES EDUCACIONAIS**

**JOÃO PESSOA
2017**

**UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE EDUCAÇÃO
CURSO DE LICENCIATURA EM PEDAGOGIA**

JÉSSIKA LIMA TAVARES

**MODELOS, TÉCNICAS E INSTRUMENTOS DE ANÁLISE DE
SOFTWARES EDUCACIONAIS**

Trabalho de Conclusão de Curso, apresentado ao Curso de Licenciatura em Pedagogia do Centro de Educação da Universidade Federal da Paraíba, em cumprimento à exigência para obtenção do título de Licenciada em Pedagogia.

Orientadora: Profa. Dra. Lebiam Tamar Gomes Silva

**JOÃO PESSOA
2017**

T231m Tavares, Jéssika Lima.

Modelos, técnicas e instrumentos de análise de *softwares* educacionais / Jéssika Lima Tavares. – João Pessoa: UFPB, 2017.
97f. : il.

Orientador: Lebiam Tamar Gomes Silva
Trabalho de Conclusão de Curso (graduação em Pedagogia) –
Universidade Federal da Paraíba/Centro de Educação

1. Aprendizagem - computador. 2. Educação. 3. Softwares educacionais. I. Título.

JÉSSIKA LIMA TAVARES

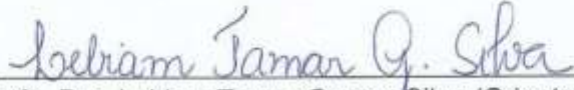
**MODELOS, TÉCNICAS E INSTRUMENTOS DE ANÁLISE DE SOFTWARES
EDUCACIONAIS**

Trabalho de Conclusão de Curso,
apresentado ao Curso de Licenciatura em
Pedagogia do Centro de Educação da
Universidade Federal da Paraíba, em
cumprimento à exigência para obtenção
do título de Licenciada em Pedagogia.

Área de concentração: Pedagogia

Aprovado em: 08/06/2017.

BANCA EXAMINADORA



Profa. Dra. Lebiam Tamar Gomes Silva (Orientadora)
Universidade Federal da Paraíba (UFPB)

Profa. Dra. Danielle Rousy Dias da Silva
Universidade Federal da Paraíba (UFPB)

Profa. Dra. Signe Dayse Castro de Melo e Silva
Universidade Federal da Paraíba (UFPB)

À minha mãe e ao meu pai, pelo carinho,
apoio e suporte dado ao longo da minha
vida para que tudo que tenho planejado
possa ser realizado. DEDICO.

AGRADECIMENTOS

Anos se passaram, conhecimentos foram adquiridos e desafios foram superados, entretanto, sozinha seria impossível contabilizar mais essa vitória em minha vida. Hoje agradeço a todos que contribuíram para a realização desse sonho.

Agradeço aos meus pais, que acompanharam cada dia dessa trajetória, o incentivo constante do meu pai Ailton, a dedicação incondicional da minha mãe Suely. Agradeço aos meus irmãos, Jessiane e Jerffeson, que me apoiaram com palavras e gestos ao longo do processo.

Ao meu noivo, Cássio, que me acompanhou nessa trajetória com todo apoio, carinho e incentivo. Não poderia deixar de mencionar meus tios (as), primos (as) e amigos (as) que sempre me incentivaram a não desistir.

A minha querida orientadora, Professora Dra. Lebiam, pela sua rica contribuição acadêmica e por sempre acreditar em mim e ter aceitado me orientar. A banca, formada pela Professora Dra. Danielle Rousy e a Professora Dra. Signe Dayse, pela aceitação de fazer parte dessa conclusão de curso e por adicionarem seus conhecimentos ao meu trabalho.

E principalmente, agradeço a Deus, porque sem a presença Dele eu não chegaria até aqui, vem Dele tudo o que sou, tudo o que tenho e o que espero.

"[...]Onde você quer chegar?
Ir alto?
Sonhe alto.
Queira o melhor do melhor.
Se pensarmos pequeno.
Coisas pequenas teremos.
Mas se desejarmos fortemente o melhor e,
principalmente, lutarmos pelo melhor
O melhor vai se instalar em nossa vida.
Porque sou do tamanho daquilo que vejo,
e não do tamanho da minha altura."

Carlos Drummond de Andrade

RESUMO

Com a grande demanda de *softwares* educacionais na escola, os profissionais da educação devem fazer uma avaliação prévia do produto, a fim de escolher um *software* que esteja de acordo com o planejamento do professor. Assim, este estudo tem como objetivo geral sistematizar modelos e instrumentos de análise de *softwares* educacionais, identificados em publicações científicas nacionais. Para atingir o objetivo proposto neste trabalho, a metodologia adotada baseia-se em pesquisa bibliográfica, procurando explicar e discutir o tema com base em referências teóricas publicadas em plataformas digitais do *Google Acadêmico*, Periódicos Científicos da *Scielo*, Portal dos Periódicos Livres da CAPES e Banco de Teses e Dissertações da Capes. Como principais autores destacam-se: Abreu (2010), Andres e Cybis (2000), Martins (2002), Oliveira (2001), Ramos (1991), Rezende (2013), Texeira e Brandão (2003) e Vieira (2000). O trabalho identificou diferentes modelos e instrumentos de análise de *softwares* educacionais adotados nas pesquisas nacionais, no campo de educação e tecnologia. Os modelos destacados da literatura consultada foram: a norma ISO de produto de *softwares*, o modelo TUP, o modelo de Reeves e o modelo de avaliação segundo Campos. E os instrumentos de avaliação foram: Questionário de Nokelainen, Questionário de Ssemugabi, Questionário de Reitz, *Checklist* do PROINFO, Avaliação de Oliveira. Ambos, modelos e instrumentos, contribuem na avaliação de *softwares* educacionais, identificando as qualidades, os problemas e as limitações, segundo critérios específicos para a análise de um *software* educacional.

Palavras-chave: Aprendizagem mediada por computador. Educação. Modelos e instrumentos de análise de *softwares* educacionais. *Softwares* educacionais.

ABSTRACT

With the high demand for educational software in the school, education professionals must make a prior evaluation of the product in order to choose software that is in accordance with the teacher's planning. Thus, this study has as general objective to systematize models and instruments of analysis of educational software, identified in national scientific publications. In order to reach the objective proposed in this study, the methodology adopted is based on bibliographic research, trying to explain and discuss the theme based on theoretical references published in academic platforms of Google Scholar, Scientific Periodicals of Scielo, Portal of CAPES Free Newspapers and Bank Of Thesis and Dissertations of Capes. Abreu (2010), Andres and Cybis (2000), Martins (2002), Oliveira (2001), Ramos (1991), Rezende (2013), Texeira and Brandão (2003) and Vieira (2000) . The work identified different models and instruments of analysis of educational software adopted in the national researches, in the field of education and technology. Outstanding models of the literature consulted were: the ISO software product standard, the TUP model, the Reeves model and the evaluation model according to Fields. And the evaluation instruments were: Nokelainen questionnaire, Ssemugabi questionnaire, Reitz questionnaire, PROINFO checklist, Oliveira's assessment. Both models and instruments contribute to the evaluation of educational software, identifying qualities, problems and limitations, according to specific criteria for the analysis of educational software.

Keywords: Computer-mediated learning. Education. Models and instruments of analysis of educational software. Educational software.

LISTA DE SIGLAS

| | |
|----------|--|
| CE | Centro de Educação |
| DENATRAN | Diretrizes Nacionais da Educação para o Trânsito |
| IEC | <i>Electrotechnical Commission</i> |
| ISSO | <i>International Organization of Standardization</i> |
| NTICs | Novas Tecnologias de Informação e Comunicação |
| PBLE | Programa banda larga nas escolas |
| PDE | Plano Nacional de Desenvolvimento da Educação |
| PPGIE | Programa de Pós-Graduação em informática na educação |
| PPP | Projeto Político Pedagógico |
| PROINFO | Programa Nacional de Tecnologia Educacional |
| PROUCA | Programa um computador por aluno |
| TIC | Tecnologia de Informação e Comunicação |
| TUP | Tecnologia, Usabilidade, Pedagogia |
| UCA | Um computador por aluno |
| UFPB | Universidade Federal da Paraíba |
| UFSCar | Universidade Federal de São Carlos |

LISTA DE ILUSTRAÇÕES

| | | |
|------------|--|----|
| Figura 1 - | Tela do <i>Software</i> de Exercitação | 20 |
| Figura 2- | Tela do <i>Software</i> de Simulação | 21 |
| Figura 3 - | Exemplo de <i>Software</i> Modelagem... .. | 22 |
| Figura 4 - | Exemplo de <i>Software</i> Aplicativo | 22 |
| Figura 5 - | Exemplo de <i>Softwares</i> Jogos Educativos | 23 |
| Figura 6 - | Exemplo de <i>Softwares</i> Tutoriais | 24 |
| Figura 7 - | Exemplo de <i>Software</i> de Linguagem de Programação | 24 |
| Figura 8 - | Exemplo de <i>Softwares</i> de Investigação | 25 |
| Figura 9 - | <i>Classificação dos Softwares</i> Educacionais por Objetivo | 26 |
| Figura 10- | <i>Classificação dos Softwares</i> Educacionais por Aprendizagem..... | 30 |
| Figura 10- | Esquema do modelo TUP | 33 |
| Figura 11- | Tela do <i>Software</i> Google Tradutor | 35 |
| Figura 12- | Tela do <i>Monster Math 2</i> | 36 |
| Figura 13- | Tela do <i>software Sistema do Corpo Humano 3D</i> | 37 |
| Figura 14- | Tela do <i>software Duolingo</i> | 37 |
| Figura 15- | Informações sobre <i>software Excel</i>,..... | 38 |
| Figura 16- | Ferramenta TUP <i>online</i> | 39 |
| Figura 17- | Modelo de Reeves aspectos pedagógicos | 41 |
| Figura18 - | Modelo de Reeves de interface do <i>software</i> | 44 |
| Figura 19- | Procedimento gráfico na metodologia Reeves | 45 |
| Figura 20- | Avaliação de softwares segundo Campos | 47 |
| Figura 21- | Procedimento gráfico da Técnica de Mucchielli | 52 |
| Figura 22- | Critérios e respectivas questões de usabilidade técnica e pedagógica sistematizadas e adaptadas por Reitz..... | 62 |
| Figura 23- | Avaliação Final do Instrumento de Oliveira..... | 65 |

SUMÁRIO

| | | |
|----------|--|-----------|
| 1 | INTRODUÇÃO | 12 |
| 2 | METODOLOGIA..... | 15 |
| 3 | TIPOS E CLASSIFICAÇÕES DE SOFTWARES EDUCACIONAIS..... | 19 |
| 3.1 | CLASSIFICAÇÕES DOS SOFTWARES EDUCACIONAIS POR OBJETIVOS..... | 19 |
| 3.2 | CLASSIFICAÇÕES DOS SOFTWARES EDUCACIONAIS EDUCACIONAIS POR APRENDIZAGEM..... | 27 |
| 4 | MÉTODOS DE AVALIAÇÃO DE SOFTWARES EDUCACIONAIS..... | 31 |
| 4.1 | NORMA ISO DE PRODUTO DE SOFTWARE..... | 31 |
| 4.2 | O MODELO TUP..... | 33 |
| 4.3 | O MODELO DE REEVES..... | 40 |
| 4.4 | MODELO DE AVALIAÇÃO SEGUNDO CAMPOS..... | 46 |
| 5 | TÉCNICAS DE AVALIAÇÃO DE SOFTWARES EDUCACIONAIS..... | 49 |
| 5.1 | TESTES DE USABILIDADE..... | 49 |
| 5.2 | TESTES PEDAGÓGICOS..... | 51 |
| 6 | INSTRUMENTOS AVALIATIVOS DE SOFTWARES EDUCACIONAIS. | 57 |
| 6.1 | QUESTIONÁRIO DE NOKELAINEN..... | 57 |
| 6.2 | QUESTIONÁRIO DE SSEMUGABI | 58 |
| 6.3 | QUESTIONÁRIO DE REITZ | 61 |
| 6.4 | CHECKLIST DO PROINFO | 63 |
| 6.5 | AVALIAÇÃO DE OLIVEIRA | 64 |
| 6 | CONSIDERAÇÕES FINAIS..... | 67 |
| | REFERÊNCIAS | 70 |

| | |
|----------------------|-----------|
| ANEXO A..... | 74 |
| ANEXO B | 81 |
| ANEXO C..... | 87 |
| ANEXO D..... | 91 |
| ANEXO E..... | 92 |

1 INTRODUÇÃO

De acordo com Penha (2013), a inserção de *softwares* na educação nacional foi motivada pelo que ocorria em outros países como Estados Unidos e França. Educadores de universidades brasileiras, acompanhando o que vinha ocorrendo fora do país, demonstraram interesse em fazer o mesmo. Em meados de 1970, alguns *softwares* começaram a ser inseridos na área educacional. Esse processo iniciou-se na Universidade Federal de São Carlos (UFSCar), onde em um seminário, ministrado por E. Huggins, professor da Universidade de Dartmouth - Estados Unidos, foi proposto o uso de computadores no ensino de Física (PENHA, 2013).

Um *software* educacional é um “produto [...] adequadamente utilizado pela escola, mesmo que não tenha sido produzido com a finalidade de uso no sistema escolar” (OLIVEIRA, 2001, p. 73). Concordo com Oliveira (2001) que, independente do objetivo para o qual o *software* foi criado, ele pode se tornar um *software* educacional, entretanto, isso dependerá da forma como ele será utilizado no contexto educacional e com quais objetivos.

Com o passar do tempo, foram crescentes os resultados obtidos através da relação entre *softwares* e a educação para os envolvidos em processos de ensino e aprendizagem mediados por *softwares* educacionais. Segundo Prensky (2001 *apud* Pescador, 2010, p. 2), os “jovens estão acostumados a obter informações de forma rápida e costumam recorrer primeiramente a fontes digitais [...]. Por causa desses comportamentos e atitudes [...] Prensky os descreve como Nativos Digitais, uma vez que 'falam' a linguagem digital desde que nasceram”.

Segundo, Fialho e Matos (2010, p.123), os alunos aprendem melhor quando lhe são oferecidos um ambiente familiar ao seu cotidiano, vocabulário adequado e liberdade para descobrir a relação entre aquilo que se aprende e a realidade onde vive. Com isto, o uso de *softwares* educacionais é considerado uma contribuição ao processo de ensino e aprendizagem.

Assim sendo, o uso de *softwares* educacionais pode propiciar “experiências educacionais novas e ricas, ou pelo menos tornará muito mais eficiente o ensino efetivado nos moldes tradicionais” (RAMOS, 1991, p. 125). Para que isto ocorra, o *software* deve possuir características específicas, como ser de fácil utilização,

favorecer a compreensão e a assimilação dos conteúdos, deve despertar o interesse e manter a atenção do usuário.

Entretanto, verifica-se que os *softwares* trazem “embutidos certas qualidades e defeitos provenientes principalmente da concepção de ensino e de aprendizagem de quem o elaborou” (OLIVEIRA, 2001, p. 5). Assim como todo recurso utilizado em sala de aula, o *software* também deve passar por uma análise prévia do professor. Há que se avaliar as características técnicas e pedagógicas, de modo que a escolha de um *software* esteja de acordo com o Projeto Político Pedagógico (PPP) da escola e com o planejamento do professor.

O problema desta pesquisa e o interesse por investigá-lo surge a partir da participação no Projeto de Extensão Sobre Aprendizagem Móvel: Formação docente e aplicação de dispositivos móveis na Educação, sob a coordenação da Professora Doutora Lebiam Tamar Gomes Silva, realizado no Centro de Educação, da Universidade Federal da Paraíba (CE/UFPB), no ano de 2015. Foi possível verificar a relevância do estudo do tema, uma vez que, a entrada das Tecnologias da Informação e Comunicação nas escolas trouxe gradativamente a inserção dos *softwares* educacionais nas aulas.

Com o crescente uso de *softwares* educacionais nas escolas, os profissionais da educação precisam de métodos e de instrumentos que lhes permitam realizar a análise da qualidade do *software* a ser utilizado no processo de ensino e aprendizagem. Assim, este estudo tem como objetivo geral sistematizar métodos e instrumentos de análise de *softwares* educacionais, identificados em publicações científicas nacionais. Buscando responder quais os métodos e os instrumentos de análise de *softwares* educacionais desenvolvidos e aplicados no campo de pesquisa em educação e tecnologias?

Como objetivos específicos, estabelecemos:

- Conhecer os tipos de *softwares* educacionais, apresentados na literatura científica nacional;
- Identificar diferentes métodos de análise de *softwares* educacionais adotados nas pesquisas nacionais, no campo de educação e tecnologias;
- Apresentar instrumentos de análise de *softwares*, elaborados e aplicados por pesquisadores nacionais.

Este trabalho se debruça sobre os métodos e os instrumentos de análise de *softwares* educacionais, pretendendo dar uma contribuição para a reflexão teórica e para a prática educativa no campo da Pedagogia, colaborando com os professores em formação e em exercício profissional, que deseja adotar em suas práticas educativas *softwares* educacionais, e não sabem como selecionar um *software* adequado. e instrumentos de avaliação que possam auxiliá-los nas escolhas dos *softwares* educacionais, em prol do aprendizado do aluno. Por consequência, espera-se colaborar para uma elevação no nível de qualidade dos *softwares* educacionais utilizados nas escolas.

Esta monografia apresenta a seguinte estrutura:

Capítulo 1, introduz o tema e os conceitos de autores que discutem o uso de *softwares* educacionais no cenário nacional, a definição do problema de pesquisa e aponta os objetivos geral e específicos.

Capítulo 2, descreve os tipos e as classificações dos *softwares* educacionais.

Capítulo 3, apresenta os métodos de avaliação de *softwares* educacionais identificados.

Capítulo 4, expõe as técnicas de avaliação de *softwares* educacionais.

Capítulo 5, destaca instrumentos de avaliação de *softwares* educacionais.

Capítulo 6, propõe conclusões e recomendações para futuros trabalhos.

Ao final, encontra-se a bibliografia utilizada para a construção deste trabalho.

2 METODOLOGIA

Este trabalho resulta de uma pesquisa bibliográfica, a qual se realiza por meio da

[...] busca de informações bibliográficas, seleção de documentos que se relacionam com o problema de pesquisa (livros, verbetes de enciclopédia, artigos, de revistas, trabalhos de congressos, teses etc.) e o respectivo fichamento das referências para que sejam posteriormente utilizadas (na identificação do material referenciado ou na bibliografia final) (MACEDO, 1994, p. 13).

A pesquisa bibliográfica procura explicar e discutir um tema com base em referências teóricas já publicadas, entretanto, este tipo de pesquisa não é apenas uma mera repetição do que já foi dito ou escrito sobre determinado assunto, mas, sim, proporciona o exame de um tema sob novo enfoque ou abordagem, chegando a conclusões específicas a partir de um problema proposto (LIMA e MIOTO, 2007).

Como estratégia metodológica, foram aplicados os níveis de leitura e os procedimentos descritos por Lima e Mioto (2007) para a coleta, a organização e a categorização dos dados extraídos das publicações integrantes da amostra e para a produção das análises e da síntese integradora do estudo.

Os níveis de leitura descritos por Lima e Mioto (2007), são:

- a) Leitura de reconhecimento do material bibliográfico – Refere-se a uma leitura rápida, têm como objetivo a localização dos materiais que podem apresentar informações e/ou dados referentes ao tema.
- b) Leitura exploratória – Baseia-se na leitura de partes do estudo, de modo a comprovar a existência das informações pertinentes a pesquisa.
- c) Leitura seletiva – “Procura determinar o material que de fato interessa, relacionando-o diretamente aos objetivos da pesquisa” (LIMA; MIOTO, 2007, p. 41).
- d) Leitura reflexiva ou crítica – Após a seleção dos textos, realiza a leitura buscando a compreensão das afirmações dos autores, destacando os pontos mais importantes e o porquê dos destaques.
- e) Leitura interpretativa – “Implica na interpretação das idéias do autor, acompanhada de uma interrelação destas com o propósito do pesquisador” (LIMA; MIOTO, 2007, p. 41).

E os procedimentos descritos por Lima e mioto (2007), são:

- a) Elaboração do projeto de pesquisa – consiste na escolha do tema, na formulação do problema e objetivos da pesquisa.
- b) Investigação das soluções – É dividida em dois momentos: o primeiro consiste no levantamento da bibliografia e o segundo é o levantamento das informações contidas na bibliografia.
- c) Análise explicativa das soluções – Refere-se à análise das informações e dos dados contidos nos estudos, “é construída sob a capacidade crítica do pesquisador para explicar ou justificar os dados e/ou informações contidas no material selecionado” (LIMA; MIOTO, 2007, p. 41).
- d) Síntese integradora – É resultado final, construído a partir da análise e reflexão dos documentos.

Fez-se, em primeiro lugar a investigação das soluções, por meio do levantamento da bibliografia (LIMA e MIOTO, 2007), a busca por fontes científicas nacionais ocorreu a partir das palavras *softwares* educacionais, avaliação de *softwares* educacionais, modelos, técnicas e instrumentos de análise de *softwares* educacionais, nas plataformas digitais do *Google Acadêmico*, *Periódicos Científicos da Scielo*, *Portal dos Periódicos Livres da CAPES* e *Banco de Teses e Dissertações da Capes*. Como resultado, foram localizados 31 publicações, por meio da leitura de reconhecimento do material bibliográfico. Estes estudos variam do ano de 1991 a 2016, entre eles estão, artigos, monografias, dissertações e teses.

Em segundo lugar, fez-se uma leitura seletiva dos resumos, a fim de separar os estudos que tinham relevância para a pesquisa em tela, relacionando-os diretamente ao objetivo do estudo, o que possibilitou a seleção de oito publicações que condiziam com a investigação proposta neste trabalho. Os autores selecionados foram:

- Abreu (2010): Dissertação de Mestrado em Computação Aplicada. Seus estudos tem como foco a Informática Educativa.
- Andres e Cybis (2000): Artigo científico, disponível na Web. Respectivamente, Andres é Mestre em Ciências da Computação, suas pesquisas tem ênfase nos seguintes temas: engenharia de usabilidade, interfaces homem-computador, programação, informática na educação. Cybis é Mestre em Engenharia de Produção, atua nos seguintes temas: interface humano-

computador, ergonomia, usabilidade, avaliação, teste e monitoramento de interfaces.

