

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
BACHARELADO EM ENGENHARIA ELÉTRICA**

**ÉRIKA DUARTE DE SOUZA
MAURICI ONESSEKEN**

**QUAL O MÉTODO PARA A ESCOLHA DOS PROGRAMAS MAIS
ADEQUADO NO ENSINO DE ENGENHARIA ELÉTRICA**

TRABALHO DE CONCLUSÃO DE CURSO

**PONTA GROSSA
2020**

**ÉRIKA DUARTE DE SOUZA
MAURICI ONESSEKEN**

**QUAL O MÉTODO PARA A ESCOLHA DOS PROGRAMAS MAIS
ADEQUADO NO ENSINO DE ENGENHARIA ELÉTRICA**

Trabalho de Conclusão de Curso apresentado(a) como requisito parcial à obtenção do título de Bacharel(a) em Engenharia Elétrica, do Departamento Acadêmico de Eletrônica, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Josmar Ivanqui

**PONTA GROSSA
2020**



Ministério da Educação
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
 DEP. ACADEMICO DE ELETRONICA-PG

TERMO DE APROVAÇÃO

Trabalho de Conclusão de Curso - TCC

QUAL O MÉTODO PARA A ESCOLHA DOS PROGRAMAS MAIS ADEQUADO NO ENSINO DE ENGENHARIA ELÉTRICA

Por

Érika Duarte de Souza e Maurici Onesseken

Monografia apresentada às 15 horas 30 min. do dia 11 de dezembro de 2020 como requisito parcial, para conclusão do Curso de Engenharia Elétrica da Universidade Tecnológica Federal do Paraná, Câmpus Ponta Grossa. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação e conferidas, bem como achadas conforme, as alterações indicadas pela Banca Examinadora, o trabalho de conclusão de curso foi considerado APROVADO.

Banca examinadora:

Prof. Jeferson José Gomes	Membro
Profª. Marcella Scoczynski Ribeiro Martins	Membro
Prof. Josmar Ivanqui	Orientador e Professor responsável TCCII



Documento assinado eletronicamente por (Document electronically signed by) **MARCELLA SCOCZYNSKI RIBEIRO MARTINS, PROFESSOR DO MAGISTERIO SUPERIOR**, em (at) 11/12/2020, às 16:35, conforme horário oficial de Brasília (according to official Brasilia-Brazil time), com fundamento no (with legal based on) art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por (Document electronically signed by) **JEFERSON JOSE GOMES, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em (at) 11/12/2020, às 16:36, conforme horário oficial de Brasília (according to official Brasilia-Brazil time), com fundamento no (with legal based on) art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por (Document electronically signed by) **JOSMAR IVANQUI, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em (at) 11/12/2020, às 16:48, conforme horário oficial de Brasília (according to official Brasilia-Brazil time), com fundamento no (with legal based on) art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site (The authenticity of this document can be checked on the website) https://sei.utfpr.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador (informing the verification code) **1796310** e o código CRC (and the CRC code) **D3763598**.

Dedico este trabalho as nossas famílias,
namorado e namorada, e aos meus
amigos, pelos nossos momentos de
ausência.

AGRADECIMENTOS

Este trabalho não poderia ser terminado sem a ajuda de diversas pessoas e a esta instituição às quais presta-se a homenagem. Certamente esses parágrafos não irão atender a todas as pessoas que fizeram parte dessa importante fase. Portanto, desde já pedimos desculpas àquelas que não estão presentes entre estas palavras, mas que elas estejam certas de que fazem parte dos nossos pensamentos e de nossa gratidão.

As nossas famílias, pelo carinho, incentivo e total apoio em todos os momentos.

Um agradecimento especial ao nosso orientador, que nos mostrou os caminhos a serem seguidos, pela confiança depositada e pela atenção que sempre nos foi dada neste período difícil.

Aos professores e colegas do departamento de engenharia elétrica, que ajudaram de forma direta e indireta na conclusão deste trabalho.

Enfim, a todos os que de alguma forma contribuíram para a realização deste trabalho.

Os engenheiros têm uma
responsabilidade coletiva para melhorar a
vida das pessoas a nível planetário.”
(QUADRADO, 2013).

RESUMO

SOUZA, Érika Duarte de; ONESSEKEN, Maurici. **Qual o método para a escolha dos programas mais adequado no ensino de Engenharia Elétrica.** 2020. 83 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Elétrica) – Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2020.

O crescimento da utilização de *softwares* no mercado de Engenharia Elétrica fez com que os cursos desta área se especializassem. E também, utilizassem os *softwares* requeridos pelo mercado para a formação de profissionais mais preparados. Assim, as empresas desenvolvedoras desses *softwares* se preocupam em criar licenças e materiais para a didática e os docentes tem como responsabilidade definir qual será o programa utilizado. O presente trabalho apresenta métodos de avaliação e seleções de *softwares* para o âmbito educacional, considerando quatro *softwares* largamente utilizados na Engenharia Elétrica: Proteus, MatLab, AutoCAD e OrCAD. Foram selecionadas várias metodologias utilizadas para o ensino dessas ferramentas e os tipos de *softwares* educacionais. Este estudo utilizou-se de três métodos: Banks, Davis-Williams e Oliveira e dois questionários: Nokelainen e Ssemugabi para serem aplicados aos *softwares*, com o objetivo de auxiliar os docentes na escolha de um método adequado para escolha de um *software* em sua disciplina. Aplicando os *softwares* nos métodos chegamos à conclusão que o método Davis-Williams, que utiliza cálculo, é mais objetivo, mesmo que possibilite uma escolha de comparações, os cálculos realizados permitem uma avaliação para que o método não se torne falho.

Palavras-chave: Programa. Engenharia Elétrica. Didática. Educacional. Método.

ABSTRACT

SOUZA, Érika Duarte de; ONESSEKEN, Maurici. **What is the method for choosing the most suitable programs in the teaching of Electrical Engineering.** 2020. 83 p. Final Coursework (Bachelor's Degree in Electrical Engineer) – Federal University of Technology – Paraná. Ponta Grossa, 2020.

The growth in the use of software in the Electrical Engineering market has made courses in this area specialize. And also, use the software required by the market to train more prepared professionals. Therefore, the companies that develop these software are concerned with creating licenses and materials for didactics and teachers have the responsibility to define which program will be used. The present work presents methods of evaluation and selection of software for the educational scope, considering four software widely used in Electrical Engineering: Proteus, MatLab, AutoCAD and OrCAD. Various methodologies used for teaching these tools and the types of educational software were selected. This study used three methods: Banks, Davis-Williams and Oliveira and two Questionnaires: Nokelainen and Ssemugabi to be applied to software, with the objective of assisting teachers in choosing an appropriate method for choosing software in their discipline. Applying the software in the methods we came to the conclusion that the Davis-Williams method, which uses calculations, is more objective, even though it allows a choice of comparisons, the performed calculations allow an evaluation so that the method does not become flawed.

Keywords: Software. Electrical Engineering. Didactics. Educational. Method.

LISTA DE ILUSTRAÇÕES

Figura 1 – Comparação entre as duas metodologias de aula:	20
Figura 2 – <i>Software</i> educacional.	22
Figura 3 – Faixas de análise.	23
Figura 4 – Método de análise <i>Analytic Hierarchy Process</i> (AHP).	27

LISTA DE TABELAS

Tabela 1 – Soma dos atributos	25
Tabela 2 – Avaliação final	25
Tabela 3 – Escala de julgamentos para comparação paritária	27
Tabela 4 – Método de Oliveira - resultado final	35
Tabela 5 – Método de Banks - resultado final.	42
Tabela 6 – Matriz de comparação - critérios de 1 a 7.	43
Tabela 7 – Matriz de comparação - critérios de 8 a 14.	43
Tabela 8 – Matriz normalizada - critérios de 1 a 8.	44
Tabela 9 – Matriz normalizada - critérios de 9 a 14 e percentual.	44
Tabela 10 – Matriz consistência - critérios de 1 a 8.	45
Tabela 11 – Matriz consistência - critérios de 9 a 14 e soma dos pesos.	46
Tabela 12 – Cálculo de λ_{max}	46
Tabela 13 – Índice randômico	47
Tabela 14 – Resultado final.	47
Tabela 15 – Sensibilidade I - Mesmo peso.	48
Tabela 16 – Sensibilidade II - Mesmo peso.	48
Tabela 17 – Comparação entre os três métodos.	50
Tabela 18 – Apresentação e funcionalidade do produto - aplicado	58
Tabela 19 – Confiabilidade - aplicado	58
Tabela 20 – Eficiência - aplicado	59
Tabela 21 – Manutenibilidade - aplicado	59
Tabela 22 – Portabilidade - aplicado	59
Tabela 23 – Conteúdo do <i>software</i> , relacionado com a disciplina I - aplicado	60
Tabela 24 – Conteúdo do <i>software</i> , relacionado com a disciplina II - aplicado	60
Tabela 25 – Método de Banks - aplicado.	61
Tabela 26 – Matriz comparação aplicada.	62
Tabela 27 – Matriz normalizada aplicada.	64
Tabela 28 – Cálculo de consistência aplicado.	66
Tabela 29 – Cálculo de consistência estratificado aplicado.	68
Tabela 30 – Apresentação e funcionalidade do produto	71
Tabela 31 – Confiabilidade	71
Tabela 32 – Eficiência	71
Tabela 33 – Manutenibilidade	72
Tabela 34 – Portabilidade	72
Tabela 35 – Conteúdo do <i>software</i> relacionado com a disciplina parte I	72
Tabela 36 – Conteúdo do <i>software</i> relacionado com a disciplina parte II	73
Tabela 37 – Método de Banks.	74
Tabela 38 – Exemplo de cinco <i>softwares</i> aplicados no método Davis-Williams.	75
Tabela 39 – Adaptação dos cinco <i>softwares</i> aplicados no método Davis-Williams.	75
Tabela 40 – Questionário de Nokelainen.	76
Tabela 41 – Questionário de Ssemugabi.	80

LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

SIGLAS

EUA	Estados Unidos da América
IEEE	<i>Institute of Electrical and Electronic Engineers</i>
MEC	Ministério de Educação

ACRÔNIMOS

AHP	<i>Analytic Hierarchy Process</i>
AIA	Ambientes Interativos de Aprendizagem
CAD	<i>Computer Aided Desing</i>
CAI	<i>Computer Assisted Instruction</i>
IC	Índice de Consistência
IR	Índice Randômico
ITS	<i>Intelligent Tutoring System</i>
PCI	Placa de Circuito Impressa
RC	Razão de Consistência
SDT	<i>Schematic Desing Tools</i>
VP	Vetor de prioridade

SUMÁRIO

1	INTRODUÇÃO	13
1.1	PROBLEMA	15
1.2	HIPÓTESE	15
1.3	OBJETIVOS	15
1.3.1	Objetivo geral	15
1.3.2	Objetivos específicos	15
1.4	JUSTIFICATIVA	16
1.5	METODOLOGIA	16
2	REVISÃO DA LITERATURA	17
2.1	HISTÓRIA DO USO DA INFORMÁTICA E DE <i>SOFTWARES</i> NO ENSINO	17
2.2	METODOLOGIAS DE ENSINO RELACIONADA AOS <i>SOFTWARES</i>	18
2.2.1	Modelagem e simulação	19
2.3	TIPOS DE <i>SOFTWARES</i> EDUCACIONAIS	20
2.3.1	Classificação dos <i>softwares</i> por objetivos	21
2.3.2	Classificação dos <i>softwares</i> por aprendizagem	22
2.4	MÉTODOS DE AVALIAÇÃO E SELEÇÃO DE UM <i>SOFTWARE</i> DIDÁTICO	23
2.4.1	Método avaliativo e seletivo Oliveira (2001)	24
2.4.2	Método avaliativo e seletivo Banks (1991)	25
2.4.3	Método Davis-Williams	26
2.4.4	Questionário de Nokelainen	27
2.4.5	Questionário de Ssemugabi	28
2.5	OS <i>SOFTWARES</i> MAIS UTILIZADOS NO ENSINO DA ENGENHARIA ELÉTRICA	29
2.5.1	Proteus	29
2.5.2	MatLab	30
2.5.3	AutoCAD	30
2.5.4	OrCAD	31
3	MATERIAL E MÉTODOS	32
3.1	<i>SOFTWARES</i> APLICADOS NO MÉTODO OLIVEIRA	32
3.2	<i>SOFTWARES</i> APLICADOS NO MÉTODO BANKS	35
3.2.1	Proteus	40
3.2.2	MatLab	41
3.2.3	AutoCAD	42
3.2.4	OrCAD	42
3.3	<i>SOFTWARES</i> APLICADOS NO MÉTODO DE DAVIS-WILLIAMS	42
4	RESULTADOS E DISCUSSÃO	50
5	CONCLUSÕES E PERSPECTIVAS	51
5.1	CONCLUSÃO	51

5.2	TRABALHOS FUTUROS	52
	REFERÊNCIAS	54
	APÊNDICES	57
	APÊNDICE A – MÉTODO DE OLIVEIRA - APLICADO	58
	APÊNDICE B – MÉTODO DE BANKS - APLICADO	61
	APÊNDICE C – MÉTODO DE DAVIS-WILLIAMS - APLICADO . . .	62
	ANEXOS	70
	ANEXO A – MÉTODO DE OLIVEIRA	71
	ANEXO B – MÉTODO DE BANKS	74
	ANEXO C – MÉTODO DE DAVIS E WILLIAMS	75
	ANEXO D – QUESTIONÁRIO DE NOKELAINEN	76
	ANEXO E – QUESTIONÁRIO DE SSEMUGABI	80

1 INTRODUÇÃO

A importância da utilização de *softwares* no processo de ensino e de aprendizagem se torna cada dia mais evidente, tanto pela facilidade de uso, precisando apenas de um computador, quanto pela importância dos alunos estarem de fato familiarizados com estes, como o mercado de trabalho irá cobrá-los para tal, nesse aspecto já se dá um excelente motivo para que as aulas explorem a utilização de *softwares* para o processo de aprendizagem.

Segundo Silva (2019), com a crescente globalização e as novas descobertas tecnológicas, o mercado de trabalho da engenharia vem se modificando ao decorrer dos anos, tornando-se mais competitivo, diferenciado e irreverente. O engenheiro deve estar pronto para ser um ser humano capaz de realizar múltiplas facetas, compreender, aceitar, defender e melhorar a sua percepção tecnológica.

O perfil do engenheiro é montado na universidade, porém é preciso compreender de que os anos despendidos de estudo no curso superior, não passam de uma primeira fase de um processo contínuo de aprendizagem. Assim, os cursos de engenharia estão em um processo contínuo de atualização, voltado para perspectivas melhores a serem oferecidas à sociedade, formando profissionais capacitados para o mercado tecnológico, conforme Silva (2019).

Essa crescente necessidade de um profissional mais qualificado e antenado com as tecnologias exigidas pelo mercado, fez com que as universidades se preocupassem no ensino de *softwares* que servem como ferramenta no dia a dia do engenheiro. Além da utilização de *softwares* para auxílio do domínio do conteúdo ministrado. Assim as aulas tem cada vez mais se passado na frente de um computador e uma avaliação de quais *softwares* são didáticos para o entendimento do discente é fundamental.

Outra justificativa para a aplicação de *softwares* educacionais na Engenharia Elétrica é a simulação de um experimento de laboratório. Uma das disciplinas que exige um *software* de simulação é Microcontroladores. Neste curso, há a necessidade de computadores e *kits* práticos, que são placas com microcontroladores e periféricos que tem como objetivo exemplificar fisicamente o que foi programado no *software* de compilação. Em um laboratório, geralmente não há a quantidade necessária de *kits* didáticos para todos os alunos, o que dificulta a afirmação se a programação feita foi correta. Para solucionar esse problema são utilizados os *softwares* de simulação, o mais usual sendo o Proteus. Utilizando em conjunto os *softwares* de compilação e simulação, cria-se um laboratório virtual.

Segundo Junior e Coutinho (2007), a criação dos laboratórios virtuais surgiu da necessidade do uso do laboratório em tempo real, com acesso a qualquer momento

e por um grande número de pessoas. Também é um ótimo recurso pré laboratorial, pois o discente antes de ir para o laboratório real, pode fazer todos os testes virtualmente, fazendo com que os conhecimentos sejam melhor fixado e evitando possíveis inconvenientes que possam ocorrer por má utilização dos equipamentos.

Segundo Sanusb (2011), considerado o aumento dos discentes, os laboratórios também precisam ser aumentados, mas os custos dos *kits* didáticos são altos, o que não permite que cada aluno tenha o seu próprio kit para estudar. Assim, é mais viável a utilização de *softwares* para que haja o desenvolvimento individual do aluno.

Segundo Su e Wang (2010), o curso de microcontroladores é técnico e muito abrangente, os experimentos são a parte mais importante do processo. Por motivos de falta de recurso financeiros e *hardware*, os discentes não fazem testes o suficiente.

Como cita em seu artigo, Oliveira (2001) diz que um *software* educacional se trata de um produto, mas que deve ser utilizado adequadamente pela instituição de ensino indiferente deste ter sido ou não criado para um ambiente escolar, e desta forma auxiliará no processo de ensino e aprendizagem. Valendo destacar que para esta utilização de forma adequada o profissional fará uso do mesmo deve ter uma forma de selecionar um *software* sem ter que testar de cabo a rabo todos os que lhe são ofertados.

Com a evolução das tecnologias computacionais, os docentes tem que assimilar o impacto dessas tecnologias no processo de ensino e aprendizagem. A utilização dessas ferramentas tem mostrado um ótimas vantagens e um excelente rendimento dos discentes conforme afirma Mariani e Martim (2003).

Para Oliveira (2001), a utilização do *software* no ensino deve ser implementada quando este oferece vantagens em relação à outros recursos, porém não devem ser aplicados por modismo, assim o discente assume mais responsabilidades no processo de aprendizagem, isso faz com que aumente o estímulo de pesquisas por meio da interação com a máquina.

