

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

DAVI MIGUEL CORDEIRO KUBIÇA
ISRAEL EDUARDO KUSS

**APLICAÇÃO DO PROCESSO ANALÍTICO HIERÁRQUICO (AHP)
PARA AUXÍLIO À TOMADA DE DECISÃO NA MUDANÇA DE
SOFTWARE LEVANDO EM CONSIDERAÇÃO OS CUSTOS DE
TROCA**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2015

**DAVI MIGUEL CORDEIRO KUBIÇA
ISRAEL EDUARDO KUSS**

**APLICAÇÃO DO PROCESSO ANALÍTICO HIERÁRQUICO (AHP)
PARA AUXÍLIO À TOMADA DE DECISÃO NA MUDANÇA DE
SOFTWARE LEVANDO EM CONSIDERAÇÃO OS CUSTOS DE
TROCA**

Trabalho apresentado à disciplina Trabalho de Conclusão de Curso 2 do Bacharelado em Sistemas de Informação do Departamento Acadêmico de Informática – DAINF – da Universidade Tecnológica Federal do Paraná como requisito parcial para a obtenção do título de Bacharel em Sistemas de Informação.

Orientador: Prof. Alexandre Reis Graeml

CURITIBA

2015

AGRADECIMENTOS

Agradecemos ao nosso orientador, professor Alexandre Reis Graeml, pelo apoio, correções e disponibilidade durante a realização deste trabalho.

À professora Mariângela de Oliveira Gomes Setti, pelo auxílio na pesquisa durante a disciplina de Trabalho de Conclusão de Curso 1.

Aos demais professores da instituição que colaboraram com nossa formação, tornando possível a criação do sistema.

E aos nossos familiares, pelo apoio durante todo o processo de pesquisa e criação deste trabalho.

“O sucesso parece ser, em grande parte, uma questão de persistir quando os outros desistiram.”

(William Feather)

RESUMO

KUBIÇA, Davi Miguel Cordeiro; KUSS, Israel Eduardo. APLICAÇÃO DO PROCESSO ANALÍTICO HIERÁRQUICO (AHP) PARA AUXÍLIO À TOMADA DE DECISÃO NA MUDANÇA DE SOFTWARE LEVANDO EM CONSIDERAÇÃO OS CUSTOS DE TROCA. 70 f. Trabalho de Conclusão de Curso – Curso de Bacharelado em Sistemas de Informação, Universidade Tecnológica Federal do Paraná. Curitiba, 2015.

As tecnologias da informação estão sujeitas aos custos de troca de um fornecedor para outro. Este trabalho discorre sobre os diversos tipos de custos de troca que influenciam o mercado de TI, apresentando também o Processo Analítico Hierárquico. Este será utilizado na elaboração de um software que auxilie os tomadores de decisão nas suas trocas de artefatos tecnológicos levando em consideração os custos de troca apontados.

Palavras-chave: Custos de Troca, Aprisionamento Tecnológico, AHP, Tomada de Decisão.

ABSTRACT

KUBIÇA, Davi Miguel Cordeiro; KUSS, Israel Eduardo. APPLICATION OF THE ANALYTICAL HIERARCHY PROCESS (AHP) TO AID DECISION MAKING IN CHANGING SOFTWARE TAKING INTO ACCOUNT SWITCHING COSTS. 70 f. Trabalho de Conclusão de Curso – Curso de Bacharelado em Sistemas de Informação, Universidade Tecnológica Federal do Paraná. Curitiba, 2015.

The information technologies are subject to switching costs from one supplier to another. This paper discusses the various types of switching costs that influence the IT market, also introducing the Analytical Hierarchy Process. This will be used in preparing a software that assists decision makers in the process of replacing technological artifacts considering the switching costs mentioned.

Keywords: Switching Costs, Vendor Lock-in, AHP, Decision-making.

LISTA DE FIGURAS

FIGURA 1	– Cronograma das atividades	15
FIGURA 2	– Graus de custos de troca	16
FIGURA 3	– Modelo de custos de troca em TI	17
FIGURA 4	– Estrutura do AHP	26
FIGURA 5	– Diagrama de casos de uso	38
FIGURA 6	– Diagrama de pacotes	42
FIGURA 7	– Diagrama de classes: visão geral	43
FIGURA 8	– Diagrama de classes: pacote principal	44
FIGURA 9	– Diagrama de classes: pacote ahp	45
FIGURA 10	– Diagrama de classes: pacote dao	46
FIGURA 11	– Diagrama de classes: pacote visualização (parte 1)	47
FIGURA 12	– Diagrama de classes: pacote visualização (parte 2)	48
FIGURA 13	– Interface inicial com os critérios e as alternativas já estabelecidos	50
FIGURA 14	– Barra de ferramentas	52
FIGURA 15	– Interface novo projeto	53
FIGURA 16	– Interface nova avaliação	54
FIGURA 17	– Interface avaliação	54
FIGURA 18	– Interface gerenciar avaliações	55
FIGURA 19	– Aviso para avaliações incompletas.	56
FIGURA 20	– Interface resultado final	56
FIGURA 21	– Interface modelo hierárquico	57
FIGURA 22	– Interface conteúdo	57
FIGURA 23	– Interface com as características dos desenvolvedores.	58
FIGURA 24	– Resultado dos cálculos das avaliações com duas alternativas	61
FIGURA 25	– Resultado final com duas alternativas	62
FIGURA 26	– Resultado dos cálculos das avaliações com quatro alternativas	68
FIGURA 27	– Resultado final com quatro alternativas	69
FIGURA 28	– Resultado dos cálculos das avaliações com dez alternativas	69
FIGURA 29	– Resultado final com dez alternativas	70