- Martins (2002): Monografia de Pós-graduação *lato sensu* em Informática Educativa, pela Universidade Federal do Ceará.
- Oliveira (2001): Dissertação de Mestrado em Engenharia de Produção, pela Universidade Federal de Santa Catarina.
- Ramos (1991): Artigo publicado em eventos científico. Doutora em tecnologias de informação e comunicação na educação. As áreas de interesse são: Informática Educacional; Informática aplicada ao Ensino de Matemática.
- Rezende (2013): Dissertação de Mestrado Ciências pela Universidade Federal de Itajubá. Em sua dissertação pesquisou sobre modelo de avaliação de qualidade de *software* educacional para o ensino de Ciências.
- Texeira e Brandão (2003): Artigo publicado em revista científica. Doutor em Informática na Educação. Suas pesquisas enfatizam inclusão digital, informática educativa, software livre e formação docente.
- Vieira (2000): Artigo publicado em evento científico Doutora em Educação, trabalha os seguintes temas: tecnologias educacionais, educação à distância, ambiente virtual de aprendizagem, redes colaborativas de aprendizagem.

A terceira etapa consistiu na leitura reflexiva ou crítica dos estudos selecionados, para compor a análise explicativa das soluções (LIMA e MIOTO, 2007). À medida que ocorriam as leituras, foram elaborados fichamentos de transcrição separados sobre as seguintes categorias de analíticas:

- Classificações dos *softwares* educacionais: Nesta categoria de análise, buscou-se identificar os *softwares* educacionais suas características em relação ao objetivo e ao processo de ensino aprendizagem.
- Modelos de avaliação de *software* educacional: Foram destacados os modelos para a avaliação de um *software* educacional, abordando os aspectos de funcionalidade e os aspectos pedagógicos de um *software* educacional.
- Técnicas de avaliação de *software* educacional: Foram elencadas as técnicas de avaliação de *software* educacional mais utilizadas. Desde os testes de usabilidade até os testes pedagógicos.

- Instrumentos de avaliação de *software* educacional: Por fim, fez-se a seleção dos instrumentos de avaliação de *software* educacional destacados nas pesquisas selecionadas.

Os fichamentos tiveram como objetivo, sistematizar a coleta das informações pertinentes à pesquisa. Com a conclusão da leitura do material, da seleção e da organização das informações em fichamentos, iniciou-se a organização da redação da monografia, ou seja, da síntese integradora (LIMA e MIOTO, 2007) discutindo e analisando os dados obtidos.

3 CLASSIFICAÇÕES DOS SOFTWARES EDUCACIONAIS

A bibliografia consultada apresenta vários tipos de *softwares* utilizados no campo da educação. Segundo Teixeira e Brandão (2003, p. 2), *software* educacional, “é todo aquele *software* que possa ser usado com algum objetivo educacional, pedagogicamente defensável, por professores e alunos, qualquer que seja o objetivo para o qual ele foi criado”.

Oliveira (2001), Teixeira e Brandão (2003), afirmam que qualquer *software* pode ser utilizado para fins educacionais, mesmo não sendo explicitamente projetado para tal objetivo, entretanto, acrescento, para que um *software* seja considerado educacional, ele deve atender aos objetivos que estão sendo propostos no contexto educacional, independente dos objetivos para qual foram projetados.

Os *softwares* educacionais mais destacados nos estudos científicos reunidos nesta produção são: *software* de exercício e prática ou exercitação, *software* de simulação, *softwares* aplicativos, *softwares* de jogos educativos, *softwares* tutoriais, *softwares* de linguagem de programação, *software* de investigação.

3.1 CLASSIFICAÇÕES DOS SOFTWARES EDUCACIONAIS POR OBJETIVOS

Nas pesquisas realizadas foram encontrados diferentes *softwares*, classificados de acordo com seus objetivos, são eles: *software* de exercitação, *software* de simulação, *software* de modelagem, *software* aplicativos, *software* jogos educacionais, *software* tutoriais, *software* de linguagem de programação, *software* de investigação.

Os *softwares* de **exercício e prática ou exercitação** apresentam exercícios para a revisão de conteúdos, “buscam reforçar fatos e conhecimentos e têm como principais características a memorização e a repetição” (OLIVEIRA, 2001, p. 40), não tendo a preocupação de como o aluno está compreendendo o que está fazendo.

Esse tipo de *software*, além de apresentar o exercício, faz um apanhado das respostas, de modo a verificar o desempenho do usuário. Então, o professor terá à sua disposição dados importantes sobre como a aprendizagem é realizada a partir do ensino dos conteúdos curriculares. Um exemplo de *software* de exercitação é o *Math Master - Brain Quizzes* (Figura 1).

Figura 1 - Exemplo de *Software* de Exercitação

Fonte: *Print screen* do software na *Play Store*¹.

O *Math Master* é um jogo gratuito que propõe ao usuário perguntas sobre conteúdos da Matemática, oferecendo um conjunto de testes desafiadores, dividido em 12 livros. São eles: Adição, Subtração, Multiplicação, Divisão, Normal - média e mediana, Potência, Estatísticas, Menor e Maior, Equações, Sequência e Séries. Cada livro tem 10 capítulos exclusivos com níveis de dificuldade crescentes. Cada pergunta, possui 4 alternativas. O *software* não aceita respostas erradas. Para cada questão, o cronômetro é acionado. (GOOGLE PLAY, 2017).

Os *softwares* de **simulação** permitem ao aluno realizar atividades das quais normalmente não poderia participar, ou seja, por meio da simulação, é criada uma situação que se assemelha com a realidade, onde o “aluno pode testar, tomar decisões, analisar, sintetizar e aplicar conhecimentos”. (GAMEZ 1998 *apud* OLIVEIRA, 2001, p.55).

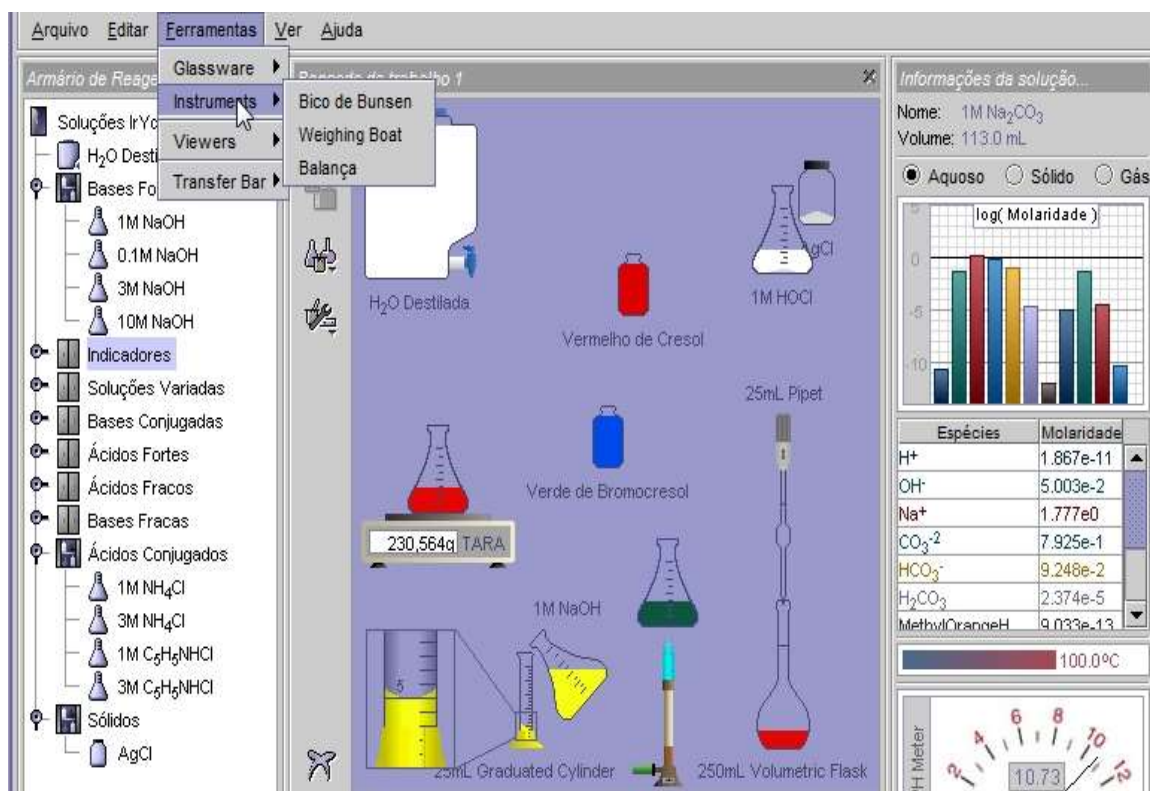
Para Valente (1993 *apud* OLIVEIRA, 2001, p. 41), a utilização desse *softwares* na educação é proveitosa para tomar decisões, testar diferentes hipóteses, e assim, ter um contato mais “real” com os conceitos envolvidos no problema em estudo.

Um exemplo desses *softwares* é o *IrYdium - Virtual Chemistry Lab* (Figura 2). Ele permite que os usuários possam selecionar centenas de reagentes e os

¹ Disponível em: https://play.google.com/store/apps/details?id=com.mathmaster&hl=pt_BR>
Acesso em: 01 de Março de 2017.

manipular, como se estivessem em um laboratório real de química executando experiências diversas. (BAIXAKI, 2010).

Figura 2 - Exemplo de Software de Simulação



Fonte: Baixaki.²

Outro tipo de *software* é o de **modelagem**, o usuário pode simular um determinado acontecimento por meio do *software*. Entretanto, o *software* de modelagem e o *software* de simulação são distintos.

Ao usuário da simulação, cabe a alteração de certos parâmetros e a observação do comportamento do fenômeno, de acordo com os valores atribuídos. Na modelagem, o modelo do fenômeno é criado pelo aprendiz, que utiliza recursos de um sistema computacional para implementá-lo. Uma vez implementado, o aprendiz pode utilizá-lo como se fosse uma simulação. (BORNATTO, 2002, p. 68).

O *software* de modelagem mais conhecido é o *Modellus* (Figura 3). A ideia básica do *Modellus* é a de facilitar a realização de experiências (ou simulações) com o auxílio do computador nas aprendizagens de Matemática, Física e Química.

² Disponível em: <<http://www.baixaki.com.br/download/irydium-chemistry-lab.htm>> Acesso em 01 de Março de 2017.

Figura 3 - Exemplo de Software de Modelagem



Fonte: Expressando a Física³.

Os **softwares aplicativos** são voltados para aplicações em atividades específicas como: processadores de texto, planilhas eletrônicas, de apresentação (Figura 4). Segundo Martins (2002, p. 11), “eles não foram desenvolvidos para uso educacional, porém, vem sendo adaptados com esse objetivo”.

Figura 4 - Exemplo de Software Aplicativo



Fonte: Google Imagens⁴.

Os **jogos educativos** são outro tipo de *softwares* têm o objetivo de ensinar às pessoas sobre determinado assunto ou conceito de forma lúdica. (MARTINS, 2002, p. 12). Os jogos educativos possibilitam ao aluno aprender de prazerosa e dinâmica, porque possuem desafios que despertam interesse e a motivação no processo de ensino aprendizagem.

Um exemplo de jogo educativo é o *Vrum* Aprendendo sobre o trânsito (Figura 5). Esse *software* está baseado nas Diretrizes Nacionais da Educação para o

³ Disponível em: <<http://expressandoafisica.blogspot.com.br/2014/05/modelagem-do-movimento-retilineo.html>> Acesso em 09 de Junho de 2017.

⁴ Disponível em: <<http://www.tudocelular.com/android/noticias/n61738/Parceria-trara-pacote-Office-pre-instalado-em-smartphones-da-Asus.html>> Acesso em 01 de Março de 2017.

Trânsito do DENATRAN e tem o objetivo de apresentar, de maneira lúdica, as principais regras de trânsito para os usuários.

Figura 5 - Exemplo de *Softwares* Jogos Educativos



Fonte: Vrum⁵.

Há também os **softwares tutoriais**, utilizados para apresentar informações novas aos seus usuários, direcionar o aprendizado. As atividades são organizadas de acordo com uma sequência pedagógica particular e apresentadas aos usuários, seguindo essa sequência. No caso de tutoriais, os dispositivos utilizados (computador, *tablet*) assumem o papel de máquina de ensinar (VIEIRA, 2000 *apud* MARTINS, 2002, p. 11).

As mensagens de erro que aparecem têm o intuito de conduzir o aluno às respostas corretas, desejadas ou adequadas, ou seja, direcionar o aprendizado. Ao final de cada atividade, os *softwares* tutoriais dão *feedback* imediato e avaliam o desempenho do usuário. (MARTINS, 2002).

Um exemplo de *software* tutorial é a tabela periódica virtual (Figura 6). Ele apresenta todos os elementos da tabela periódica e exibe dados e classificações dos elementos químicos.

⁵ Disponível em: <<http://www.jogovrum.com.br/>> Acesso em 01 de Março de 2017.

Figura 6 - Exemplo de Softwares Tutoriais

Para os elementos sem isótopos estáveis, o número de massa do isótopo com a meia-vida mais longa está entre parênteses.

Fonte: Print screen do software na Play Store⁶

Nas literaturas foram identificados os **softwares de linguagem de programação** que, segundo VIEIRA (2000 *apud* MARTINS, 2002, p. 11) “são softwares que permitem que as pessoas, professores ou alunos, criem seus próprios protótipos de programas”. Com eles, são possíveis a criação de diferentes softwares, sem precisar de conhecimentos avançados sobre programação, com a implementação de várias mídias (som, vídeo, movimento etc.) que favoreçam a interação do usuário com o software projetado. Um exemplo desse tipo de software *Construct* (Figura 7) é destinado para não-programadores, permitindo a criação rápida de jogos.

Figura 7 - Exemplo de Software de linguagem de programação



Fonte: Fábrica de Aplicativos⁷

⁶ Disponível em: <<https://play.google.com/store/apps/details?id=jqsoft.apps.periodictable.hd&hl=pt-BR>> Acesso em 01 de Março de 2017.

Por fim, os **softwares de investigações**, nesta categoria se enquadram todos os *softwares* capazes de localizar informações complementares. São exemplos deles, as Enciclopédias e os Dicionários (Figura 8).

Figura 8 - Exemplo de *Softwares* de Investigação



Fonte: Baixaki⁸

Para auxiliar na compreensão das informações apresentadas neste capítulo, foram elaborados dois infográficos, o primeiro (Figura 9) com as classificações dos *softwares* educacionais por objetivo.

⁷ Disponível em: <<http://fabricadeaplicativos.com.br/>> Acesso em 01 de Março de 2017.

⁸ Disponível em: <<http://www.baixaki.com.br/windows-8/dicionario-aurelio.htm>> Acesso em 01 de Março de 2017.

Figura 9 – Classificação dos softwares educacionais por objetivos



Fonte: Elaborado pela Autora (2017)

3.2 CLASSIFICAÇÕES DOS SOFTWARES EDUCACIONAIS POR APRENDIZAGEM

Os *softwares* educacionais apresentam características com ênfase no processo de ensino e aprendizagem, três modalidades foram destaques nos estudos selecionados, são elas: nível de aprendizagem (MARTINS, 2002); atividade do sujeito de aprendizagem (RAMOS, 1991); e os paradigmas educacionais de um *software* (RAMOS, 1991).

Segundo Vieira (2000 *apud* MARTINS, 2002, p. 12), os *softwares* educacionais são classificados de acordo com o **nível de aprendizagem** que cada um é capaz de proporcionar. Nesse caso, temos os seguintes tipos, com base em Martins (2002, p. 12.):

- **Sequencial** – tem o objetivo de apenas transferir a informação para o usuário, de forma sequencial, o aluno “memoriza e repete conteúdos quando solicitado, o que resulta em um aprendizado passivo e sem reflexão” (MARTINS, 2002, p.12). Como exemplos desta categoria temos os *softwares* de exercício e prática ou exercitação, aplicativos e os *softwares* de tutoriais.
- **Relacional** – É centrado no usuário, tem como objetivo a aquisição de habilidade e interação com a tecnologia, “permite que o aluno possa fazer relações com outros fatos ou faça uso de outras fontes de informação” (MARTINS, 2002, p.12). Assim, aprendizagem se processa a partir da interação do aprendiz com a tecnologia. A principal característica do *software* é um aprendiz isolado, uma vez que, a interação ocorre apenas entre o *software* e o usuário, não há interação com outros sujeitos. São exemplos deles, os *softwares* de investigação.
- **Criativo** – Tem como objetivo criar novos esquemas mentais. Sua principal característica é o aluno participativo, por meio da tecnologia, “o aluno pode interagir com outras pessoas compartilhando objetivos comuns” (MARTINS, 2002, p.12), proporcionando o desenvolvimento da criatividade, da interação entre os integrantes do grupo e destes com a tecnologia, são exemplos de *softwares* de simulação.

Quanto à **atividade do sujeito de aprendizagem**, Galvis (1988 *apud* RAMOS, 1991, p.123) cita a classificação proposta por Thomas Dwyer (1988), dividido em dois grupos. O primeiro, **software com enfoque do tipo algoritmo**,

é predominante a ênfase na transmissão de conhecimento do sujeito que sabe para o sujeito que deseja aprender, sendo neste caso função do criador do *software*, projetar uma sequência bem planejada de atividades que conduzam o aprendiz ao objetivo esperado. (RAMOS, 1991, p. 123).

Este enfoque utiliza basicamente *softwares* do tipo tutoriais e exercício e prática, centrados na simples transmissão de conteúdos, através de uma sequência de atividades. Logo, em termos de nível de aprendizagem são classificados como sequenciais.

O segundo grupo são os **softwares com enfoque do tipo heurístico**. Nestes, o aspecto predominante “é a aprendizagem experimental ou por descobrimento, devendo o *software* criar um ambiente rico em situações que o aluno deve explorar conjecturalmente” (RAMOS, 1991, p.124). Fazem parte desse grupo as simulações e os jogos, nos quais predominam a aprendizagem experimental ou por descoberta.

Ramos (1991, p.124) aborda os paradigmas educacionais dos *softwares*. Destaca o **comportamentalista**, na educação está associado ao trabalho de Burrhus Frederic Skinner (1968), que vendo a necessidade de um instrumento que auxiliasse o professor em sala de aula, propôs a “máquina de ensinar”.

[Essa máquina] nada mais era do que meios ou dispositivos mecânicos para a apresentação de seqüências de aprendizagem programadas de acordo com os princípios comportamentais descritos anteriormente, elas avançavam linearmente ao toque de alavancas com a ocorrência de respostas certas (CARRAHER, 1992, p. 161).

Segundo Skinner e seu “ensino programado” (1968 *apud* MARTINS, 2002, p.17),

as pessoas aprendem mais facilmente quando o conteúdo é:

- Apresentado em seções breves;
- Testa o estudante após cada seção;
- Apresenta *feedback* imediato para as respostas dadas.

Martins (2002), afirma que o uso desses *softwares* têm como consequência:

- ❑ Alunos passivos, uma vez que, realizam o exercício sem refletir sobre seus procedimentos;
- ❑ Os alunos não podem fazer uso de outras alternativas para a resolução de problemas, a não ser as que estão disponíveis nos *softwares*;
- ❑ O aluno não tem a opção de discordar do *software*;

Entretanto, os *softwares* com essa abordagem comportamentalista podem ser utilizados na sistematização de informações, uma vez que, são capazes de levar a uma compreensão mais profunda sobre conceitos estudados. O paradigma comportamentalista abrange o enfoque algorítmico.

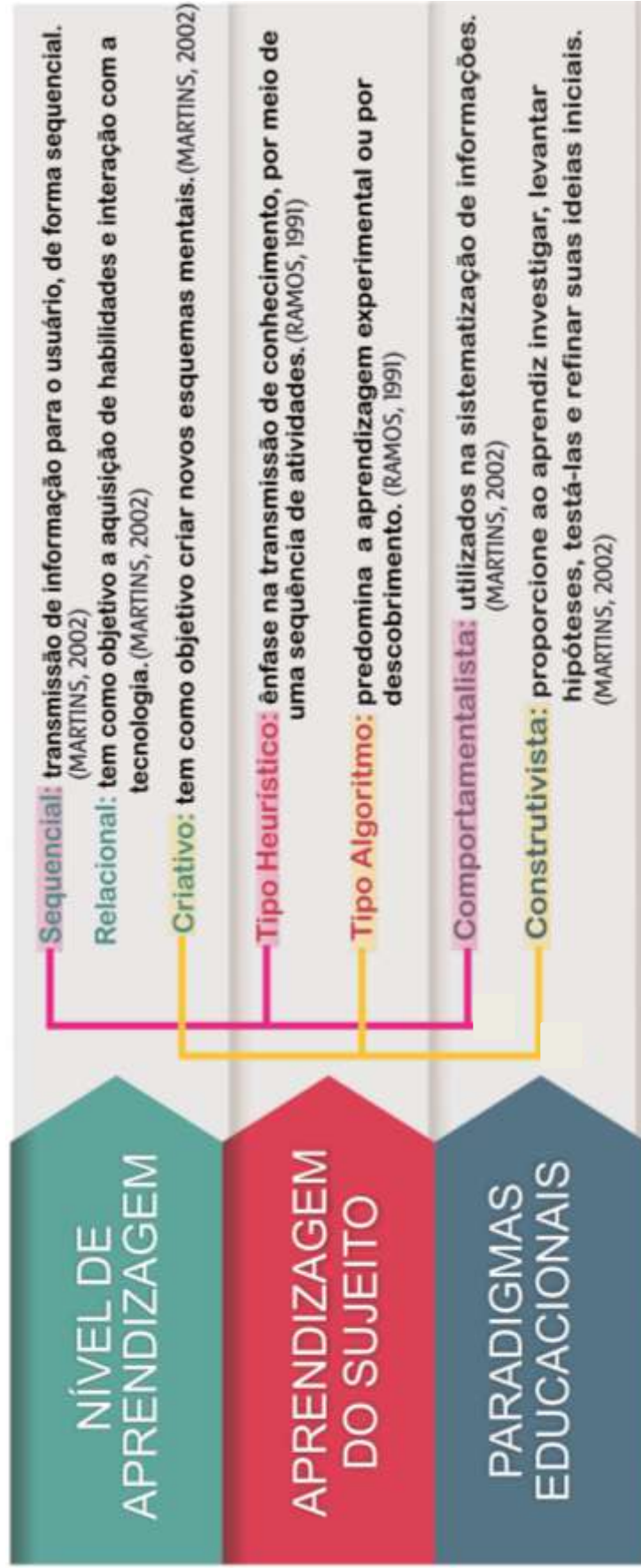
Outro paradigma citado por Ramos (1991, p.124) é o **construtivista**, é uma teoria educacional (ou de aprendizagem) desenvolvida pelo matemático Seymour Papert, que se baseia, principalmente, na teoria epistemológica desenvolvida por Jean Piaget (MALTEMPI, 2005, p. 2). Para Papert, a atividade de programação permite ao aluno resolver problemas que envolvem abstrações, aplicação de estratégias, estruturas e conceitos já construídos, ou a criação de novas estratégias, estruturas e conceitos.