A consequência do desenvolvimento de *softwares* na educação faz com que os discentes da graduação obtenham um aproveitamento otimizado durante o curso. Este aproveitamento é quantitativo, pois há um melhor entendimento de alguns fenômenos físicos que são recorrentes na engenharia, isto é apresentado em diversos segmentos do conhecimento Silva, Lima e Ferreira (2004).

A pandemia do *SARS COV-19* impossibilitou a presença dos alunos às salas de aula, sendo uma realidade as aulas a distância, desta forma, algumas disciplinas que contavam com as aulas práticas tiveram que se adaptar a este cenário. Uma dessas disciplinas do curso de engenharia elétrica foi a de microcontroladores. Uma solução é a utilização de um *software* de simulação para suprir as aulas de laboratório.

Isso ressaltou ainda mais a importância do ensino no ambiente virtual, como *softwares* de simulação de projetos, possibilitando o continuo aprendizado dos acadê-

micos de forma remota, fortalecendo a capacidade de ensino em momentos de crise como o que estamos vivendo.

Com essa demanda, foram estudados métodos e questionários que classificam e selecionam *softwares* educacionais, com objetivo de auxiliar os docentes na definição de *softwares* para engenharia elétrica. Para este trabalho, definimos o Proteus, MatLab, AutoCAD e OrCAD, por serem alguns dos *softwares* largamente utilizados nas disciplinas técnicas da Engenharia Elétrica. A avaliação de mais *softwares* não se fez necessária, pois esses quatro *softwares* são suficientes para compreender a dinâmica dos métodos e questionários aplicados. Esses métodos propõem um resultado comparativo, mas isso não se aplicará ao trabalho, pois o objetivo deste trabalho é mostrar como esses métodos funcionam e como *softwares* de diferentes áreas reagem a eles.

1.1 PROBLEMA

Especificar um método a fim de optar pelos *softwares* mais adequados no ensino de Engenharia Elétrica

1.2 HIPÓTESE

Existe a possibilidade de escolher *softwares* mais adequados ao curso de engenharia elétrica, dentro dos métodos existentes, conhecidos e dominados dentro do escopo científico, analisando o perfil didático pedagógico do nosso discente.

1.3 OBJETIVOS

1.3.1 Objetivo geral

Analisar os *softwares* mais adequados no processo ensino-aprendizagem, e na melhoria do trabalho docente de ensino. Pretende estreitar a relação teoria-prática e, assim, identificar as necessidades dos docentes para a adoção de *softwares* educacionais, voltados ao ensino.

1.3.2 Objetivos específicos

- Analise bibliográfica sobre métodos existentes para avaliação de *software*;
- Identificar os *softwares* mais utilizados no curso de Engenharia Elétrica;

- Analisar o resultado dos *software* nas mais diversas situações;
- Aplicar os métodos encontrados nos *softwares* de engenharia; e
- Análise e comparação de resultado para verificar a viabilidade de aplicação no curso de Engenharia Elétrica.

1.4 JUSTIFICATIVA

Colaborar com os docentes na efetiva escolha do *software* mais adequado, a fim de facilitar ao acadêmico o processo ensino-aprendizado.

1.5 METODOLOGIA

Este trabalho foi realizado utilizando embasamento teórico existente na literatura, com o objetivo de classificar e selecionar os *softwares* pela didática na Engenharia Elétrica. Estes materiais foram encontrados em acervos e repositórios de universidades, revistas acadêmicas e acervo do *Institute of Electrical and Electronic Engineers* (IEEE), essa busca resultou em artigos e trabalhos acadêmicos que foram utilizados como embasamento, recorrendo ao estudo de três métodos e dois questionários como método de avaliação e seleção dos *softwares* mais adequados. Os questionários não foram utilizados, por não serem adequados aos *softwares* estudados, assim, utilizamos apenas os três métodos para avaliar os *softwares*, contudo os questionários podem ser analisados nos Anexo D e Anexo E. Os principais autores estudados foram: Banks, Davis e Williams e Oliveira, onde seus métodos serão explicados em seção 2.4. Os *softwares* avaliados nesse estudo foram: Proteus, MatLab, AutoCAD e OrCAD, escolhidos pelos autores deste trabalho como os *softwares* mais utilizados no curso de Engenharia Elétrica. Foram criadas planilhas com todos os métodos utilizados para facilitar o trabalho com dados, em seguida foi analisado cada método critério a critério em relação aos *softwares* a serem avaliados.

2 REVISÃO DA LITERATURA

2.1 HISTÓRIA DO USO DA INFORMÁTICA E DE *SOFTWARES* NO ENSINO

A utilização de *softwares* e informática são ferramentas relativamente novas no ensino. Estas tecnologias tem evoluído e popularizado nas últimas décadas, o que fez com que a inserção delas como ferramenta didática fosse algo orgânico e necessário no ramo acadêmico.

Segundo Valente (1996), o termo “Informática na Educação” refere-se à inserção do computador no processo de ensino-aprendizagem de conteúdos curriculares de todos os níveis e modalidades de Educação. O interesse dos educadores brasileiros em algumas Universidades desencadeou o começo da informática na educação brasileira, isso foi incentivado pois esta prática já ocorria em outros países, como Estados Unidos e França.

“Diferentemente do que aconteceu na França e nos Estados Unidos, as políticas e propostas pedagógicas da Informática na Educação, no Brasil, sempre foram fundamentadas em pesquisas realizadas nas Universidades, em função das escolas da rede pública.” Valente (1996)

Pode ser notado que a bastante tempo o Brasil já vem estudando, aprendendo e de forma conjunta criando um método, de fato, para que esta implementação ocorra de forma gradual e científica.

Nos Estados Unidos, o uso de computadores no ensino é descentralizado e não depende das decisões governamentais. O uso de computadores no ensino é pressionado pelo desenvolvimento tecnológico, pela necessidade de profissionais qualificados e pela competição estabelecida pelo mercado das empresas que produzem *softwares* das Universidades e das escolas, como descrito em Valente (1999).

Na França, o uso de computadores no ensino é centralizado. Isso ocorreu por terem uma forte identidade cultural e um estado centralizador e planejador. Uma exemplificação é a interação da indústria, comércio, cultura e saúde com a rede escolar. No início da implementação, final dos anos 60, o questionamento era se o aluno iria dominar a Informática ou aprender por meio dela. Conforme cita Valente (1999), o objetivo de introduzir a informática na França não era de provocar mudanças de ordem pedagógicas.

No Brasil, a utilização do computador na educação teve início em Universidades, no começo da década de 70. A diferença entre o programa de informática na educação do Brasil e da França e Estados Unidos é a relação que se estabeleceu entre os órgãos de pesquisa e a escola pública. Na França, as medidas do adotadas pelo governo não vieram da pesquisa e não houve uma ligação entre os centros de

pesquisa e a escola pública. Nos Estados Unidos da América (EUA) foram produzidas diversas pesquisas, porém estas podiam ou não ser adotadas pelas escolas, dependia apenas do interesse da escola em implementar a informática. No Brasil, as decisões e propostas sobre informática no ensino não é centralizada no Ministério de Educação (MEC). Estas têm ocorrido por meio de discussões e propostas feitas pela comunidade técnica e de pesquisadores da área. Cabe ao MEC acompanhar, viabilizar e implementar as decisões. Isso difere da França, que tem todas as decisões tomadas somente pelo governo e também difere dos EUA, que possui uma demanda direta do mercado Valente (1999).

A introdução da informática na educação no Brasil foi influenciada pelos acontecimentos da França e dos Estados Unidos, mas isso ocorreu no sentido de minimizar os pontos negativos e enfatizar os pontos positivos em vez de servir como modelo para uma reprodução.

2.2 METODOLOGIAS DE ENSINO RELACIONADA AOS *SOFTWARES*

Na Engenharia Elétrica, a utilização de *softwares* não é aplicada como recurso alternativo aos livros. Nesse ensino, os *softwares* são aplicados como recursos necessários para o desenvolvimento do profissional. Isso não diminui o fato de que os softwares são apresentados dentro de sala de aula e ainda precisam ser ensinados de maneira didática. Dessa forma, isto faz com que o discente seja responsável por parte do aprendizado, já que este estará interagindo diretamente com o *software*.

“A mudança pedagógica que todos almejam é a passagem de uma Educação totalmente baseada na transmissão da informação, na instrução, para a criação de ambientes de aprendizagem nos quais o aluno realiza atividades e constrói o seu conhecimento.” (MEC)

Segundo Valente (1999), a aprendizagem pode ocorrer de duas maneiras, a informação é memorizada ou é processada pelos esquemas mentais. Neste caso, o conhecimento é construído.

Segundo Baranauskas et al. (1999), o aprendizado é a construção individual do conhecimento a partir de atividades de exploração, investigação e descoberta. Existem os programas *Computer Assisted Instruction* (CAI), similares aos livros, onde o discente terá acesso à informação por meio de módulos, que são apresentados ao estudante de forma gradual e sequencial. Também existem os programas *Intelligent Tutoring System* (ITS), em que o discente tem acesso à informação, porém também recebe um *feedback* de sua interação com o programa. E existem os programas que implementam a aprendizagem construcionista, chamamos de Ambientes Interativos de Aprendizagem (AIA).

As AIA incluem:

- Construção e não instrução;
- Controle do estudante e não controle do sistema;
- Individualização é determinada pelo estudante e não pelo sistema;
- *Feedback* rico, gerado a partir da interação do estudante com o ambiente de aprendizagem e não pelo sistema.

Exemplos dessa classe são: sistemas de modelagem e simulação, micro-mundos, o uso de linguagens de programação e sistemas de autoria.

2.2.1 Modelagem e simulação

É uma técnica usada para estudar o comportamento de fenômenos reais. Este processo de modelar um fenômeno real ou hipotético para observar/analisar seu comportamento consiste em três fases:

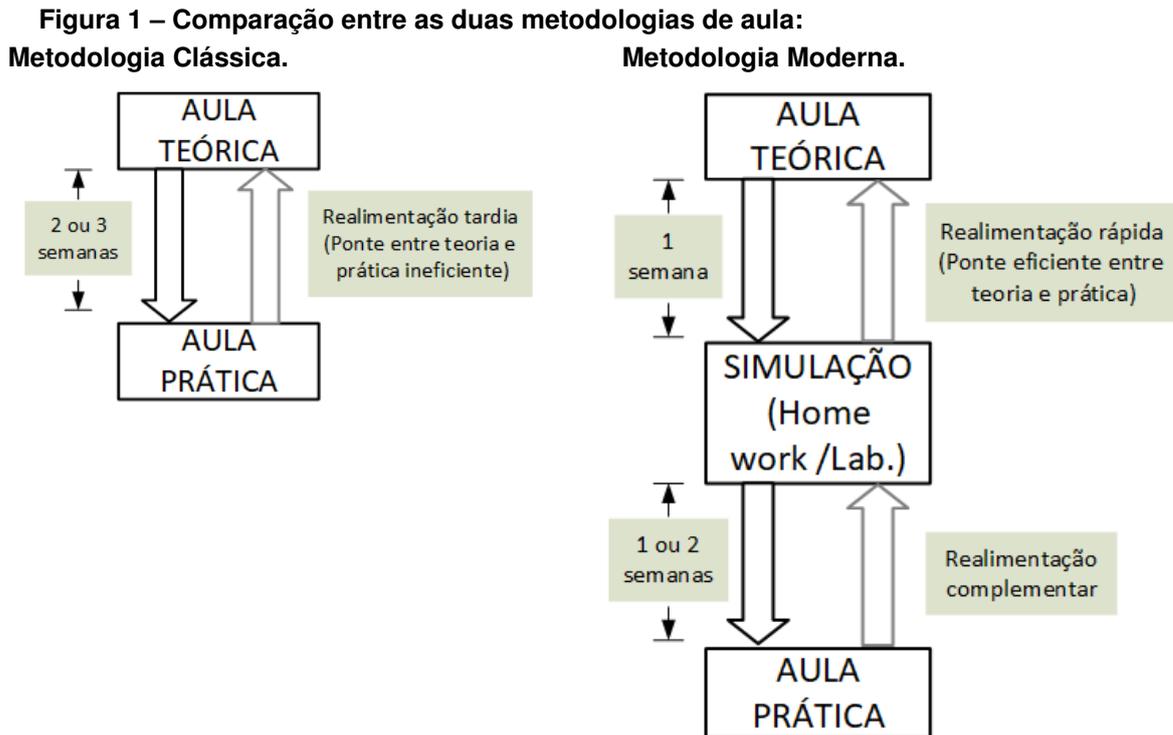
- Construção de um modelo que represente aspectos relevantes do sistema a ser estudado;
- Experimentação e análise do modelo criado;
- Comparação do modelo construído com sistemas reais.

Existem sistemas puramente de simulação, neles, há um modelo embutido e o usuário apenas altera a entrada do modelo e observa/analisa os resultados da simulação. A diferença entre os sistemas de simulação e modelagem é quem escolhe o fenômeno a estudar e quem desenvolve o seu modelo. Na simulação, isso já está feito e é apenas fornecido ao usuário. Na modelagem, é o usuário que escolhe o fenômeno, desenvolve o modelo e implementa. Um bom exemplo disso na Engenharia é o Proteus e o PicSimLab, sendo o Proteus um *software* de modelagem e o PicSimLab, de simulação.

Sistemas de modelagem tendem a ser ótimos ambientes de aprendizagem, por envolverem o usuário no ciclo básico de expressão, avaliação e reflexão sobre o domínio considerado. Este tipo de sistema tem boa visibilidade no desenvolvimento de habilidades em resolução de problemas, tomadas de decisão, apresentação e comunicação de entendimento e do conhecimento que o estudante tem em um dado domínio.

Segundo Silveira et al. (1998), a metodologia de ensino utilizada nos cursos de engenharia, é baseada, em sua maior parte, em duas etapas: teoria e prática. Conforme pode ser observado na Figura 1. A teoria é caracterizada por utilizar textos que apresentam uma divisão tradicional dos tópicos, encontrada em livros didáticos.

O ensino da prática segue a sequência cronológica da teoria. São necessárias muitas aulas práticas até que o discente adquira uma experiência de bancada, como manusear equipamentos de medida e dominar a depuração de ligação de componentes em uma matriz de contato, por exemplo. Isto ocasiona uma defasagem entre o conteúdo teórico ministrado e o observado em laboratório, impedindo que o discente faça a conexão do conteúdo ministrado.



Fonte: Silveira et al. (1998).

2.3 TIPOS DE *SOFTWARES* EDUCACIONAIS

Nas grades curriculares das engenharias existem disciplinas muito diversificadas, assim, também existem variados tipos de *softwares* que podem ser utilizados. Oliveira (2001) descreve que a informática é um mercado amplo e para classificar de maneira geral, há onze tipos de *softwares* educacionais.

Segundo Tavares (2017), a bibliografia apresenta tipos diversificados de *softwares* educacionais. Segundo Teixeira e Brandão (2003), *software* educacional é aquele que pode ser usado com algum objetivo educacional, pedagogicamente defensável, por professores e alunos, não importando o objetivo com que este foi criado. Ela então classifica por objetivos e por aprendizagem.

2.3.1 Classificação dos *softwares* por objetivos

“Cada um dos diferentes softwares usados na Educação, como os tutoriais, a programação, o processador de texto, os softwares multimídias, as simulações, modelagens e jogos, apresenta características que podem favorecer o processo de construção do conhecimento.”Valente (1999).

Na literatura, especificamente em Oliveira (2001) é encontrada várias maneiras de classificar os *softwares* educacionais, neste trabalho vamos listar uma metodologia que os classifica em oito categorias.

- **Tutorial:** são semelhantes aos livros, no quesito de apresentar a informação, porém conseguem proporcionar um *feedback* imediato e avaliam o desempenho.
- **Exercício e Prática:** são utilizados para revisão de conteúdo. Na maior parte, são apresentados em forma de jogos, para fixação de um conteúdo de forma repetitiva.
- **Simulação:** permite a participação do aluno em atividades que normalmente ele não poderia participar. O discente pode testar, tomar decisões, analisar, sintetizar e aplicar conhecimentos.
- **Software de programação:** são *softwares* que permitem a programação e compilação de um determinado programa, há uma vasta gama de tipos de linguagem. Essas linguagens são acessíveis para o entendimento do ser humano e também mais perto do tipo de linguagem do computador.

Aplicativos: são *softwares* que aplicam em atividades específicas como: processadores de texto, planilhas eletrônicas e de apresentação.

Investigação: são *softwares* com capacidade de localizar informações complementares.

Jogos: são *softwares* que ensinam sobre determinado assunto/conteúdo de forma lúdica.

Modelagem: são *softwares* que permitem o usuário criar o modelo.

Assim que estiver implementado o modelo, o usuário pode usar o *software* como um de simulação. Tavares (2017), propõe a Figura 2 para deixar mais claro as informações passadas em texto.

Figura 2 – *Software* educacional.



Fonte: Tavares (2017).

2.3.2 Classificação dos *softwares* por aprendizagem

Para Martins (2002), os *softwares* educacionais apresentam ênfase no processo de ensino aprendizagem e também são classificados de acordo com o nível de aprendizagem que cada *software* é capaz de proporcionar.

São eles:

- Sequencial: esses *softwares* transferem a informação para o usuário de maneira sequencial. Exemplo: *softwares* de exercício e prática, aplicativos e tutoriais.
- Relacional: esses *softwares* permitem o desenvolvimento de habilidades e interação com a tecnologia, isso significa que o aprendizado ocorre por meio da interação do usuário com a tecnologia. Exemplo: *software* de investigação.
- Criativo: esses *softwares* permitem a criação de esquemas mentais, exigindo um usuário participativo que acaba desenvolvendo a criatividade. Exemplo: *software*

de simulação.