LISTA DE TABELAS

TABELA 1	– Quantidade de comparações paritárias no método AHP	26
TABELA 2	– Escala de relativa importância	27
TABELA 3	– Matriz de comparação com três critérios	28
TABELA 4	– Normalização da matriz de comparação	28
TABELA 5	– Vetor de prioridade (Eigen)	29
TABELA 6	– Valores de RI propostos por Saaty	30
TABELA 7	– Matriz de comparação das alternativas em relação a custo de troca 1	30
TABELA 8	– Vetor de prioridade (Eigen) das alternativas em relação a custo de troca 1 .	31

LISTA DE QUADROS

QUADRO 1	–	Requisito funcional RF001	33
QUADRO 2	–	Requisito funcional RF002	33
QUADRO 3	–	Requisito funcional RF003	34
QUADRO 4	–	Requisito funcional RF004	34
QUADRO 5	–	Requisito funcional RF005	34
QUADRO 6	–	Requisito funcional RF006	35
QUADRO 7	–	Requisito funcional RF007	35
QUADRO 8	–	Requisito funcional RF008	35
QUADRO 9	–	Requisito não-funcional NF001	36
QUADRO 10	–	Requisito não-funcional NF002	36
QUADRO 11	–	Requisito não-funcional NF003	36
QUADRO 12	–	Requisito não-funcional NF004	36
QUADRO 13	–	Requisito não-funcional NF005	36
QUADRO 14	–	Requisito não-funcional NF006	37
QUADRO 15	–	Requisito não-funcional NF007	37
QUADRO 16	–	Requisito não-funcional NF008	37
QUADRO 17	–	Caso de uso manter critérios	39
QUADRO 18	–	Caso de uso manter alternativas	40
QUADRO 19	–	Caso de uso realizar decisão	41
QUADRO 20	–	Casos de teste da interface	60

LISTA DE SIGLAS

MCDA	<i>Multiple-Criteria Decision Analysis</i>
TI	Tecnologia da Informação
SO	Sistema Operacional
MAUT	<i>Multi-Attribute Utility Theory</i>
AHP	<i>Analytic Hierarchy Process</i>
CBR	<i>Case-Based Reasoning</i>
DEA	<i>Data Envelopment Analysis</i>
SMART	<i>Simple Multi-Attribute Rating Technique</i>
ELECTRE	<i>Elimination and Choice Expressing the Reality</i>
PROMETHEE	<i>Preference Ranking Organization Method for Enrichment Evaluations</i>

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVO GERAL	12
1.2	OBJETIVOS ESPECÍFICOS	13
1.3	JUSTIFICATIVA	13
1.4	METODOLOGIA	13
1.5	RECURSOS DE HARDWARE E SOFTWARE	14
1.6	CRONOGRAMA	14
2	CUSTOS DE TROCA	16
2.1	CUSTOS DE SELEÇÃO	17
2.2	CUSTOS DE ATIVAÇÃO	18
2.2.1	Aquisição de Bens Duráveis	18
2.2.2	Aquisições Complementares	18
2.2.3	Aquisição de Fornecedores Especializados	19
2.2.4	Compatibilidade de Sistemas	19
2.2.5	Compatibilidade de Formatos de Informações	19
2.2.6	Aquisições Ligadas aos Efeitos de Rede e Comprometimento da Cadeia	20
2.3	CUSTOS DE CONSTRUÇÃO	20
2.3.1	Custos de Aprendizado e Treinamento	20
2.3.2	Custos do Estabelecimento de um Novo Relacionamento	21
2.4	CUSTOS FORMAIS	21
2.4.1	Custos Contratuais	21
2.4.2	Custos de Troca Relacionados a Programas de Fidelidade	21
2.5	CUSTOS PSICOLÓGICOS	22
2.6	CUSTOS DE OPORTUNIDADE	22
2.7	CUSTOS DE VOLTA	22
3	TOMADA DE DECISÃO COM MÚLTIPLOS CRITÉRIOS	23
3.1	PRINCIPAIS MÉTODOS	23
3.2	<i>ANALYTIC HIERARCHY PROCESS</i>	25
3.2.1	Avaliação dos Critérios	26
3.2.2	Normalização das Comparações	28
3.2.3	Avaliação das Alternativas	30
3.2.4	Escolha da Melhor Alternativa	31
4	MÉTODO DE DESENVOLVIMENTO	32
4.1	PROCESSO DE SOFTWARE	32
4.2	LEVANTAMENTO DE REQUISITOS	33
4.2.1	Requisitos Funcionais	33
4.2.2	Requisitos Não-Funcionais	35
4.3	CASOS DE USO	37
4.4	DIAGRAMA DE PACOTES	41
4.5	DIAGRAMA DE CLASSES	42
5	RESULTADOS OBTIDOS	49

5.1	INTERFACE INICIAL	49
5.1.1	Barra de Menu	49
5.1.2	Barra de Ferramentas	52
5.2	INTERFACE NOVO PROJETO	52
5.3	INTERFACE NOVA AVALIAÇÃO	53
5.4	INTERFACE AVALIAÇÃO	54
5.5	INTERFACE GERENCIAR AVALIAÇÕES	55
5.6	INTERFACE RESULTADO FINAL	56
5.7	INTERFACE MODELO GRÁFICO HIERÁRQUICO	57
5.8	INTERFACE CONTEÚDO	57
5.9	INTERFACE SOBRE	58
5.10	TESTES	58
5.10.1	Testes de Interface	59
5.10.2	Testes de Cálculos	61
6	CONCLUSÃO	63
	REFERÊNCIAS	65
	Apêndice A – TESTES DOS CÁLCULOS DO AHP	68

1 INTRODUÇÃO

Custos de troca são despesas relacionadas ao processo de troca de um fornecedor por outro, de um mesmo produto, e que não necessariamente ocorrem imediatamente após a troca (BURNHAM et al., 2003). Existem custos dos mais diversos tipos, como o de busca por novos fornecedores, aprendizado com o novo produto/serviço e perda de benefícios de programas de fidelidade.