Diante de uma proposta construtivista, o trabalho pedagógico realizado com computador, o utilizará enquanto uma ferramenta, pois a atitude construtivista implica na meta de ensinar de forma que o máximo de aprendizagem seja produzida, com o mínimo de ensino (MARTINS, 2002, p. 21).

Assim, um *software* construtivista deve ser um ambiente interativo que proporcione ao aprendiz investigar, levantar hipóteses, testá-las e refinar suas ideias iniciais. Dessa forma, o aprendiz estará construindo o seu próprio conhecimento. O paradigma construtivista abrange o enfoque heurístico.

Para auxiliar na compreensão das informações apresentadas neste capítulo, foi elaborado infográfico (Figura 10) classificações dos *softwares* educacionais por aprendizagem.

Figura 10 – Classificação dos softwares educacionais por aprendizagem



Fonte: Elaborado pela Autora (2017)

4. MODELOS DE AVALIAÇÃO DE SOFTWARES EDUCACIONAIS

Segundo Rezende, (2013, p.20) “os modelos orientados para a avaliação de *software* educacional, além de levarem em conta os requisitos exigidos por *softwares* em geral, devem também atender a exigências específicas do ensino e aprendizagem”. Então, pode-se entender que, modelos de avaliação de softwares educacionais, são um conjunto de requisitos exigidos em um *software*, que devem ser adotadas para uso no contexto educacional.

Existem diversas metodologias para avaliar *softwares* educacionais. Pode-se destacar: (a) a norma ISO de produto de *softwares*; (b) o modelo TUP (Tecnologia, Usabilidade, Pedagogia), (c) o modelo de Reeves e (d) o modelo de avaliação segundo Campos.

4.1 NORMA ISO DE PRODUTO DE SOFTWARE

A International Organization of Standardization - ISO e a Electrotechnical Commission - IEC⁹ elaboraram um conjunto de normas que tratam sobre a atual padronização mundial para a qualidade de produtos de todos os tipos de *softwares*:

As várias partes que compõem a norma NBR ISO/IEC apresentam, de maneira geral, os instrumentos necessários para realizar uma avaliação de *software*, descrevendo como medir qualitativamente e quantitativamente a qualidade do produto (REZENDE, 2013, p. 33).

De acordo com a NBR ISO/IEC 9126-1¹⁰(ABNT, 2003 *apud* REZENDE, 2013, p. 30), a qualidade diz respeito à "totalidade de características de uma entidade que lhe confere a capacidade de satisfazer as necessidades explícitas e implícitas". As características explícitas abrangem os objetivos, as funções e o desempenho esperado dos *softwares*. As implícitas são os princípios básicos e óbvios, necessários para que o usuário execute a sua tarefa.

⁹ Normas da ISO e IEC publicadas entre os anos de 2014 - 2017, disponíveis em <<http://www.abnt.org.br/normalizacao/lista-de-publicacoes/normas-iso-e-iec-publicadas>> Acesso em: 13 de Maio de 2017

¹⁰ Engenharia de *software* - Qualidade de produto, disponível em <http://jkolb.com.br/wp-content/uploads/2014/02/NBR-ISO_IEC-9126-1.pdf> Acesso em: 13 de Maio de 2017

A NBR ISO/IEC 25010¹¹ é uma norma que “define as características de qualidade que todos os softwares devem ter” (REZENDE, 2013, p. 30). Este modelo apresenta oito características de qualidade a serem abordadas por qualquer tipo de *software*. São elas:

- ❑ **Funcionalidade:** Completude funcional, correção funcional, adequação funcional;
- ❑ **Eficiência:** Tempo comportamental, utilização dos recursos, capacidade;
- ❑ **Compatibilidade:** Coexistência, Interoperável;
- ❑ **Usabilidade:** Capacidade de aprendizado, operabilidade, proteção contra erros do usuário, estética de interface de usuário, acessibilidade;
- ❑ **Confiabilidade:** Maturidade, tolerância a falhas, recuperabilidade;
- ❑ **Segurança:** Confidencialidade, integridade, não repúdio, responsabilidade, autenticidade;
- ❑ **Modularidade:** Modularidade, reusabilidade, analisabilidade, testabilidade;
- ❑ **Portabilidade:** Adaptabilidade, instabilidade, substituição. (ABNT, 2011 apud REZENDE, 2013, p. 31).

O uso da norma ISO/IEC 25010 estabelece as características e as subcaracterísticas de qualidade para uma avaliação de *software*, permitindo verificar os atributos necessários a cada tipo de *softwares*. As características e as subcaracterísticas, definidas pelos modelos ISO/IEC, representam um referencial relevante para os estudos sobre avaliação da qualidade de *softwares* em geral, independente da aplicação dos mesmos.

A proposta apresentada por essa norma está fora do escopo deste trabalho, que tem por objetivo, apresentar modelos para avaliação de *softwares* educacionais. Entretanto, pode ser interessante aplicá-la para avaliar um *software* educacional, uma vez que, a norma ISO/IEC apresenta os requisitos de qualidade para a avaliação de *softwares* em geral, auxiliando os usuários na avaliação da adequação do *software* à sua necessidade.

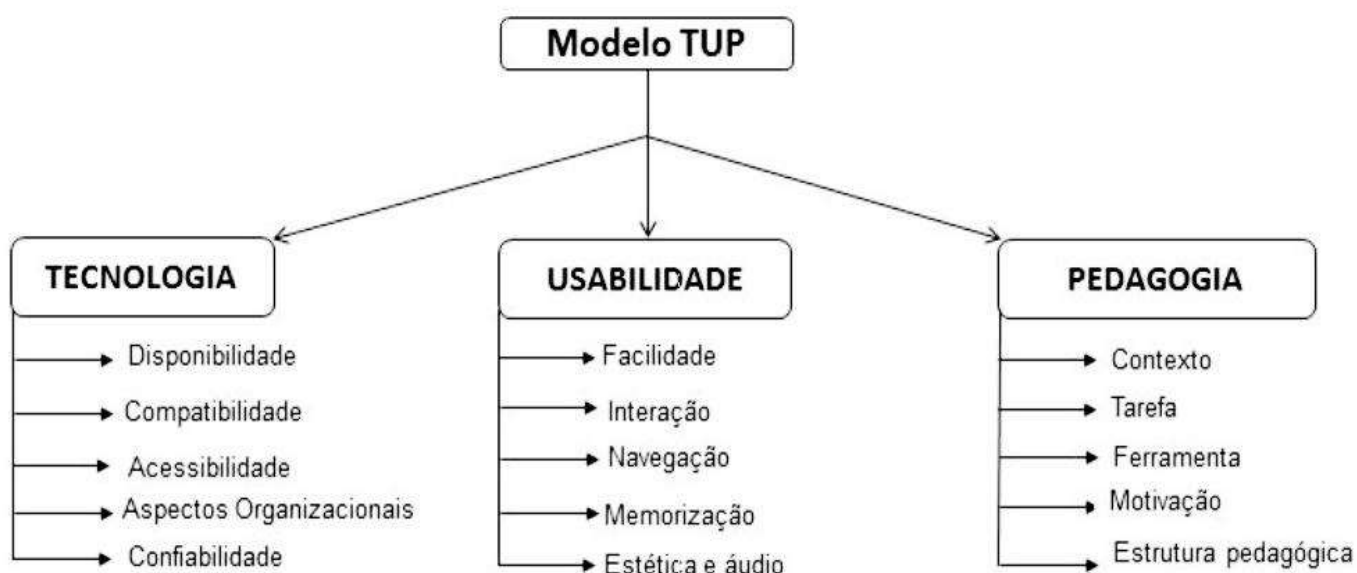
¹¹ Característica de qualidade, disponível em <<https://prezi.com/k6m84f3u5jrk/iso-25010/>> Acesso em: 13 de Maio de 2017

4.2 O MODELO TUP (BEDNARIK, 2004 *apud* REZENDE, 2013)

Bednarik (2004 *apud* REZENDE, 2013, p. 21) propõe um modelo chamado TUP (Figura 11), do inglês *Technology, Usability and Pedagogy*, surgiu da necessidade de criação de um modelo de avaliação de *software* educacional, que permitisse aos profissionais da educação a seleção de *softwares* de acordo com um conjunto estruturado de critérios definidos.

Esse modelo destaca a importância de se avaliar e selecionar cuidadosamente um *software* educacional, trabalhando de forma interdisciplinar, compreendendo aspectos de tecnologia, usabilidade e pedagógicos, integrados em um método de avaliação, conforme mostra os requisitos abaixo:

Figura 11 - Esquema do modelo TUP



Fonte: BEDNARIK, 2004 *apud* REZENDE, 2013, p. 22)

No ato de avaliar um *software* educacional é importante analisar os **aspectos da tecnologia**, assim, Rezende (2013) pontua aspectos importantes na análise de um *software* educacional relacionado à tecnologia, bem como: disponibilidade, compatibilidade, acessibilidade, aspectos organizacionais e a confiabilidade.

A disponibilidade diz respeito ao *software* ficar disponível o maior tempo possível para uso, durante um determinado período, por exemplo, o *software* deverá estar disponível 24h por 7 dias na semana, ou seja, deve estar disponível para

quando o usuário precisar. Outro aspecto importante é a compatibilidade, significa que o *software* deve funcionar em diferentes sistemas operacionais.

O terceiro atributo é a acessibilidade, segundo Rezende (2013) o *software* deve atender diferentes usuários com distintos perfis e necessidades, desse modo, pode-se entender que, usuários diferentes têm necessidades diferentes e os *softwares* precisam ser projetados de acordo com tais necessidades.

O próximo atributo são os aspectos organizacionais, engloba as questões de planejamento, monitoramento e integração das tarefas apoiadas pelo *software*. Por fim, a confiabilidade, é geralmente definida pela segurança, confidencialidade e autenticidade do *software*.

Em uma análise de um *software* educacional, deve-se também avaliar as **características de usabilidade**, que é a questão relacionada a quão bem os usuários podem usar a funcionalidade definida pelo sistema do *software*. Segundo Rezende (2013), associada a cinco atributos para avaliação da usabilidade.

O primeiro é a facilidade, o *software* deve levar em conta que os seus usuários não precisa ter conhecimentos específicos acerca do *software*, desse modo, à necessidade da existência de instruções claras e objetivas e opções para o esclarecimento de dúvidas e ajuda durante a utilização do *software*, com a existência desses elementos resultará em uma facilidade para o usuário utilizar o *software* e realizar suas tarefas, aproveitando o máximo do *software*.

O próximo atributo é a interação, está ligada a “maneira como o usuário se relaciona com o *software*, entende os comandos e realiza as tarefas de seu interesse.” (REZENDE, 2013, p.25.) Assim, o *software* deve estar sempre se comunicando com o usuário, cada elemento presente na interface do *software* (ícones, botões, sons, palavras) têm o potencial de comunicar algo.

O terceiro atributo é a navegação, refere-se aos ““caminhos” exigidos para explorar o sistema” (REZENDE, 2013, p.25), ou seja, o *software* deve informar para os usuários, a localização deles no *software* e indicar quais os caminhos já percorridos e quantos ainda estão disponíveis.

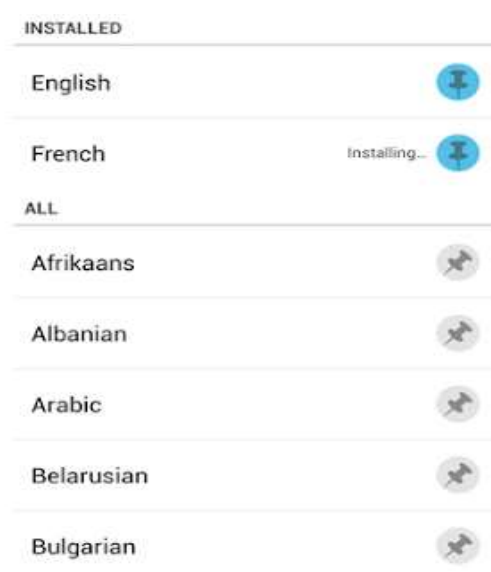
O atributo seguinte é a memorização, refere-se a capacidade de manter o registro do ponto onde cada usuário estava no momento de interrupção do uso do *software*, garantindo-lhe dessa forma a possibilidade de dar continuidade num outro momento a utilização do *software*.

O último atributo em relação a usabilidade é a estética e áudio, os *softwares* devem ter uma “visualização agradável e recursos de áudio adequados, que deixem o usuário satisfeito” (REZENDE, 2013, p. 25), ou seja, o *software* deve ter um *layout* esteticamente adequado, fazendo uso de texto bem distribuído, imagens e animações pertinentes ao contexto proposto no *software*, efeitos sonoros oportunos, entretanto, não devem afetar a atenção do usuário durante a realização da atividade proposta no *software*.

Os *Softwares* educacionais possuem características que os diferenciam dos demais *softwares*, uma vez que sua ênfase está no aprendizado. A avaliação da qualidade de um *software* educacional deve levar em conta, principalmente, as características relacionadas à qualidade didático-pedagógica. Neste aspecto, os atributos mais importantes a serem considerados segundo Rezende (2013), são: contexto, tarefa, ferramenta, motivação e estrutura pedagógica.

O contexto, “refere-se aos aspectos multiculturais e multilingues” (REZENDE, 2013, p. 26), ou seja, o *software* deve buscar atender diferentes culturas e promover o uso de vários idiomas, possibilitando o uso do *software* por um público mais amplo. Por exemplo, *Google Tradutor* (Figura 12), este *software* faz uso de 103 idiomas, atendendo um grande número de pessoas.

Figura 12 -Tela do *Software Google Tradutor*



Fonte: *Print screen* do *software* na *Play Store*¹².

¹² Disponível em:
<<https://play.google.com/store/apps/details?id=com.google.android.apps.translate&hl=pt-br>> Acesso em 17 de Maio de 2017.

A tarefa, diz respeito às atividades que compõem o *software*, elas devem possuir ajustes de modo que atenda os níveis de conhecimentos esperados dos usuários e apresentar condições para o alcance dos objetivos, por meio da compatibilidade dos desafios, das simulações.

Por exemplo, o *software Monster Math 2* (Figura 13), é um jogo educativo, que tem como objetivo aprender matemática de forma divertida. O jogo é personalizável, permite alternar entre matemática básica e avançada, além de optar pelos conteúdos que deseja jogar, de acordo com as necessidades dos usuários.

Figura 13 -Tela do *Monster Math 2*



Fonte: *Print screen* do *software* na *Play Store*¹³.

As ferramentas, “refere-se a instrumentos que possibilitam a compreensão dos processos pedagógicos específicos de aprendizagem” (REZENDE, 2013, p. 26) desse modo, alguns dos instrumentos que facilitam a aprendizagem sem sobrecarga ou deficiência de informações são as imagens, texto, sons e animações.

Por exemplo, o *software* Sistemas do Corpo Humano 3D (Figura 14), apresenta as principais características do corpo humano através de do uso de imagens detalhadas, pequenos textos explicativo e animações 3D, o que auxilia no processo de ensino aprendizagem.

¹³ Disponível em: <<https://play.google.com/store/apps/details?id=com.makkajai.monstermath2free&hl=pt-br>> Acesso em: 17 de Maio de 2017.

Figura 14 -Tela do *software Sistema do Corpo Humano 3D*

Fonte: *Print screen do software na Play Store*¹⁴.

Em relação à motivação, o *software* deve despertar o interesse dos usuários para que se possam cumprir os objetivos da aprendizagem, sejam por meio de premiações (estrela, medalhas entre outros) ou pela presença de desafios capazes de provocar e de manter o nível de interesse dos usuários.

Por exemplo, o *software Duolingo* (Figura 15), para motivar os usuários a cumprirem as atividades, faz uso de corações, que representam as vidas do jogador e a cada atividade concluída com sucesso o usuário acumula pontos.

Figura 15 -Tela do *software Duolingo*

Fonte: *Print screen do software na Play Store*¹⁵.

A estrutura pedagógica “refere-se ao suporte oferecido pelo *software* para o gerenciamento de materiais instrucionais, que ofereçam perspectivas dos estilos e processos de aprendizagem”. (REZENDE, 2013, p.26). O *software* deve conter a

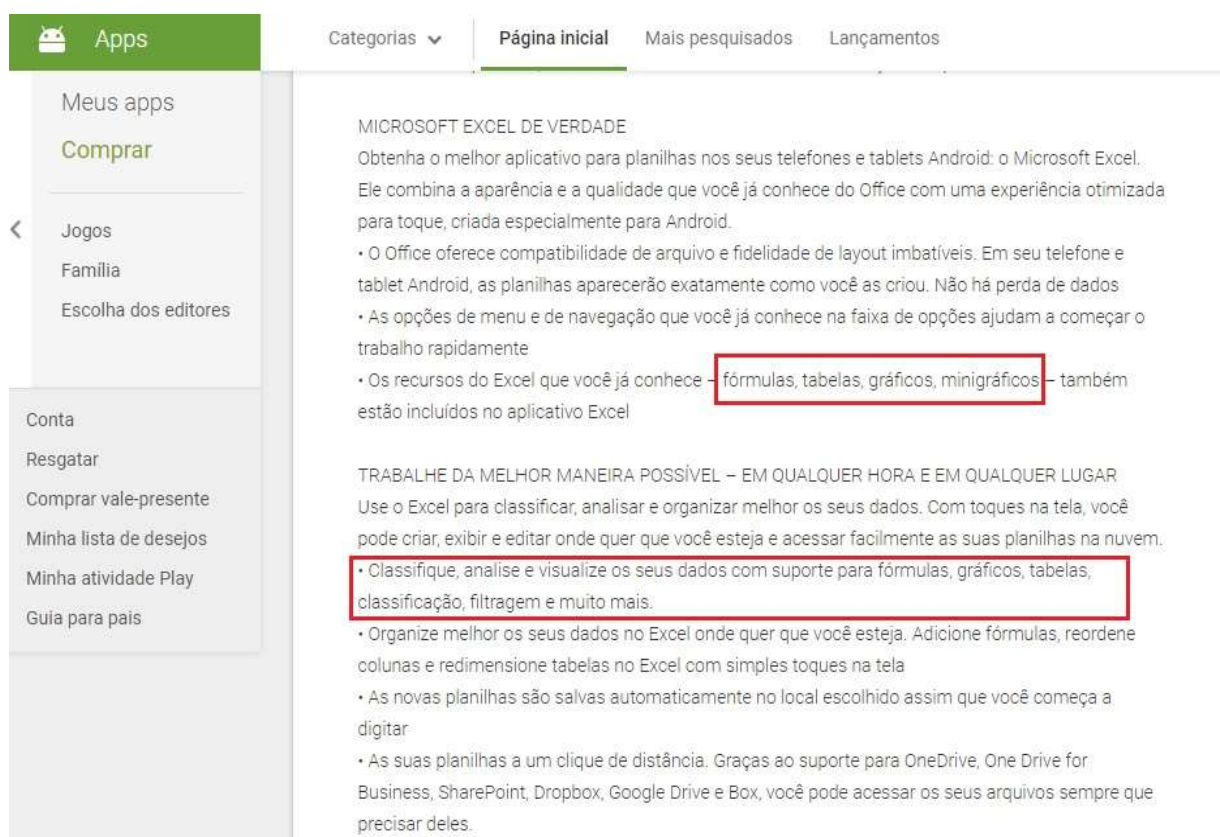
¹⁴ Disponível em: <<https://play.google.com/store/apps/details?id=com.EvoBooks.SistemasCorpoHumano&hl=pt-br>> Acesso em 17 de Maio de 2017.

¹⁵ Disponível em: <<https://play.google.com/store/apps/details?id=com.duolingo&hl=pt-br>> Acesso em 17 de Maio de 2017.

explicitação dos objetivos pedagógicos e definição do público-alvo do *software* educacional ou até mesmo sugestões de utilização que favoreçam a integração do *software* às atividades em sala de aula.

Por exemplo, o *software Excel* (Figura 16) tem como objetivo classificar, analisar e organizar melhor os seus dados, na loja *Play Store*, nas informações apresentadas para o usuário, o seu desenvolvedor, sugere utilizar o *software* para classificar, analisar e visualizar dados com suporte para fórmulas, gráficos, tabelas, classificação, filtragem entre outros.

Figura 16 - Informações sobre *software Excel*



Fonte: *Print screen do software na Play Store*¹⁶.

Segundo Bednarik (2002 *apud* REZENDE, 2013, p.26), o motivo do desenvolvimento do modelo TUP foi

a necessidade de avaliar adequadamente a interseção entre a computação e a educação, para a criação de um método de

¹⁶ Disponível em: <<https://play.google.com/store/apps/details?id=com.microsoft.office.excel&hl=pt-br>> Acesso em 17 de Maio de 2017.

avaliação de software educacional que permitisse aos educadores selecionar softwares de seu interesse, a partir de um conjunto estruturado de critérios. Tal objetivo é importante, uma vez que, devido à diversidade de produtos de software disponíveis, o professor, muitas vezes, não possui informações suficientes para uma seleção coerente e que leve a impactos positivos para o ensino e aprendizagem.

A partir do modelo TUP, foi desenvolvida uma ferramenta de *software*, também denominada TUP (Figura 17). Segundo Bednarik (2004 *apud* REZENDE, 2013, p.26), essa ferramenta “tem por finalidade apoiar a realização de avaliações de *softwares* educacionais, de acordo com o modelo proposto”. A ferramenta TUP está acessível via *online*. Os usuários que acessarem a ferramenta deverão realizar um cadastro e, a partir deste, fazer suas avaliações e também consultar outras avaliações já realizadas e armazenadas na ferramenta.