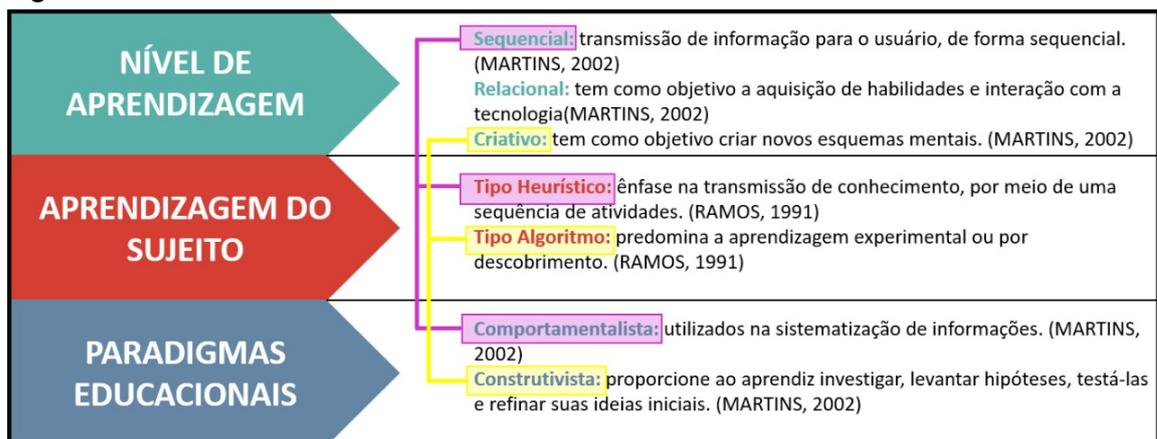
Para Ramos (1991) a atividade do sujeito de aprendizagem é dividida em dois grupos. O primeiro sendo o *software* com enfoque do tipo algoritmo, este trata-se de *softwares* que possuem como objetivo a simples transmissão de conteúdo, por meio de uma sequência de atividades. Como por exemplo *softwares* tutoriais e de exercício e prática. O segundo grupo são os *softwares* com enfoque do tipo heurístico, onde o objetivo é a aprendizagem experimental ou por descobrimento. Como por exemplo os *softwares* de simulação e os jogos educacionais.

Na literatura já foi abordado também os paradigmas educacionais dos *softwares*. Há afirmações sobre o paradigma comportamentalista que afirmam que os discentes possuem mais facilidade de aprender quando o conteúdo é apresentado em seções curtas, testando o usuário após cada seção e apresentando o *feedback* instantâneo para as respostas. Isso resultaria em discentes passivos, que não podem utilizar outra alternativa além da determinada pelo *software*. Porém, estes tipos de *softwares* podem ser utilizados na sistematização de informações como afirma Tavares (2017).

O outro paradigma é o construtivista. Na literatura há indicações de que a atividade de programação permite ao discente a resolução de problemas que envolvam abstrações, aplicação de estratégias, estruturas e conceitos já construídos ou a criação delas. Ramos (1991).

A Figura 3 proposta por Tavares exemplifica esses conceitos.

Figura 3 – Faixas de análise.



Fonte: Tavares (2017).

2.4 MÉTODOS DE AVALIAÇÃO E SELEÇÃO DE UM SOFTWARE DIDÁTICO

Há diversos métodos de avaliação e seleção de *softwares* didáticos, neste capítulo citaremos os que achamos mais relevantes na literatura.

2.4.1 Método avaliativo e seletivo Oliveira (2001)

Para Oliveira (2001), as conquistas tecnológicas de *hardware* e *software*, levaram os docentes a reverem os seus conceitos sobre a educação e a fazerem uma análise sobre o papel dessas tecnologias e as vantagens que elas podem trazer para o ensino. Porém, há a necessidade de utilizar um método de avaliação para a seleção desses *softwares*, tendo em vista o contexto pedagógico.

Este tipo de análise é de suma importância, pois como é apresentada na literatura, já ocorreram erros de programação que custaram a vida de pessoas ou prejuízos financeiros. Este fator de qualidade se salienta ainda mais quando direcionado para o ambiente educacional. No desenvolvimento de *softwares* educacionais há alguns requisitos que devem atender, como: questões culturais, éticas, filosóficas e psicopedagógicas.

Oliveira (2001) lista quais características devem ser aplicadas à um *software*. São elas: conjunto de multimídias que permitam a interatividade disponibilizada pelos sistemas, a agilidade em acessar a informação, a contextualização das informações, a adequação das ferramentas aos processos metacognitivos e a mobilidade de adaptação dos sistemas aos usuários. Ele também lista as preocupações que uma equipe de desenvolvedores deve levar em consideração. São elas: a construção de caminhos mais úteis, antecipar necessidades, considerar preferências e a transferência de conhecimento presentes nos processos de metacognição do usuário e também mediar sistemas de representações mais aprimorados.

Segundo Oliveira (2001), há de tomar certos cuidados no desenvolvimento de um programa educacional, tais como: definição de algumas estimativas, entre elas recursos, custos e cronogramas; fornecer dados sobre os usuários, restrições externas, limitações do produto e outros relevantes que influenciam o *hardware* onde será implementado o programa.

Oliveira (2001) propõe um modelo para auxiliar projetistas e educadores nos quesitos de concepção, avaliação e aplicação de *software* educacional. Este modelo é composto de três fichas. São elas: identificação do produto, apresentação e funcionalidade do produto e confiabilidade, usabilidade, eficiência, manutenibilidade, portabilidade, conteúdo do *software* relacionado à disciplina.

Para as fichas técnicas de apresentação e funcionalidade do produto, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade, ele atribui 0, como não, 1, para às vezes e 2, para sim. E para as fichas de conteúdo do *software* relacionado à disciplina, ele atribui 0, como não, 3, para às vezes e 5, para sim.

Conforme podem ser visto no Anexo A este métodos com os critérios distribuídos da Tabela 30 até a Tabela 36 devidamente preenchidas, pode se organizar e finalizar a soma fazendo uso Tabela 1, desta forma, obtendo-se um resultado final e

mesmo pode ser analisado com o auxílio da Tabela 2, ou seja, estabelecendo uma avaliação final em função da pontuação obtida.

Tabela 1 – Soma dos atributos

Característica do software	Atributo
Apresentação e funcionalidade do produto	
Confiabilidade	
Usabilidade	
Eficiência	
Manutenibilidade	
Portabilidade	
Conteúdo do software relacionado com a disciplina I	
Conteúdo do software, relacionado com a disciplina II	
Total (Máximo 187 pontos)	

Fonte: Oliveira (2001).

Tabela 2 – Avaliação final

AVALIAÇÃO FINAL	
Pontuação	Classificação
De 000 a 075	Insatisfatório
De 075 a 115	Regular
De 115 a 150	Bom
De 150 a 187	Excelente

Fonte: Oliveira (2001).

2.4.2 Método avaliativo e seletivo Banks (1991)

O método de Banks (1991) é utilizado para avaliar e selecionar *softwares* de simulação, assim, ele apresenta um conjunto de critérios para a análise de eventos discretos classificados em cinco categorias:

- Critérios de entrada;
- Critérios de processamento;
- Critérios de saída;
- Critérios de suporte;
- Critérios de custo.

Banks comenta a importância de destacar quais critérios são relevantes para cada caso. Assim, o autor sugere a utilização de um modelo de pontuação para de 0 a 10 para ser definido como peso de cada item. Depois deve ser feita uma pontuação de zero a um para cada *software* avaliado, conforme pode ser apurado em Silva (2006). Os itens a serem pontuados por Banks se encontram no Anexo B.

2.4.3 Método Davis-Williams

O método Davis-Williams, assim como o método Banks, foi criado originalmente para avaliar *softwares* de simulação. Ele propõe a utilização do AHP para definição dos pesos dos critérios. Os autores propuseram oito critérios a serem avaliados, com alguns subcritérios.

Mas diferentemente do método Banks, eles não utilizaram o modelo de pontuação, do termo em inglês: *scoring models*, para a pontuação dos pesos. O AHP é um método avaliativo criado por Saaty (1987).

Ao aplicar o AHP, eles pretendem eliminar a inconsistência do julgamento humano que o método de decisão por múltiplos critérios, utilizado em pesos e pontuações, acarreta.

Este método foi criado originalmente para fins socioeconômicos e situações políticas, porém obteve bons resultados em julgamentos de outras áreas, como seleção de *software* de manufatura e programação de *software* como trata Davis e Williams (1994).

Segundo Silva (2006) o método do AHP possui as seguintes características:

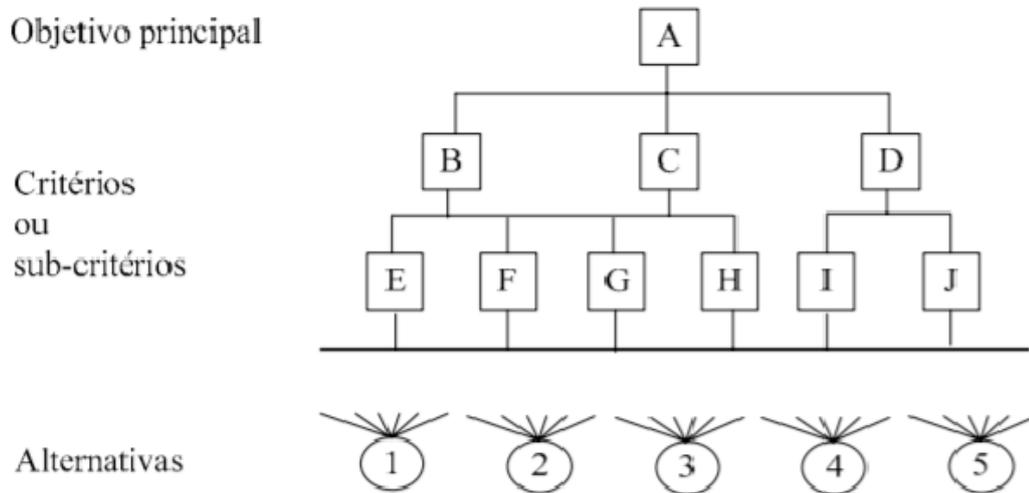
- Aplicação em problemas orientados por múltiplos critérios estruturados hierarquicamente;
- Capacidade de considerar simultaneamente critérios quantitativos e qualitativos;
- Julgamento dos critérios por meio de comparações paritárias;
- Resultado final que permite ordenar os critérios e alternativas em função de sua importância, com graduações (notas, pontuações ou medidas);
- Aplicação em questões complexas ou que envolvam julgamentos subjetivos;
- Resultado que não extrapola os limites do conjunto de alternativas pré-estabelecidas;
- Não obrigatoriedade dos fatores serem representados em termos de valores econômico-financeiros, técnicos ou físicos.

Para a aplicação do AHP, é necessário definir o objetivo principal e os critérios e subcritérios. Como exemplificado na Figura 4.

Saaty (1987) propôs a Tabela 3 para definir os pesos de importância dada à uma característica.

Este método será aplicado no Capítulo 3, onde será feito passo a passo a proposta do AHP.

Figura 4 – Método de análise AHP.



Fonte: Silva (2006).

Tabela 3 – Escala de julgamentos para comparação paritária

Intensidade de importância	Definição
1	Critérios essenciais com mesma importância
3	Fraca importância de um critério essencial sobre o outro
5	Importância forte de um critério essencial sobre o outro
7	Importância muito forte de um critério essencial sobre o outro
9	Importância extrema de um critério essencial sobre o outro
2, 4, 6, 8	Intensidades intermediárias entre valores adjacentes
Recíprocos dos valores acima de zero	Por exemplo, se o critério essencial C_1 recebe uma das designações diferentes acima de zero, quando comparada com o critério essencial C_2 , então C_2 tem o valor recíproco quando comparado com C_1 .

Fonte: Adaptado de Saaty (1987).

Na Tabela 38, Davis e Williams apresentam os resultados da comparação de 5 *softwares* no método desenvolvido por eles, conforme pode ser observado no Anexo C.

2.4.4 Questionário de Nokelainen

Segundo Tavares (2017), Nokelainen desenvolveu um estudo baseado no conceito de usabilidade pedagógica definindo critérios pedagógicos. Ele propôs um questionário composto de 56 questões com os seguintes critérios pedagógicos:

- Controle do aluno: o discente deve possuir controle e liberdade sobre o conteúdo por meio de um *software* que seja dividido em unidades;
- Atividade do aprendiz: o *software* deve propor atividade de solução de problemas e atividades colaborativas;

- Aprendizagem colaborativa/cooperativa: o *software* deve permitir a colaboração e cooperação entre usuários;
- Orientação de metas: o *software* deve proporcionar ao usuário interferir na definição de objetivos, fazendo com que o discente busque seus interesses;
- Aplicabilidade: os *softwares* devem proporcionar conteúdos que correspondam às necessidades do discente;
- Valor agregado: o *software* deve atingir uma série de requisitos, como: adaptabilidade para as necessidades individuais, controle do aprendiz, conteúdo interessante, comunicação e participação ativa dos usuários.
- Motivação: o *software* deve possuir alguns conceitos com que haja uma motivação ao usuário, como: incentivos, autonomia, expectativas, atribuições de sucesso ou fracasso e alcance de objetivos;
- Avaliação de conhecimento anterior: o *software* deve ser flexível em relação à vasta diferença entre as habilidades dos usuários;
- *Feedback*: o *software* deve fornecer *feedbacks* imediatos.

O questionário de Nokelainen está disponível no Anexo D.

2.4.5 Questionário de Ssemugabi

Segundo Tavares (2017), foi elaborado com o objetivo de fazer uma medição da usabilidade técnica e pedagógica. É composto por dez heurísticas para usabilidade técnica, contendo 56 questões e dez heurísticas para usabilidade pedagógica, contendo 53 questões, que estão divididas em três categorias.

- Categoria 1: Critério geral de usabilidade de interface com dez heurísticas que propõe avaliar:
 - Visibilidade do status do sistema;
 - Correlação entre o sistema e o mundo real;
 - Controle do usuário e liberdade;
 - Consistência e aderência às normas;
 - Prevenção de erros;
 - Reconhecimento ao invés de evocação;
 - Flexibilidade e eficiência do uso;

- Desing estético e minimalista;
 - Reconhecimento, diagnóstico e recuperação de erros;
 - Ajuda e documentação.
- Categoria 2: Critério específico para aplicações web educacionais, com duas heurísticas que propõe avaliar:
 - Simplicidade de navegação, organização e estrutura;
 - Relevância do conteúdo do site para a aprendizagem.
 - Categoria 3: Critério específico sobre design com foco em aprendizado, com oito heurísticas que propõe avaliar:
 - Claridade das metas;
 - Aprendizagem colaborativa;
 - Aplicabilidade do nível de controle do aluno;
 - Suporte para abordagens significativas de aprendizagem;
 - Identificação de erros cognitivos, diagnósticos e restabelecimento;
 - Feedback, orientação e avaliação;
 - Contexto significativo;
 - Motivação, criatividade e aprendizagem ativa.

Questionário de Ssemugabi está disponível no Anexo E.

2.5 OS *SOFTWARES* MAIS UTILIZADOS NO ENSINO DA ENGENHARIA ELÉTRICA

Visando limitar o escopo do trabalho, foi selecionado, entre vários *softwares* que são utilizados no ensino da Engenharia Elétrica, quatro deles como amostra para aplicação nos métodos. Estes programas possuem objetivos diferentes, por isso foram propositalmente selecionados, assim desta forma, obteríamos uma amostra de como *softwares* de diferentes áreas dentro da Engenharia Elétrica, irão desempenhar nos métodos escolhidos.

2.5.1 Proteus

O Proteus é um *software* desenvolvido pela empresa Labcenter® que é utilizado para criação de escopos de projetos eletrônicos analógicos e digitais na área de

trabalho ISIS. Permite a criação de Placa de Circuito Impressa (PCI) com o programa integrado ARES.

Segundo Su e Wang (2010), utilizando-se de métodos mais tradicionais, o desenvolvimento do *software* e os testes do sistema não podem ser executados antes que haja uma PCI ou um protótipo físico, o que pode atrasar o processo de 2 a 3 semanas. Utilizando o Proteus, há a possibilidade de desenvolver o *software* assim que o esquemático é desenhado nele. E o conjunto *software/hardware* pode ser testado por completo antes da fabricação da PCI.

2.5.2 MatLab

“O MatLab é um sistema gráfico que integra a capacidade de se fazer cálculos, programação e visualização gráfica em um ambiente interativo bastante agradável, onde os problemas e suas soluções são expressos em uma linguagem matemática familiar.” Silva, Lima e Ferreira (2004)

O MatLab[®] tem vasta aplicabilidade na Engenharia, abrangendo o ponto de vista técnico e educacional. Os diversos recursos disponíveis podem ser utilizados na resolução de problemas em diferentes áreas. Também permite que o docente transmita tópicos da disciplina complexos, que são mais difíceis de serem apresentados apenas no quadro. No segmento matemático ele abrange matérias como: cálculo numérico, métodos matemáticos aplicados, álgebra linear e cálculo. Na maior parte dos casos dessas disciplinas, há o conteúdo em sala de aula e listas de exercício. Os discentes resolvem alguns exercícios manualmente e com auxílio de calculadoras gráficas, após isso, eles utilizam o MatLab para a resolução dos exercícios. O MatLab possui vantagens em relação aos outros métodos, a principal sendo a otimização de tempo por não precisar de realizar as tarefas manuais exaustivas e repetitivas, fazendo com que o discente tenha um melhor entendimento físico dos problemas conforme descreve Mariani e Martim (2003).

2.5.3 AutoCAD

Os *softwares Computer Aided Desing (CAD)*, desenho assistido por computador, têm como objetivo acelerar e aperfeiçoar desenhos e projetos técnicos nas áreas da engenharia, por meio de diversos recursos que permitem a execução de forma eficiente Silva (2019).

O AutoCAD[®] foi desenvolvido e lançado pela Autodesk em 1982, propondo uma concepção inovadora, considerando que antes do surgimento deste, os desenhos técnicos eram executados manualmente. Ele proporciona: a criação, anotação

e projeção geométrica em 2D e modelos 3D com objetos sólidos, superfícies e objetos de malha; automatização de tarefas como comparar desenhos, adicionar blocos e criar agendamentos e personalizar com aplicativos complementares.