Em tecnologia da informação não é diferente. Usuários estão notoriamente sujeitos a custos de troca e aprisionamento tecnológico: uma vez que é escolhida determinada tecnologia, a troca pode ser extremamente cara. Segundo Shapiro e Varian (1999), o aprisionamento não é absoluto, já que novas tecnologias realmente substituem as antigas, mas os custos de troca podem alterar dramaticamente as estratégias e opções de uma empresa.

Quando um usuário migra de um sistema operacional para outro, por exemplo, ele também tem que trocar a maioria de seus aplicativos. Em muitos casos, documentos, arquivos e demais conteúdos devem ser convertidos para formatos diferentes, e muitos deles não são apenas incompatíveis, como também não há como convertê-los, como é o caso do conjunto Microsoft Windows/Office (HARRIS, 2012).

Portanto, diversos critérios devem ser avaliados para estimar os custos de troca e verificar se a mudança é realmente necessária e quanto esforço seria preciso. Para auxiliar neste processo, existem métodos denominados MCDA (*multiple-criteria decision analysis*), cujo objetivo é avaliar os múltiplos critérios conflitantes para que a melhor solução seja determinada (WEISTROFFER; NARULA, 1989).

1.1 OBJETIVO GERAL

Desenvolver um programa computacional que auxilie a tomada de decisão relacionada à troca de softwares por meio da utilização de um método de tomada de decisão com múltiplos critérios (MCDA).

1.2 OBJETIVOS ESPECÍFICOS

- Estudar os custos de troca envolvidos em tecnologia da informação;
- Propor um modelo que relacione os custos, por meio de um estudo de métodos de apoio a decisão;
- Implementar um sistema com base nas informações levantadas.

1.3 JUSTIFICATIVA

Com diversas tecnologias trazendo diferentes soluções para o mesmo problema, as empresas devem buscar a que atenda melhor as suas necessidades organizacionais. Assim, avaliar os custos de troca da modificação para uma ferramenta que melhor atenda essas necessidades se torna uma tarefa essencial para que a empresa consiga perceber se realmente será benéfica essa troca de tecnologia, podendo definir seu sucesso ou fracasso (JUNIOR, 2006).

Para facilitar essa decisão vários estudos foram realizados descrevendo o problema e os diversos custos relacionados, mas poucos tratam de um problema específico aplicado à algum método de tomada de decisão. Estudos como Engelbert (2010) tratam de custos de troca em tecnologia de informação, e Junior (2006) mostra o sucesso da aplicação do método de tomada de decisão multicritério AHP (*analytic hierarchy process*) em pequenas empresas para problemas genéricos. A criação de uma ferramenta específica permite uma maior facilidade para o tomador de decisão, já que este não precisará um conhecimento prévio de todos os possíveis custos de troca.

1.4 METODOLOGIA

Para a realização deste trabalho, foi primeiramente efetuada uma pesquisa sobre os diferentes custos de troca existentes, suas características e como influenciam os usuários, com a respectiva explicação da influência dos custos de troca no meio tecnológico.

Em sequência, foi percorrido sobre os diferentes métodos de tomada de decisão com múltiplos critérios, com a escolha do melhor modelo a ser utilizado para o escopo deste trabalho. Uma vez determinado, foi implementado um software que auxilia a tomada de decisão, unindo os custos avaliados ao método escolhido.

O propósito deste trabalho foi desenvolver um artefato que aplicasse um método de tomada de decisão considerando os custos de troca em tecnologias de informação levantados -

focado primariamente em software - para auxiliar o tomador de decisão na sua escolha.

1.5 RECURSOS DE HARDWARE E SOFTWARE

Para a execução deste trabalho foram utilizados dois computadores pessoais com processador Intel Core i5 e 4GB de memória, um para cada participante. Estes serviram tanto para a pesquisa quanto para o desenvolvimento do software.

Na confecção do programa, primeiramente as modelagens foram feitas nos softwares *Astah Community* e *Altova UModel*, já a programação realizada utilizado o Java versão 8 junto à plataforma de desenvolvimento NetBeans 8.0. O software é capaz de armazenar decisões passadas em arquivos de texto, não requerendo banco de dados para maior praticidade ao usar a ferramenta. Para cálculos do AHP, foi utilizada a biblioteca de álgebra linear *jblas*, de licença *BSD Revised*. Ícones da interface gráfica são da *FatCow Web Hosting*, cuja licença é *Creative Commons Attribution 3.0*. Para comparação e validação dos cálculos, foi utilizada como ferramenta auxiliar a versão *trial* do software *Expert Choice*.

Máquinas que irão executar o software não necessitam de conectividade com a rede nem quaisquer outros recursos de hardware adicionais, apenas que possuam a JVM (*Java Virtual Machine*) versão 7 ou superior instalada.

1.6 CRONOGRAMA

O cronograma desenvolvido está representado pelo diagrama de Gantt, na Figura 1. Na primeira parte do projeto ocorreu um estudo teórico, enquanto posteriormente o foco foi no desenvolvimento da ferramenta, com base na revisão bibliográfica levantada.