Figura 17 - Ferramenta TUP *online*

Fonte: <http://cs.joensuu.fi/~tup/>

Conforme REZENDE, (2013, p. 33), a ferramenta TUP coleta as informações necessárias para a avaliação do *software*, por meio de um questionário, utilizando técnicas de avaliação baseadas nos conceitos concordo e discordo associados a uma Escala *Likert* (LIKERT, 1932 *apud* REZENDE, 2013, p. 33).

A Escala *Likert* é uma escala em que os usuários são solicitados não só a concordarem ou discordarem das afirmações, mas também informarem qual o seu grau de concordância ou discordância. A cada resposta, é atribuído um valor que reflete a direção (positiva ou negativa) da atitude do usuário em relação a cada afirmação ou questionamento. A pontuação da atitude de cada usuário é dada pela somatória das pontuações obtidas de cada afirmação (KINNEAR, 1991; SARAIVA, 2006 *apud* REZENDE, 2013, p. 33).

O questionário adotado na ferramenta TUP é do tipo *checklist* “contendo 94 questões: 26 questões envolvem o aspecto tecnológico; 26 questões tratam do aspecto da usabilidade; 40 questões abordam o aspecto pedagógico; e 2 questões abrangem aspectos gerais” (FREITAS; KIRNER, 2013, p. 9). Desse modo, os requisitos e atributos que devem estar presentes no *software* educacional são postos em análise, para, a partir daí, se identificar o seu nível de qualidade.

4.3 O MODELO DE REEVES (REEVES, 1998 *apud* REZENDE, 2013, p. 33)

Professor Thomas C. Reeves, desenvolveu e avaliou inúmeros programas de aprendizagem interativos para a educação e formação. Seu interesse de pesquisa incluem: avaliação da tecnologia instrucional, modelos mentais e multimídia interativa, ambientes de aprendizagem autênticos. Suas áreas de ensino incluem avaliação de programas e *design* instrucional.

Em 1994, propôs o modelo de Reeves, que “tem sido amplamente usado em avaliações de *software*, em diferentes domínios, abordando principalmente *software* interativo e com uso de recursos de multimídia” (FREITAS; KIRNER, 2013, p.10).

O modelo define duas abordagens para avaliar um *software* educacional. A primeira abordagem está relacionada com aspectos pedagógicos e baseia-se em quatorze critérios. A segunda abordagem avalia a qualidade do *software*, referente aos aspectos da interação humano-computador, utilizando dez critérios.

Os quatorze critérios pedagógicos (Figura 18) dos *softwares* educacionais são:

Figura 18 - Modelo de Reeves aspectos pedagógicos dos *softwares* educacionais
Figura 18 - Modelo de Reeves aspectos pedagógicos dos *softwares* educacionais

Figura 18 - Modelo de Reeves aspectos pedagógicos dos *softwares* educacionais

| | | | |
|---------------------------------------|-------------------------|--------|--------------------|
| Epistemologia | Objetivista | ←————→ | Construtivista |
| Filosofia pedagógica | Instrutivista | ←————→ | Construtivista |
| Psicologia subjacente | Comportamental | ←————→ | Cognitiva |
| Objetividade | Precisamente focalizado | ←————→ | Não focalizado |
| Seqüenciamento instrucional | Reducionista | ←————→ | Construtivista |
| Validade Experimental | Abstrato | ←————→ | Concreto |
| Papel do instrutor | Provedor de materiais | ←————→ | Agente facilitador |
| Valorização do erro | Aprendizagem sem erro | ←————→ | Com a experiência |
| Motivação | Extrinseca | ←————→ | Intrinseca |
| Estruturação | Alta | ←————→ | Baixa |
| Acomodação das diferenças individuais | Não existente | ←————→ | Multifacetada |
| Controle do aluno | Não existente | ←————→ | Irrestrito |
| Atividade do usuário | Matemagênica | ←————→ | Generativa |
| Aprendizado cooperativo | Não suportado | ←————→ | Integral |

Fonte: FRESCKY, 2008, p.18

- **Epistemologia:** De objetivista a construtivista. Na epistemologia objetivista “o conhecimento é adquirido de forma objetiva por meio dos sentidos, com a aprendizagem consistindo em adquirir verdades” (FRESCKY, 2008, p.16). Na epistemologia construtivista “o conhecimento da realidade vai sendo construído individualmente, subjetivamente, com base em experiências anteriores, em reflexão” (FRESCKY, 2008, p.16).
- **Filosofia Pedagógica:** Vai da filosofia instrutivista a filosofia construtivista. A filosofia instrutivista enfatiza a importância da transmissão de informações. O usuário é visto como um ser passivo. Já na filosofia construtivista, o sujeito constrói seus conhecimentos nas interações com os *softwares*, de acordo com as possibilidades e interesses dele.
- **Psicologia Subjacente:** De comportamental a cognitiva. Quando se fala em comportamental significa que “os fatores do aprendizado são comportamentos que podem ser diretamente observados; a instrução consiste na modelagem do comportamento desejável obtido através de estímulo resposta” (FRESCKY, 2008, p.16).

[Cognitiva é a] psicologia que dá ênfase aos estados mentais internos ao invés do comportamento psicológico, reconhece que uma ampla variedade de estratégias de aprendizagem deve ser empregada, considerando o tipo de conhecimento a ser construído (FRESCKY, 2008, p.16).

- **Objetividade:** Vai do focalizado ao não focalizado. A objetividade focalizada, está relacionada a direcionar a atenção do usuário a algo específico, como por exemplo, os *softwares* tutoriais. A não focalização em um software, é quando permite que o usuário investigue, levante hipóteses, podendo testá-las, redefinindo suas ideias iniciais.
- **Sequenciamento Instrucional:** Do reducionista ao construtivista.

É reducionista quando o aprendizado sobre determinado conteúdo requer que todos os seus componentes sejam previamente entendidos. O seqüenciamento é construtivista quando o aluno é colocado em um contexto realístico, que irá requerer soluções de problemas e o apoio será oferecido conforme a necessidade individual de cada usuário. (FRESCKY, 2008, p.16-17).
- **Validade Experimental:** Do abstrato ao concreto. Quando são utilizadas situações que não fazem parte da realidade do aluno, é abstrato. É concreto quando contextualiza o conteúdo apresentado em situações reais.
- **Papel Instrutor:** Do provedor de materiais a agente facilitador. “No primeiro caso, o instrutor é considerado o detentor do conhecimento, enquanto no segundo, ele é visto como uma fonte de orientação e consulta” (FRESCKY, 2008, p.17).
- **Valorização do Erro:** De aprendizado sem erro a aprendizado com a experiência. O aprendizado sem erro ocorre quando o usuário é induzido a responder corretamente às perguntas e, enquanto não acertar a resposta, não passará para outro nível. O aprendizado com experiência prevê que os alunos aprendam com seus erros, uma vez que o software permite e aceita os erros do usuário.
- **Motivação:** De extrínseca, quando vem de fora do ambiente de aprendizado, a intrínseca, quando a motivação parte do próprio usuário.

- **Estruturação:** De alta a baixa. O programa é altamente estruturado quando segue uma sequência, ou seja, seus caminhos já foram determinados previamente. Possui baixa estruturação quando o aluno pode escolher a ordem que desejar para seguir no programa.
- **Acomodação de diferenças individuais:** De não existente a multifacetada. O primeiro considera que todos os indivíduos são iguais, não considera as diferenças dos usuários. A segunda considera as diferenças entre os sujeitos.
- **Controle do Aluno:** De não existente a irrestrito. O primeiro é quando todo o controle pertence ao *software*. O segundo significa que o aluno é quem “decide que sessões estudar, que caminhos seguir e qual material utilizar” (FRECKY, 2008, p.17).
- **Atividade do Usuário:** De matemagênico a generativo.

Matemagênico refere-se aos ambientes de aprendizagem onde se quer capacitar o aluno para acessar as várias representações do conteúdo. Generativo está significando que os estudantes estão envolvidos num processo de construção ou representação do conteúdo. (FRECKY, 2008, p.17).
- **Aprendizado Cooperativo:** De não suportado a integral. O primeiro é quando não permite trabalho em pares ou grupos, e o segundo quando permite o trabalho cooperativo, fazendo com que os objetivos sejam compartilhados.

Os dez critérios de interface (Figura 19) do *software* são:

Figura 19 - Modelo de Reeves de interface do *software*

| | | | |
|--|---------------------|--------|------------------------|
| Facilidade | Difícil | ←————→ | Fácil |
| Navegação | Difícil | ←————→ | Fácil |
| Carga cognitiva | Não gerenciável | ←————→ | Gerenciável/Intuitiva |
| Mapeamento | Nenhum | ←————→ | Poderoso |
| Design da tela | Princípios violados | ←————→ | Princípios respeitados |
| Compatibilidade espacial do conhecimento | Incompatível | ←————→ | Compatível |
| Apresentação da informação | Confusa | ←————→ | Clara |
| Integração das mídias | Não coordenada | ←————→ | Coordenada |
| Estética | Desagradável | ←————→ | Agradável |
| Funcionalidade geral | Não funcional | ←————→ | Altamente funcional |

Fonte: FRESKY, 2008, p.16

- **Facilidade de Uso:** De difícil a fácil. Se o *software* é de fácil entendimento.
- **Navegação:** De difícil a fácil. Refere-se ao ato de ir de um tópico a outro dentro do *software*.
- **Carga Cognitiva:** De não gerenciável/confusa a intuitiva/gerenciável. Estão relacionados com o esforço mental realizado pelo usuário, durante a execução das tarefas no *software*.
- **Mapeamento:** De nenhum a poderoso. Referente à habilidade do programa em rastrear os caminhos percorridos pelo aluno.
- **Design de Tela:** Vai de princípios violados a princípios respeitados. Referente à aparência, ao visual, à disposição dos elementos do *software*.
- **Compatibilidade Espacial do Conhecimento:** De incompatível a compatível. Verifica a compatibilidade do sistema com as expectativas e necessidades do usuário em sua tarefa.
- **Apresentação da Informação:** De confusa a clara. Referente ao modo como as informações estão apresentadas.
- **Integração das Mídias:** De não coordenada a coordenada. Refere-se a utilização de recursos (imagens, sons, animações e textos) que auxiliam o processo de ensino-aprendizagem por meio do *software*.

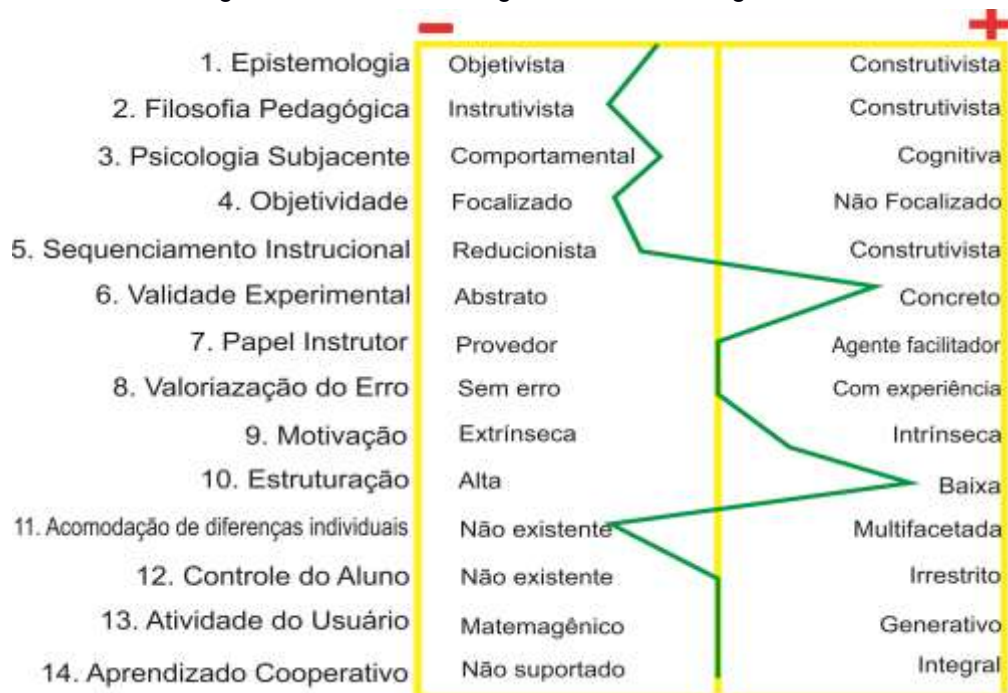
- **Estética:** De desagradável a agradável. Referente a estética do software, se visualmente é apresentável.
- **Funcionalidade Geral:** De não funcional a funcional. Diz respeito ao comportamento do software desde seu início até o final, ou seja, se é passível de execução.

O modelo de Reeves colhe informações de avaliadores sobre a qualidade e os aspectos pedagógicos do *software*, utilizando cada um dos critérios propostos e,

a partir dos valores atribuídos a cada critério, é construído um gráfico ligando os pontos demarcados, relacionando-os em uma escala linear. Esta escala baseia-se em extremos descritos como negativo (esquerda) e positivo (direita), indicando conceitos antagônicos, relativos a cada critério. A conclusão da avaliação se dá ao ligar, verticalmente, os pontos assinalados em cada escala e a análise da disposição do gráfico indicam os conceitos mais positivos ou mais negativos. (REZENDE, 2013, p.33)

A seguir um exemplo do modelo Reeves, com os pontos marcados e ligados.

Figura 20- Procedimento gráfico na metodologia Reeves



Fonte: ANDRES; CYBIS, 2000, p.6.

O fato deste modelo não usar uma escala numérica para quantificar resultados, apenas setas graficamente com marcações visuais, “torna a avaliação muito subjetiva e difícil para avaliadores inexperientes, que ficam inseguros sobre como representar as fragilidades dos *softwares* educacionais” (SILVA et. al., 2016, p.7). Para os profissionais da educação, que não tem um conhecimento acerca da avaliação de *softwares* educacionais, dificulta a chegar a uma conclusão sobre o software avaliado, ou seja, percebe-se que o resultado final pode ser inconclusivo para indicar se o *software* é recomendável ou não.

4.4 MODELO DE AVALIAÇÃO SEGUNDO CAMPOS (CAMPOS, 1994 *apud* FRESCKI, 2008)

Campos é professora do Departamento de Educação, Coordenadora Central de Educação a Distância- PUC-Rio. Suas pesquisas estão relacionadas a Educação, ambientes virtuais interativos, avaliação, Educação a Distância.

O modelo proposto por Campos (1994 *apud* FRESCKI, 2008) versa sobre um manual para avaliação da qualidade de um *software* educacional, objetivando fornecer diretrizes para desenvolvedores e usuários. Este modelo de avaliação é importante para qualquer tipo de *software* educacional, como os de exercício e prática, tutorial, simulação, investigação, jogos entre outros.

O modelo é composto por fatores, subfatores, critérios, processos de avaliação (Figura 21). “A avaliação é feita por meio de uma pesquisa de campo com os professores, onde é levada em conta a ordem de importância atribuída pelos professores aos critérios mais gerais” (FRESCKI, 2008, p.18). Percebe-se desse modo, que esta avaliação está mais próxima da visão do professor, onde ele faz o julgamento com relação ao *software*.

Figura 21- Avaliação de softwares segundo Campos (1994)



Fonte: (FRECKI, 2008, p.18)

Avaliar um *software* educacional significa analisar não só suas características de qualidade técnica, mas também, os aspectos educacionais envolvidos. Segundo Campos (1994 *apud* FRECKI, 2008, p. 18) na avaliação de um *software* educacional devem ser levados em consideração os seguintes aspectos:

- **Funcionalidade**

- **Manutenibilidade:** são as características técnicas de um software, referente à facilidade, precisão, além da capacidade de ser atualizado de maneira fácil.
- **Portabilidade:** Refere-se à característica das aplicações serem executáveis em diferentes plataformas.
- **Operacionalidade:** Relacionado a possibilidade do usuário operar e controlar o *software*.
- **Eficiência:** tempo de execução e os recursos envolvidos são compatíveis com o nível de desempenho do *software*.

- **Confiabilidade conceitual**

- **Integridade:** Referente a precisão das informações e ao grau de segurança e proteção delas.
- **Robustez:** É a capacidade do *software* funcionar mesmo em condições anormais.
- **Adequabilidade:** O conjunto de funcionalidades deve ser adequado às necessidades do usuário.
- **Adequação ao ambiente:** O conjunto de conteúdos deve ser adequado ao contexto no qual o aluno está inserido.

Os modelos de avaliação de *softwares* educacionais apresentados durante esta pesquisa são uma amostra da amplitude e complexidade da ação de avaliar a qualidade pedagógica e de usabilidade dos programas educacionais.

Pode-se perceber a importância de utilizar a norma ISO/IEC para avaliação de um *software* educacional, uma vez que apresentam os requisitos de qualidade técnica para a avaliação dos *softwares*, colaborando para uma elevação no nível de qualidade dos *softwares* educacionais utilizados pelos profissionais da educação.

Outra metodologia destaca foi a de Reeves, a qual tem como ponto positivo a utilização de critérios para a avaliação da usabilidade e critérios para a avaliação pedagógica em um único método, entretanto, uma das desvantagens é a utilização de setas com marcações para a avaliação do *software*, o que torna a avaliação subjetiva e de difícil interpretação dos resultados, dificultando o avaliador a indicar se o *software* é recomendável ou não.

Em relação ao modelo de avaliação segundo Campos, tem como vantagem a utilização de critérios para a avaliação da usabilidade e critérios para a avaliação pedagógica, em um único modelo, no entanto, deixa a desejar quanto ao número de critérios apresentados para a análise pedagógica do *software* educacional. Este modelo poderia fazer uso de outros critérios pedagógicos, além dos que foram apresentados, tais como: motivação, ferramentas, *feedback*, objetividade, entre outros.

Diferente dos outros modelos, a metodologia TUP aborda três categorias, a tecnologia, a usabilidade e a pedagogia, cada uma com critérios bem definidos, o que auxilia na seleção de um bom *software* educacional. Além disso, o modelo TUP por ser dividido em três categorias, facilita na hora de analisar os resultados obtidos, uma vez que o avaliador poderá observar o resultado por categorias e de modo geral.

De acordo com a base teórica, considero o modelo TUP o mais adequado para ser utilizados pelos professores que desejem analisar os *softwares* educacionais, que serão utilizados em sala de aula, visto que, o modelo é de fácil compreensão e interpretação, além de abordar um bom número de critérios essenciais em uma avaliação.

5 TÉCNICAS DE AVALIAÇÃO DE SOFTWARES EDUCACIONAIS

A avaliação da qualidade de um *software* educacional deve levar em conta as técnicas, segundo Rezende (2013) técnicas são as formas de avaliação de um *software* educacional, para se chegar a obtenção de um determinado resultado. Por meio das pesquisas, pode-se perceber que as principais técnicas são: técnicas objetivas (ou interpretativas); técnicas prospectivas; técnicas preditivas; Técnica de Mucchielli; TICESE (Técnica de Inspeção de Conformidade Ergonômica de *Software* Educacional).

5.1 TESTES DE USABILIDADE

Os testes de usabilidade é uma das estratégias de avaliação existentes para aferir a qualidade de uso dos softwares. O teste de usabilidade pode

ser aplicada desde testes de grandes tamanhos de amostra e de modelos complexos a estudos qualitativos informais, com apenas um único participante. Pode-se utilizar abordagens de teste diferentes de acordo com os objetivos, tempo e recursos necessários. (ABREU, 2010, p.24)

Os testes de usabilidade podem ser classificadas quanto ao seu objetivo. Para esta classificação, Cybis (2003 *apud* Ongaro; Canal, 2004, p. 169) dividiu-as em três grupos:

O primeiro grupo é nomeado **técnicas objetivas (ou interpretativas)**, que são aplicadas a simulações do uso do *software*, aos usuários, a fim de coletar e interpretar os dados. Como exemplo, temos o teste empírico tradicional, que consiste em observar a interação do usuário com o *software*, “normalmente ocorre em um laboratório, no qual as atividades são elaboradas para analisar a funcionalidade da aplicação” (PRATES; BARBOSA, 2003, p.4).

Outro exemplo é o Protocolo *Think-Aloud*. Durante a interação com o *software*, o participante realiza uma determinada atividade elaborada pelo avaliador, Na utilização do *software* para a realização da tarefa, o usuário deve ir expressando em voz alta “os seus pensamentos, sensações e opiniões enquanto interage com o produto” (OLIVEIRA, 2006);

O *Focus Groups* é mais um exemplo de teste de usabilidade. Sua característica principal é a discussão em grupo. Nesses debates, levantam-se as necessidades das pessoas, os desejos, os sentimentos, os valores, as ideias sobre os *softwares* avaliados. Com base nessas discussões, o avaliador reformular um produto, quando necessário.

O último exemplo desse grupo é o teste de comunicabilidade. Ele “deve ser executado em laboratório, para avaliar a *interface* com relação à qualidade da comunicação do *designer* para os usuários” (PRATES; BARBOSA, 2003, p. 5), o avaliador busca coletar as expressões dos usuários durante a sessão do teste.

O segundo grupo são as **técnicas prospectivas**. Por meio destas, faz-se uma sondagem das opiniões dos usuários, a partir de questionários ou entrevistas, com o objetivo de o usuário avaliar sua satisfação em relação ao *software*. Neste grupo são utilizados os questionários de Satisfação dos Usuários. Segundo Winckler (2001 *apud* Santana, 2013, p.23), esses instrumentos “são úteis na avaliação da interação entre o usuário e a aplicação, permitindo conhecer as experiências, opiniões e preferências dos usuários, coletando informações sobre a qualidade da interface”.

O último grupo é o das **técnicas preditivas**. Elas são realizadas pelos próprios especialistas de *interface* e tem como objetivo prever problemas que os usuários possam vir a ter. Um exemplo desta técnica é a avaliação heurística. Esta avaliação foi desenvolvida pelos pesquisadores Jakob Nielsen e Rolf Molich e “consiste em um exame de um produto ou sistema feito por um especialista em usabilidade ou em fatores humanos” (SANTANA, 2013, p. 23).

Outro exemplo é o *checklist*, que consiste em um conjunto de questões, que são aplicadas diretamente aos *softwares*. É uma técnica de fácil interpretação, pode ser aplicada pelos usuários. Como enfatiza Winckler (2001 *apud* SANTANA, 2013, p. 24), “nessa técnica, a qualidade do *checklist* é que faz a diferença no resultado da avaliação”.