2.5.4 OrCAD

A OrCAD® foi fundada em 1985 e o primeiro produto da empresa foi o *Schematic Desing Tools* (SDT) SDT. A linha de produtos da OrCAD posteriormente foi adquirida pela Cadence®, que faz a distribuição até os dias de hoje. O OrCAD permite que engenheiros eletricitas/eletrônicos realizem suas ideias de criação de produtos, por meio de uma plataforma que permite executar projetos em PCI. Fonte: <<https://www.orcad.com/pt/about/overview>>.

3 MATERIAL E MÉTODOS

Estudamos três métodos, Banks, Oliveira e Davis-Williams, e dois questionários, Nokelainen e Ssemugabi. Nos três métodos observamos metodologias diferentes aplicadas à eles, o que pode gerar uma discussão interessante. Já nos dois questionários, a maioria das perguntas não se aplicava aos *softwares* propostos neste estudo, ambos questionários possuem perguntas mais específicas para *softwares* dos tipos: tutoriais, de exercitação e jogos. Assim, não aplicamos os *softwares* estudados nesses questionários.

3.1 *SOFTWARES* APLICADOS NO MÉTODO OLIVEIRA

O método Oliveira é objetivo e para cada tópico deve ser respondido com Sim, Não ou Às vezes. Ele também já possui seus próprios pesos e, com o resultado final, ele classifica esse *software* quanto a didática dele. Assim, atribuímos as notas propostas de acordo com a didática da Engenharia Elétrica.

Neste método são apresentadas 8 categorias em que o *software* deve ser avaliado, cada uma delas apresenta características relevantes para observar se o programa está atendendo essas diretrizes didáticas. No Apêndice A encontram-se a Tabela 18 à Tabela 24 apresentam as notas propostas para cada característica, conforme já descrito em subseção 2.4.1.

Na categoria Apresentação e funcionalidade o MatLab e o AutoCAD possuem maior destaque. Esses dois *softwares* possuem instalação mais difíceis, por necessitarem de mais *hardware*, o que acaba tomando mais tempo. Enquanto que o Proteus e o OrCAD são programas que exigem menos *hardware*, logo facilitam a instalação.

Em relação à metodologia, o Proteus e o OrCAD em alguns casos possuem recursos físicos mais proveitosos para o ensino. Por exemplo no curso de microcontroladores, onde a utilização de kits didáticos é mais eficaz, por ser um contato físico do discente com os componentes. Já o MatLab e AutoCAD, no que se propõem a fazer, são um dos métodos mais recomendados.

Avaliando se o *software* gera resultados corretos de acordo com o que foi especificado, o Proteus e o OrCAD são mais suscetíveis à erros para algumas aplicações. Já o MatLab e o AutoCAD, quando utilizados corretamente, proporcionam uma resposta assertiva.

Os quatro *softwares* não apresentam uma faixa etária específica para a sua utilização. Principalmente porque todos eles podem ser utilizados no ensino médio, graduação e pós-graduação.

Na categoria Confiabilidade o MatLab e o AutoCAD possuem maior destaque.

Todos os *softwares* estão suscetíveis à erros. O Proteus não mantém o nível de desempenho na ocorrência de falhas. O MatLab mantém o nível. O AutoCAD e o OrCAD mantêm o nível em alguns casos. Em relação a recuperação de dados quando ocorrem falhas, o Proteus, o MatLab e o OrCAD, em alguns casos, não conseguem salvar as alterações quando há falha. O AutoCAD mantém o nível.

Na Categoria Usabilidade, o Proteus possui o maior destaque atendendo todos os requisitos desejados. O Proteus, AutoCAD e OrCAD, são *softwares* que proporcionam o entendimento do conceito de aplicação facilmente, por serem mais simples de trabalhar. Já o MatLab, por ser mais complexo e possuir vários ramos de estudo em um único *software*, em alguns casos é claro e em outros não. O Proteus, AutoCAD e OrCAD, são *softwares* que possibilitam uma aplicação mais simples. O matlab é mais complexo.

Em relação ao controle da operação, o Proteus e o OrCAD são *softwares* mais intuitivos, logo deixam o controle fácil. O MatLab e o AutoCAD necessitam de um aprofundamento maior para o controle e a operação. O Proteus, MatLab e OrCAD permitem a realização de diferentes observações incorporadas ao objeto, de forma imediata. O AutoCAD não permite isso.

O Proteus proporciona uma plataforma estudantil, onde os alunos podem acessar juntamente ao professor uma mesma área de trabalho.

Na categoria eficiência MatLab e AutoCAD receberam a nota máxima. Em relação ao tempo de resposta, o Proteus e o OrCAD possuem tempo de resposta rápido em alguns casos. O MatLab e o AutoCAD possuem o tempo de resposta rápido considerando as ações que eles estão sendo aplicados. E todos os quatro *softwares* não necessitam de recursos como impressora e discos flexíveis.

Na categoria Manutenibilidade, os quatro *softwares* apresentaram pontuações semelhantes. Todos os *softwares* possuem notificações de aviso de erro, porém nem todas as vezes é de fácil identificação. Por serem *softwares* de simulação, o Proteus e o OrCAD têm bastante mobilidade para fazer alterações. Já o MatLab e o AutoCAD são um pouco mais complexos em relação a modificações. E todos os *softwares* permitem a aplicação de modificações sem que ocorram efeitos inesperados e possuem facilidade para executar um teste quando há modificações.

Na categoria Portabilidade, o MatLab e o AutoCAD conseguiram um maior destaque na pontuação. Em relação à adaptabilidade em outros ambientes especificados, todos os *softwares* atendem esse requisito. Para instalar em outros ambientes, o Proteus e o OrCAD não possuem essa opção. Já o MatLab e o AutoCAD possuem integração melhor com outros ambientes, entretanto, os quatro *softwares* possuem facilidade para substituição dentro do ambiente.

Na primeira parte do Conteúdo do *software* relacionado com a disciplina, o AutoCAD obteve o maior resultado entre os *softwares* avaliados. No quesito de trans-

mitir conceitos adequados e utilizar uma linha pedagógica adequada, os *softwares* avaliados atendem esses requisitos. O Proteus não possui uma variedade de níveis de dificuldade por ser um *software* mais limitado quanto ao seu tipo de programação. Já os outros três *softwares* possuem uma variedade maior de dificuldade por poderem ser utilizados de formas diferentes. O tópico sobre apresentações adicionais, não se aplica à esses *softwares*, pois eles não foram criados com objetivo de serem didáticos.

Avaliando a clareza das instruções, o Proteus e o Autocad são mais claros que o MatLab e o OrCAD, estes dois, tendem a ter instruções algumas vezes mais complexas. Considerando a parte de recursos estimulantes, os quatro *softwares* utilizam bem os recursos gráficos e as cores. O AutoCAD é o único que não utiliza recurso sonoro.

O MatLab é um *software* muito bom quando tratado em relação a utilização de banco de dados. O AutoCAD permite a utilização de dados em alguns casos e o Proteus e OrCAD não utilizam bases de dados externas. Os *softwares* apresentam exercícios de níveis diferentes, proporcionando a alavancagem do aprendizado dos discentes. Apenas o AutoCAD apresenta correção de ortografia e gramática.

Na segunda parte do Conteúdo do *software* relacionado com a disciplina, todos obtiveram a mesma nota, porém essa nota não foi suficiente para obter mais que a metade dos pontos dedicados a esta seção. Em relação à simulação, todos possuem ótimos resultados, tirando o AutoCAD, que não é um *software* de simulação. Avaliando a interação de diferentes observações incorporadas ao objeto, todos possuem essa característica. Assim como no quesito de ajustar-se aos objetivos curriculares.

Em relação à interação com o usuário, como, fazer questionamentos, apresentar avaliação final, apresentar quais assuntos a serem revisados e disponibilizar um *feedback*, os *softwares* não apresentam essas características, pois não foram desenvolvidos com esse objetivo.

Considerando o público alvo, estes *softwares* apresentam linguagem adequada, material visual bom, links que permitem o usuário a buscar mais informações e trabalhos com conteúdo interdisciplinares.

Com o entendimento específico de todos estes critérios, aplicamos os valores nas Tabela 18 a Tabela 24, conforme apresentado no Apêndice A.

Como visto na Tabela 4, o *software* que foi mais bem adequado, considerando a didática na Engenharia Elétrica foi o MatLab com 128 pontos, em segundo o AutoCAD com 126 pontos. Seguido por OrCAD com 122 pontos e Proteus com 120 pontos. Todas essas notas segundo Oliveira se classificam como um *software* educacional Bom. Mesmo estes *softwares* não tendo sido criados com o objetivo educacional, eles tiveram um desempenho bom neste método seletivo.

Esta Tabela 4 é a síntese das Tabela 18 até a Tabela 24 que se encontram no Apêndice A.

Tabela 4 – Método de Oliveira - resultado final

Característica do <i>software</i>	Proteus	Matlab	AutoCAD	OrCAD
Total	120	128	126	122

Fonte: A autoria própria

3.2 SOFTWARES APLICADOS NO MÉTODO BANKS

Utilizamos o método Banks, que é um método originalmente criado para avaliar *softwares* de simulação, e de acordo com o método de avaliação proposto por Banks de *scoring model*, definimos os pesos para cada uma das 56 características levantadas pelo autor.

O critério usado para a escolha de pesos foi a didática na engenharia elétrica, de forma subjetiva baseadas na visão dos autores deste trabalho.

Neste método podem ser analisados vários aspectos como os aspectos de entrada, formas com a qual o usuário insere informações no *software*, cria modelos, escreve códigos ou ainda adiciona componentes a uma simulação, por exemplo.

Neste método são colocados também os aspectos de processamento, por exemplo tempo de execução do programa, tempo de espera até os *softwares* de fato entregar algum resultado, ou ainda capacidades mais específicas e técnicas como por exemplo capacidade de aleatoriedade, a qual não é simples. Continuando a citar os critérios, tem-se ainda os critérios de saída onde este avalia os dados que o *software* fornece para uma análise de resultados seja simulação, gráficos, planilhas ou mesmo dados brutos.

Uma parte muito importante é a condição de suporte, no mesmo é levando em conta o quanto do suporte ao *software* auxilia aos usuários em várias situações como em aprendizagem, ou problemas que possam ocorrer, ou mesmo fatores mais estimulantes, tais como animação do *softwares*. Este podem impactar de forma positiva no processo de ensino e de aprendizagem.

Por último e não menos importante reúne os critérios de custo do programa analisado, nesse caso não de forma financeira especificamente, mas de custo de poder de processamento do computador para ser capaz de utilizar o programa ou tempo para iniciar e aprender todas ou a mais importantes funcionalidades.

Nesta linha foi feita a análise de quatro softwares como pode ser visto na Tabela 25 conforme pode ser visto no Apêndice B. Segue uma breve explicação do porquê as notas foram pensadas como um exemplo de análise.

Segue a baixo a listagem de pesos:

- Interface com outros *softwares*

De um ponto de vista didático, é interessante que haja uma interface com outros *softwares*, por permitirem câmbio de informações, porém de forma focada que o

software deve “ensinar” um conteúdo isso acaba não se tornando tão relevante, assim definimos a nota 5 como peso.

- Capacidade de análise de dados de entrada

Poder observar e analisar os dados de entradas é importante para o aprendizado, assim *softwares* que possibilitam isso acabam ajudando na didática. Definimos 8 como peso para esse critério.

- Portabilidade

Foi considerado a portabilidade que os programas criados naquele *software* tenham em outras plataformas. Assim, para a didática isso é um pouco relevante quando considerado o a possibilidade de um aluno apenas ter acesso a outro *software* e ainda sim ser capaz de desenvolver seu conhecimento na disciplina. Definimos peso 6.

- Sintaxe

Considerando a sintaxe como a metodologia em que o aluno utiliza o *software*, consideramos como relevante para a didática no ensino por ser a metodologia que o docente terá que passar ao discente. Definimos 8 como peso.

- Flexibilidade de entrada

A flexibilidade permite que o aluno aprenda mais de um método para programar em certo *software*. Isso é relevante na didática, pois influencia em como o professor pode ensinar o *software*. Definimos 7 como peso.

- Depuração interativa

Na visão da didática na Engenharia Elétrica, consideramos que a depuração interativa é relevante por possibilitar que o discente aprenda a cada passo, assim, definimos peso 8 para essa característica.

- Flexibilidade de modelagem

É uma característica relevante, considerando que em uma sala há diversos tipos de discentes e um *textitsoftware* mais flexível que possibilite maneiras diferentes de modelagem é um recurso didático interessante. Definimos peso 8.

- Concisão na modelagem

Essa característica é relevante por ser necessário saber qual o tempo dispendido por um *software*. Nesse caso, quanto mais conciso ele será mais objetivo e gastará menos tempo com repetições mecânicas. Definimos peso 9.

- Velocidade de execução Para a realidade da didática esse textitsoftware não é tão relevante, pois o foco no ensino da disciplina é o ganho de conhecimento e não de desempenho. Definimos 5 como peso.
- Tamanho do modelo Não é um ponto muito relevante em frente à didática, pois o tempo e processamento não são tão relevantes para o objetivo final. Assim, definimos peso 5.
- Capacidade de manuseio de materiais
É um ponto relevante pois, a quantidade de bibliotecas possibilita mais conteúdo para a didática, e faz o textitsoftware ficar mais interdisciplinar. Definimos peso 8 para este critério.
- Gerador de variáveis aleatórias
Para a didática não é muito relevante, porém pode ser útil para algumas aplicações, assim tomamos o peso como 6.
- Período de aquecimento
É um critério de desempenho não muito importante para a didática, assim tomamos o peso como 4.
- Replicações independentes
Não se aplicou a classificação de didática na Engenharia Elétrica.
- Configuração de atributos às transações
Esse critério é relevante, pois, pode influenciar nas atividades de um textitsoftware na simulação dá ao aluno um controle maior e ele pode aplicar mais partes do seu conhecimento. Definimos o peso 8.
- Variáveis globais
Relevante, pois a utilização deste tipo de variáveis são um bom recurso para a utilização de *softwares*, na didática se destaca em alguns casos. Definimos peso 6.
- Possibilidade de programação para incorporar características especiais
Consideramos esse critério relevante, pois incorporar características especiais supõe que o textitsoftware possibilita que haja maneiras diferentes de programá-lo e usá-lo. Isso na didática é uma característica interessante, pois permite o discente que queira se aprofundar a conhecer mais funções no *software*. Definimos peso 8.

- Roteamento condicional

É uma característica relevante na didática, pois permite retirar muitas tarefas mecânicas tornando o processo de mexer no *software* mais automatizado. Definimos peso 7.

- Relatórios padrão com medidas de desempenho

Não se aplicou a classificação de didática na Engenharia Elétrica.

- Relatórios personalizados

Não se aplicou a classificação de didática na Engenharia Elétrica.

- Análise estatística

É uma característica relevante na didática, pois há possibilidade de receber as informações estaticamente do que o discente utilizou no *software*. Definimos peso 7.

- Geração de gráficos

É uma característica importante, pois permite ao aluno a observação dos dados de uma forma mais visual. Definimos peso 9.

- Criação de arquivos compatíveis com planilhas eletrônicas, banco de dados etc

Muito relevante para a didática por fornecer integratividade ao *software*, podendo cambiar informações e dados para outros programas. Definimos peso 9.

- Capacidade de rastreamento

Não se aplicou a classificação de didática na Engenharia Elétrica.

- Manutenção do banco de dados Pouco relevante didaticamente, esta é uma característica de desempenho exigido no mercado com pouca influência no aprendizado do *software*. Definimos peso 2.

- Facilidade de uso

Este critério é essencial, pois quanto mais fácil o *software* é para usar, mais otimizado fica a utilização do aluno quando estiver aplicando seus conhecimentos no *software*, poupando tempo de ensino. Definimos o peso 10.

- Facilidade de aprendizagem

Assim como o critério anterior, é muito importante para o retorno em tempo, já que o docente não precisa gastar tempo da disciplina explicando o *software*. Definimos peso 10.

- Qualidade da documentação

Este critério é relevante para a didática, pois parte do aprendizado é após aula, assim ter uma boa documentação do *software* ajuda o discente a firmar o conteúdo aprendido em sala e adquirir o conhecimento no seu próprio tempo. Definimos peso 10.

- Capacidade de animação A capacidade de animação é necessária, pois os discentes tendem a ter uma melhor performance de aprendizado quando recebem estímulos visuais e sonoros. Definimos um peso 8.

- Ajuda *on-line*

Definimos ajuda online como matérias, artigos, trabalhos e comunidades de discussão online sobre o *software*. Assim como na qualidade da documentação, é importante no auto desenvolvimento do discente. Definimos peso 9.

- Tutorial *on-line*

Definimos tutorial online como vídeos, cursos e tutorias presentes nos softwares. Também é importante para o discente no auto desenvolvimento. Definimos peso 9.

- Suporte ao cliente

Não se aplicou a classificação de didática na Engenharia Elétrica.

- Aquisição da licença do *software*:

Não se aplicou a classificação de didática na Engenharia Elétrica.

- Requisitos de *on-line*

O tempo gasto com a aprendizagem é um tópico crucial quando focando na didática. Saber se o tempo gasto ensinando um *software* será maior do que ensinando a disciplina que este *software* irá auxiliar é importante para o docente organizar o tempo despendido nessas duas tarefas. Assim, definimos o peso 10 para este critério.

- Tempo gasto com aprendizagem do *software*

O tempo gasto com a aprendizagem é um tópico crucial quando focando na didática. Saber se o tempo gasto ensinando um *software* será maior do que ensinando a disciplina que este *software* irá auxiliar é importante para o docente organizar o tempo despendido nessas duas tarefas. Assim, definimos o peso 10 para este critério.

- Tempo necessário para construção de modelos

Este critério é importante para o planejamento das aulas a serem ministradas. O docente deve levar em consideração o tempo que os alunos terão para realizar cada atividade quando estiverem criando os modelos. Definimos o peso 9 para este critério.

De acordo com os pesos definidos, a nota máxima possível da soma de todos os pesos é 225.

3.2.1 Proteus

Foi feita a avaliação do Proteus segundo o método de Banks, alguns aspectos que devem ser levados em consideração quando se avalia os *softwares*.