Em *Levantamento de requisitos*, as funcionalidades do sistema foram verificadas. Com base nos requisitos levantados, o sistema foi projetado e implementado. Os testes foram realizados durante sua implementação e, posteriormente, a verificação foi feita por meio de uma comparação de uma outra ferramenta que já é bastante difundida (*Expert Choice*).

A escrita da monografia não foi considerada no cronograma pois foi realizada em paralelo a todas as atividades representadas.

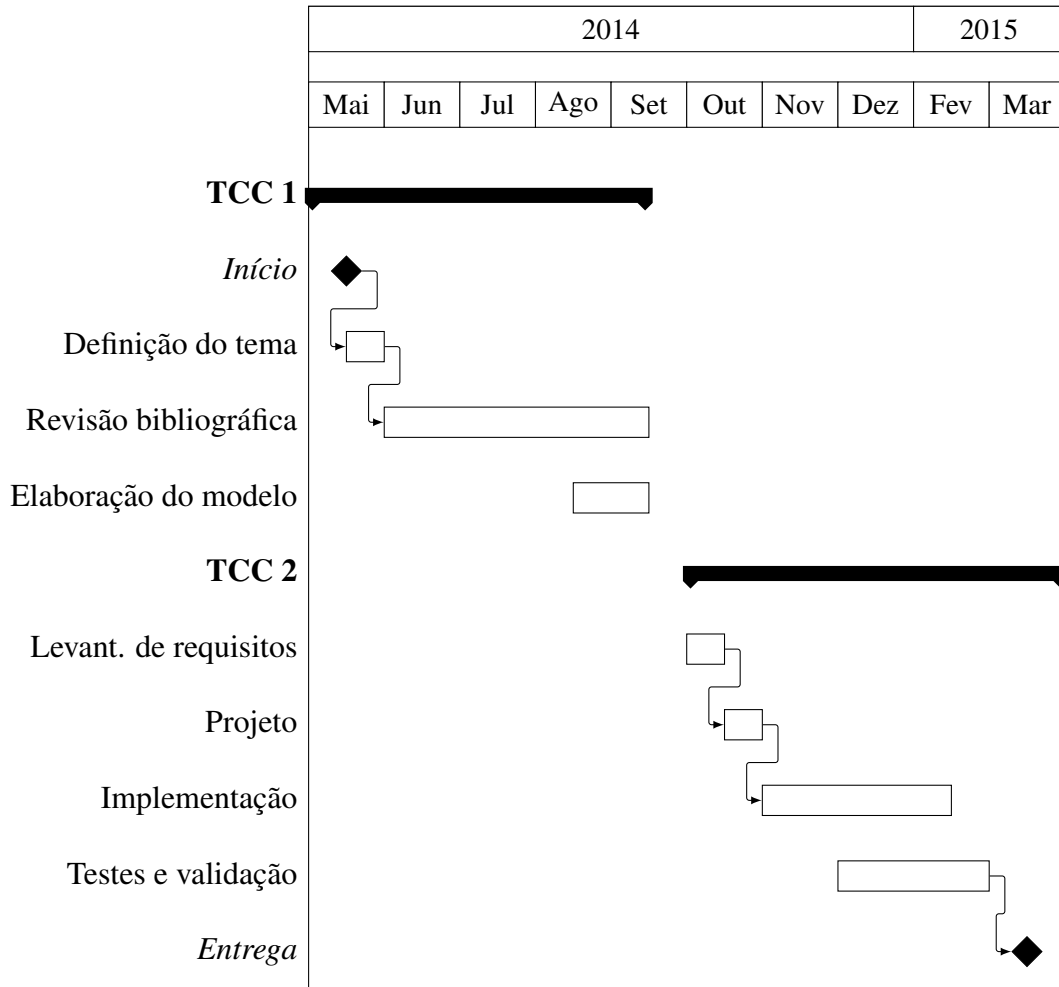


Figura 1: Cronograma das atividades

Fonte: autoria própria.

2 CUSTOS DE TROCA

Durante a transição entre tecnologias utilizadas em uma companhia os respectivos custos de troca devem ser reconhecidos, analisados e gerenciados para que esta organização consiga realizar da melhor maneira possível essa mudanças necessárias à sua sobrevivência no mercado. A análise dos custos de troca tem assim um papel relevante no processo de mudança organizacional no nível estratégico, tático e operacional (ENGELBERT; GRAEML, 2011).

De acordo com Harris (2012) não há uma categorização única para os custos de troca, pois dependendo do produto/serviço que está sendo analisado diferentes custos podem ser observados, estes também possuindo prioridades diferentes dependendo do tomador de decisão e das suas necessidades.

Estes custos estão relacionados tanto com o consumidor, ao determinar o quão custosa é a utilização de uma nova/outra tecnologia, quanto pelo fornecedor, ao fazer com que sua tecnologia possua o menor custo possível para que os clientes migrem para seu produto (ENGELBERT, 2010).

Mas, caso a clientela de determinado fornecedor já seja fiel, este pode aumentar os custos de troca de seus produtos para aprisionar os seus clientes àquela tecnologia, dificultando então a troca para um novo fornecedor, mesmo que a nova solução seja melhor tecnicamente do que a já utilizada. A Figura 2 conceitualiza como os graus dos custos de troca podem variar de acordo com as decisões gerenciais, podendo levar ao aprisionamento à uma tecnologia.