Dentre os testes já mencionados para avaliar a usabilidade dos *softwares*, existe um atrativo na utilização do *checklist* como ferramenta para avaliação. O *checklist* é um conjunto de perguntas e/ou afirmações sobre determinada circunstância, podem ser utilizado para diversos fins. Essas perguntas “são

passíveis de descrição e análise estatística. Isto torna tal técnica uma ferramenta para sondagem de aplicação rápida” (NIELSEN, 1994 *apud* ABREU, 2010, p. 27).

A utilização dessa ferramenta não exige maiores conhecimentos técnicos, apenas será necessário uma noção acerca do conteúdo abordado. A ferramenta *checklist* tem como objetivos, descobrir erros de função, de lógica ou de implementação para qualquer representação do *software*, verificar se o *software* em questão atende aos requisitos especificados e aos objetivos propostos;

Segundo Pressman (1995 *apud* ALVES, et. al., 2004, p. 7), as principais vantagens do formato de avaliação do *checklist* são a sua fácil organização e o custo baixo. As desvantagens são que os *checklists* “não levam em conta a integração de relações entre capacidade de utilização e aprendizagem”, ou seja, assuntos ligados à educação não são considerados.

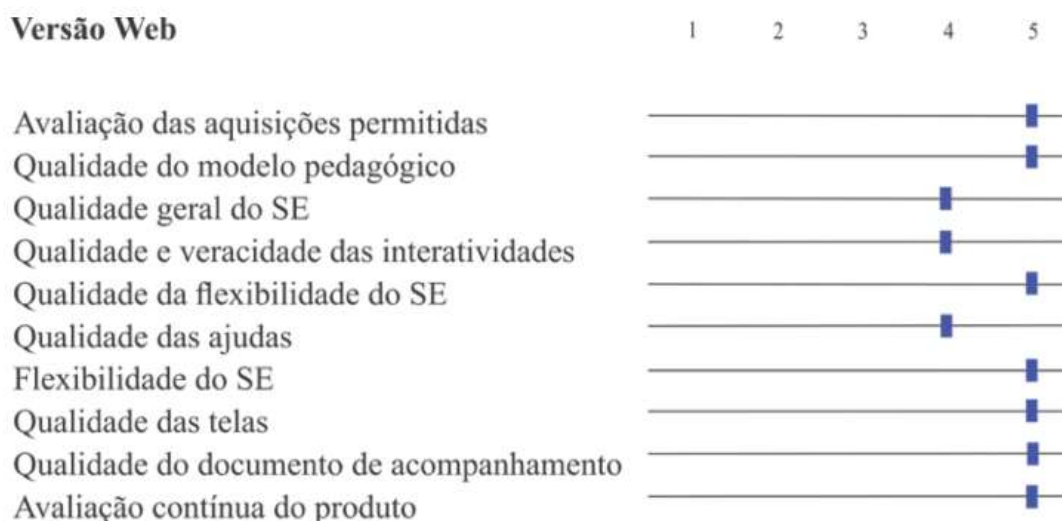
5.2 TÉCNICAS PEDAGÓGICAS

A avaliação da qualidade de um *software* educacional deve levar em conta, principalmente, as características relacionadas à qualidade didático-pedagógica. Nesse aspecto, os objetivos dos estudiosos da ergonomia de *software* e dos educadores convergem para um mesmo ponto, ou seja, “garantir a adaptação do trabalho ao homem e aos meios didáticos a fim de obter a satisfação e produtividade dos alunos no processo de ensino-aprendizagem” (SILVA, 1998 *apud* ANDRES, 1999, p. 8).

A **Técnica de Mucchielli** (MUCCHIELLI, 1987 *apud* ANDRES; CYBIS, 2000, p. 7) consiste em avaliar o potencial do *software* educacional com o público para o qual ele foi desenvolvido. Por meio dela, é possível perceber os aspectos técnicos e principalmente os pedagógicos de um *software* educacional.

Mucchielli (1987 *apud* ANDRES; CYBIS, 2000, p. 7) propôs dez perspectivas gerais a serem avaliadas a partir de um *checklist*, cujos itens estão relacionados com uma escala numérica com pontuação variando entre 1 (um) e 5 (cinco). O resultado de sua aplicação resulta em um conjunto de barras para cada qual um critério foi associado a um valor da escala adotada, contendo a marcação dos valores atribuídos pelo avaliador (Figura 22).

Figura 22- Procedimento gráfico da Técnica de Mucchielli



Fonte: SILVA et. al., 2016 p.9.

Os critérios são:

- Avaliação das aquisições permitidas, concernentes aos elementos de conhecimento retido ou a medida das *performances* evolutivas, resultado dos testes de avaliações;
- Qualidade do modelo pedagógico adotado: Deve ser levado em conta a clareza dos objetivos pedagógicos; a coerência da organização pedagógica; a eficácia provável do modelo pedagógico em relação aos objetivos; a originalidade do modelo pedagógico adotado;
- Qualidade da ideia geral do *software*: Segundo Andres (1999, p.13), será “apreciada a existência da ideia diretriz, sua originalidade e matriz de adaptação no domínio e ainda realizar testes com o público-alvo verificando o interesse global manifestado pelo princípio geral do *software*”.
- Qualidade e variedades dos procedimentos de interatividade utilizadas:
 - avalia-se o questionário, as páginas, os procedimentos de designação, a adaptação destes procedimentos ao conteúdo e às fases pedagógicas; a flexibilidade ou rigidez dos procedimentos, a pertinência das questões postas, a clareza e a não-ambiguidade destas questões (ANDRES, 1999, p.13).
- Qualidade da flexibilidade do *software*: um *software* é flexível quando oferece caminhos e propõe escolhas. Avalia-se, também, a possibilidade de corrigir a resposta dada, a potência dos modelos de análise de respostas.

- Natureza e qualidade das ajudas: deve oferecer ajuda no problema que lhe é posto ao aluno. Ajudar em relação às dificuldades encontradas, número de erros cometidos, sua originalidade e sua variedade.
- Grau de flexibilidade do *software*: deve-se avaliar a facilidade com a qual o usuário pode parar a atividade no *software* e retomar o ensino de onde se parou e, a maneira de poder adaptar os conteúdos com as necessidades específicas dos usuários.
- Qualidade das telas: o *software* educacional possui imagens, sons, animações, entre outros efeitos, que auxiliam na construção do conhecimento, desse modo, este aspecto visa analisar a estética geral do *software*, de modo que, o conjunto de elementos que o constitui esteja em harmonia com o conteúdo apresentado.
- Qualidade do documento de acompanhamento: refere-se ao suporte oferecido pelo *software*, se ele contém informações sobre a indicação do público-alvo, informação sobre atualização e principalmente se apresenta sugestões de uso do *software* educacional em sala de aula.
- Avaliação contínua do produto: um *software* educacional deve-se atualizar continuamente, buscando melhorias, evoluções, a fim de corrigir falhas, entretanto “o usuário deve ser informado dessa capacidade” (ABREU, 1999, p.14).

Esta técnica não separa por grupos os critérios em pedagógicos e de usabilidade, o que exige do avaliador mais atenção ao analisar os resultados, antes de afirmar que os problemas encontrados no *softwares* educacionais são prioritariamente de interface ou pedagógicos. Além disso, “é interessante observar que 7 dos 10 critérios utilizam o termo qualidade, que por si só é subjetivo” (SILVA et. al., 2016 p. 9).

A segunda técnica é a **TICESE** (Técnica de Inspeção de Conformidade Ergonômica de *Software* Educacional). Ela é uma técnica que foi desenvolvida por Gamez (1998) e propõe a integração entre aspectos pedagógicos e de usabilidade no processo de avaliação de um *software* educacional.

“A técnica é formada por um conjunto específico de critérios de análise e tem seu suporte teórico nas ciências cognitivas, ergonomia de *software*, psicologia da aprendizagem e pedagogia” (ANDRES, 1999, p.15). Aos critérios adotados está

associado um conjunto de questões que visa orientar o avaliador na difícil tarefa de inspecionar a avaliação das qualidades ergonômicas e pedagógicas do *software* educacional.

Segundo Andres (1999, p.15), os objetivos da TICESE são:

- Fornecer diretrizes para avaliadores na tarefa de avaliar qualidade em *software* educacional;
- Orientar na realização de inspeção de conformidade ergonômica do *software*;
- Tem um enfoque particular sobre a ergonomia de *software* aplicada produtos educacionais informatizados;
- São considerados tanto os aspectos pedagógicos como os aspectos referentes à interface e usabilidade.

Antes de iniciar o processo de avaliação, o avaliador precisa ter um primeiro contato com o *software* para melhor compreender o seu funcionamento e deve ter conhecimento da técnica. Assim, o avaliador poderá dar início à avaliação do *software* educacional. Sugerem-se os seguintes passos:

1. Resposta às questões: implica em ler cada questão do *checklist* e verificar a existência do atributo no *software*.

2. Atribuição de pesos às questões: significa classificá-las segundo uma ordem de importância, como por exemplo:

- À questão *Não se Aplica*, atribuir o peso 0 (zero);
- À questão *Muito importante*, atribuir o peso 1.5 (um e meio);
- À questão *Importante*, atribuir o peso 1 (um).

3. Tratamento quantitativo dos dados: esta atividade implica nas seguintes sub atividades:

- Atribuição de valores às questões:
 - Para as questões com resposta sim, atribuir o valor 1 (um);
 - Para as questões com resposta parcialmente, atribuir o valor 0.5 (meio);
 - Para questões com resposta não, atribuir o valor 0 (zero);
- Cálculo da média de cada sub-critério ou critério isolado

Segundo Andres (1999, p.16), para encontrar o valor médio de cada sub-critério ou critério isolado, aplicar a seguinte equação:

$$X(j) = \frac{a(i) \cdot p(i)}{q(j)} \times 100$$

| |
|---|
| <p>Onde</p> <p>$(i=1)$</p> <p>j= critério</p> <p>q = número de questões</p> <p>a(i) = valor da questão</p> <p>p(i) = (peso da questão)</p> |
|---|

O resultado indicará o percentual de conformidade ergonômica do *software* ao critério em avaliação. Entretanto, o avaliador deverá aplicar esta equação a todos os critérios e subcritérios e encontrar o percentual de conformidade.

- Cálculo da média final dos critérios que estão subdivididos em subcritérios (opcional).

Quando um critério possui sub-critérios, caso o avaliador queira obter um resultado isolado para o critério, sugere-se a aplicação da seguinte equação:

$$Y = \frac{X(mr) + X(i)}{2}$$

Onde:

- X(mr) é resultado da média do menor sub-critério
- X(i) é o resultado da média dos sub-critérios

Uma vez encontrados os valores para cada critério e sub-critério, o avaliador poderá construir gráficos e comparar os resultados obtidos (ANDRES, 1999, p.16).

A análise dos gráficos permite identificar quais pontos devem ser melhorados no *software*.

Três módulos compõem a técnica: o primeiro módulo de classificação, tem como objetivo determinar o tipo de *software* educacional (tutorial, exercício e prática, simulador, jogos educativos, entre outros), o nível de aprendizagem (sequencial, relacional e criativo), quanto à atividade do sujeito de aprendizagem (*software* com enfoque do tipo algoritmo e *software* com enfoque do tipo heurístico) e, por fim, a identificação do paradigma educacional (Comportamentalista e Construtivista).

O segundo módulo é o de avaliação e têm como objetivo avaliar a conformidade do *software* educacional aos padrões ergonômicos de qualidade, analisando a capacidade do *software* em auxiliar o aprendizado específico. O último módulo é o de contextualização e complementa o critério anterior, tendo como

objetivo avaliar se o *software* é adequado a um determinado contexto escolar, uma vez que, “cada instituição de ensino possui características e contextos próprios, que se diferenciam das demais” (ANDRES, CYBIS, 2000, p.10).

O detalhamento de critérios em subcritérios guia o avaliador e minimiza o nível de abstração envolvido na aplicação do método. No entanto, esta técnica

exige um maior tempo de dedicação por parte do avaliador, é necessário o uso de planilhas auxiliares para guiar o acompanhamento de cada critério, e para a execução dos cálculos exigidos pela técnica (SILVA et. al., 2016 p. 9).

A utilização da planilha para a realização dos cálculos e para a organização dos critérios facilita a leitura e a visualização do resultado da avaliação para o avaliador.

Para avaliação do *software* educacional, exige um conhecimento sobre as técnicas existentes. As principais técnicas apresentadas na biografia nacional foram apresentadas durante a escrita deste trabalho. De acordo com a base teórica, considero a técnica *checklist* a mais recomendada para avaliação da usabilidade dos *softwares* educacionais, uma vez que, seu formato pode facilitar a coleta e o entendimento da avaliação, podendo ser aplicada pelo próprio usuário do programa.

E para a avaliação pedagógica do *software* educacional, recomendo a utilização da técnica de Mucchielli, apesar dela não separa os critérios em categorias pedagógicas e de usabilidade, é composta por dez critérios essenciais em um *software* educacional, cada critério pode variar em uma escala de 1 (um) a 5 (cinco), o que a torna a análise dos resultados mais fácil.

6 INSTRUMENTOS DE AVALIAÇÃO DE *SOFTWARES* EDUCACIONAIS

Na literatura existem várias fichas já elaboradas para a avaliação de *softwares* educacionais. A seguir, relacionamos algumas mais referendadas nas pesquisas consultadas neste estudo.

6.1 QUESTIONÁRIO DE NOKELAINEN

Nokelainen (2006 apud *apud* ABREU, 2010, p. 36) é Ph.D., Professor (Pedagogia de Engenharia), umas das suas áreas de atuação é a Tecnologia Educacional (usabilidade pedagógica, ambientes de aprendizagem). Ele desenvolveu um estudo baseado no conceito de usabilidade pedagógica que,

está associado aos critérios e recomendações de usabilidade técnica com a finalidade de satisfazer as necessidades dos estudantes e professores para a realização de tarefas propostas através de componentes da interface do material de aprendizagem. (MATOS, 2013, p. 94).

A contribuição de Nokelainen para a avaliação de *softwares* educacionais é o estabelecimento de critérios pedagógicos. Nokelainen (2006 apud ABREU, 2010, p. 36) propôs um questionário para avaliação da usabilidade pedagógica composto de 56 questões (Anexo A). Este instrumento apresenta os seguintes critérios pedagógicos:

- Controle do aluno: O conteúdo presente no *software* deve ser dividido em unidades significativas para o aluno, considera-se que, assim, o aluno adquire controle e liberdade sobre o conteúdo.
- Atividade do aprendiz: As atividades propostas devem ser baseadas em aprendizagem por meio de solução de problemas e atividades colaborativas.
- Aprendizagem colaborativa/cooperativa: Permitir realizar atividades em conjunto com outros usuários com metas de aprendizagem comuns.
- Orientação de metas: Os objetivos de aprendizagem devem ser claros para o usuário, além disso, deve permitir aos alunos interferir na definição de objetivos, buscando seus interesses.
- Aplicabilidade: Os conteúdos devem corresponder às necessidades dos alunos e serem transferíveis para outros contextos.

- Valor agregado: Adaptabilidade para necessidades individuais, controle do aprendiz, interesse do conteúdo, comunicação, participação ativa dos alunos.
- Motivação: Incentivos, autonomia, expectativas, atribuições de sucesso ou fracasso e alcance de objetivos são conceitos que devem estar presentes para gerar motivação.
- Avaliação de conhecimento anterior: O *software* deve adaptar-se às diferenças de conhecimento e habilidades entre os alunos.
- *Feedback*: O *software* deve fornecer *feedbacks* imediatos e encorajadores.

Esse instrumento tem uma excelente abordagem das necessidades pedagógicas, uma vez que as funções avaliadas facilitam a construção do conhecimento. Entretanto, o questionário proposto por Nokelainen (2006), não aborda os aspectos inerentes à usabilidade técnica e pedagógica de forma separada, o que dificulta o avaliador a identificar aonde o *software* deixa a desejar. Além disso, as maiorias das questões estão relacionadas à avaliação pedagógica, poucas são as questões que abordam a funcionalidade de um *software*.

6.2 QUESTIONÁRIO DE SSEMUGABI

Foi elaborado por Ssemugabi (2006 *apud* ABREU, 2010, p. 36) com o objetivo de medir a usabilidade técnica e pedagógica. É composto por 20 heurísticas. São 10 heurísticas para a usabilidade técnica (56 questões) e 10 heurísticas para a usabilidade pedagógica (53 questões), divididas em 03 categorias (Anexo B). Cada heurística possui uma certa quantidade de questões, que devem ser respondidas por meio de um (X), em uma das alternativas: () Concordo plenamente () Concordo () Talvez () Discordo () Discordo plenamente.

As três categorias são:

Categoria 1: Critério geral de usabilidade de *interface*, com as 10 heurísticas:

1. Visibilidade do *status* do sistema: O *software* precisa manter o usuário atualizado do que está acontecendo, ou seja, todas as ações precisam de *feedback* instantâneo para orientá-lo.

2. Correlação entre o sistema e o mundo real: O *software* precisa ser coerente ao contexto do usuário, utilizando, por exemplo, uma linguagem que faça sentido para o indivíduo.
3. Controle do usuário e liberdade: O *software* deve apresentar opções de desfazer ou refazer a ação, ou retornar ao ponto anterior, quando o usuário estiver perdido ou em situações inesperadas.
4. Consistência e aderência às normas: O *software* deve fazer uso da mesma língua o tempo todo, e nunca identificar uma mesma ação com ícones ou palavras diferentes.
5. Prevenção de erros: Os *softwares* devem ser programados de maneira que possa prevenir erros.
6. Reconhecimento ao invés de evocação: A interface dos *softwares* deve oferecer ajuda, por meio de informações capazes de orientar as ações do usuário.
7. Flexibilidade e eficiência do uso: Os *softwares* precisam ser fáceis para usuários leigos, mas flexíveis o bastante para se tornar ágeis para usuários avançados.
8. *Design* estético e minimalista: Deve evitar que os textos e o *design* falem mais do que o usuário necessita saber.
9. Reconhecimento, diagnóstico e recuperação de erros: As mensagens de erro do *software* devem possuir uma redação simples e clara, identificando uma saída construtiva ou uma possível solução.
10. Ajuda e documentação: O *software* deve possuir documentos e ajuda, para orientação do usuário, em caso de dúvida. A documentação deve ser visível, facilmente acessada e oferecer uma ferramenta de busca na ajuda.

Categoria 2: Critério específico para aplicações web educacionais, com 02 heurísticas.

1. Simplicidade de navegação, organização e estrutura: O *software* precisa ser fácil de aprender e de usar, sua organização e sequência devem ser elaboradas de modo a proporcionar satisfação de uso.

2. Relevância do conteúdo do *site* para a aprendizagem: Deve se manter sempre atualizado e com conteúdo motivador, relevante, apropriado e claro aos alunos.

Categoria 3: Critério específico sobre *design* com foco em aprendizado, focando em aprendizado efetivo, com 08 heurísticas.

1. Clareza das metas, objetivos e saídas: As metas e os objetivos devem ser claros para o usuário, possibilitando que o mesmo encontre saídas para resolver problemas, superar imprevistos ou até mesmo sair do *software*.
2. Aprendizagem colaborativa: Permite o trabalho cooperativo, fazendo com que os objetivos sejam compartilhados.
3. Aplicabilidade do nível de controle do aluno: Indica o nível de controle que o usuário tem. O *software* permite que o aluno decida sobre que sessões estudarem, que caminhos seguir e qual material utilizar?
4. Suporte para abordagens significativas de aprendizagem: O *software* deve possibilitar ferramentas para que ocorra uma aprendizagem significativa para o aluno.
5. Identificação de erros cognitivos, diagnóstico e restabelecimento: O *software* permite que os alunos aprendam com seus erros? Ou não aceita os erros do usuário?
6. *Feedback*, orientação e avaliação: É parte fundamental em um *software*. Por meio do *feedback*, o usuário saberá seus erros e acertos e quais caminhos seguir .
7. Contexto significativo: Os conteúdos presentes nos *softwares* devem ser adequados ao contexto no qual o aluno está inserido.
8. Motivação, criatividade e aprendizagem ativa: O *software* deve despertar o interesse dos usuários para que ele possa cumprir os objetivos da aprendizagem.

A abordagem do questionário de Ssemugabi é muito boa, uma vez que são separadas por categorias, o que facilita na hora da avaliação, pois o avaliador terá uma visão dos resultados de modo geral e separados por categorias. Entretanto, por utilizar as 10 heurísticas para a medição da usabilidade técnica, ou seja, 56

questões só para a medição da usabilidade geral, torna a avaliação da usabilidade longa, demandando assim, tempo para ser respondida.

6.3 QUESTIONÁRIO DE REITZ

Este questionário (Anexo C) foi desenvolvido por Doris Simone Reitz (2009) em sua tese apresentada ao Programa de Pós Graduação em Informática na Educação, do Centro Interdisciplinar de Novas Tecnologias na Educação, da Universidade Federal do Rio Grande do Sul. As suas pesquisas possuem ênfase nos seguintes temas: usabilidade, usabilidade técnica e pedagógica, ergonomia cognitiva, interação humano computador (IHC), *websites*, *e-learning* e usabilidade de interfaces educacionais.

O instrumento apresentado foi construído com base no “mapeamento dos critérios e heurísticas de usabilidade técnica e pedagógica dos autores Nokelainen (2006) e Ssemugabi (2006)” (REITZ, 2009, p.73). Reitz realizou uma síntese e adaptação dos questionários de usabilidade técnica e pedagógica dos autores citados anteriormente, algumas questões mostraram-se repetitivas quando da união dos instrumentos, portanto, essas foram excluídas. O que resultou em 72 questões de usabilidade técnica e pedagógica, separadas em três categorias (Figura 23), para a análise de um *software* educacional.