Um fator importante é a capacidade de interação com outros *softwares*, no campo do ensino esta capacidade se faz de grande valia levando em consideração a interdisciplinaridade na qual o Proteus perde pontos neste quisto pela sua incapacidade de interagir com outros programas, tendo em vista que por ser um *software* bastante completo não se nota este ímpeto de adicionar esta funcionalidade. Observando ainda para outras características é possível perceber um *software* bastante transparente no que se diz respeito a capacidade de análise dos dados de entrada tendo em vista que de maneira bastante intuitiva é possível ver todos os dados que estão interagindo durante a utilização dos *softwares* como por exemplo sinais de tensão, medidas de corrente valor das unidades dos componentes mais simples como resistores, capacitores e indutores.

A forma como se faz a utilização dos *softwares* se mostra muito pratica onde é possível através de grupos e subgrupos localizar os componentes desejados e ainda os salvar em uma lista de utilização rápida, ainda é possível verificar passo a passo a execução do programa por meio da opção de *debug*, esta sendo muito interativa proporcionando voltar e passar a cada instante da execução.

Um aspecto muito interessante reside no fato de que toda a simulação feita no Proteus pode ser feita de inúmeras formas, ou seja, cada pessoa pode criar um modelo de sua própria forma seguindo as regras básicas dos circuitos construídos, entretanto pode ser notado que em alguns aspectos o Proteus não alcança seus resultados almejados quando comparado ao circuito real.

O Proteus possui um execução leve do ponto de vista requisito computacional porem ao se aumentar a complexidade do modelo nota-se uma dificuldade em conciliar velocidade e acurácia na utilização do *softwares* fazendo o mesmo ficar instável e as vezes até mesmo paradas de funcionamento, possui um tempo de aquecimento (tempo desde o início do comando de executar até a execução de fato) este tempo em

média não passando de 10 segundos, os modelos gerados também são de tamanhos variados nunca sendo maiores que 100KB (kilobytes) muito menor que um arquivo de texto por exemplo.

Proteus tem a capacidade de gerar gráficos usando o osciloscópio e ainda gera arquivos pós simulação com vários dados da execução, não possui facilidade em gerar dados para planilha, não de forma direta.

Se trata de um *softwares* muito interativos e com animações em vários componentes o que estimula os alunos a utilizarem um *software* com essas vantagens, com isso ainda pode ser notado que várias abas do programa levam o usuário a fazer o que deseja, ajudando de forma excelente no processo de aprendizagem de uso do Proteus desta forma somando pontos muito positivos neste quesito tão importante.

3.2.2 MatLab

Fazendo a análise do Matlab segundo o método de Banks, podem ser notadas algumas características bem interessantes, tendo em vista que o Matlab se trata de um *software* muito prático e poderoso.

Possuindo várias formas de se utilizar este pode ser feito por blocos ou programação tendo a capacidade de se simular qualquer evento dependendo da complexidade desejada este ainda interage com vários *softwares* e, gera arquivos compatíveis com outros *softwares* sua sintaxe não é tão simples pois pode confundir o usuário menos familiarizado com o ambiente de matrizes, vetores, os mais utilizados no Matlab, mas por outro lado pode ser o mais preciso quanto se desejar dependendo do grau de complexidade a qual o usuário é capaz de trabalhar no programa.

Por se tratar de um *software* com muito poder e possibilidades é já esperado um *software* que tenha seu tempo de execução bastante variado nas tarefas mais simples este trabalha de forma rápida, mas mudando a complexidade este tempo pode variar assim como tamanho dos arquivos gerados por ele. Sua capacidade de processamento é de fato muito superior aos *softwares* aqui analisados.

Os dados que o Matlab gera com a sua utilização são muito versáteis sendo possível gerar gráficos com detalhes incríveis, exportar todos os dados para planilhas ou ainda lidar com esses dados e trabalhar eles no próprio Matlab.

É vasto a quantidade de material para a correta utilização do programa tanto externamente ao programa quanto de fato dentro do mesmo, pois com a função *help* existem todas as explicações para cada função e juntamente com estas explicações ainda têm exemplos de utilização auxiliando muito no processo de utilização de *software* para o ensino.

O Matlab possui um custo elevado, tanto no tempo de aprendizagem quanto na capacidade do computadores para este funcionar de forma fluida e podendo buscar

soluções complexas nesse quesito o Matlab perde pontos no entanto é de se esperar que tantos benefícios tenham seu preço, não justificando mas é o que ocorre.

3.2.3 AutoCAD

Quando se trata de *software* para desenhos tem-se um muito conhecido e bastante utilizado que é o AutoCAD, *software* muito utilizado em projetos elétricos residenciais e industriais substituindo de fato o desenho técnico à mão.

O AutoCAD possui uma sintaxe muito intuitiva os seus painéis são de fácil compreensão e ainda cada um deles conta com explicações, existe a possibilidade de utilizar o AutoCAD juntamente com outros programas de forma a facilitar o trabalho e ainda abrir seus arquivos em outros programas de desenho sendo uma vantagem de fato importante. Seus modelos são tão reais quanto for capaz o operador do programa.

3.2.4 OrCAD

Para o OrCAD foi feita a análise este por mais que tenha grandes diferenças com o Proteus ainda sim é um *softwares* voltado para simulação de circuitos eletrônicos e elétricos, possui grande biblioteca e muitos componentes, é mais conhecido por se comportar de maneira mais fiel em situação e de dados analógicas como corrente alternada e utilização das curvas de atuação de componentes por exemplo não lineares ainda possível captar sinais e trabalhar com estes na simulação, claramente dessa forma aumentando a complexidade de operação com o programa.

Finalizando a aplicação do critérios de Banks, transportamos o resultado final, deste método aplicado na Tabela 5

Tabela 5 – Método de Banks - resultado final.

Critérios	Descrição	Peso	Proteus	MatLab	AutoCAD	OrCAD
Total:		225	146,65	184,85	152,2	157,75

Fonte: Adaptado de Oliveira-2006 - método de Banks (1991).

3.3 SOFTWARES APLICADOS NO MÉTODO DE DAVIS-WILLIAMS

Estudamos o método Davis-Williams e preenchemos a matriz de comparação de acordo com a didática na Engenharia Elétrica.

O primeiro passo foi criar a matriz de comparação dos 14 critérios, os números

Tabela 6 – Matriz de comparação - critérios de 1 a 7.

MATRIZ DE COMPARAÇÕES	1	2	3	4	5	6	7
1. Custo	1,000	0,111	0,125	0,250	0,200	0,111	0,111
2. Flexibilidade	9,000	1,000	2,000	6,000	5,000	1,000	1,000
3. Recursos gráficos	8,000	0,500	1,000	5,000	4,000	0,500	0,500
4. Distribuições estatísticas	4,000	0,167	0,200	1,000	2,000	0,167	0,167
5. Integração com outros sistemas	5,000	0,200	0,250	0,500	1,000	0,200	0,200
6. Manual de instruções	9,000	1,000	2,000	6,000	5,000	1,000	1,000
7. Manual de referência	9,000	1,000	2,000	6,000	5,000	1,000	1,000
8. Treinamento	9,000	1,000	2,000	6,000	5,000	1,000	1,000
9. Conhecimento prévio requerido	9,000	1,000	2,000	6,000	5,000	1,000	1,000
10. Para o usuário novo	9,000	1,000	2,000	6,000	5,000	1,000	1,000
11. Para o usuário experiente	9,000	1,000	2,000	6,000	5,000	1,000	1,000
12. Para o usuário final	1,000	0,111	0,125	0,250	0,333	0,200	0,200
13. Hardware e instalação	7,000	0,333	0,500	3,000	2,000	0,333	0,333
14. Questões ligadas à credibilidade do fornecedor	8,000	0,500	1,000	5,000	4,000	0,500	0,500
Soma	97,000	8,922	17,200	57,000	48,533	9,011	9,011

Fonte: Autoria própria.

indicados nas colunas representam os respectivos critérios apresentados nas linhas. Fazendo uma comparação critério a critério, onde compara-se a importância de um critério sobre o outro, considerando as classificações de Saaty na Tabela 39 contida no Apêndice C. Pode se observar que critérios iguais ou com mesma importância recebem a nota um. No caso de comparações diferentes aplicasse a nota desejada. Por exemplo, no caso da comparação entre Custo e Flexibilidade, nota-se que Flexibilidade tem nota 9, logo Custo em relação a Flexibilidade tem nota 1 a 9. Seguindo o método, é apresentada a soma dessas comparações na vertical. Chegamos nas Tabela 6 e Tabela 7.

Tabela 7 – Matriz de comparação - critérios de 8 a 14.

MATRIZ DE COMPARAÇÕES	8	9	10	11	12	13	14
1. Custo	0,111	0,111	0,111	0,111	1,000	0,143	0,125
2. Flexibilidade	1,000	1,000	1,000	1,000	9,000	3,000	2,000
3. Recursos gráficos	0,500	0,500	0,500	0,500	8,000	2,000	1,000
4. Distribuições estatísticas	0,167	0,167	0,167	0,167	4,000	0,500	0,200
5. Integração com outros sistemas	0,200	0,200	0,200	0,200	5,000	0,333	0,250
6. Manual de instruções	1,000	1,000	1,000	1,000	9,000	3,000	2,000
7. Manual de referência	1,000	1,000	1,000	1,000	9,000	3,000	2,000
8. Treinamento	1,000	1,000	1,000	1,000	9,000	3,000	2,000
9. Conhecimento prévio requerido	1,000	1,000	1,000	1,000	9,000	3,000	2,000
10. Para o usuário novo	1,000	1,000	1,000	1,000	9,000	3,000	2,000
11. Para o usuário experiente	1,000	1,000	1,000	1,000	9,000	3,000	2,000
12. Para o usuário final	0,200	0,200	0,200	0,200	1,000	0,143	0,125
13. Hardware e instalação	0,333	0,333	0,333	0,333	7,000	1,000	0,500
14. Questões ligadas à credibilidade do fornecedor	0,500	0,500	0,500	0,500	8,000	2,000	1,000
Soma	9,011	9,011	9,011	9,011	97,000	27,119	17,200

Fonte: Autoria própria.

Para criar a matriz normalizada, dividimos o valor de cada comparação critério a critério pela soma dos critérios da coluna. Por exemplo, na célula comparativa Custo o Custo será calculado seu valor normalizado atribuindo valor 1 dividido por 97, resultando em 0,010.

Conseqüentemente, aplicando este cálculo para todas as comparações, conseguimos calcular o vetor prioridade, que é a média de cada critério na horizontal. Isso é mostrado na Tabela 8 e Tabela 9.

Tabela 8 – Matriz normalizada - critérios de 1 a 8.

MATRIZ NORMALIZADA	1	2	3	4	5	6	7	8
1. Custo	0,010	0,012	0,007	0,004	0,004	0,012	0,012	0,012
2. Flexibilidade	0,093	0,112	0,116	0,105	0,103	0,111	0,111	0,111
3. Recursos gráficos	0,082	0,056	0,058	0,088	0,082	0,055	0,055	0,055
4. Distribuições estatísticas	0,041	0,019	0,012	0,018	0,041	0,018	0,018	0,018
5. Integração com outros sistemas	0,052	0,022	0,015	0,009	0,021	0,022	0,022	0,022
6. Manual de instruções	0,093	0,112	0,116	0,105	0,103	0,111	0,111	0,111
7. Manual de referência	0,093	0,112	0,116	0,105	0,103	0,111	0,111	0,111
8. Treinamento	0,093	0,112	0,116	0,105	0,103	0,111	0,111	0,111
9. Conhecimento prévio requerido	0,093	0,112	0,116	0,105	0,103	0,111	0,111	0,111
10. Para o usuário novo	0,093	0,112	0,116	0,105	0,103	0,111	0,111	0,111
11. Para o usuário experiente	0,093	0,112	0,116	0,105	0,103	0,111	0,111	0,111
12. Para o usuário final	0,010	0,012	0,007	0,004	0,007	0,022	0,022	0,022
13. Hardware e instalação	0,072	0,037	0,029	0,053	0,041	0,037	0,037	0,037
14. Questões ligadas à credibilidade do fornecedor	0,082	0,056	0,058	0,088	0,082	0,055	0,055	0,055

Fonte: Autoria própria.

Tabela 9 – Matriz normalizada - critérios de 9 a 14 e percentual.

MATRIZ NORMALIZADA	9	10	11	12	13	14	VP*	%
1. Custo	0,012	0,012	0,012	0,010	0,005	0,007	0,010	1%
2. Flexibilidade	0,111	0,111	0,111	0,093	0,111	0,116	0,108	11%
3. Recursos gráficos	0,055	0,055	0,055	0,082	0,074	0,058	0,065	6%
4. Distribuições estatísticas	0,018	0,018	0,018	0,041	0,018	0,012	0,022	2%
5. Integração com outros sistemas	0,022	0,022	0,022	0,052	0,012	0,015	0,024	2%
6. Manual de instruções	0,111	0,111	0,111	0,093	0,111	0,116	0,108	11%
7. Manual de referência	0,111	0,111	0,111	0,093	0,111	0,116	0,108	11%
8. Treinamento	0,111	0,111	0,111	0,093	0,111	0,116	0,108	11%
9. Conhecimento prévio requerido	0,111	0,111	0,111	0,093	0,111	0,116	0,108	11%
10. Para o usuário novo	0,111	0,111	0,111	0,093	0,111	0,116	0,108	11%
11. Para o usuário experiente	0,111	0,111	0,111	0,093	0,111	0,116	0,108	11%
12. Para o usuário final	0,022	0,022	0,022	0,010	0,005	0,007	0,014	1%
13. Hardware e instalação	0,037	0,037	0,037	0,072	0,037	0,029	0,042	4%
14. Questões ligadas à credibilidade do fornecedor	0,055	0,055	0,055	0,082	0,074	0,058	0,065	6%

VP* = Vetor de prioridade.

Fonte: Autoria própria.

O vetor prioridade será utilizado como o valor a ser aplicado em relação a cada *software*, porém, antes de seguir para este passo, fizemos o cálculo da consistência, para garantir que não houve erros de inconsistência aplicados no método.

Para Saaty (1991), o valor máximo que pode haver de inconsistência é de 10%. Para a realização deste cálculo, pegamos os valores da matriz de comparação e multiplicamos pelo vetor prioridade, que transposto. Por exemplo, na célula comparativa Custo e Custo será calculado pegando valor 1 e multiplicando por 0,010, resultando em 0,010. Obtivemos a soma dos pesos na vertical.

Como apresentados nas Tabela 10 e Tabela 11.

Tabela 10 – Matriz consistência - critérios de 1 a 8.

CÁLCULO DE CONSISTÊNCIA	1	2	3	4	5	6	7	8
1. Custo	0,010	0,012	0,008	0,006	0,005	0,012	0,012	0,012
2. Flexibilidade	0,087	0,108	0,131	0,134	0,118	0,108	0,108	0,108
3. Recursos gráficos	0,077	0,054	0,065	0,112	0,094	0,054	0,054	0,054
4. Distribuições estatísticas	0,039	0,018	0,013	0,022	0,047	0,018	0,018	0,018
5. Integração com outros sistemas	0,048	0,022	0,016	0,011	0,024	0,022	0,022	0,022
6. Manual de instruções	0,087	0,108	0,131	0,134	0,118	0,108	0,108	0,108
7. Manual de referência	0,087	0,108	0,131	0,134	0,118	0,108	0,108	0,108
8. Treinamento	0,087	0,108	0,131	0,134	0,118	0,108	0,108	0,108
9. Conhecimento prévio requerido	0,087	0,108	0,131	0,134	0,118	0,108	0,108	0,108
10. Para o usuário novo	0,087	0,108	0,131	0,134	0,118	0,108	0,108	0,108
11. Para o usuário experiente	0,087	0,108	0,131	0,134	0,118	0,108	0,108	0,108
12. Para o usuário final	0,010	0,012	0,008	0,006	0,008	0,022	0,022	0,022
13. Hardware e instalação	0,068	0,036	0,033	0,067	0,047	0,036	0,036	0,036
14. Questões ligadas à credibilidade do fornecedor	0,077	0,054	0,065	0,112	0,094	0,054	0,054	0,054
Vetor Prioridade Transposto	0,010	0,108	0,065	0,022	0,024	0,108	0,108	0,108

Fonte: Autoria própria.

Dividindo a soma dos pesos pelo vetor prioridade e fazendo a média desses valores encontramos o lambda máximo λ_{max} , apresentado na Tabela 12.

Para o cálculo do Razão de Consistência (RC), calculamos primeiramente o Índice de Consistência (IC), em que **n** é a ordem da matriz, conforme a Equação 1:

$$IC = \frac{\lambda_{max} - n}{n - 1} \quad (1)$$

Aplicando o valor obtido da Tabela 12, chega-se ao valor numérico de IC.

$$IC = \frac{14,899 - 14}{14 - 1} = 0,06917$$

Utilizamos o Índice Randômico (IR) que Saaty definiu de acordo com a ordem da matriz, como visto na Tabela 13. Neste caso, uma matriz de ordem 14.

Tabela 11 – Matriz consistência - critérios de 9 a 14 e soma dos pesos.

CÁLCULO DE CONSISTÊNCIA	9	10	11	12	13	14	Soma dos Pesos
1. Custo	0,012	0,012	0,012	0,014	0,006	0,008	0,141
2. Flexibilidade	0,108	0,108	0,108	0,127	0,127	0,131	1,611
3. Recursos gráficos	0,054	0,054	0,054	0,113	0,085	0,065	0,990
4. Distribuições estatísticas	0,018	0,018	0,018	0,056	0,021	0,013	0,338
5. Integração com outros sistemas	0,022	0,022	0,022	0,070	0,014	0,016	0,352
6. Manual de instruções	0,108	0,108	0,108	0,127	0,127	0,131	1,611
7. Manual de referência	0,108	0,108	0,108	0,127	0,127	0,131	1,611
8. Treinamento	0,108	0,108	0,108	0,127	0,127	0,131	1,611
9. Conhecimento prévio requerido	0,108	0,108	0,108	0,127	0,127	0,131	1,611
10. Para o usuário novo	0,108	0,108	0,108	0,127	0,127	0,131	1,611
11. Para o usuário experiente	0,108	0,108	0,108	0,127	0,127	0,131	1,611
12. Para o usuário final	0,022	0,022	0,022	0,014	0,006	0,008	0,201
13. Hardware e instalação	0,036	0,036	0,036	0,099	0,042	0,033	0,640
14. Questões ligadas à credibilidade do fornecedor	0,054	0,054	0,054	0,113	0,085	0,065	0,990
Vetor Prioridade Transposto	0,108	0,108	0,108	0,014	0,042	0,065	

Fonte: Autoria própria.