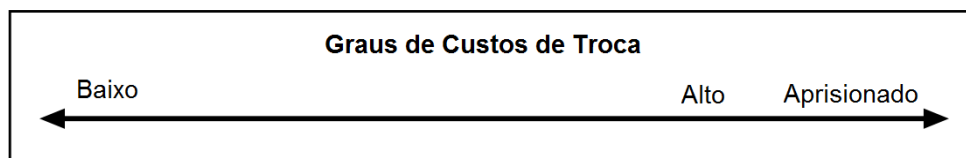


Figura 2: Graus de custos de troca

Fonte: Hess e Ricart (2002).

No estudo de Engelbert (2010), da relação entre os custos de troca em TI, os seguintes custos de troca foram levantados: custos de seleção, custos de ativação, custos de construção,

custos formais, custos psicológicos, custos de oportunidade e custos de volta. A Figura 3 demonstra o modelo de estudo obtido em sua dissertação.

Custos	Investimentos na situação atual	Custos potenciais
de seleção	custo (incluindo tempo e esforço) de buscar, avaliar, testar e selecionar uma solução e fornecedor.	
de ativação	custo (incluindo tempo e esforço) para adquirir, customizar, adaptar, complementar, participar de rede e compatibilizar uma solução ou fornecedor.	
de construção	custo (incluindo tempo e esforço) de aprender a utilizar uma solução e estabelecer relacionamento com o fornecedor.	
formais	taxas (ou depósitos) de inicialização não reembolsáveis.	multas contratuais, perda de benefícios de programas de fidelidade, taxas (ou depósitos) de inicialização, oferta de novas garantias.
psicológicos	custo psicológico de deixar uma marca à qual se é fiel.	custos relacionados a percepções de tempo, esforço, incerteza sobre resultados, risco pessoal e risco organizacional.
de oportunidade	não se aplicam	custo de abandonar benefícios e ganhos potenciais relacionados a complementos e rede de usuários da solução (mesmo que ainda não usufruídos).
de volta	não se aplicam	custo de ter que retornar para a solução anterior no caso de a nova se mostrar insatisfatória.

Figura 3: Modelo de custos de troca em TI

Fonte: Engelbert (2010).

A partir deste modelo serão detalhados os diferentes tipos de custos de troca associados com tecnologias da informação.

2.1 CUSTOS DE SELEÇÃO

Estes custos fazem referência ao processo inicial de busca, avaliação, teste e seleção de um novo fornecedor. Levando em consideração o esforço para determinar o novo fornecedor e os riscos dessa negociação (BURNHAM et al., 2003).

Assim, o simples fato de realizar a pesquisa na busca de um novo sistema a ser utilizado já gera um custo de troca relevante, isso devido ao esforço necessário para buscar as diferentes ferramentas existentes no mercado que possivelmente resolvam aquele problema, além de testá-las e avaliar qual é a melhor solução.

Com relação a essa categoria, percebe-se que quanto maior for esse custo mais difícil é a mudança. A compra contínua com o mesmo fornecedor faz com que os custos de busca e transação sejam reduzidos (FORMAN; CHEN, 2003).

2.2 CUSTOS DE ATIVAÇÃO

São os custos relacionados com a aquisição e adaptação da solução desejada para que seja adequada ao cotidiano da empresa. Assim incluindo os complementos, a infraestrutura e os acessórios necessários para o seu funcionamento (ENGELBERT, 2010).

Com isso, durante a aquisição de uma nova solução, deve ser levada em conta a quantidade de fornecedores daquela tecnologia e dos seus complementos e acessórios, para que o cliente consiga pré-determinar as suas necessidades em busca de um melhor acordo com o fornecedor (SHAPIRO; VARIAN, 1999).

Dentre os custos de ativação definidos por Engelbert (2010), tem-se: aquisição de bens duráveis, aquisições complementares, aquisição de fornecedores especializados, custos relacionados à incompatibilidade de sistemas, à incompatibilidade de formatos de informações, efeitos de rede e comprometimento da cadeia.

2.2.1 AQUISIÇÃO DE BENS DURÁVEIS

Estes custos estão relacionados com o gasto inicial necessário para a compra, implantação e utilização da solução (ENGELBERT; GRAEML, 2011). No caso de sistemas de TI, há especificações mínimas de recursos de hardware para que um software consiga funcionar. Assim, em determinados casos a mudança para um novo sistema gera a necessidade de mudança para um novo hardware mais robusto que consiga manter aquele sistema em funcionamento.

2.2.2 AQUISIÇÕES COMPLEMENTARES

Quando existem produtos complementares, a decisão da compra de um certo produto pode acarretar em uma “trava” na compra de outros produtos relacionados ou, no mínimo, criar novos custos quanto aos diferentes fornecedores desses complementos (HARRIS, 2012).

Um exemplo característico desse tipo de custo são as impressoras de jato tinta. Usualmente são feitas promoções do produto, diminuindo seu preço, mas dificilmente são realizadas promoções dos cartuchos de tinta, essenciais para o funcionamento da impressora (BACIC, 2003).

2.2.3 AQUISIÇÃO DE FORNECEDORES ESPECIALIZADOS

Com a aquisição de produtos disponibilizados por um único fornecedor, ao longo prazo a organização se torna dependente desse fornecedor, o que gera altos custos de troca para aquele respectivo consumidor (FORMAN; CHEN, 2003).

2.2.4 COMPATIBILIDADE DE SISTEMAS

Na computação, usualmente sistemas se utilizam de outros para realizar diversas operações, essa característica gera um vínculo que deve ser mantido ou recriado na troca do sistema para a sua correta utilização pois, com as diferentes soluções tecnológicas para a área de TI, é necessário que diferentes sistemas consigam “conversar” entre si, com uma transferência e leitura de dados de forma simples e eficiente entre os sistemas envolvidos (ENGELBERT, 2010).