Figura 23 - Critérios e respectivas questões de usabilidade técnica e pedagógica sistematizadas e adaptadas por Reitz.

| 1. Usabilidade geral da interface | |
|---|------------------------|
| Visibilidade do status do sistema | 13 |
| Modelo do projetista e modelo do aluno, comparação entre o sistema e o mundo real | 15, 16, 17, 18, 22 |
| Controle do usuário e liberdade | 23, 24 |
| Consistência e aderência às normas | 30, 31, 32, 33, 34, 35 |
| Prevenção de erros | 36, 37, 38 |
| Flexibilidade e eficiência de uso | 39, 40 |
| Design estético e minimalista | 41 |
| Reconhecimento, diagnóstico e recuperação de erros | 42, 43, 44, 45, 67 |
| Ajuda e documentação | 3, 46, 47 |
| 2. Usabilidade técnica e pedagógica específicas para <i>websites</i> | |
| Simplicidade de navegação, organização e estrutura | 48, 49, 50, 51, 52 |
| Relevância do conteúdo para a aprendizagem | 53, 54, 55 |
| 3. Usabilidade pedagógica | |
| Atividade do aluno | 1, 2, 4, 5 |
| Controle do aluno | 25, 26 |
| Aprendizagem colaborativa e cooperativa | 58, 59, 60, 61, 62, 63 |
| Orientação a objetivos | 6 |
| Aplicabilidade | 27, 29 |
| Valor agregado | 8, 9, 19, 20, 21 |
| Avaliação do conhecimento prévio | 7, 10, 11, 64 |
| <i>Feedback</i> | 12, 14 |
| Clareza das metas, objetivos e saídas | 56, 57 |
| <i>Feedback</i> , orientação e avaliação | 28 |
| Contexto significativo ao domínio do aluno | 65, 66, 68 |
| Motivação, criatividade e aprendizagem ativa | 69, 70, 71, 72 |

Fonte: REITZ, 2009, p.73

O instrumento de Reitz classifica as questões em concordância com grau de severidade, ou seja, de acordo com sua importância para a avaliação de um *software* educacional. A classificação do grau de severidade, conforme Nielsen (2006 *apud* REITZ, 2009, p. 84) são as seguintes:

- Severidade Alta: os problemas impõem um custo e/ou perda inaceitável, por impedir o uso do *site* pelos usuários ou por efetivamente conduzi-los a sair do *site*;
- Severidade Média: causam confusão e frustração aos usuários. Há perda da qualidade de interação;
- Severidade Baixa: os problemas são estéticos ou irritantes, mas os usuários conseguem contornar o problema.

Para definir o grau de severidade das questões, Reitz contou com a colaboração do professor Roberto Cabral de Mello Borges e a professora Cíntia Inês Boll, ambos doutorandos no Programa de Pós-Graduação em informática na educação (PPGIE). “Pode-se verificar que em relação ao número total de questões, 55,55% (40 questões) apresentam severidade média; 25% (18 questões)

apresentam severidade baixa e 19,45% (14 questões) apresentam severidade alta” (REITZ, 2009, p.85).

Para responder o questionário utiliza a escala de Likert com pontuação de 1 a 5, que corresponde a: Discordo totalmente (1 ponto), Discordo parcialmente (2 pontos), Talvez (3 pontos), Concordo parcialmente (2 pontos) e concordo totalmente (5 pontos).

Uma das vantagens do questionário de Reitz é que faz uma comparação dos instrumentos de Nokelainen e Ssemugabi, resultando nas questões mais relevantes abordadas nos dois questionários. Outro ponto positivo é, por atribuir valores às questões, ao final do questionário, o usuário pode analisar os resultados pelo grau de severidade, separadamente, ou de forma geral.

As desvantagens da utilização do questionário de Reitz é a grande quantidade de questões, o que torna a avaliação cansativa, o avaliado precisará voltar ao software, para ver se atende ou não a questão abordada, além da demanda de tempo necessária para utilização do modelo.

6.4 CHECKLIST DO PROINFO

O Programa Nacional de Tecnologia Educacional (PROINFO) é um projeto governamental federal, criado pelo Ministério da Educação (MEC), através da Portaria nº 522 em 09/04/1997, que tem como objetivo a inserção das Novas Tecnologias de Informação e Comunicação (NTICs) nas escolas públicas.

“Realizado no âmbito do Plano Nacional de Desenvolvimento da Educação – PDE, o PROINFO busca promover o letramento digital de estudantes e contribuir para avanços nos processos de ensino e de aprendizagem com a ajuda das TIC (Tecnologia de Informação e Comunicação)”. (CARVALHO; MONTEIRO, 2013, p.13). Desse modo, criou o projeto um computador por aluno (UCA), programa um computador por aluno (PROUCA), programa banda larga nas escolas (PBLE), além do uso de *tablets* no ensino público, além de oferecer diversas formações para os profissionais da educação.

Em relação a análise de um *software* educacional, o PROINFO parte do pressuposto que

a tarefa de avaliação de *software* educativo é complexa, uma vez que põe em julgamento vários aspectos inter-relacionados e que envolvem uma articulação de conteúdos de Informática, Computação, Comunicação, Psicologia, Pedagogia etc. (PROINFO, 1987 *apud* BRITO; ALMEIDA; CAVALCANTI, 2003, p. 335).

Neste sentido, o III Encontro Nacional do PROINFO (1998) teve como objetivo discutir os critérios para o desenvolvimento, uso e seleção de *softwares* educacionais. O PROINFO propôs um *checklist* composto por 20 questões (ANEXO D), abordando a:

- Interatividade: Diz respeito a comunicação entre o software educacional e o usuário, seja por meio de mensagens, *feedback*, entre outros recursos.
- Flexibilidade na resolução dos problemas: o *software* educacional deve permitir múltiplas formas para a resolução de problemas.
- Criatividade: Em relação se o *software* permite o usuário construir seu próprio conhecimento.
- Usabilidade: Refere-se à facilidade do uso do *software* educacional.
- Ética: Em geral diz respeito à segurança, confidencialidade e autenticidade do *software*.
- Tratamento do erro: Como o software reage com os erros dos usuários? Não aceita erros? Permite que o usuário corrija os erros cometidos?
- *Feedback*: Em relação ao software fazer uma avaliação das ações do usuário.

O *checklist* do PROINFO é uma sequência de questões abertas, capaz de identificar situações que acarretem problemas na utilização dos *softwares* educacionais. O instrumento é de fácil aplicação. Mas, por outro lado, tem como desvantagens o pequeno número de questões, para avaliar os princípios de usabilidade técnica e pedagógica de um *software* educacional. Além disso, o *checklist* por ser construído com questões abertas, não possui uma escala de classificação, o que dificulta a visualização do resultado.

6.5 AVALIAÇÃO DE OLIVEIRA

O instrumento proposto por Noé de Oliveira (2001) tem como objetivo,

auxiliar projetistas e educadores, quando da concepção, avaliação e aplicação de software educacional. Nele busca-se a integração de abordagens pedagógicas, que permitam analisar o produto que se

pretende construir ou aplicar, dentro do processo ensino aprendizagem (OLIVEIRA, 2001, p. 83).

O questionário foi concebido a partir da dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção, da Universidade Federal de Santa Catarina. Constitui objetivo deste trabalho

o propósito de chegar-se a uma proposta de avaliação de software educacional em forma de listas de avaliações (checklist) que permita a orientação teórico-metodológica na avaliação e utilização destes softwares (OLIVEIRA, 2001, p. 05).

A proposta é de um *checklist* (ANEXO E) divididos em critérios essenciais para a análise de um *software* educacional, que são: Apresentação e funcionalidade; Confiabilidade Usabilidade; Eficiência; Manutenibilidade; Portabilidade; Conteúdo dos softwares, relacionado com a disciplina.

Para cada critério são apresentadas questões, que devem ser respondidas com valores iguais a: 0, 1, ou 2, respectivamente, não, às vezes e, sim. Ao final de cada critério, são somadas as pontuações correspondentes. E ao final do *checklist*, pede-se para somar os resultados obtidos em cada critério, resultando na classificação final do *software* educacional (Figura 24), em insatisfatório, regular, bom e excelente.

Figura 24 - Avaliação final do instrumento de Oliveira

| AVALIAÇÃO FINAL | |
|------------------------|----------------------|
| Pontuação | Classificação |
| De 00 a 75 | Insatisfatório |
| De 75 a 115 | Regular |
| De 115 a 150 | Bom |
| De 150 a 187 | Excelente |

Fonte: OLIVEIRA, 2001, p.92

Os pontos positivos na utilização deste *checklist* são que, permitem registrar características do avaliador, informações contidas no *software* e informações pedagógicas indicadas pelo fornecedor. Além disso, por atribuir valores a cada

critério avaliado, facilita a obtenção do resultado mais preciso em relação à qualidade técnica e pedagógica dos *softwares* avaliados.

Os principais instrumentos de avaliação de softwares educacionais identificados nas literaturas científica nacionais foram apresentados neste capítulo, por meio deles pode-se perceber que avaliar um software educacional é uma tarefa complexa, uma vez que devem ser analisados os aspectos técnicos e pedagógicos.

De acordo com a análise teórica, considero que os profissionais da educação, quando forem avaliarem um *software* educacional, utilizem o instrumento de *checklist* proposto por Oliveira, composto por 27 questões de usabilidade técnica e 24 questões sobre o aspecto pedagógico, nas quais abordam as características essenciais que um *software* educacional deve conter.

Através da análise dos instrumentos apresentados, destaco o questionário de Oliveira como sendo o de mais fácil aplicação, não há necessidade de conhecimentos específicos sobre determinado conteúdo. O instrumento é separado por características essenciais em um *software* educacional, gerando um resultado ao final de cada uma, e ao final do *checklist*, se propõe somar os valores obtidos, para a classificação final do *software* educacional em insatisfatório, regular, bom e excelente. Contribuindo para a seleção de softwares educacionais qualificados.

7 CONCLUSÃO

Com este trabalho, concluímos que a utilização de *softwares* educacionais nas escolas é crescente. Eles são utilizados como recursos didáticos que apoiam as atividades realizadas nas aulas, visto que ajudam os alunos na compreensão dos conteúdos, de forma lúdica. Mas, para que o *software* educacional atinja os objetivos de ensino e aprendizagem propostos, ele deve passar por uma avaliação da sua usabilidade técnica e pedagógica.

Desse modo, o objetivo geral deste trabalho foi sistematizar modelos e instrumento de análise de *softwares* educacionais, identificados em publicações científicas nacionais. Para isso, fez-se necessário atender aos objetivos específicos de: Conhecer os tipos de *softwares* educacionais, apresentados na literatura científica nacional; Identificar diferentes modelos de análise de *softwares* educativos adotados nas pesquisas nacionais, no campo de educação e tecnologias; Apresentar instrumento de análise de *softwares*, elaborados e aplicados por pesquisadores nacionais.

De acordo com os resultados obtidos, pode-se afirmar que, os principais tipos de *softwares* educacionais apresentados na literatura, são: de exercício e prática ou exercitação, simulação, programação, aplicativos, jogos educativos, tutoriais, *softwares* de autoria e de investigação. Estes *softwares* educacionais despertam o interesse, motivam os usuários, estimulam a criatividade, enfim contribuem para o processo de ensino e aprendizagem.

Este estudo identificou diferentes modelos de análise de *softwares* educacionais adotados nas pesquisas nacionais, no campo de educação e tecnologias. Os mais destacados na literatura foram: a norma ISO de produto de *softwares*, o modelo TUP, o modelo de Reeves e o modelo de avaliação segundo Campos. Cada modelo apresenta critérios específicos para a análise de um *software* educacional.

Entretanto, por meio da análise teórica dos modelos, cheguei a conclusão de o modelo Tup é o de mais fácil compreensão e interpretação para os professores em formação ou em atuação. Por meio dele, é possível escolher um *software* apropriado para o contexto educacional, uma vez que, aborda um grande de número de critérios pedagógicos essenciais em uma análise de um *software* educacional.

Por fim, apresentamos instrumentos de avaliação dos aspectos da usabilidade e dos aspectos pedagógicos de um *software* educacional, elaborados e aplicados por pesquisadores nacionais. São eles: Questionário de Nokelainen, Questionário de Ssemugabi, Questionário de Reitz, *Checklist* do PROINFO, Avaliação de Oliveira. Esses instrumentos trazem em si contribuições à reflexão sobre o tema abordado e colaboram com uma avaliação técnica e pedagógica de *softwares* educacionais, identificando as qualidades, os problemas e as limitações, segundo critérios estabelecidos nos instrumentos apresentados.

Entretanto, através da análise dos instrumentos apresentados, destaco o questionário de Oliveira como sendo o de mais fácil aplicação, porque não há necessidade de domínio de conhecimentos técnicos específicos da computação, contribuindo para que os professores possam escolher *softwares* adequados para uso em sala de aula.

Como fruto desta pesquisa, espera-se oferecer contribuições para a reflexão teórica e para a prática educativa no campo da Pedagogia, apresentando para os professores em formação e em exercício profissional informações sobre métodos e instrumentos de avaliação que possam auxiliá-los nas escolhas dos *softwares* educacionais que adotarão em suas práticas educativas. Por consequência, pretende-se colaborar para a elevação do nível de qualidade dos *softwares* educacionais utilizados nas escolas.

Considerando-se a temática abordada de extrema importância, julga-se relevante a continuidade deste trabalho para que as pesquisas futuras possam ser ainda mais especializadas e que permitam acompanhar o crescente desenvolvimento na área da avaliação de *softwares* educacionais.

Para aprofundar o tema, poder-se-ia propor uma terceira categorização ou classificação para os *softwares* educacionais em função de critérios pedagógicos específicos. Assim, seria possível realizar uma análise mais crítica e especializada, do ponto de vista pedagógico, de qualquer um dos modelos elencados neste estudo, ou ainda, sugerir alterações para os instrumentos de avaliação de *software* educacional descritos na pesquisa. Por fim, os resultados obtidos podem ainda ser empregados por outro pesquisador para elaborar orientações específicas para a avaliação de *softwares* educacionais para que os professores possam aplicar, a

exemplo de um, guia ou manual para o professor avaliar um *software* educacional a partir de qualquer um dos modelos, técnicas ou instrumentos aqui estudados.

REFERÊNCIAS

- ABREU, Célia. **Avaliação de Usabilidade em Softwares Educativos**. 2010. 109 f. Dissertação (Mestrado) - Curso de Mestrado Integrado Profissional em Computação Aplicada, Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Fortaleza, 2010. Disponível em: <http://www.uece.br/mpcomp/index.php/arquivos/doc_download/231-dissertacao-72-avaliacao-de-usabilidade-em-softwares-educativos>. Acesso em: 19 nov. 2016.
- ALVES, Juliano et al. **Metodologia para Avaliação de Software de Autoria como uma Ferramenta Computacional para auxílio no Desenvolvimento de Conteúdos Didático-Pedagógicos**. 2004. Disponível em: <<http://nlx.di.fc.ul.pt/~guelpeli/Arquivos/Artigo16.pdf>>. Acesso em: 25 mar. 2017..
- ANDRES, Daniele. **Técnicas de Avaliação de Software Educacional**. 1999. Disponível em: <www.inf.ufsc.br/~edla.ramos/infoedu/alunos/alunos99/trabfinal/Daniele1.doc>. Acesso em: 19 mar. 2017.
- ANDRES, Daniele; CYBIS, Walter. **Um estudo teórico sobre as técnicas de avaliação de software educacional**. 2000. Disponível em: <<http://sedici.unlp.edu.ar/handle/10915/23499>>. Acesso em: 19 nov. 2006.
- BAIXAKI. **IrYdium - Virtual Chemistry Lab**. 2010 Disponível em: <<http://www.baixaki.com.br/download/irydium-chemistry-lab.htm>>. Acesso em 01 de mar. de 2017.
- BORGE, Luciano. **Ensino de lógica de programação de forma visual**. 2012. Disponível em: <<http://proflborges.blogspot.com.br/2012/06/blockly-ensino-de-logica-de-programacao.html>>. Acesso em: 29 de abr. de 2017
- BORNATTO, Gilmar. **Modelagem - simulação - informática e a matemática**. Rev. PEC, Curitiba, v.2, n.1, p.67-71, jul. 2001-jul. 2002.
- BRITO, Cristina; ALMEIDA, Iolanda; Cavalcanti, Lialda. **O que se avalia e o que é preciso avaliar em um software educativo**. In: Workshop de Informática na Escola - WIE2003, XXIII Congresso da Sociedade Brasileira de Computação. 2003.
- CARRAHER, D. **A aprendizagem de conceitos com o auxílio do computador**. Em Alencar, M. E. Novas Contribuições da Psicologia aos Processos de Ensino-Aprendizagem. São Paulo: Cortez, 1992.
- CARVALHO, Liliane; MONTEIRO, Carlos. Análise de aspectos do PROINFO e possibilidades para a educação estatística. **Revista Tópicos Educacionais**, Recife, v. 1, p.10-31, jul. 2013.

FIALHO, Neusa; MATOS, Elizete. **A arte de envolver o aluno na aprendizagem de ciências utilizando softwares educacionais**. Educar em Revista, Curitiba, n. especial 2, p. 121-136, 2010. Editora UFPR.

Disponível em: <<http://www.scielo.br/pdf/er/nspe2/07.pdf>> Acesso: 20 de Fevereiro de 2017

FREITAS, Lilian Carla, KIERNER, Tereza Gonçalves. **Rumo ao Sucesso na Utilização de Softwares Educacionais para o Ensino e Aprendizagem de Ciências**. Revista de Informática Aplicada , v. 9, n.1, 2013.

FRESCKI, Franciele. **Avaliação da qualidade de softwares educacionais para o ensino de álgebra**. Monografia, Universidade Estadual do Oeste do Paraná - Cascavel, 2008. Disponível em:

<http://www.educadores.diaadia.pr.gov.br/arquivos/File/2010/artigos_teses/MATEMATICA/Monografia_Fran.pdf> Acesso em: 02 de Março de 2017

Gamez, L. **Ergonomia Escolar e as Novas Tecnologias no Ensino: Enfoque na Avaliação de Software Educacional**. Dissertação (Mestrado em Engenharia Humana). Braga, Portugal, Universidade do Minho, 1998.

GOOGLE PLAY. **Math Master - Brain Quizzes**, 2017. Disponível em:

<https://play.google.com/store/apps/details?id=com.mathmaster&hl=pt_BR> Acesso em 01 de Março de 2017.

LIMA, Telma Cristiane S. de; MIOTO, Regina Célia T. **Procedimentos metodológicos na construção do conhecimento científico: a pesquisa bibliográfica**. Rev. Katálises, Florianópolis, v. 10, n. especial, p. 37-45, 2007.

Disponível em: <

<http://www.periodicos.ufsc.br/index.php/katalysis/article/view/1030/5742>>. Acesso em: 01 mar. 2017.

MACEDO, Neusa. **Iniciação à pesquisa bibliográfica: guia do estudante para a fundamentação do trabalho de pesquisa**. 2. ed. revista. São Paulo: Edições Loyola, 1994.

Maltempi, M.V. (2005). **Novas Tecnologias e Construção de Conhecimento: Reflexões e Perspectivas**. In: *V Congresso Ibero-americano de Educação Matemática (CIBEM)*. Porto, Portugal, 17 a 22 de julho. Anais em CD. Disponível em: <<http://www.rc.unesp.br/igce/demac/maltempi/Publicacao/Maltempi-cibem.pdf>> acesso em: 18 de Fevereiro de 2017.

MARTINS, Kerley. **Teorias de aprendizagem e avaliação de software educativo**. Monografia (Informática Educativa) Universidade Federal do Ceará. 2002. Disponível em: <http://www.multimeios.ufc.br/arquivos/pc/monografias/Monografia_kerley.pdf> acesso em: 18 de Fevereiro de 2017.

MATOS, Ecivaldo. **Dialética da Interação Humano – Computador: tratamento didático do diálogo midiático**. Tese (doutorado em educação). Universidade de São Paulo. 2013

OLIVEIRA, Álvaro. **Avaliação de software: O caso de editores de textos.** Dissertação (Mestrado em Engenharia de Produção e sistemas) Universidade Federal de Santa Catarina. 1992.

OLIVEIRA, João. **Apoio à Avaliação de Usabilidade na Web – desenvolvimento do USEWEB.** Dissertação de Mestrado Profissional em Computação. Instituto de Computação - Universidade Estadual de Campinas. p. 116. 2006.

OLIVEIRA, Noé. **Uma proposta de avaliação de Softwares educacionais.** Dissertação (Mestrado em Engenharia de Produção). Florianópolis, Universidade Federal de Santa Catarina, 2001.
Disponível em: <<https://core.ac.uk/download/pdf/30362580.pdf>> acesso em: 18 de Fevereiro de 2017.

ONGARO, Edineia; CANAL, Ana. **Técnicas de usabilidade aplicadas aos softwares da empresa Zipline Tecnologia.** Disc. Scientia. Série: Ciências Naturais e Tecnológicas, S. Maria, v. 5, n. 1, p. 163-183, 2004. Disponível em: <<http://sites.unifra.br/Portals/36/tecnologicas/2004/Usabilidade.pdf>> acesso em: 18 de Dezembro de 2016.

PENHA, Daniel. **Software Educacional para o Ensino-Aprendizagem de Equação do 2º Grau: Desenvolvimento e Avaliação.** Lavras - MG, 2013. Disponível em: <http://repositorio.ufla.br/jspui/bitstream/1/5249/1/MONOGRAFIA_Software%20educacional%20para%20o%20ensino-aprendizagem%20de%20equa%C3%A7%C3%A3o%20do%20o%20grau%3A%20Desenvolvimento%20e%20avalia%C3%A7%C3%A3o.pdf> acesso em: 18 de Fevereiro de 2017.

PESCADOR, Cristina. **Tecnologias digitais e ações de aprendizagem dos nativos digitais.** V Congresso Internacional de Filosofia e Educação, Caxias do Sul/RS, ISSN 2177-644X, Maio 2010.
Disponível em: <http://www.ucs.br/ucs/tp/cinfe/eventos/cinfe/artigos/artigos/arquivos/eixo_tematico7/TECNOLOGIAS%20DIGITAIS%20E%20ACOES%20DE%20APRENDIZAGEM%20DOS%20NATIVOS%20DIGITAIS.pdf> Acesso em: Agosto de 2016

PRATES, Raquel; BARBOSA, Simone. **Avaliação de Interfaces de Usuário – Conceitos e Métodos.** Disponível em: <http://www.inf.puc-rio.br/~inf1403/docs/JAI2003_PratesBarbosa_avaliacao.pdf> Acesso em : 03/01/2017

RAMOS, Edla. **O fundamental na avaliação da qualidade do software educacional.** In: II Simpósio Brasileiro de Informática na Educação. Porto Alegre: SBC. p.122-131. 1991. Disponível em: <<http://www.inf.ufsc.br/~edla.ramos/publicacoes/Qualid.pdf>> acesso em: 07 de Janeiro de 2016.