Tabela 12 – Cálculo de λ_{max}

Crítérios	Soma dos Pesos	Vetor Prioridades	Peso/Prioridade
1. Custo	0,141	0,010	14,540
2. Flexibilidade	1,611	0,108	14,888
3. Recursos gráficos	0,990	0,065	15,160
4. Distribuições estatísticas	0,338	0,022	15,137
5. Integração com outros sistemas	0,352	0,024	14,949
6. Manual de instruções	1,611	0,108	14,888
7. Manual de referência	1,611	0,108	14,888
8. Treinamento	1,611	0,108	14,888
9. Conhecimento prévio requerido	1,611	0,108	14,888
10. Para o usuário novo	1,611	0,108	14,888
11. Para o usuário experiente	1,611	0,108	14,888
12. Para o usuário final	0,201	0,014	14,293
13. Hardware e instalação	0,640	0,042	15,134
14. Questões ligadas à credibilidade do fornecedor	0,990	0,065	15,160
		Total	208,590
		λ_{max}	14,8992796

Fonte: Autoria própria.

Na Equação 2, apresentamos o cálculo do RC.

$$RC = \frac{IC}{IR} = \frac{0,06917}{1,57} = 0,04406 = 4,41 \quad (2)$$

Com o resultado de 4,41% de inconsistência, nós pudemos seguir com a análise.

O passo seguinte foi fazer a comparação *software* a *software* em relação a cada um dos critérios. Ou seja, foram feitas 14 matrizes comparando os *softwares*,

Tabela 13 – Índice randômico

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0,00	0,00	0,58	0,9	1,12	1,24	1,32	1,41	1,45	1,49	1,51	1,48	1,56	1,57	1,59

Fonte: Autoria própria.

onde cada uma delas considerou um critério proposto pelo método Davis-Williams.

Seguimos os mesmos passos anteriores e chegamos nos resultados indicados nas tabelas apresentadas no Apêndice C.

No Apêndice C, apresentado na Tabela 27, chegamos no vetor prioridade de cada *software*, com esse resultado já temos qual *software* se destaca mais para cada critério.

Observando a avaliação *software* a *software* para cada critério, nota-se que todos os programas apresentaram o RC menor que 10%, validando a comparação feita.

Assim chegamos na última parte, onde fizemos uma matriz *softwares* versus Critérios, utilizando o vetor prioridade de cada critério em relação a cada *software* e multiplicamos pelo vetor de critérios das características, apresentado na Tabela 14.

Tabela 14 – Resultado final.

Resultado Final	Proteus	MatLab	AutoCAD	OrCAD
1	0,002	0,002	0,002	0,002
2	0,025	0,025	0,046	0,013
3	0,016	0,005	0,029	0,016
4	0,002	0,017	0,002	0,002
5	0,002	0,013	0,006	0,002
6	0,021	0,056	0,021	0,011
7	0,020	0,038	0,038	0,012
8	0,039	0,009	0,039	0,022
9	0,033	0,008	0,033	0,033
10	0,039	0,009	0,039	0,022
11	0,006	0,044	0,044	0,014
12	0,004	0,004	0,004	0,004
13	0,021	0,004	0,004	0,014
14	0,011	0,022	0,022	0,011
Resultado	0,241	0,254	0,327	0,177
%	24%	25%	33%	18%

Fonte: Autoria própria.

Como resultado, chegamos na comparação de qual desses *softwares* é mais adequado na Engenharia Elétrica. O AutoCAD apresentou 33%, o MatLab 25%, seguido por OrCAD com 24% e Proteus com 18%.

Os resultados que obtivemos são influenciados pelos pesos das comparações que nós definimos subjetivamente. É necessário fazer uma análise de sensibilidade previsto no método AHP de Saaty, fazendo com que os pesos tenham valores diferen-

tes, assim, é possível avaliar quais critérios foram mais relevantes.

Primeiramente, utilizamos todos os critérios com o mesmo peso. A Tabela 15 mostra essa hipótese.

Tabela 15 – Sensibilidade I - Mesmo peso.

Sensibilidade 1	Proteus	MatLab	AutoCAD	OrCAD
1	0,018	0,018	0,018	0,018
2	0,016	0,016	0,030	0,009
3	0,018	0,005	0,031	0,018
4	0,006	0,054	0,006	0,006
5	0,007	0,040	0,018	0,007
6	0,014	0,037	0,014	0,008
7	0,014	0,025	0,025	0,008
8	0,026	0,006	0,026	0,014
9	0,022	0,005	0,022	0,022
10	0,026	0,006	0,026	0,014
11	0,004	0,029	0,029	0,009
12	0,018	0,018	0,018	0,018
13	0,036	0,006	0,006	0,023
14	0,012	0,024	0,024	0,012
Resultado	0,235	0,288	0,292	0,184
%	23%	29%	29%	18%

Fonte: Autoria própria.

Na segunda hipótese, definimos que o critério 8, treinamento, teria metade da importância e os outros critérios ficariam com 3,846% de importância. A Tabela 16, mostra essa hipótese.

Tabela 16 – Sensibilidade II - Mesmo peso.

Sensibilidade 2	Proteus	MatLab	AutoCAD	OrCAD
1	0,010	0,010	0,010	0,010
2	0,009	0,009	0,016	0,005
3	0,009	0,003	0,017	0,009
4	0,003	0,029	0,003	0,003
5	0,004	0,021	0,010	0,004
6	0,007	0,020	0,007	0,004
7	0,007	0,013	0,013	0,004
8	0,179	0,041	0,179	0,100
9	0,012	0,003	0,012	0,012
10	0,014	0,003	0,014	0,008
11	0,002	0,016	0,016	0,005
12	0,010	0,010	0,010	0,010
13	0,019	0,003	0,003	0,012
14	0,006	0,013	0,013	0,006
Resultado	0,292	0,193	0,323	0,192
%	29%	19%	32%	19%

Fonte: Autoria Própria.

Isso significa que quando o critério treinamento possuir mais de 50% da importância a tendência será o *software* AutoCAD e Proteus.

O teste de sensibilidade mostra como podemos alterar o método Davis-Williams de maneira a atender certamente o critério mais relevante para cada disciplina e de acordo com o objetivo do discente. As hipóteses podem ser diversas e aplicáveis de acordo com a necessidade.

4 RESULTADOS E DISCUSSÃO

Na Tabela 17 podemos observar os resultados finais, dá comparação entre os três métodos.

Tabela 17 – Comparação entre os três métodos.

	Banks	Oliveira	Davis-Williams
Proteus	146,65	120	24%
MatLab	184,85	128	25%
AutoCAD	152,2	126	33%
OrCAD	157,75	122	18%

Fonte: Autoria própria.

Estes resultados apresentam diferenças entre si, não havendo uma unanimidade entre eles. O método Banks, é um método mais subjetivo e criado com o objetivo de avaliar *softwares* de simulação. Os autores do trabalho foram responsáveis por atribuir os pesos. Este método pode ser aplicado por docentes que querem ter um maior controle sobre o processo. Permitindo que eles definam os critérios mais importantes baseados em experiência que estes possuem, e também imponham notas subjetivas para cada *software* avaliado.

No método Oliveira, as notas possíveis são objetivas, sendo as mesmas: possui, não possui ou possui as vezes uma determinada característica. Este método foi criado com o objetivo de categorizar *softwares* educacionais, logo suas perguntas estão focadas no âmbito do ensino. Esse método é bom para os docentes que desejam avaliar como que os discentes reagem em relação à aprendizagem de um *software*.

Já no método Davis-Williams, assim como em Banks, foi desenvolvido para a avaliação de *softwares* de simulação. Porém, este método é mais complexo que o Banks. Há a possibilidade do usuário de dar notas em forma comparativa de forma subjetiva, porém essas notas serão aplicadas no método AHP, o que torna este método mais científico. Pois o método consiste em normalização dados e também um cálculo de consistência. É um bom método para os docentes poderem aplicar suas prioridades, com a ressalva de poder tirar uma parte dessa subjetividade.

5 CONCLUSÕES E PERSPECTIVAS

5.1 CONCLUSÃO

Os questionários de Nokelainen e Ssemugabi, não foram utilizados para este trabalho, pois possuem características avaliativas para os tipos de *software*: tutorias, de exercitação e jogos. Por conseguinte, concluímos que questionários não se aplicam para a maior parte dos *softwares* da Engenharia Elétrica. Este tipo de pesquisa é mais utilizado para *softwares* que substituem livros e tem uma maneira sequencial de apresentar o conteúdo. A maior parte dos *softwares* utilizados na Engenharia são programas aplicados para simulação de eventos ou para facilitação de cálculo e trabalhos manuais.

O método Oliveira é um método avaliativo e seletivo para *softwares* educacionais que propõe resultados classificatórios numa escala de: Ruim a Ótimo, passando por Bom e Regular. Isso possibilita a análise de apenas um *software* sem a necessidade de fazer a comparação com outros. O docente pode utilizar-se desse método para classificar se o *software* utilizado por ele está dentro das classificações didáticas propostas por Oliveira. Quando foi aplicado os *softwares* selecionados nesse método, foi possível observar que eles tiveram um desempenho similar, todos foram classificados como Bons (pontuação obtida de 115 a 150) para *softwares* didáticos. Isso ocorreu, pois, estes programas tiveram desempenhos bons e ótimos nas categorias: apresentação e funcionalidade do produto, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade. Porém, na categoria conteúdo do *software* relacionado a disciplina, os critérios são voltados para *softwares* dos tipos: tutoriais, de exercitação e jogos. Isso nos faz concluir que os programas aplicados nesse método, são excelentes *softwares* educacionais para Engenharia Elétrica, pois, nos quesitos em que há a necessidade no cursos os mesmos possuem excelentes notas.

O método Banks é um método que se faz necessário uma comparação para sua análise, ao contrário do método do Oliveira, ele não fornece uma classificação, impõe que o usuário do método escolha os pesos e as notas, assim só é possível chegar a um resultado quando há mais de um *software* para a comparação. Quando se trata de uma pesquisa com intuito científico a opinião de uma pessoa não mostra confiança no método que dependem de critérios subjetivos, é notado que, para um resultado conciso e rigoroso deve ser usada uma pesquisa do tipo formulário para obter uma amostragem em uma população específica de usuário do *software* que será avaliado, melhorando a percepção geral.

Desta forma neste trabalho os resultados tiveram o intuito de demonstrar alguns métodos e assim facilitar a disseminação do seu uso. Na aplicação do método nos

softwares selecionado foi notório observar como o MatLab sobressaiu-se em relação aos demais, por possuir vários pontos positivos em relação a didática e simulações na qual este método foi desenvolvido com objetivo de avaliar *softwares* de simulação e assim os autores da aplicação do método tiveram seu pesos ponderados por um objetivo específico que é o do melhor *software* para utilização na Engenharia Elétrica, por conseguinte tem-se OrCAD, AutoCAD e Proteus nesta ordem.

Na Utilização do método Davis-Williams foram obtidos resultados muito interessantes pois são vários os passos para de fato a seleção de um *software*, falando mais especificamente do método AHP, pôde ser notado que este método de seleção pode ser usado para varias decisões no mundo acadêmico e profissional, este método buscando retirar a subjetividade da seleção mostra que o *software* teve maior destaque foi o AutoCAD contrariando os resultado dos métodos anteriores já citados anteriormente, deixando na sequência MatLab em seguida Proteus e por fim OrCAD, mostrando o método pode de fato mudar escolhas e ainda de forma científica guiar os profissionais do ensino superior na área da Engenharia Elétrica.

Este trabalho apresenta e aplica os métodos de Davis-Williams, Oliveira e Banks onde Davis-Williams se destaca por apresentar uma metodologia científica diminuindo, mas não excluindo, a interferência pessoal de cada avaliador. Apresenta como, por meio de pesos, é possível alterar os resultados do método de Banks tornando a seleção do *software* muito pessoal caso a caso ou ainda, a necessidade de uma pesquisa por amostragem tornando o método mais assertivo. Isto mostra como estes métodos já existentes possuem deficiências que podem ser corrigidas, não os tornando ineficazes, mas, para a aplicação dos métodos, tomar cuidado para não fazer uma avaliação majoritariamente subjetiva.

Comparando os três métodos, acreditamos que o Davis-Williams seja o mais indicado, por possibilitar que o docente coloque mais peso no que lhe é conveniente para sua disciplina, mas sem perder a objetividade de um método científico.

Foi feito um apanhado mostrando os *softwares* amplamente usados no ensino da Engenharia Elétrica, e os aplicamos em três métodos de avaliação e seleção de programas. Entretanto, todos esses métodos seriam mais proveitosos se os *softwares* em questão tivessem o mesmo objetivo, assim quando o docente fosse fazer a comparação ele escolheria programas que pudessem ser utilizados na sua disciplina e selecionaria o mais adequado de acordo com os seus interesses didáticos.

5.2 TRABALHOS FUTUROS

Ao fazer as pesquisas para este trabalho foi notada a carência de métodos, para avaliação de *softwares*, mais modernos. Podendo ser um trabalho futuro a criação de um método novo onde baseado em uma distribuição estatística entre alunos e

professores, fundamentar e criar um método já pensado no mercado de trabalho para os futuros engenheiros eletricitas e até mesmo de outras áreas do ensino superior.

Para este trabalho, serão desenvolvidos dois questionários, um para discentes e outro para docentes. Esses questionários terão notas subjetivas de uma população em relação à quais critérios eles avaliam como relevante num *software* com empregabilidade na Engenharia. Com os resultados da pesquisa, será feita uma distribuição desses valores na matriz de comparação do método AHP. Será seguido depois, os passos propostos por Saaty (1987). Com o AHP estruturado, será necessário que o docente escolha apenas os *softwares* que ele deseja comparar.

Outra proposta seria um método mais moldável. Com uma proposta de separar os critérios de avaliação por tipo de *software*, como tratado na seção subseção 2.3.1, assim, caso uma categoria não se aplique ao *software* estudado, o usuário pode não aplicar esta categoria de critérios ao *software* avaliado. Essa proposta também possuirá uma classificação, parecida com o método Oliveira, para que os programas utilizados na educação possam ser avaliados individualmente, possibilitando ao docente checar quão indicado é o *software* que ele usará como recurso didático.

REFERÊNCIAS

BARANAUSKAS, Maria Cecília Calani et al. Uma taxonomia para ambientes de aprendizado baseado no computador. In: **O computador na sociedade do conhecimento**. [S.l.: s.n.], 1999. Citado na página 18.

DAVIS, Lesley; WILLIAMS, Glyn. Evaluating and Selecting Simulation Software Using the Analytic Hierarchy Process. **Integrated Manufacturing Systems**, 1994. ISSN 09576061. Citado 2 vezes nas páginas 26 e 75.

JUNIOR, João Batista Bottentuit; COUTINHO, Clara Pereira. Análise da Usabilidade de um Laboratório Virtual de Química Orgânica. **6ª Conferência Ibero-americana em Sistemas, Cibernética e Informática (CISCI)**, v. 1, p. 91–95, 2007. Disponível em: <<http://repositorium.sdum.uminho.pt/bitstream/1822/6718/1/usabilidade.pdf>>. Citado na página 13.

MARIANI, Viviana Cocco; MARTIM, Emerson. Aplicações Do Matlab No Ensino De Disciplinas Básicas Nos Cursos De Engenharia. **COBENGE**, 2003. Citado 2 vezes nas páginas 14 e 30.

MARTINS, Kerley. Teorias de aprendizagem e avaliação de software educativo. v. 133, n. 57, p. 1–5, 2002. Citado na página 22.

OLIVEIRA, Noé de. **Uma proposta para a avaliação de software educacional**. 2001. 105 p. Tese (Doutorado) — Universidade Federal de Santa Catarina, 2001. Disponível em: <<http://repositorio.ufsc.br/xmlui/handle/123456789/81485>>. Citado 12 vezes nas páginas 14, 20, 21, 24, 25, 58, 59, 60, 71, 72, 73 e 74.

QUADRADO, José Carlos. Organizações de Engenharia no mundo. **Revista de Ensino de Engenharia**, v. 32, n. 3, p. 19–24, 2013. Disponível em: <<http://revista.educacao.ws/revista/index.php/abenge/article/view/233>>. Citado na página 5.

RAMOS, Edla Maria Faust. **O fundamental na Avaliação da Qualidade do Software Educacional**. 1991. Citado na página 23.

SAATY, Thomas Lorie. The analytic hierarchy process-what it is and how it is used. **Mathematical Modelling**, 1987. ISSN 02700255. Citado 3 vezes nas páginas 26, 27 e 53.

SANUSB, Grupo. Aplicações Práticas de Eletrônica e Microcontroladores em Sistemas Computacionais Aprenda de forma simples a gravação wireless e via USB de microcontroladores. 2011. Citado na página 14.

SILVA, André Koide da. Método para avaliação e seleção de softwares de simulação de eventos discretos aplicados à análise de sistemas logísticos. p. 191, 2006. Citado 3 vezes nas páginas 25, 26 e 27.

SILVA, Alisson Sousa da. Utilização De Software Autocad Como Instrumento Didático Para a Formação Acadêmica No Ensino De Engenharia. **Revista Produção Industrial e Serviços**, 2019. Citado 2 vezes nas páginas 13 e 30.

SILVA, José G. Santos da; LIMA, Luciano R. Ornelas de; FERREIRA, Alessandra R. Uma Experiência Didática Com Base no Emprego do Matlab nos Cursos de Graduação da Faculdade de Engenharia da Uerj. In: **Congresso Brasileiro de Ensino de Engenharia**. Brasília: [s.n.], 2004. Citado 2 vezes nas páginas 14 e 30.