A exemplo disso tem-se o fato dos computadores tipicamente funcionarem com apenas um SO, limitando então a gama de aplicativos disponíveis para aquela empresa. Isso faz com que uma mudança em uma aplicação que não exista para o SO da organização seja muito mais custosa pela possível necessidade de modificação do próprio SO.

Assim é necessário levar em consideração todas essas “amarras” que podem surgir na troca para um produto novo.

2.2.5 COMPATIBILIDADE DE FORMATOS DE INFORMAÇÕES

O armazenamento das informações representa uma função importante dentro das organizações. Assim, a troca do formato atual para outro pode se tornar extremamente custosa devido aos riscos de perda de dados e a quantidade de informações já armazenadas (HESS; RICART, 2002).

Para diminuir esses custos para os clientes, os fornecedores podem fazer com que seus sistemas leiam e gravem no formato dos seus concorrentes que possuem uma base maior de clientes do que a sua (ENGELBERT; GRAEML, 2011). A Microsoft fez isso quando o mercado de editores de texto era dominado pelo WordPerfect, habilitando a manipulação dos arquivos gerados pelo WordPerfect por meio do seu editor de texto (SHAPIRO; VARIAN, 1999).

2.2.6 AQUISIÇÕES LIGADAS AOS EFEITOS DE REDE E COMPROMETIMENTO DA CADEIA

Com as redes de usuários de uma tecnologia, esta se mantém mais forte no mercado devido ao grande número de clientes e à criação de novas tecnologias que complementem aquela em específico. Como é o caso das *softhouses*, que tendem a criar softwares que sejam compatíveis com o padrão dominante, assim, se este for o padrão *Windows*, serão criados mais sistemas para este padrão reforçando assim a sua influência no mercado (ENGELBERT, 2010).

Com isso, os custos de troca são mais marcantes nos ambientes com rede, pois trocar para um sistema que não faz parte da grande rede de usuários pode significar um aumento nos custos; pelo contrário, a adoção da tecnologia que possui uma maior rede pode significar maior aprisionamento àquela solução.

2.3 CUSTOS DE CONSTRUÇÃO

São os custos relacionados com as pessoas, o aprendizado e o tempo necessário para aperfeiçoar a utilização da ferramenta escolhida (BURNHAM et al., 2003). Assim tem-se custos de treinamento e aprendizagem, para garantir o mínimo de eficiência na utilização do novo sistema e o custo de geração do relacionamento com o novo fornecedor (ENGELBERT, 2010).

2.3.1 CUSTOS DE APRENDIZADO E TREINAMENTO

Muitos dos sistemas não são simplesmente consumidos pela empresa e descartados. Normalmente, ao serem incorporados na organização passam a fazer parte do seu dia-a-dia, sendo necessária uma aprendizagem contínua para que seja possível obter todos os benefícios do novo produto (HARRIS, 2012).

Assim, com diferenças relevantes em interfaces e metodologias, a troca de um sistema gera a necessidade de uma re-aprendizagem de um processo interno da empresa, agora com a utilização da nova ferramenta, sendo que, quanto maior for o tempo de utilização da tecnologia anterior, maior será esse custo. Isso devido à facilidade já estabelecida anteriormente na execução daquela atividade.

2.3.2 CUSTOS DO ESTABELECIMENTO DE UM NOVO RELACIONAMENTO

Com contínuas aquisições do mesmo fornecedor, vínculos de conforto e confiança são criados o que pode beneficiar ambas as partes com melhores condições de negociações (BURNHAM et al., 2003). Mas com a troca de um produto esse vínculo é perdido e serão necessário mais gastos, tempo e dinheiro, para garantir tal relação com o novo fornecedor.

2.4 CUSTOS FORMAIS

São custos que estão formalizados, seja em contratos, planos de fidelidade, etc. definidos entre o cliente e o fornecedor (ENGELBERT, 2010). Com tais formalidades os fornecedores conseguem prender os clientes através de custos adicionais, como multas ou perda de vantagens, caso o consumidor venha a trocar o produto/serviço por outro.

2.4.1 CUSTOS CONTRATUAIS

Cada serviço e produto possui o seu respectivo contrato de aquisição ou utilização, este devendo ser analisado e levado em consideração pois determinará como poderá ser utilizado, com a descrição dos respectivos direitos e deveres da organização (HARRIS, 2012).

Por exemplo, em uma venda de um plano de telefonia móvel, alguns benefícios podem ser dados aos clientes caso escolham fechar o contrato com a empresa, mas se por ventura decidirem cancelar o contrato antes de um prazo estipulado estarão sujeitos ao pagamento de uma multa previamente determinada.

Esse tipo de custo pode prender o cliente em um produto ou serviço durante o período de duração do contrato. Mas, caso o cliente e o vendedor conheçam um ao outro e não dependam tanto da outra parte, os dois podem sair ganhando com uma formulação de contrato que atenda as necessidades das duas partes interessadas. O problema é quando uma das partes é dependente da outra, fazendo com que fique a mercê de um contrato que possa aumentar ainda mais o custo da utilização de um determinado produto ou serviço.

2.4.2 CUSTOS DE TROCA RELACIONADOS A PROGRAMAS DE FIDELIDADE

Estes custos são parte de uma estratégia totalmente empresarial (SHAPIRO; VARIAN, 1999) que busca fidelizar os seus clientes por meio de brindes, vantagens, recompensas, etc. Descontos em atrações culturais caso a pessoa possua o cartão de determinado banco, vantagens

em compras acima de determinado valor, bônus para próximas compras, etc. enquadram-se nessa categoria.