REITZ, Simone. **Avaliação do Impacto da Usabilidade Técnica e Pedagógica no desempenho de Aprendizagens em “E-learning”.** Tese (Doutorado em Informática na Educação) – Programa de Pós-Graduação em Informática na Educação, Centro Interdisciplinar de Novas Tecnologias na Educação, Universidade

Federal do Rio Grande do Sul, Porto Alegre, 2009. Disponível em:
<<http://www.lume.ufrgs.br/bitstream/handle/10183/18253/000727594.pdf?sequence>>
Acesso em: 01 de Abril de 2017

REZENDE, Cristina. **Modelo de avaliação de qualidade de software educacional para o ensino de ciências**. Dissertação (Mestrado em Ensino de Ciências) Universidade Federal de Itajubá. 2013.

SANTANA, Leandro. **Utilizando a User Experience na Avaliação do Banco Internacional de Objetos Educacionais: um Estudo de Caso com Graduandos do Curso de Licenciatura em Matemática**. Serra Talhada, 2013. Disponível em:<
<http://docplayer.com.br/2345256-Leandro-cavalcanti-de-santana.html>> acesso em:
05 de Janeiro de 2017.

SILVA et. al. **Aplicação comparativa de diferentes abordagens de avaliação para o software educativo Duolingo**: A complexidade de escolher uma abordagem adequada. Revista Tecnologias na Educação – Ano 8 – Número/Vol.16 – Edição Temática – Congresso Regional sobre Tecnologias na Educação (Ctrl+E 2016) – Setembro 2016.
Disponível em:
<http://tecedu.pro.br/wp-content/uploads/2016/09/Art8-Aplica%C3%A7%C3%A3o-comparativa-de-diferentes-abordagens-de-avalia%C3%A7%C3%A3o-para-o-software-Duolingo-.pdf> Acesso: 26 de Março de 2016

TEIXEIRA, Adriano; BRANDAO, Edemilson. **Software educacional: difícil começo**. In: Revista Novas Tecnologias na Educação – RENOTE. Vol 1. n° 1. CINTED/UFRGS, fev/2003. Disponível em
<http://www.cinted.ufrgs.br/renote/fev2003/artigos/adriano_software.pdf> Acesso em:
12 de Dezembro de 2016. acesso em: 05 de Janeiro de 2017

VIEIRA, Fábila Magali Santos. **Avaliação de software educativo: reflexões para uma análise criteriosa**. Minas Gerais: Proinfo: 2000. Disponível em:<<http://www.connect.com.br/~ntemg7/avasoft.htm>> acesso em: 12 de Dezembro de 2016

ANEXO A - Questionário de Nokelainen

Respostas para as questões:

() Concordo totalmente () Concordo parcialmente () Indeciso () Discordo parcialmente () Discordo totalmente

Questões:

1. Quando eu trabalho nesta tarefa eu sinto que eu, não o programa, tenho controle sobre a responsabilidade de minha aprendizagem. (Definição: Eu não repito os mesmos tipos de passos nos meus estudos, mas as tarefas me fazem pensar e alcançar soluções diferentes para cada uma),

Critério: CONTROLE DO ALUNO

2. Quando eu usei este material senti que tive controle sobre o que realizei e não ao contrário. (Definição: O programa não me conduz de um passo para outro, mas eu posso controlar sozinho qual a ordem na qual quero finalizar minhas tarefas).

Critério: CONTROLE DO ALUNO

3. Eu tenho que pensar e tomar minhas próprias resoluções para aprender este material de aprendizagem. (Definição: Eu tenho que me concentrar no material, eu não posso completar as tarefas simplesmente para com a função de aprendizagem).

Critério: ATIVIDADE DO ALUNO

4. Este material de aprendizagem tem sido dividido dentro de seções, minha tarefa é aprendê las em uma ordem pré-definida (e possivelmente responder às tarefas).

Critério: ATIVIDADE DO ALUNO

5. Este material de aprendizagem proporciona questões de aprendizagem sem um modelo pré-definido para a sua resolução.

Critério: ATIVIDADE DO ALUNO

6. Este material não possui somente o material, mais *links* para várias outras fontes, as quais eu tenho de usar para aprender. (Definição: A aprendizagem inicia, por exemplo, com uma dica do professor, o qual conta qual tipo de ações são necessárias para a construção de "algo". Algumas das informações que você irá precisar estão no sistema, mas você terá de procurar a maioria delas em periódicos, livros ou na Internet para fazer sua própria apresentação).

Critério: ATIVIDADE DO ALUNO

7. Eu me aprofundi tanto neste material de aprendizagem que esqueci tudo o que estava acontecendo ao meu redor e de quanto tempo se passou.

Critério: ATIVIDADE DO ALUNO

8. Quando eu trabalho com este material de aprendizagem sinto que sei mais sobre alguns tópicos do que outros. Eu "sou um perito". (Definição: O material de aprendizagem pode envolver uma informação específica reunida a uma tarefa, por exemplo, uma entrevista com vizinhos ou a medida da altura da neve no jardim da casa em um período de um mês).

Critério: ATIVIDADE DO ALUNO

9. Quando eu trabalho neste material de aprendizagem, eu (ou nós, se um grupo de trabalho) tenho de encontrar soluções próprias em a solução de modelos dos professores ou do programa.

Critério: ATIVIDADE DO ALUNO

10. Estou orgulhoso com as minhas soluções ou uma solução realizada com outros para o problema apresentado no material de aprendizagem. (Definição: Eu sinto que eu ou fizemos algo significativo).

Critério: ATIVIDADE DO ALUNO

11. Este material de aprendizagem me deixa conversar com meus colegas (Definição: Por exemplo, as mensagens em chat ou quadro de avisos).

Critério: APRENDIZAGEM COOPERATIVA/COLABORATIVA

12. Eu posso fazer trabalhos em grupo com meus colegas neste material de aprendizagem. (Definição: Se eu quiser, eu posso realizar tarefas junto com meus colegas de aula de modo que ambos utilizem seus próprios computadores).

Critério: APRENDIZAGEM COOPERATIVA/COLABORATIVA

13. É agradável usar o material de aprendizagem com outro estudante no mesmo computador.

Critério: APRENDIZAGEM COOPERATIVA/COLABORATIVA, MOTIVAÇÃO

14. Este material de aprendizagem me permite saber o que os outros usuários têm feito no sistema. (Definição: Por exemplo, quais materiais de aprendizagem mais lidos ou as tarefas mais consagradas).

Critério: APRENDIZAGEM COOPERATIVA/COLABORATIVA

15. Este material de aprendizagem permite saber o que os outros usuários estão fazendo quando eu estou usando o sistema. (Definição: Por exemplo: a maior parte do material de aprendizagem no momento ou as tarefas com as quais a maior parte das pessoas estão trabalhando).

Critério: APRENDIZAGEM COOPERATIVA/COLABORATIVA

16. Este material de aprendizagem oferece programas de utilidade simples, como por exemplo, a calculadora.

17. Este material de aprendizagem oferece programas de utilidade prática (por exemplo, Tabelas Excel, editor HTML, processador de textos, etc..).

18. Neste material de aprendizagem os programas de aplicações possuem uma função fundamental. (Definição: Eu tenho que, por exemplo, editar uma tabela *Excel* para resolver um problema).

19. Eu posso salvar meu trabalho neste material de aprendizagem e usar ou avaliar o trabalho dos outros. (Definição: Eu posso, por exemplo, explorar ou avaliar o trabalho de outros grupos de trabalho e usá-los em meus estudos).

20. Este material de aprendizagem avisa claramente o que estou esperando saber (ou aprender) após tê-lo utilizado. (Definição: As metas de aprendizagem estão claramente definidas, por exemplo, "Após esta tarefa você saberá como dividir em frações decimais" ou "Após estas tarefas você pode formar perguntas na língua inglesa").

Critério: ORIENTAÇÃO DE OBJETIVO

21. Este material de aprendizagem mostra claramente porque é útil aprendê-lo. (Definição: Os objetivos de aprendizagem são justificados, por exemplo, "Esta tarefa irá ajudá-lo a fazer frases interrogativas na língua inglesa").

Critério: ORIENTAÇÃO DE OBJETIVOS

22. O material de aprendizagem avalia meus desempenhos com classificações. (Definição: Por exemplo, o sistema dá uma classificação ao final de uma tarefa e mostra a classificação máxima).

Critério: ORIENTAÇÃO DE OBJETIVOS

23. Este material de aprendizagem mostra quanto progresso eu realizei em meus estudos. (Definição: Eu sei no que ou experiente ou tenho que aprender mais).

Critério: ORIENTAÇÃO DE OBJETIVOS

24. Este material de aprendizagem é estritamente limitado. (Definição: Por exemplo, o tópico do material de aprendizagem de matemática é "Cálculo do significado").

Critério: ORIENTAÇÃO DE OBJETIVOS

25. Este material de aprendizagem ensina habilidades que necessitarei. (Definição: Eu sou capaz de, por exemplo, converter Euros em dólares, ou ajudar meus pais a escolher entre diferentes tamanhos de pacotes de acordo com seus diferentes preços).

Critério: APLICABILIDADE

26. Eu sinto que estou apto a usar as habilidades e conhecimento que este material de aprendizagem tem me ensinado no futuro.

Critério: APLICABILIDADE

27. Este material de aprendizagem é baseado na ideia que "alguém aprende melhor fazendo por si mesmo". (Definição: O material oferece mais tarefas do que por exemplo, as apresentações de *Power Point*).

Critério: APLICABILIDADE

28. Eu sinto que este material de aprendizagem ajudará a realizar melhor o teste. (Definição: Eu penso que as tarefas no material são similares às tarefas que nós frequentemente temos nos testes).

Critério: APLICABILIDADE

29. Este material de aprendizagem é adequadamente desafiador para mim. (Definição: As tarefas não são tão fáceis ou tão difíceis).

Critério: APLICABILIDADE

30. Eu sinto que este material de aprendizagem foi projetado para mim. (Definição: O material satisfaz minhas necessidades e isso não dá a sensação de que você seja muito ou pouco competente).

Critério: APLICABILIDADE

31. Este material de aprendizagem ajusta-se às dificuldades para adaptar-se às minhas habilidades. (Definição: Eu posso praticar algo que é difícil para mim até que eu o tenha aprendido e antes de partir para o próximo tópico).

Critério: APLICABILIDADE

32. As imagens neste material de aprendizagem ajudam a aprender.

Critério: VALOR AGREGADO

33. Os sons neste material de aprendizagem ajudam a aprender.

Critério: VALOR AGREGADO

34. As animações neste material de aprendizagem ajudam a aprender.

Critério: VALOR AGREGADO

35. É mais útil aprender tópicos com este material de aprendizagem do que com métodos convencionais em uma sala de aula. (Definição: Pense se você estaria mais disposto a fazer estas tarefas com um computador ou com um livro de estudos normal ou livro de exercícios).

Critério: VALOR AGREGADO

36. Eu tento alcançar uma alta classificação tanto quanto posso neste material de aprendizagem.

Critério: MOTIVAÇÃO

37. Eu quero aprender os tópicos deste material de aprendizagem tão profundamente quanto posso.

Critério: MOTIVAÇÃO

38. Estou interessado nos tópicos deste material de aprendizagem.

Critério: MOTIVAÇÃO

39. Este material de aprendizagem requer que eu saiba algo que tem sido pensado em algum outro material de aprendizagem. (Definição: Este material faz referência a algum outro material de aprendizagem).

Critério: AVALIAÇÃO DO CONHECIMENTO PRÉVIO

40. Eu posso usar meus conhecimentos prévios quando estudo com este material.

Critério: CONTROLE DO ALUNO, APLICABILIDADE, AVALIAÇÃO DO CONHECIMENTO PRÉVIO

41. Este material de aprendizagem revê materiais anteriores antes de iniciar a ensinar um novo tópico. (Definição: Por exemplo, na matemática, o material primeiro inicia com simples cálculos que são necessários para aprender um tópico mais difícil).

Critério: AVALIAÇÃO DO CONHECIMENTO PRÉVIO

42. Este material de aprendizagem oferece caminhos opcionais para o meu progresso. (Definição: Eu posso escolher diferentes tarefas cada vez que eu uso o sistema).

Critério: FLEXIBILIDADE

43. Este material de aprendizagem não me deixa prosseguir para o próximo ponto antes de ter respondido corretamente a cada questão. (Definição: Por exemplo, em um exercício de língua inglesa deve-se ter respondido corretamente a todas as questões, mesmo que com a ajuda do programa, antes de deixá-lo prosseguir para o próximo tópico).

Critério: CONTROLE DO ALUNO

44. Este material de aprendizagem apresenta muitas similaridades, tarefas consecutivas. (Definição: Por exemplo, uma tarefa de preencher que possui muitas tarefas consecutivas para sentenças do verbo "to be").

Critério: FLEXIBILIDADE

45. Este material de aprendizagem torna rápido e fácil o aprendizado de um novo tópico ou o recapitular de um tópico anterior.

Critério: VALOR AGREGADO

46. Se eu não posso me lembrar de uma palavra específica ou conceito enquanto uso este material de aprendizagem posso retomar e verificar seu significado no material anterior.

Critério: FLEXIBILIDADE

47. Quando uso este material de aprendizagem sinto que tenho de lembrar muitas coisas ao mesmo tempo. (Definição: Eu sinto em alguns momentos que deveria ter usado o papel para escrever algumas anotações).

Critério: CONTROLE DO ALUNO

48. Este material de aprendizagem apresenta informações em formato que o torna fácil de aprender. (Definição: A informação é apresentada de forma significativa e interconectada e não em partes separadas que são difíceis de entender).

Critério: CONTROLE DO ALUNO, APLICABILIDADE

49. Este material de aprendizagem apresenta novos materiais (ou recapitula antigos) em "porções" adequadas para mim. (Definição: Não há muitas novas coisas apresentadas de uma só vez, eu tenho tempo de aprendê-las antes de mover-me para o próximo tópico).

Critério: CONTROLE DO ALUNO, APLICABILIDADE

50. Eu posso fazer certo número de erros com este material (por exemplo, respostas erradas em tarefas de cálculos).

Critério: *FEEDBACK*

51. Quando eu erro uma solução de uma tarefa, o programa me envia um aviso amigável.

Critério: *FEEDBACK*

52. Este material de aprendizagem me dá um *feedback* motivador. (Definição: Eu estou querendo pôr a prova as funções menos usadas no material de aprendizagem porque eu sei irá me dar todos os avisos que eu preciso).

Critério: *FEEDBACK*

53. Este material de aprendizagem apresenta um *feedback* imediato das minhas atividades. (Definição: Quando eu escrevo minha resposta a uma tarefa de cálculo, o sistema mostra imediatamente se a resposta é correta ou não).

Critério: *FEEDBACK*

54. Este material de aprendizagem dá inicialmente um exemplo da solução correta. (Definição: Multiplicar com frações decimais é iniciado com um modelo de desempenho, e depois posso calcular por conta própria).

Critério: APLICABILIDADE

55. Neste material de aprendizagem, eu transporto a responsabilidade para a solução de uma tarefa em pequenas porções. (Definição: Por exemplo, em uma tarefa de matemática, será apresentada primeiro a tarefa e então o resultado. A seguir, eu vejo a tarefa mas não o resultado, o qual eu tenho de resolver por conta própria).

Critério: APLICABILIDADE

56. Eu penso que aprendo mais rapidamente com este material do que normalmente. (Definição: Este material de aprendizagem proporciona o tipo certo de suporte quando preciso).

Critério: APLICABILIDADE

ANEXO B - Questionário de Ssemugabi

Categoria 1: Projeto *interface* em geral

1 Visibilidade do *status* do sistema

1.1 O Sistema me mantém informado através de *feedback*: sobre o que está acontecendo.

Resposta:

() Concordo plenamente () Concordo () Talvez () Discordo () Discordo plenamente.

1.2 Eu entendo o que significa *feedback*.

1.3 Eu obtenho o *feedback* dentro de um tempo razoável.

1.4 Posso ver ou ouvir os resultados de cada ação que eu realizo.

1.5 O sistema não reage de forma que me surpreenda e não faz nada inesperado.

2 Modelo do projetista e modelo do aluno, isto é, correlação entre o sistema e o mundo real

2.1 A linguagem usada é natural, os termos, frases, conceitos são similares àqueles usados no meu dia-a-dia ou no ambiente de estudo.

2.2 Eu não fico confuso com o uso dos termos.

2.3 Eu não fico confuso com a forma que os símbolos, ícones, imagens são usadas.

2.4 Não há jargões usados (jargão significa uma abreviatura ou uma expressão desenvolvida e usada por um grupo de pessoas).

2.5 A metáfora usada corresponde aos objetos ou conceitos do mundo-real, por exemplo, o ícone para salvar se parece com um disquete.

2.6 As informações são organizadas em uma ordem natural e lógica.

3 Controle do usuário e liberdade

3.1 Eu controlo o sistema, ao invés deste me controlar.

3.2 O sistema trabalha da forma que eu quero que trabalhe.

3.3 Cada página apresenta todos os botões de navegação ou *hiperlinks* necessários, tais como, anterior (voltar), próxima e página inicial (*home*).

3.4 Quando eu cometo um erro eu não posso escolher sair do sistema usando um botão de saída de emergência claramente sinalizado.

4 Consistência e aderência às normas

4.1 A mesma convenção (normas ou o caminho pelo qual o conteúdo é organizado e apresentado) é usado através do sistema.

4.2 É fácil de entender as convenções usadas através do sistema.

4.3 A convenção usada é similar àquelas em outros sistemas que utilizei.

4.4 As mesmas palavras, frases, situações ou ações referem-se às mesmas coisas através do sistema.

4.5 As cores são usadas de forma consistente (mesma forma) através do sistema.

- 4.6 Os gráficos, ícones e imagens são consistentemente usados através do sistema.
- 4.7 Há consistência no *layout* do sistema.
- 4.8 Há consistência no uso dos menus.
- 4.9 Há consistência no uso do tipo e tamanhos das fontes.
- 4.10 Os *links* das páginas são consistentes com os títulos das páginas as quais estão vinculadas.

5 Prevenção de erros, prevenção especificamente de erros relacionados à usabilidade periférica

- 5.1 O sistema me dá suporte de forma que se torna difícil cometer erros graves.
- 5.2 A qualquer hora que um erro é cometido uma mensagem de erro é apresentada.
- 5.3 O sistema lisa uma interface de usuário gráfica, tais como, listas suspensas (*drop down*), com *hiperlink* ou interfaces baseadas em ícones, que podem ser clicadas por mouse, ao invés de baseada em comandos, onde os comandos têm de ser digitados pelo uso de teclado.
- 5.4 Sou requisitado a confirmar minhas entradas antes de levar adiante ações "potencialmente perigosas" como a de "apagar".
- 5.5 Acho fácil entrar com as informações no sistema.

6 Reconhecimento ao invés de evocação

- 6.1 Estão disponíveis instruções de como utilizar o sistema.
- 6.2 Há uma relação óbvia entre os controles e suas ações.
- 6.3 Os objetos usados, como os gráficos em barras de ferramentas, são fáceis de reconhecer.
- 6.4 Ao trabalhar em uma tarefa eu não preciso lembrar as informações de outras tarefas.

7 Flexibilidade e eficiência do uso

- 7.1 O site oferece diferentes níveis de usuários, desde o novato até o experiente.
- 7.2 Atalhos em forma de abreviações, questões especiais, macros e comandos escondidos estão disponíveis para usuários experientes.
- 7.3 O site guia os usuários novatos de forma competente.
- 7.4 Há a opção de usar somente o teclado para realizar as tarefas.
- 7.5 O sistema é o bastante flexível para permitir que os usuários ajustem as configurações adequando-as, isto é, personalizando o sistema.

8 Design estético e minimalista

- 8.1 As páginas contêm a informação requisitada.
- 8.2 A informação em cada página não é muito grande para confundir-me ou distrair-me.
- 8.3 Não há o uso excessivo de gráficos e imagens no *site*.
- 8.4 As caixas de diálogo proporcionam informações adequadas ao desempenho das tarefas.
- 8.5 Menus e listas suspensas possuem as opções exigidas para as escolhas.

9 Reconhecimento, diagnóstico e recuperação de erros

- 9.1 As mensagens de erros são expressas em linguagem simples.
- 9.2 As mensagens de erros indicam precisamente qual é o problema.
- 9.3 Cada mensagem proporciona um procedimento para a correção do erro.
- 9.4 O procedimento para 'corrigir um erro é específico, rápido e eficiente.
- 9.5 Se um erro de digitação (dados) resulta em um erro, não é necessário digitar o comando por inteiro, mas de preferência reparar somente a parte errada.
- 9.6 O *site* proporciona uma rápida mudança de ação pela qual é possível, por exemplo, disponibilizar ambos Desfazer (*Undo*) e Refazer (*Redo*).

10 Ajuda e documentação

- 10.1 Eu acho os recursos de ajuda, tais como ajuda *online* e o glossário, úteis.
- 10.2 Os recursos de ajuda são fáceis de usar.
- 10.3 Eu acho fácil procurar pela ajuda solicitada.
- 10.4 Os links para outras fontes são de g:'.mdc auxílio.

Categoria 2: Design específico para *websites* educacionais

11 Simplicidade de navegação, organização e estrutura

- 11.1 Eu sempre sei onde estou e quais opções seguir, isto é, o que concluí e o que ainda devo fazer.
- 11.2 Não há a necessidade de ter o Menu do Curso desde que eu tenha o Mapa do Curso, o qual satisfaz à mesma proposta.
- 11.3 O site sempre direciona para os documentos e páginas corretas,
- 11.4 Eu gostaria de ter *links* às seções dentro da mesma página.
- 11.5 As cores para os links são consistentes com as convenções Web, isto é, links não visitados em azul e os visitados em verde ou roxo.
- 11.6 Informações correlacionadas são colocadas juntas.
- 11.7 Informações importantes são colocadas no topo da página.
- 11.8 A função de rolar páginas é minimizada, isto é, eu não tenho rolar várias páginas para encontrar as informações requisitadas.