SILVEIRA, Maurício et al. Ferramentas de Software de análise e síntese de circuitos como apoio didático na melhoria do ensino de engenharia. **Indefinido**, p. 1242 – 1250, 1998. Disponível em: <<http://www.abenge.org.br/cobenge/arquivos/20/st/t/t085.pdf>>. Citado 2 vezes nas páginas 19 e 20.

SU, Bo; WANG, Li. Application of Proteus Virtual System Modelling (VSM) in teaching of microcontroller. In: **2010 International Conference on E-Health Networking, Digital Ecosystems and Technologies, EDT 2010**. [S.l.: s.n.], 2010. ISBN 9781424455157. Citado 2 vezes nas páginas 14 e 30.

TAVARES, Jéssika Lima. **Modelos, técnicas e instrumentos de análise de softwares educacionais**. 2017. 97 p. Tese (Doutorado) — Universidade Federal da Paraíba, 2017. Disponível em: <<https://repositorio.ufpb.br/jspui/handle/123456789/2563>>. Citado 6 vezes nas páginas 20, 21, 22, 23, 79 e 83.

TEIXEIRA, Adriano Canabarro; BRANDÃO, Edenilson Jorge Ramos. Software educacional: o difícil começo. **Novas Tecnologias na Educação**, v. 1, n. 1, p. 1 –

7, 2003. Disponível em: <http://penta2.ufrgs.br/edu/ciclopalestras/artigos/adriano{_}software.> Citado na página 20.

VALENTE, J. A. O Uso Inteligente Do Computador Na Educação. **Pátio - revista pedagógica**, 1996. Citado na página 17.

VALENTE, José Armando. **O computador na sociedade do conhecimento**. [S.l.: s.n.], 1999. Citado 3 vezes nas páginas 17, 18 e 21.

APÊNDICES

APÊNDICE A – MÉTODO DE OLIVEIRA - APLICADO

Tabela 18 – Apresentação e funcionalidade do produto - aplicado

Característica do software	Proteus	MatLab	AutoCAD	OrCAD
O manual é adequado e claro, quanto aos conteúdos?	2	2	2	2
Apresenta as exigências de hardware?	2	2	2	2
É de fácil instalação?	2	1	1	2
Está de acordo com as finalidades e objetivos propostos?	2	2	2	2
É mais eficaz do que outra metodologia para atingir os objetivos a que se propõe?	1	2	2	1
Interage com os sistemas especificados?	2	2	2	2
Suas funções são apropriadas para as tarefas especificadas?	2	2	2	2
Gera resultados ou efeitos corretos de acordo com a especificação?	1	2	2	1
evita acesso não autorizado, acidental ou deliberado, a programas e dados?	2	2	2	2
O software declara a sua corrente pedagógica?	2	2	2	2
O software apresenta qual a faixa etária do seu público alvo?	0	0	0	0
Subtotal (Máximo 22 pontos)	18	19	19	18

Fonte: Oliveira (2001).

Tabela 19 – Confiabilidade - aplicado

Característica do software	Proteus	MatLab	AutoCAD	OrCAD
Há ausência de falhas?	1	1	1	1
Mantém o nível de desempenho na ocorrência de falhas?	0	2	1	1
Recupera os dados quando ocorrem falhas?	1	1	2	1
Subtotal (Máximo 6 pontos)	2	4	4	3

Fonte: Oliveira (2001).

Tabela 20 – Eficiência - aplicado

Característica do <i>software</i>	Proteus	MatLab	AutoCAD	OrCAD
O programa proporciona facilidade no entendimento do conceito e aplicação?	2	1	2	2
É fácil a sua aplicação?	2	1	2	2
É fácil o controle e a operação?	2	1	1	2
Permite a realização de diferentes observações incorporadas ao objeto?	2	2	0	2
As diferenças observações incorporadas ao objeto são imediatas?	2	2	0	2
Permite desenvolver e testar hipóteses?	2	2	2	2
Permite a análise de resultados depurando os conceitos?	2	2	2	2
Permite o uso simultâneo por grupos de alunos?	2	0	0	0
Subtotal (Máximo 16 pontos)	16	11	9	14

Fonte: Oliveira (2001).

Tabela 21 – Manutenibilidade - aplicado

Característica do <i>software</i>	Proteus	MatLab	AutoCAD	OrCAD
O tempo de resposta ao comando das funções é rápido?	1	2	2	1
Dispensa outros recursos (impressora, discos flexíveis) para ser usado?	2	2	2	2
Subtotal (Máximo 4 pontos)	3	4	4	3

Fonte: Oliveira (2001).

Tabela 22 – Portabilidade - aplicado

Característica do <i>software</i>	Proteus	MatLab	AutoCAD	OrCAD
É fácil perceber quando houve uma falha?	1	1	1	1
É fácil modificar, remover defeitos ou adaptá-lo a mudanças do ambientes?	2	1	1	2
Permite a aplicação de alguma modificação sem que ocorram efeitos inesperados?	2	2	2	2
É fácil o teste quando se faz alguma modificação?	2	2	2	2
Subtotal (Máximo 8 pontos)	7	6	6	7

Fonte: Oliveira (2001).

Tabela 23 – Conteúdo do *software*, relacionado com a disciplina I - aplicado

Característica do <i>software</i>	Proteus	MatLab	AutoCAD	OrCAD
Transmite conceitos adequados?	5	5	5	5
O <i>software</i> utiliza adequadamente a linha pedagógica proposta?	5	5	5	5
Tem uma variedade de níveis de dificuldade?	0	5	5	5
Os alunos podem chegar a determinados pontos sem seguir uma sequência obrigatória?	5	5	5	5
Estão previstas apresentações adicionais se necessárias?	0	0	0	0
As instruções são claras e lógicas?	5	3	5	3
O <i>software</i> utiliza bem o gráfico?	5	5	5	5
O <i>software</i> utiliza bem o som?	5	5	0	5
O <i>software</i> utiliza bem a cor?	5	5	5	5
Permite manipular vários dados com a utilização de um eventual banco de dados possivelmente modificáveis?	0	5	3	0
Apresenta exercícios de níveis diferentes, relacionados com o conteúdo estudado?	5	5	5	5
Utiliza a correção da ortografia e gramática?	0	0	5	0
Subtotal (Máximo 60 pontos)	40	48	48	43

Fonte: Oliveira (2001).

Tabela 24 – Conteúdo do *software*, relacionado com a disciplina II - aplicado

Característica do <i>software</i>	Proteus	MatLab	AutoCAD	OrCAD
De acordo com o conteúdo, o <i>software</i> utiliza alguma simulação?	5	5	0	5
Permite a interação de diferentes observações incorporadas ao objeto?	5	5	5	5
Ajusta-se aos objetivos curriculares?	5	5	5	5
Faz questionamentos ao usuário relacionados às respostas dadas durante a sua utilização?	0	0	0	0
Apresenta a avaliação final com os resultados obtidos pelo usuário?	0	0	0	0
A avaliação é compatível com a corrente pedagógica?	0	0	0	0
Apresenta "feedback"?	0	0	0	0
A linguagem está adequada para a faixa etária a que se propõe?	5	5	5	5
O material visual (figuras, gráficos, simulações) está adequado à faixa etária?	5	5	5	5
Apresenta links que permitam ao usuário buscar mais informações sobre o assunto?	5	5	5	5
Os conteúdos são trabalhados de forma interdisciplinar?	5	5	5	5
De acordo com os resultados obtidos, o programa apresenta quais assuntos a serem revisados?	0	0	0	0
Subtotal (Máximo 65 pontos)	35	35	30	35

Fonte: Oliveira (2001).

APÊNDICE B – MÉTODO DE BANKS - APLICADO

Tabela 25 – Método de Banks - aplicado.

Cr	Descrição	Peso	Proteus	MatLab	AutoCAD	OrCAD
Entrada	Interface com outros softwares	5	0,5	0,9	0,6	0,5
	Capacidade de análise de dados de entrada	8	0,8	0,95	0,9	0,8
	Portabilidade	6	0,2	0,9	0,8	0,7
	Sintaxe	8	0,9	0,7	0,8	0,7
	Flexibilidade de entrada	7	0,2	0,9	0,8	0,85
	Depuração interativa	8	0,6	0,9	0	0,8
	Flexibilidade de modelagem	8	0,9	1	1	0,9
	Concisão na modelagem	9	0,7	1	1	0,8
Processamento	Velocidade de execução	5	0,6	0,8	0,8	0,6
	Tamanho do modelo	5	0,5	1	0,9	0,6
	Capacidade de manuseio de materiais	8	0,9	0,95	0,9	0,9
	Gerador de variáveis aleatórias	6	0,4	1	0	0,5
	Período de aquecimento	4	0,5	0,8	0,8	0,6
	Replicações independentes	0	0	0	0	0
	Configuração de atributos às transações	8	0,8	0,9	1	0,8
	Variáveis globais	6	0,8	1	0	0,8
	Possibilidade de programação para incorporar características especiais	8	0,4	1	1	0,8
	Roteamento condicional	7	0,6	0,95	0	0,6
Saída	Relatórios padrão com medidas de desempenho	0	0	0	0	0
	Relatórios personalizados	0	0	0	0	0
	Análise estatística	7	0	0,7	0	0
	Geração de gráficos	9	0,6	1	0	0,9
	Criação de arquivos compatíveis com planilhas eletrônicas, banco de dados etc.	9	0	1	0,8	0,6
	Capacidade de rastreamento	0	0	0	0	0
	Manutenção do banco de dados	2	0	0,9	0,9	0,6
Suporte	Facilidade de uso	10	0,8	0,5	0,8	0,5
	Facilidade de aprendizagem	10	0,9	0,5	0,9	0,7
	Qualidade da documentação	10	0,6	0,9	0,9	0,7
	Capacidade de animação	8	0,9	0,6	0,9	0,8
	Ajuda on-line	9	0,9	0,9	0,9	0,7
	Tutorial on-line	9	1	1	1	0,7
	Suporte ao cliente	0	0	0	0	0
Custo	Aquisição da licença do software	0	0	0	0	0
	Requisitos de hardware	7	0,85	0,5	0,5	0,7
	Tempo gasto com aprendizagem do software	10	0,9	0,4	0,6	0,8
	Tempo necessário para construção de modelos	9	0,7	0,5	0,5	0,7
Total:		225	146,65	184,85	152,2	157,75

Fonte: Adaptado de Oliveira-2006 - método de Banks (1991).

APÊNDICE C – MÉTODO DE DAVIS-WILLIAMS - APLICADO

Tabela 26 – Matriz comparação aplicada.

MATRIZ DE COMPARAÇÕES				
1	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	1,000	1,000	1,000
MatLab	1,000	1,000	1,000	1,000
AutoCAD	1,000	1,000	1,000	1,000
OrCAD	1,000	1,000	1,000	1,000
Soma	4,000	4,000	4,000	4,000
2	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	1,000	0,500	2,000
MatLab	1,000	1,000	0,500	2,000
AutoCAD	2,000	2,000	1,000	3,000
OrCAD	0,500	0,500	0,333	1,000
Soma	4,500	4,500	2,333	8,000
3	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	4,000	0,500	1,000
MatLab	0,250	1,000	0,200	0,250
AutoCAD	2,000	5,000	1,000	2,000
OrCAD	1,000	4,000	0,500	1,000
Soma	4,250	14,000	2,200	4,250
4	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	0,111	1,000	1,000
MatLab	9,000	1,000	9,000	9,000
AutoCAD	1,000	0,111	1,000	1,000
OrCAD	1,000	0,111	1,000	1,000
Soma	12,000	1,333	12,000	12,000
5	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	0,200	0,333	1,000
MatLab	5,000	1,000	3,000	5,000
AutoCAD	3,000	0,333	1,000	3,000
OrCAD	1,000	0,200	0,333	1,000
Soma	10,000	1,733	4,667	10,000
6	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	0,333	1,000	2,000
MatLab	3,000	1,000	3,000	4,000
AutoCAD	1,000	0,333	1,000	2,000
OrCAD	0,500	0,250	0,500	1,000
Soma	5,500	1,917	5,500	9,000
7	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	0,500	0,500	2,000
MatLab	2,000	1,000	1,000	3,000
AutoCAD	2,000	1,000	1,000	3,000
OrCAD	0,500	0,333	0,333	1,000

(continua)

Tabela 26 – Matriz comparação aplicada.

(continuação)

MATRIZ DE COMPARAÇÕES				
Soma	5,500	2,833	2,833	9,000
8	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	4,000	1,000	2,000
MatLab	0,250	1,000	0,250	0,333
AutoCAD	1,000	4,000	1,000	2,000
OrCAD	0,500	3,000	0,500	1,000
Soma	2,750	12,000	2,750	5,333
9	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	4,000	1,000	1,000
MatLab	0,250	1,000	0,250	0,250
AutoCAD	1,000	4,000	1,000	1,000
OrCAD	1,000	4,000	1,000	1,000
Soma	3,250	13,000	3,250	3,250
10	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	4,000	1,000	2,000
MatLab	0,250	1,000	0,250	0,333
AutoCAD	1,000	4,000	1,000	2,000
OrCAD	0,500	3,000	0,500	1,000
Soma	2,750	12,000	2,750	5,333
11	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	0,167	0,167	0,333
MatLab	6,000	1,000	1,000	4,000
AutoCAD	6,000	1,000	1,000	4,000
OrCAD	3,000	0,250	0,250	1,000
Soma	16,000	2,417	2,417	9,333
12	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	1,000	1,000	1,000
MatLab	1,000	1,000	1,000	1,000
AutoCAD	1,000	1,000	1,000	1,000
OrCAD	1,000	1,000	1,000	1,000
Soma	4,000	4,000	4,000	4,000
13	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	5,000	5,000	2,000
MatLab	0,200	1,000	1,000	0,250
AutoCAD	0,200	1,000	1,000	0,250
OrCAD	0,500	4,000	4,000	1,000
Soma	1,900	11,000	11,000	3,500
14	Proteus	MatLab	AutoCAD	OrCAD
Proteus	1,000	0,500	0,500	1,000
MatLab	2,000	1,000	1,000	2,000
AutoCAD	2,000	1,000	1,000	2,000
OrCAD	1,000	0,500	0,500	1,000
Soma	6,000	3,000	3,000	6,000

Fonte: Autoria própria.

Tabela 27 – Matriz normalizada aplicada.

MATRIZ NORMALIZADA					
1	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,250	0,250	0,250	0,250	0,250
MatLab	0,250	0,250	0,250	0,250	0,250
AutoCAD	0,250	0,250	0,250	0,250	0,250
OrCAD	0,250	0,250	0,250	0,250	0,250
Soma	1,000	1,000	1,000	1,000	
2	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,222	0,222	0,214	0,250	0,227
MatLab	0,222	0,222	0,214	0,250	0,227
AutoCAD	0,444	0,444	0,429	0,375	0,423
OrCAD	0,111	0,111	0,143	0,125	0,123
Soma	1,000	1,000	1,000	1,000	
3	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,235	0,286	0,227	0,235	0,246
MatLab	0,059	0,071	0,091	0,059	0,070
AutoCAD	0,471	0,357	0,455	0,471	0,438
OrCAD	0,235	0,286	0,227	0,235	0,246
Soma	1,000	1,000	1,000	1,000	
4	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,083	0,083	0,083	0,083	0,083
MatLab	0,750	0,750	0,750	0,750	0,750
AutoCAD	0,083	0,083	0,083	0,083	0,083
OrCAD	0,083	0,083	0,083	0,083	0,083
Soma	1,000	1,000	1,000	1,000	
5	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,100	0,115	0,071	0,100	0,097
MatLab	0,500	0,577	0,643	0,500	0,555
AutoCAD	0,300	0,192	0,214	0,300	0,252
OrCAD	0,100	0,115	0,071	0,100	0,097
Soma	1,000	1,000	1,000	1,000	
6	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,182	0,174	0,182	0,222	0,190
MatLab	0,545	0,522	0,545	0,444	0,514
AutoCAD	0,182	0,174	0,182	0,222	0,190
OrCAD	0,091	0,130	0,091	0,111	0,106
Soma	1,000	1,000	1,000	1,000	
7	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,182	0,176	0,176	0,222	0,189
MatLab	0,364	0,353	0,353	0,333	0,351
AutoCAD	0,364	0,353	0,353	0,333	0,351
OrCAD	0,091	0,118	0,118	0,111	0,109
Soma	1,000	1,000	1,000	1,000	

(continua)

Tabela 27 – Matriz normalizada aplicada.

(continuação)

MATRIZ NORMALIZADA					
8	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,364	0,333	0,364	0,375	0,359
MatLab	0,091	0,083	0,091	0,063	0,082
AutoCAD	0,364	0,333	0,364	0,375	0,359
OrCAD	0,182	0,250	0,182	0,188	0,200
Soma	1,000	1,000	1,000	1,000	
9,000	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,308	0,308	0,308	0,308	0,308
MatLab	0,077	0,077	0,077	0,077	0,077
AutoCAD	0,308	0,308	0,308	0,308	0,308
OrCAD	0,308	0,308	0,308	0,308	0,308
Soma	1,000	1,000	1,000	1,000	
10,000	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,364	0,333	0,364	0,375	0,359
MatLab	0,091	0,083	0,091	0,063	0,082
AutoCAD	0,364	0,333	0,364	0,375	0,359
OrCAD	0,182	0,250	0,182	0,188	0,200
Soma	1,000	1,000	1,000	1,000	
11,000	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,063	0,069	0,069	0,036	0,059
MatLab	0,375	0,414	0,414	0,429	0,408
AutoCAD	0,375	0,414	0,414	0,429	0,408
OrCAD	0,188	0,103	0,103	0,107	0,125
Soma	1,000	1,000	1,000	1,000	
12,000	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,250	0,250	0,250	0,250	0,250
MatLab	0,250	0,250	0,250	0,250	0,250
AutoCAD	0,250	0,250	0,250	0,250	0,250
OrCAD	0,250	0,250	0,250	0,250	0,250
Soma	1,000	1,000	1,000	1,000	
13,000	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,526	0,455	0,455	0,571	0,502
MatLab	0,105	0,091	0,091	0,071	0,090
AutoCAD	0,105	0,091	0,091	0,071	0,090
OrCAD	0,263	0,364	0,364	0,286	0,319
Soma	1,000	1,000	1,000	1,000	
14,000	Proteus	MatLab	AutoCAD	OrCAD	Vetor Prioridade
Proteus	0,167	0,167	0,167	0,167	0,167
MatLab	0,333	0,333	0,333	0,333	0,333
AutoCAD	0,333	0,333	0,333	0,333	0,333
OrCAD	0,167	0,167	0,167	0,167	0,167
Soma	1,000	1,000	1,000	1,000	

Fonte: Autoria própria.