Estes custos podem ser observados, por exemplo, quando um cliente decide comprar do mesmo fornecedor visto que receberá brindes por compras anteriores. Assim, as experiências que o cliente tinha anteriormente irão influenciar as futuras decisões, aumentando o custo de troca para com novos fornecedores.

2.5 CUSTOS PSICOLÓGICOS

São custos relacionados com os relacionamentos pessoais, as experiências de participar de uma rede de usuários, a “sobrevalorização” de determinado produto, a aversão da utilização do novo sistema e a incerteza da real eficiência da nova solução (ENGELBERT, 2010).

Apesar da exaustiva pesquisa realizada para a compra de um novo produto, há sempre a incerteza se aquele produto será bem aceito no cotidiano da empresa e se cumprirá corretamente o que foi proposto a fazer e assim melhorando os processos da organização.

Isso porque, durante o processo de propaganda, apenas as características boas dos produtos são apresentadas pelos vendedores, fazendo com que outras fontes de informações, tais como *ratings* de qualificações de usuários sobre aquele sistema, sejam necessárias para que se possa determinar a qualidade daquele produto (HARRIS, 2012).

2.6 CUSTOS DE OPORTUNIDADE

São custos relacionados com as potenciais vantagens de outras soluções que não poderão ser obtidas devido a escolha de determinada solução. Tais custos podem ser a perda dos benefícios de se fazer parte de uma grande rede de usuários ou a perda de potenciais complementos que não poderão ser utilizados com o produto escolhido (HESS; RICART, 2002).

2.7 CUSTOS DE VOLTA

Por fim, os custos de volta fazem referência aos possíveis gastos para reverter para a metodologia utilizada anteriormente caso a solução selecionada não atenda as necessidades da organização (HESS; RICART, 2002).

3 TOMADA DE DECISÃO COM MÚLTIPLOS CRITÉRIOS

A tomada de decisão é o processo de escolher o que fazer considerando as possíveis consequências de diferentes escolhas (BROCKMAN; RUSSELL, 2009). Está relacionada a diferentes pontos de vista, alguns favoráveis e outros contra à decisão que está em análise. Esses pontos de vista podem ser definidos como critérios.

De acordo com Figueira et al. (2005), por muitos anos a única maneira de solucionar um problema de decisão foi através da utilização de um único critério, misturando os aspectos multidimensionais da decisão em uma única escala de medida, sendo esta uma maneira não natural de se olhar para o problema.

Para que a tomada de decisão fosse mais intuitiva, levando em consideração todas as pluralidades de pontos de vista, surgiram estudos da tomada de decisão com múltiplos critérios (*MCDA*). A abordagem deste método consiste em um conjunto finito ou infinito de ações (como alternativas, soluções), dois ou mais critérios e um tomador de decisão. Com esses três elementos básicos, o *MCDA* ajuda a tomar decisões em termos de escolha, ranqueamento e ordenação das ações (FIGUEIRA et al., 2005).

Vários métodos *MCDA* foram criados e utilizados nas últimas décadas. Velasquez e Hester (2013) afirmam que os mais comuns são MAUT, AHP, Fuzzy, CBR e DEA. Cada um deles possui vantagens, desvantagens e áreas de aplicação recomendadas. Uma visão geral de cada um deles será apresentada na próxima sessão.

3.1 PRINCIPAIS MÉTODOS

O método MAUT (*Multi-Attribute Utility Theory*) tem sido muito utilizado para auxiliar decisões em problemas do mundo real. Pode decidir o melhor curso de ação em um dado problema atribuindo uma utilidade para cada possível consequência e calculando a melhor utilidade possível (KONIDARI; MAVRAKIS, 2007). Este método leva em conta a incerteza, mas requer uma grande quantidade de dados de entrada, podendo não estar sempre disponível

para determinados tipos de problema. Também requer precisão das preferências do tomador de decisão.

Outro método bastante utilizado, e que também é relacionado a problemas reais é o AHP (*Analytic Hierarchy Process*). Funciona através de comparações de critérios em pares e depende do julgamento de especialistas para derivar escalas de prioridade (SAATY, 2008). Sua principal vantagem é a facilidade de uso, onde o tomador de decisão compara alternativas em pares. Possui uma estrutura hierárquica que pode se ajustar a problemas de diversos tamanhos. Embora seja necessária uma razoável quantidade de dados para efetuar comparações, não chega perto do MAUT nesse quesito. É sujeito a inconsistências no julgamento de critérios, já que a comparação é sempre feita aos demais, sem identificar os pontos fortes e fracos destes individualmente (KONIDARI; MAVRAKIS, 2007).

A teoria *fuzzy* é uma extensão da teoria clássica de conjuntos, que permite a solução de problemas relacionados à manipulação de dados imprecisos (BALMAT et al., 2011). Portanto, em suas aplicações também se estende à análise de decisão com múltiplos critérios. Com algumas regras, permite manipular problemas de alta complexidade. O problema é que pode ser difícil desenvolver um sistema fuzzy e ainda assim requer várias simulações para que possa ser usado no mundo real.

O CBR (*Case-Based Reasoning*) é uma abordagem para solução de problemas que enfatiza o papel de uma experiência anterior durante a solução de problemas. Isto é, reutilizando ou adaptando soluções de problemas semelhantes (MANTARAS et al., 2005). Essa é uma de suas principais vantagens: precisa de pouco esforço para coleta de dados e manutenção da base de dados. Além disso, melhora com o passar do tempo ao adicionar novos casos aos existentes. Mas ao mesmo tempo que a base já existente possui vantagens, deve-se cuidar com os dados inseridos já que dados inconsistentes serão considerados para solucionar problemas futuros.