12 Relevância do conteúdo do site para a aprendizagem

- 12.1 O conteúdo mantém-me engajado.
- 12.2 O conteúdo é relevante ao que deve ser aprendido.
- 12.3 O conteúdo está no nível apropriado de meu entendimento.
- 12.4 Estão definidos quais são os materiais estão protegidos por leis de direito autoral e quais não são.
- 12.5 O material no site não possui preconceitos raciais ou quanto ao gênero.
- 12.6 Eu gostaria de ver as gravações das minhas atividades, tais como, visitas feitas em cada página.

Categoria 3: Heurísticas de *design* instrucional centrado no aluno

13 Claridade das metas, objetivos e saídas

- 13.1 Eu conheço os objetivos antes de cada encontro.
- 13.2 As saídas são comunicadas antecipadamente, antes do início do encontro.
- 13.3 Eu acho o plano de estudos útil.
- 13.4 Eu obtenho informações atualizadas no quadro de avisos.
- 13.5 O calendário proporciona informações úteis.

14 Aprendizagem colaborativa

- 14.1 Eu gosto de sites que tenham meios de incentivar atividades em grupo, tais como, um projeto em grupo e a colaboração na resolução de problemas.
- 14.2 A colaboração com 'outros alunos, introduzida ou apoiada pelo sistema, permite-me aprender algo.
- 14.3 Ao colaborar com outros alunos, eu gostaria que o professor agisse como um facilitador, guia, treinador ou mentor, mas não como um controlador.
- 14.4 Quando colaboro com o professor, gostaria que o professor agisse como um parceiro, não como um controlador.
- 14.5 Embora eu tenha a ferramenta de *e-mail* proposta pela instituição, eu ainda necessito ter a ferramenta de e-mail dentro de aplicativo.
- 14.6 Eu gostaria de ter um fórum de discussão acadêmico com outros alunos usando ferramentas de discussão no *site*.
- 14.7 Eu gostaria de ter um fórum de discussão acadêmico com o professor usando ferramentas de discussão no *site*.

15 Aplicabilidade do nível de controle do aluno

- 15.1 Eu posso decidir o que aprender e o que deixar de lado, dentro do *site*.
- 15.2 Quando eu uso o *site*, sinto como se eu estivesse no controle da minha própria aprendizagem .
- 15.3 Eu sinto um senso de domínio próprio neste *site*.
- 15.4 Eu posso usar meus próprios caminhos para encontrar o que eu quero aprender dentro do *site*.
- 15.5 A partir do *site* eu gosto de acessar a internet para encontrar meu próprio material de aprendizagem.

16 Suporte para abordagens significativas de aprendizagem

- 16.1 Este *site* proporciona diferentes estratégias de suporte para a aprendizagem.
- 16.2 O *site* é usado em combinação com outros meios de instrução para dar suporte à aprendizagem.
- 16.3 O *site* me permite planejar, avaliar e questionar minhas habilidades de aprendizagem.
- 16.4 Este site me encoraja a aplicar as habilidades de aprendizagem dentro de situações do mundo prático/real.

17 Identificação de erros cognitivos, diagnóstico e restabelecimento

17.1 Quando eu obtenho respostas erradas aos problemas proporcionados pelo sistema, as soluções oferecidas pelo sistema ajudam-me a aprender.

17.2 Eu acredito que as pessoas aprendem por meio de seus erros.

17.3 O sistema reconhece que como aluno, eu posso estar errado sobre alguns conceitos e que tentar corrigi-las é parte da aprendizagem.

18 Feedback, orientação e avaliação

18.1 O sistema proporciona *feedback* em relação as minhas atividades e conhecimentos.

18.2 Eu aprecio a orientação em forma de questões de exemplo e suas soluções.

18.3 Eu aprecio o *feedback* quantitativo como forma de classificar minhas atividades.

19 Contexto significativo

19.1 O conhecimento é apresentado dentro de um contexto significativo que me ajuda a aprender.

19.2 O conhecimento é apresentado num sentido de que ser autêntico, corresponde a como as coisas são na prática.

19.3 São empregadas tarefas autênticas e contextualizadas (tarefas que se relacionam com as do mundo real), preferivelmente do que instruções abstratas (modelos puramente teóricos).

19.4 As representações simbólicas usadas são fáceis de entender e os símbolos usados são significativos dentro do contexto da tarefa de aprendizagem.

20 Motivação, criatividade e aprendizagem ativa

20.1 A aplicação fornece motivação própria (intrínseca) que me faz querer aprender.

20.2 As classificações (notas) e outros incentivos que eu obtenho são motivações externas (extrínsecas) e requisitos institucionais consumados.

20.3 A aplicação facilita meu empenho.

20.4 A aplicação mantém minha atenção.

20.5 Eu gosto do visual e da atmosfera deste *site*.

20.6 Eu prefiro atividades que são subdivididos em partes, tais como jogos e testes, pois estas não consomem muito tempo.

20.7 Eu gosto dos testes/jogos no *site*.

20.8 Eu gosto de fazer auto-avaliações neste *site*.

Conclusões

a. Eu achei o sistema fácil de usar.

b. É rápido de se trabalhar no sistema.

c. O sistema desempenha as tarefas apropriadamente.

d. Uma vez que eu aprendi a usar o sistema, será fácil de usá-la na próxima vez.

e. Eu fiquei satisfeito com o sistema.

() Concordo plenamente () Concordo () Talvez () Discordo () Discordo plenamente

f. O quão bem este site trabalha como um suplemento de instrução da aula/disciplina?

Muito boa Boa Adequado Ruim Muito ruim

g. Eu prefiro aprender usando:

Ambiente virtual de aprendizagem Aula presencial Ambos

h. Qual é sua visão geral deste *site*?

Muito boa Boa Adequado Ruim Muito ruim

i. O que mais o atrai neste *site*? (questão aberta)

j. O que você mais gostou neste *site*? (questão aberta)

k. O que você menos gostou neste *site*? (questão aberta)

l. Use o espaço que segue para preencher no mínimo cinco problemas que você encontrou como os mais problemáticos no sistema. (questão aberta)

ANEXO C - Questionário de Reitz

| Nº | Severidade | Questão |
|----|------------|---|
| 1 | Média | O material de aprendizagem é dividido em partes (seções), minha tarefa é aprendê-las em uma ordem pré-definida. |
| 2 | Média | As questões para a aprendizagem não seguem sempre o mesmo modelo para suas resoluções. |
| 3 | Baixa | Os <i>links</i> para outras fontes são de grande auxílio para minha aprendizagem. |
| 4 | Média | Esqueci de tudo o que estava acontecendo ao meu redor e de quanto tempo se passou. |
| 5 | Média | Estou orgulhoso das minhas soluções para os exercícios apresentados. Sinto que fiz algo significativo. |
| 6 | Média | O material de aprendizagem avalia meus desempenhos com classificações (notas) de minhas atividades. |
| 7 | Baixa | E mais útil aprender tópicos com este material no computador do que com livros em sala de aula. |
| 8 | Baixa | É rápido e fácil o aprendizado de um novo tópico ou o recapitular de um tópico anterior. |
| 9 | Alta | Este material exige que eu tenha conhecimentos de outro material de aprendizagem anterior. |
| 10 | Média | Este material de aprendizagem não me deixa prosseguir para o próximo ponto ou exercício antes de ter respondido corretamente a cada questão. |
| 11 | Média | Os programas utilizados (calculadora, exercícios, jogos, etc.) são simples e práticos de usar. |
| 12 | Baixa | Quando erro na solução de uma tarefa, o programa me envia um aviso amigável. |
| 13 | Alta | O <i>feedback</i> (aviso/resposta dado pelo sistema) é imediato. |
| 14 | Baixa | Recebo <i>feedback</i> (aviso) motivador. |
| 15 | Média | A linguagem usada é natural. Os termos, frases, conceitos são similares àqueles usados no meu dia-a-dia ou no ambiente de estudo. |
| 16 | Média | Não são usados jargões no <i>site</i> (jargão significa uma abreviatura ou uma expressão técnica desenvolvida e usada por um grupo de pessoas). |

| | | |
|----|-------|--|
| 17 | Alta | Não fico confuso com a forma com que os símbolos, ícones, imagens são usados. |
| 18 | Média | A metáfora usada corresponde aos objetos ou conceitos do mundo real, por exemplo, o ícone para salvar se parece com um disquete. |
| 19 | Média | As imagens ajudam a aprender. |
| 20 | Baixa | Os sons ajudam a aprender. |
| 21 | Baixa | As animações ajudam a aprender. |
| 22 | Média | As informações são organizadas em uma ordem natural e lógica. |
| 23 | Média | Sinto como se eu estivesse no controle da minha própria aprendizagem ao interagir com o <i>site</i> . |
| 24 | Média | Eu posso usar meus próprios caminhos para encontrar o que eu quero aprender dentro do <i>site</i> . |
| 25 | Alta | Tenho de lembrar muitas coisas ao mesmo tempo. Gostaria de usar o papel para escrever algumas anotações. |
| 26 | Alta | Cada página apresenta todos os botões de navegação ou <i>hiperlinks</i> necessários, tais como, anterior (voltar), próxima e página inicial (<i>homepage</i>). |
| 27 | Média | Este material de aprendizagem dá inicialmente um exemplo de exercício com uma solução correta. |
| 28 | Alta | Gosto de aprender com questões de exemplo e suas soluções. |
| 29 | Alta | Este material de aprendizagem ensina habilidades e conhecimentos que necessitarei no futuro. |
| 30 | Média | A mesma forma de apresentação e organização do conteúdo é usada em todo <i>site</i> . |
| 31 | Média | As informações são apresentadas em um formato que os torna fácil de aprender. |
| 32 | Média | As cores são usadas da mesma forma em todo o <i>site</i> . |
| 33 | Média | Os gráficos, ícones e imagens são usados da mesma forma em todo o site. |
| 34 | Média | Os menus são apresentados da mesma forma em todo o <i>site</i> . |
| 35 | Baixa | Os <i>links</i> das páginas correspondem aos títulos das páginas às quais estão vinculadas. |
| 36 | Alta | A qualquer hora que um erro é cometido uma mensagem de erro é apresentada. |
| 37 | Média | Sou requisitado a confirmar minhas ações antes de levar adiante |

| | | |
|----|-------|---|
| | | ações "perigosas" como a de "apagar". |
| 38 | Baixa | Acho fácil entrar com informações no sistema. |
| 39 | Alta | O <i>site</i> guia os usuários experientes através de atalhos em forma de abreviações, questões especiais, macros e comandos escondidos |
| 40 | Baixa | Este material de aprendizagem apresenta muitas tarefas parecidas e repetitivas. |
| 41 | Média | As caixas de diálogo proporcionam informações adequadas ao desempenho das tarefas. |
| 42 | Alta | As mensagens de erros indicam precisamente qual é o problema |
| 43 | Alta | As mensagens de erro indicam um procedimento para a correção do erro. |
| 44 | Média | Se um erro de digitação (dados) resulta em um erro, não preciso digitar novamente o comando por inteiro, mas reparar somente a parte errada. |
| 45 | Alta | O <i>site</i> proporciona uma rápida mudança de ação através dos comandos de Desfazer (<i>Undo</i>) e Refazer (<i>Redo</i>). |
| 46 | Baixa | A ajuda <i>online</i> e o glossário (recursos de ajuda) são úteis. |
| 47 | Média | Os recursos de ajuda são fáceis de usar. |
| 48 | Alta | Sei onde estou e quais opções seguir, isto é, o que concluí e o que ainda devo fazer. |
| 49 | Média | Há <i>links</i> que levam aos tópicos de estudos (seções) dentro da mesma página |
| 50 | Média | As cores para os <i>links</i> são consistentes com as convenções da <i>Web</i> , isto é, <i>links</i> não visitados em azul e os visitados em vermelho ou roxo. |
| 51 | Média | As informações importantes são colocadas no topo da página. |
| 52 | Baixa | Não preciso rolar (barra de rolagem) várias páginas para encontrar as informações que procuro. |
| 53 | Média | O conteúdo mantém minha atenção. |
| 54 | Média | O conteúdo está no nível apropriado de meu entendimento. |
| 55 | Baixa | Estão definidos quais são os materiais protegidos por leis de direito autoral e quais não são. |
| 56 | Baixa | Obtenho informações atualizadas no quadro de aviso |
| 57 | Baixa | O calendário proporciona informações úteis. |
| 58 | Média | Gosto de atividades em grupo, tais como, um projeto em grupo e a |

| | | |
|----|-------|--|
| | | colaboração na resolução de problemas. |
| 59 | Média | A colaboração com outros alunos, introduzida ou apoiada pelo sistema, permite-me aprender. |
| 60 | Média | Ao interagir com os outros alunos e o professor, eu gostaria que o professor agisse como um parceiro, guia, e não como um controlador. |
| 61 | Média | Embora eu tenha a ferramenta de <i>e-mail</i> proposta pela instituição, eu ainda necessito ter a ferramenta de e-mail dentro do aplicativo. |
| 62 | Baixa | Gosto de participar do fórum de discussão acadêmico com outros alunos. |
| 63 | Média | E agradável usar o material de aprendizagem com outro estudante no mesmo computador. |
| 64 | Baixa | Posso salvar meu trabalho neste material de aprendizagem e usar ou avaliar o trabalho dos outros alunos. |
| 65 | Média | O <i>site</i> me permite planejar, avaliar e questionar minhas habilidades de aprendizagem. |
| 66 | Média | Este <i>site</i> me encoraja a aplicar as habilidades de aprendizagem nas situações práticas da vida. |
| 67 | Média | Quando meu obtenho respostas erradas aos problemas proporcionados pelo sistema, as soluções oferecidas pelo sistema ajudam-me a aprender. |
| 68 | Alta | São empregadas tarefas autênticas e contextualizadas (tarefas práticas), mais do que instruções abstratas (conceitos teóricos). |
| 69 | Média | O material de aprendizagem me faz querer aprender, sinto-me motivado. |
| 70 | Média | Gosto de fazer auto-avaliações neste <i>site</i> . |
| 71 | Média | Gosto do visual e da aparência deste <i>site</i> |
| 72 | Média | Gosto dos testes e jogos no <i>site</i> . |

ANEXO D - *Checklist* do PROINFO

1. Qual a proposta pedagógica que permeia o *software*?
2. Proporciona um ambiente interativo entre aluno e o *software*? Como?
3. Permite uma fácil exploração?(seqüencial, não linear)
4. Apresenta conceitos de forma clara e correta?
5. Desperta o interesse do aluno, sem perder de vista os objetivos do *software* e do usuário?
6. Oferece alternativas diversificadas para a construção das ações do aluno?
7. Permite que o aluno construa seu conhecimento a partir da ação-reflexão-ação?
 - 7.1. Tem recursos de programação?
 - 7.2. Permite o registro e a consulta das ações desenvolvidas?
8. Os recursos de multimídia usados têm relevância para os objetivos do *software*?
9. Proporciona condições de abordagem sócio-cultural que contemple aspectos regionais?
10. Os aspectos técnicos especificados no *software* são compatíveis com a configuração dos equipamentos existentes na escola?
11. É de fácil instalação e desinstalação?
12. Permite a utilização em rede?
13. Apresenta uma visão interdisciplinar?
14. Apresenta encarte com explicações sobre objetivos, conteúdos, equipe de desenvolvimento do *software* e sugestões metodológicas para a sua utilização?
15. Em que idioma o *software* é apresentado? Existe uma versão em português?
16. Em relação aos demais recursos didáticos, qual o diferencial que o *software* apresenta?
17. Proporciona um ambiente de aprendizagem por descoberta?
18. Permite a integração com outros *software*?
19. Apresenta um ambiente lúdico e criativo?
20. Qual o tipo de *software* (jogo, tutorial, exercício- prática, autoria, outros)?

ANEXO E - Checklist proposto por Oliveira (2001)

Ficha de Identificação do produto:

INFORMAÇÕES GERAIS:

| | |
|-----------------------------|---|
| Nome do avaliador | |
| Título do <i>software</i> : | () simulação () demonstração () outros |
| Projetista: | |
| Distribuidor: | |
| Área do conteúdo: | |
| Objetivos: | |
| Nível da clientela: | |
| Custo: | |
| Corrente Pedagógica: | |

APRESENTAÇÃO E FUNCIONALIDADE DO PRODUTO

Atribua valores iguais a: 0, 1, ou 2, respectivamente, para não, às vezes e, sim:

| Característica do <i>Software</i> | Atributo |
|---|----------|
| O manual é adequado e claro, quanto aos conteúdos? | |
| Apresenta as exigências de <i>hardware</i> ? | |
| É de fácil instalação? | |
| Está de acordo com as finalidades e objetivos propostos? | |
| E mais eficaz do que outra metodologia para atingir os objetivos a que se propõe? | |
| Interage com os sistemas especificados? | |
| Suas funções são apropriadas para as tarefas especificadas? | |
| Gera resultados ou efeitos corretos de acordo com a especificação? | |
| Evita acesso não autorizado, acidental ou deliberado, a programas e dados? | |

| | |
|--|--|
| O <i>software</i> declara a sua corrente pedagógica? | |
| O <i>software</i> é apropriado a faixa etária do seu público alvo? | |
| Subtotal (dê a soma dos atributos). Máximo 22 pontos | |

CONFIABILIDADE

Atribua valores iguais a: 0, 1, ou 2, respectivamente, para não, às vezes e, sim:

| Característica do <i>Software</i> | Atributo |
|--|-----------------|
| Há ausência de falhas? | |
| Mantém o nível de desempenho na ocorrência de falhas? | |
| Recupera os dados quando ocorrem falhas? | |
| Subtotal (dê a soma dos atributos). Máximo 6 pontos | |

USABILIDADE

Atribua valores iguais a: 0, 1, ou 2, respectivamente, para não, às vezes e, sim:

| Característica do <i>Software</i> | Atributo |
|--|-----------------|
| O programa proporciona facilidade no entendimento do conceito e aplicação? | |
| É fácil a sua aplicação? | |
| É fácil o controle e a operação? | |
| Permite a realização de diferentes observações incorporadas ao objeto? | |
| As diferentes observações incorporadas ao objeto são imediatas? | |
| Permit desenvolver e testar hipóteses? | |
| Permite a análise de resultados depurando os conceitos? | |
| Permite o uso simultâneo por grupos de alunos? | |
| Subtotal (dê a soma dos atributos). Máximo 16 pontos | |

EFICIÊNCIA

Atribua valores iguais a: 0, 1, ou 2, respectivamente, para não, às vezes e, sim:

| Característica do Software | Atributo |
|---|-----------------|
| O tempo de resposta ao comando das funções é rápido? | |
| Dispensa outros recursos (impressora, discos flexíveis) para ser usado? | |
| Subtotal (dê a soma dos atributos). Máximo 4 pontos | |

MANUTENIBILIDADE

Atribua valores iguais a: 0, 1, ou 2, respectivamente, para não, às vezes e, sim:

| Característica do Software | Atributo |
|---|-----------------|
| É fácil perceber quando houve uma falha? | |
| É fácil modificar, remover defeitos ou adaptá-lo a mudanças de ambientes? | |
| Permite a aplicação de alguma modificação sem que ocorram efeitos inesperados ? | |
| É fácil o teste quando se faz alguma modificação? | |
| Subtotal (dê a soma dos atributos). Máximo 8 pontos | |

PORTABILIDADE

Atribua valores iguais a: 0, 1, ou 2, respectivamente, para não, às vezes e, sim:

| Característica do Software | Atributo |
|--|-----------------|
| É fácil adaptar a outros ambientes especificados? | |
| É fácil instalar em outros ambientes? | |
| É fácil substituir um outro <i>software</i> dentro do ambiente do mesmo? | |
| Subtotal (dê a soma dos atributos). Máximo 6 pontos | |

CONTEÚDO DO SOFTWARE RELACIONADO COM A DISCIPLINA

Atribua valores iguais a: 0, 3, ou 5, respectivamente, para não, às vezes e, sim:

| Característica do Software | Atributo |
|---|-----------------|
| Transmite conceitos adequados? | |
| O <i>software</i> utiliza adequadamente a linha pedagógica proposta? | |
| Tem uma variedade de níveis de dificuldade? | |
| Os alunos podem chegar a determinados pontos sem seguir uma sequência obrigatória? | |
| Estão previstas apresentações adicionais se necessárias? | |
| As instruções são claras e lógicas? | |
| O <i>software</i> utiliza bem o gráfico? | |
| O <i>software</i> utiliza bem o som? | |
| O <i>software</i> utiliza bem a cor? | |
| Permite manipular vários dados com a utilização de um eventual banco de dados possivelmente modificáveis? | |
| Apresenta exercícios de níveis diferentes, relacionados com o conteúdo estudado? | |
| Utiliza a correção da ortografia e gramática? | |
| Subtotal (dê a soma dos atributos). Máximo 60 pontos | |

CONTEÚDO DO SOFTWARE RELACIONADO COM A DISCIPLINA

Atribua valores iguais a: 0, 3, ou 5, respectivamente, para não, às vezes e, sim:

| Característica do Software | Atributo |
|--|-----------------|
| De acordo com o conteúdo, o <i>software</i> utiliz alguma simulação? | |
| Permite a interação de diferentes observações incorporadas ao objeto? | |
| Ajusta-se aos objetivos curriculares? | |
| Faz questionamentos ao usuário relacionados às respostas dadas durante a sua utilização? | |
| Apresenta a avaliação final com os resultados obtidos pelo usuário? | |
| A avaliação é compatível com a corrente pedagógica? | |
| Apresenta “ <i>feedback</i> ” ? | |
| A linguagem está adequada para a faixa etária a que se propõe? | |
| O material visual (figuras, gráficos, simulações) está adequado à faixa etária? | |

| | |
|---|--|
| Apresenta <i>links</i> que permitam ao usuário buscar mais informações sobre o assunto? | |
| Os conteúdos são trabalhados de forma interdisciplinar ? | |
| E interativo? | |
| Subtotal (dê a soma dos atributos). Máximo 65 pontos | |

SOMAS DOS ATRIBUTOS:

| Característica do Software | Atributo |
|--|----------|
| Apresentação e funcionalidade | |
| Confiabilidade | |
| Usabilidade | |
| Eficiência | |
| Manutenibilidade | |
| Portabilidade | |
| Conteúdo do software, relacionado com a disciplina | |
| Subtotal (dê a soma dos atributos). Máximo 187 pontos | |

AVALIAÇÃO FINAL

| Pontuação | Classificação |
|--------------|----------------|
| De 00 a 75 | Insatisfatório |
| De 75 a 115 | Regular |
| De 115 a 150 | Bom |
| De 150 a 187 | Excelente |