Tabela 28 – Cálculo de consistência aplicado.

CÁLCULO DE CONSISTÊNCIA					
1	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,250	0,250	0,250	0,250	1,000
MatLab	0,250	0,250	0,250	0,250	1,000
AutoCAD	0,250	0,250	0,250	0,250	1,000
OrCAD	0,250	0,250	0,250	0,250	1,000
2	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,227	0,227	0,212	0,245	0,911
MatLab	0,227	0,227	0,212	0,245	0,911
AutoCAD	0,454	0,454	0,423	0,368	1,699
OrCAD	0,114	0,114	0,141	0,123	0,491
3	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,246	0,280	0,219	0,246	0,991
MatLab	0,061	0,070	0,088	0,061	0,281
AutoCAD	0,492	0,350	0,438	0,492	1,772
OrCAD	0,246	0,280	0,219	0,246	0,991
4	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,083	0,083	0,083	0,083	0,333
MatLab	0,750	0,750	0,750	0,750	3,000
AutoCAD	0,083	0,083	0,083	0,083	0,333
OrCAD	0,083	0,083	0,083	0,083	0,333
5	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,097	0,111	0,084	0,097	0,388
MatLab	0,484	0,555	0,755	0,484	2,277
AutoCAD	0,290	0,185	0,252	0,290	1,017
OrCAD	0,097	0,111	0,084	0,097	0,388
6	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,190	0,171	0,190	0,212	0,763
MatLab	0,570	0,514	0,570	0,423	2,077
AutoCAD	0,190	0,171	0,190	0,212	0,763
OrCAD	0,095	0,129	0,095	0,106	0,424
7	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,189	0,175	0,175	0,219	0,759
MatLab	0,378	0,351	0,351	0,328	1,408
AutoCAD	0,378	0,351	0,351	0,328	1,408
OrCAD	0,095	0,117	0,117	0,109	0,438
8	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,359	0,328	0,359	0,401	1,446
MatLab	0,090	0,082	0,090	0,067	0,328
AutoCAD	0,359	0,328	0,359	0,401	1,446
OrCAD	0,179	0,246	0,179	0,200	0,805
9	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,308	0,308	0,308	0,308	1,231
MatLab	0,077	0,077	0,077	0,077	0,308
AutoCAD	0,308	0,308	0,308	0,308	1,231

(continua)

Tabela 28 – Cálculo de consistência aplicado.

(continuação)

CÁLCULO DE CONSISTÊNCIA					
OrCAD	0,308	0,308	0,308	0,308	1,231
10	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,359	0,328	0,359	0,401	1,446
MatLab	0,090	0,082	0,090	0,067	0,328
AutoCAD	0,359	0,328	0,359	0,401	1,446
OrCAD	0,179	0,246	0,179	0,200	0,805
11	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,059	0,068	0,068	0,042	0,237
MatLab	0,354	0,408	0,408	0,502	1,671
AutoCAD	0,354	0,408	0,408	0,502	1,671
OrCAD	0,177	0,102	0,102	0,125	0,506
12	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,250	0,250	0,250	0,250	1,000
MatLab	0,250	0,250	0,250	0,250	1,000
AutoCAD	0,250	0,250	0,250	0,250	1,000
OrCAD	0,250	0,250	0,250	0,250	1,000
13	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,502	0,448	0,448	0,638	2,036
MatLab	0,100	0,090	0,090	0,080	0,359
AutoCAD	0,100	0,090	0,090	0,080	0,359
OrCAD	0,251	0,359	0,359	0,319	1,287
14	Proteus	MatLab	AutoCAD	OrCAD	Soma dos Pesos
Proteus	0,167	0,167	0,167	0,167	0,667
MatLab	0,333	0,333	0,333	0,333	1,333
AutoCAD	0,333	0,333	0,333	0,333	1,333
OrCAD	0,167	0,167	0,167	0,167	0,667

Fonte: Autoria própria.

Tabela 29 – Cálculo de consistência estratificado aplicado.

CÁLCULO DE CONSISTÊNCIA ESTRATIFICADO					
Critério	Soma dos Pesos	VP	Peso/Prioridade	IC	RC
1. Custo					
Proteus	1,000	0,250	4,000		
MatLab	1,000	0,250	4,000		
AutoCAD	1,000	0,250	4,000		
OrCAD	1,000	0,250	4,000		
		λ_{max}	4,000	0	0
2. Flexibilidade					
Proteus	0,911	0,227	4,010		
MatLab	0,911	0,227	4,010		
AutoCAD	1,699	0,423	4,016		
OrCAD	0,491	0,123	4,005		
		λ_{max}	4,010	0,003455119	0,384%
3. Recursos					
Proteus	0,991	0,246	4,030		
MatLab	0,281	0,070	4,009		
AutoCAD	1,772	0,438	4,043		
OrCAD	0,991	0,246	4,030		
		λ_{max}	4,028	0,009263246	1,029%
4. Distribuição					
Proteus	0,333	0,083	4,000		
MatLab	3,000	0,750	4,000		
AutoCAD	0,333	0,083	4,000		
OrCAD	0,333	0,083	4,000		
		λ_{max}	4,000	-1,4803E-16	0,000%
5. Integração					
Proteus	0,388	0,097	4,015		
MatLab	2,277	0,555	4,103		
AutoCAD	1,017	0,252	4,041		
OrCAD	0,388	0,097	4,015		
		λ_{max}	4,044	0,01450252	1,611%
6. Instalação					
Proteus	0,763	0,190	4,017		
MatLab	2,077	0,514	4,039		
AutoCAD	0,763	0,190	4,017		
OrCAD	0,424	0,106	4,009		
		λ_{max}	4,021	0,006877511	0,764%
7. Referência					
Proteus	0,759	0,189	4,009		
MatLab	1,408	0,351	4,014		
AutoCAD	1,408	0,351	4,014		
OrCAD	0,438	0,109	4,004		
		λ_{max}	4,010	0,003459305	0,384%
8. Treinamento					
Proteus	1,446	0,359	4,029		
MatLab	0,328	0,082	4,006		
AutoCAD	1,446	0,359	4,029		
OrCAD	0,805	0,200	4,019		
		λ_{max}	4,021	0,006895031	0,766%

(continua)

Tabela 29 – Cálculo de consistência estratificado aplicado.

(continuação)

CÁLCULO DE CONSISTÊNCIA ESTRATIFICADO					
Critério	Soma dos Pesos	VP	Peso/Prioridade	IC	RC
9. Conhecimento					
Proteus	1,231	0,308	4		
MatLab	0,308	0,077	4		
AutoCAD	1,231	0,308	4		
OrCAD	1,231	0,308	4		
		λ_{max}	4	0	0,000%
10. Inicial					
Proteus	1,446	0,359	4,029		
MatLab	0,328	0,082	4,006		
AutoCAD	1,446	0,359	4,029		
OrCAD	0,805	0,200	4,019		
		λ_{max}	4,021	0,006895031	0,766%
11. Intermediário					
Proteus	0,237	0,059	4,010		
MatLab	1,671	0,408	4,099		
AutoCAD	1,671	0,408	4,099		
OrCAD	0,506	0,125	4,039		
		λ_{max}	4,062	0,020513314	2,279%
12. Avançado					
Proteus	1,000	0,250	4		
MatLab	1,000	0,250	4		
AutoCAD	1,000	0,250	4		
OrCAD	1,000	0,250	4		
		λ_{max}	4	0	0,000%
13. Hardware					
Proteus	2,036	0,502	4,058		
MatLab	0,359	0,090	4,009		
AutoCAD	0,359	0,090	4,009		
OrCAD	1,287	0,319	4,034		
		λ_{max}	4,028	0,009238467	1,026%
14. Fornecedor					
Proteus	0,667	0,167	4		
MatLab	1,333	0,333	4		
AutoCAD	1,333	0,333	4		
OrCAD	0,667	0,167	4		
		λ_{max}	4	0	0,000%

Fonte: Autoria própria.

ANEXOS

ANEXO A – MÉTODO DE OLIVEIRA

Tabela 30 – Apresentação e funcionalidade do produto

Característica do software	Atributo		
	0	1	2
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?			
É 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> apresenta qual a faixa etária do seu público alvo?			
Subtotal (Máximo 22 pontos)			

Fonte: Oliveira (2001).

Tabela 31 – Confiabilidade

Característica do software	Atributo		
	0	1	2
Há ausência de falhas?			
Mantém o nível de desempenho na ocorrência de falhas?			
Recupera os dados quando ocorrem falhas?			
Subtotal (Máximo 6 pontos)			

Fonte: Oliveira (2001).

Tabela 32 – Eficiência

Característica do software	Atributo		
	0	1	2
O tempo de resposta ao comando das funções é rápido?			
Dispensa outros recursos (impressora, discos flexíveis) para ser usado?			
Subtotal (Máximo 4 pontos)			

Fonte: Oliveira (2001).

Tabela 33 – Manutenibilidade

Característica do software	Atributo		
	0	1	2
É fácil perceber quando houve uma falha?			
É fácil modificar, remover defeitos ou adaptá-lo a mudanças do ambientes?			
Permite a aplicação de alguma modificação sem que ocorram efeitos inesperados?			
É fácil o teste quando se faz alguma modificação?			
Subtotal (Máximo 8 pontos)			

Fonte: Oliveira (2001).

Tabela 34 – Portabilidade

Característica do software	Atributo		
	0	1	2
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 (Máximo 6 pontos)			

Fonte: Oliveira (2001).

Tabela 35 – Conteúdo do *software* relacionado com a disciplina parte I

Característica do software	Atributo		
	0	3	5
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 (Máximo 60 pontos)			

Fonte: Oliveira (2001).

Tabela 36 – Conteúdo do *software* relacionado com a disciplina parte II

Característica do software	Atributo		
	0	3	5
De acordo com o conteúdo, o software utiliza 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 "feedback"?			
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?			
De acordo com os resultados obtidos, o programa apresenta quais assuntos a serem revisados?			
Subtotal (Máximo 65 pontos)			

Fonte: Oliveira (2001).

ANEXO B – MÉTODO DE BANKS

Tabela 37 – Método de Banks.

Critérios de:	Descrição	Peso	Software
Entrada	Interface com outros <i>softwares</i>		
	Capacidade de análise de dados de entrada		
	Portabilidade		
	Sintaxe		
	Flexibilidade de entrada		
	Depuração interativa		
	Flexibilidade de modelagem		
Processamento	Concisão na modelagem		
	Velocidade de execução		
	Tamanho do modelo		
	Capacidade de manuseio de materiais		
	Gerador de variáveis aleatórias		
	Período de aquecimento		
	Replicações independentes		
	Configuração de atributos às transações		
Saída	Variáveis globais		
	Possibilidade de programação para incorporar características especiais		
	Roteamento condicional		
	Relatórios padrão com medidas de desempenho		
	Relatórios personalizados		
	Análise estatística		
	Geração de gráficos		
Suporte	Criação de arquivos compatíveis com planilhas eletrônicas, banco de dados etc.		
	Capacidade de rastreamento		
	Manutenção do banco de dados		
	Facilidade de uso		
	Facilidade de aprendizagem		
Custo	Qualidade da documentação		
	Capacidade de animação		
	Ajuda <i>online</i>		
	Tutorial <i>online</i>		
Total:	Suporte ao cliente		
	Aquisição da licença do <i>software</i>		
	Requisitos de <i>hardware</i>		
	Tempo gasto com aprendizagem do <i>software</i>		
	Tempo necessário para construção de modelos		

Fonte: Adaptado de Oliveira (2001).

ANEXO C – MÉTODO DE DAVIS E WILLIAMS

Tabela 38 – Exemplo de cinco *softwares* aplicados no método Davis-Williams.

Critério	A	B	C	D	E
1. Custo	Muito baixo	Médio	Alto	Médio	Alto
2. Integralidade do sistema					
2.1. Flexibilidade	Alto	Alto	Alto	Alto	Médio
2.2. Distribuições estatísticas	Alto	Baixo	Médio	Alto	Médio
2.3. Recursos gráficos	Baixo	Alto	Alto	Médio	Médio
3. Integração com outros sistemas	Baixo	Medio	Muito Baixo	Baixo	Baixo
4. Documentação					
4.1. Manual de instruções	Alto	Medio	Baixo	Médio	Médio
4.2. Manual de referência	Alto	Medio	Alto	Médio	Médio
5. Treinamento	Baixo	Medio	Alto	Médio	Alto
6. Facilidade de uso					
6.1. Conhecimento prévio requerido	Médio	Baixo	Médio	Médio	Alto
6.2. Para o usuário novo	Médio	Baixo	Baixo	Baixo	Alto
6.3. Para o usuário experiente	Alto	Alto	Alto	Alto	Médio
6.4. Para o usuário final	Médio	Alto	Alto	Médio	Alto
7. Hardware e instalação	Baixo	Médio	Alto	Médio	Médio
8. Questões ligadas à credibilidade do fornecedor	Baixo	Médio	Alto	Médio	Alto

Fonte: Davis e Williams (1994).

Tabela 39 – Adaptação dos cinco *softwares* aplicados no método Davis-Williams.

Nº	Critério	A	B	C	D	E
1	Custo					
2	Flexibilidade					
3	Recursos gráficos					
4	Distribuições estatísticas					
5	Integração com outros sistemas					
6	Manual de instruções					
7	Manual de referência					
8	Treinamento					
9	Conhecimento prévio requerido					
10	Para o usuário novo					
11	Para o usuário experiente					
12	Para o usuário final					
13	Hardware e instalação					
14	Questões ligadas à credibilidade do fornecedor					

Fonte: Adaptado de Davis e Williams (1994).

ANEXO D – QUESTIONÁRIO DE NOKELAINEN

Tabela 40 – Questionário de Nokelainen.

Questão	Descrição	Critérios
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),	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).	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).	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).	Atividade Do Aluno
5	Este material de aprendizagem proporciona questões de aprendizagem sem um modelo pré-definido para a sua resolução.	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).	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.	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).	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.	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).	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).	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).	Aprendizagem Cooperativa/Colaborativa

(continua)

Tabela 40 – Questionário de Nokelainen.

		(continuação)
Questão	Descrição	Critérios
13	É agradável usar o material de aprendizagem com outro estudante no mesmo computador.	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).	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).	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").	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").	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).	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).	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").	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).	Aplicabilidade
26	Eu sinto que estou apto a usar as habilidades e conhecimento que este material de aprendizagem tem me ensinado no futuro.	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).	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).	Aplicabilidade

(continua)

Tabela 40 – Questionário de Nokelainen.

		(continuação)
Questão	Descrição	Critérios
29	Este material de aprendizagem é adequadamente desafiador para mim. (Definição: As tarefas não são tão fáceis ou tão difíceis).	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).	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).	Aplicabilidade
32	As imagens neste material de aprendizagem ajudam a aprender.	Valor Agregado
33	Os sons neste material de aprendizagem ajudam a aprender.	Valor Agregado
34	As animações neste material de aprendizagem ajudam a aprender.	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).	Valor Agregado
36	Eu tento alcançar uma alta classificação tanto quanto posso neste material de aprendizagem.	Motivação
37	Eu quero aprender os tópicos deste material de aprendizagem tão profundamente quanto posso.	Motivação
38	Estou interessado nos tópicos deste material de aprendizagem.	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).	Avaliação Do Conhecimento Prévio
40	Eu posso usar meus conhecimentos prévios quando estudo com este material.	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).	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).	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).	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").	Flexibilidade
45	Este material de aprendizagem torna rápido e fácil o aprendizado de um novo tópico ou o recapitular de um tópico anterior.	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.	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).	Controle Do Aluno

(continua)

Tabela 40 – Questionário de Nokelainen.

		(continuação)
Questão	Descrição	Critérios
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).	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).	Controle Do Aluno, Aplicabilidade
50	Eu posso fazer certo número de erros com este material (por exemplo, respostas erradas em tarefas de cálculos).	Feedback
51	Quando eu erro uma solução de uma tarefa, o programa me envia um aviso amigável.	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).	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).	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).	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).	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).	Aplicabilidade

Fonte: Adaptado de Tavares (2017).

Respostas para as questões:

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

ANEXO E – QUESTIONÁRIO DE SSEMUGABI

Tabela 41 – Questionário de Ssemugabi.

Questão	Categoria
Categoria 1: Projeto interface em geral	
1.	Visibilidade do status do sistema
1.1	O Sistema me mantém informado através de <i>feedback</i> : sobre o que está acontecendo.
1.2	Eu entendo o que significa <i>feedback</i> .
1.3	Eu obtenho o <i>feedback</i> 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	E 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 toma 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

(continua)

Tabela 41 – Questionário de Ssemugabi.

(continuação)

Questão	Categoria
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 relembrar 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 grande auxílio.
Categoria 2: Design específico para <i>websites</i> 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 <i>links</i> às seções dentro da mesma página.
11.5	As cores para os <i>links</i> são consistentes com as convenções <i>Web</i> , isto é, <i>links</i> 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.

(continua)

Tabela 41 – Questionário de Ssemugabi.

(continuação)

Questão	Categoria
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.	Clareza 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.	<i>Feedback</i> , orientação e avaliação
18.1	O sistema proporciona <i>feedback</i> 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 <i>feedback</i> 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.

(continua)

Tabela 41 – Questionário de Ssemugabi.

(continuação)

Questão	Categoria
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 autoavaliações neste site.

Fonte: Adaptado de Tavares (2017).

Respostas para as questões:

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