O método DEA (*Data Envelopment Analysis*) usa uma técnica de programação linear para medir as eficiências relativas de alternativas (THANASSOULIS et al., 2012). Este método avalia as eficiências entre si, com a mais eficiente tendo uma avaliação de 1.0, sendo todas as demais uma fração disso. Permite manipulação de múltiplas entradas e saídas, e a eficiência pode ser analisada e quantificada. Uma grande desvantagem é que não lida com dados imprecisos, e todos os dados de entrada e saída devem ser conhecidos. É um método utilizado quando a eficiência deve ser comparada.

Existem diversos outros métodos, mas que não são tão utilizados quanto os já citados. Por exemplo, o SMART (*Simple Multi-Attribute Rating Technique*) é uma versão simplificada do MAUT, requerendo um menor esforço dos tomadores de decisão. ELECTRE (*Elimination*

and Choice Expressing the Reality) é um método baseado em análise de concordância que leva em consideração incerteza e imprecisão, mas seus resultados não podem ser diretamente identificados. PROMETHEE (*Preference Ranking Organization Method for Enrichment Evaluations*) é semelhante ao ELECTRE, de uso mais fácil, porém não fornece um método claro de atribuição de pesos (VELASQUEZ; HESTER, 2013).

3.2 ANALYTIC HIERARCHY PROCESS

O AHP é um método de tomada de decisão com múltiplos critérios desenvolvido por Saaty (1977). Possui uma série de vantagens e desvantagens se comparado aos demais métodos (ver seção 3.1). Este foi considerado o método mais adequado para o problema da tomada de decisão em custos de troca, pelos seus seguintes benefícios:

- Fácil de usar: o tomador de decisão não precisa de conhecimentos avançados para utilizar a ferramenta ou analisar os resultados (VELASQUEZ; HESTER, 2013).
- Escalável e de estrutura hierárquica: permite se ajustar a problemas de diversos tamanhos (VELASQUEZ; HESTER, 2013).
- Suporta decisão em grupo: permite, através da média geométrica das comparações individuais, que mais de um decisor avalie o problema (ZAHIR, 1999).
- Possui uma vasta área de aplicação: inclui gerenciamento de recursos, planejamento, efetividade, análise de benefício e risco (OGUZTIMUR, 2010).
- É possível medir a consistência dos julgamentos do tomador de decisão (OGUZTIMUR, 2010).

O AHP é organizado de modo hierárquico, o que facilita sua compreensão e avaliação. O primeiro nível possui a meta, ou seja, o objetivo que se deseja atingir. Neste caso, seria a escolha do software que possui o maior benefício, levando-se em conta os custos de troca, sendo este o mais vantajoso para a organização. O segundo nível corresponde aos critérios a serem avaliados, que equivalem aos custos de troca, já levantados no presente trabalho. Estes são comparados entre si, em pares, para a definição dos pesos que terão (MARINS et al., 2009). No último nível, estão as alternativas, que seriam os softwares possíveis. A Figura 4 é um exemplo da estrutura do método no modelo proposto com quatro critérios e três alternativas, que pode variar dependendo do caso. O AHP será visto com mais detalhes na próxima subseção.

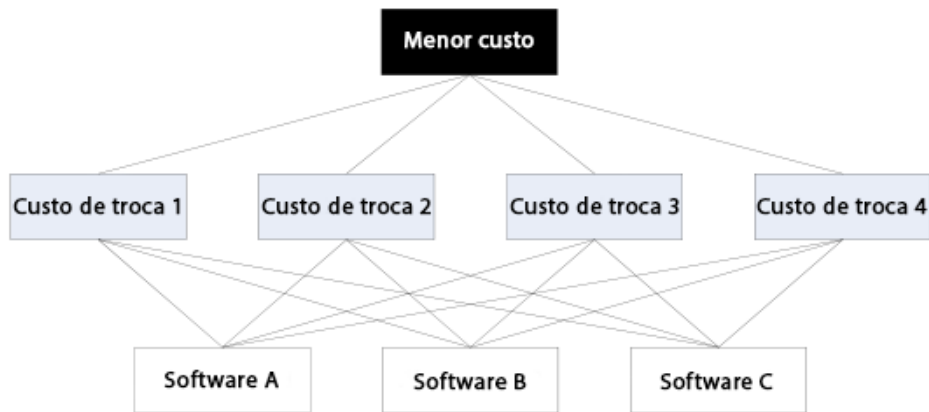


Figura 4: Estrutura do AHP

Fonte: Adaptado de Vargas (2010).

3.2.1 AVALIAÇÃO DOS CRITÉRIOS

Com os critérios definidos - sendo estes os custos de troca -, deve-se realizar julgamento paritários entre eles. O número de comparações é definido pelo número de elementos a serem comparados (MARINS et al., 2009). Sua relação de acordo com o número de critérios está representada na Tabela 1.

Número de critérios	1	2	3	4	5	6	7	n
Número de comparações	0	1	3	6	10	15	21	$\frac{n(n-1)}{2}$

Tabela 1: Quantidade de comparações paritárias no método AHP

Fonte: autoria própria.

As comparações são realizadas em uma escala (Tabela 2), proposta por Saaty (1995). Os valores variam de 1 a 9, que vão de igual importância até extrema importância de um elemento sobre o outro. É recomendável que se utilizem os valores ímpares para assegurar uma maior distinção, recorrendo aos valores pares em caso de necessidade de uma análise mais criteriosa, como o caso em que vários avaliadores discutem e chegam em consenso em um valor intermediário (VARGAS, 2010). Critérios empatados podem ser refinados através da variação do julgamento com intensidade 1 utilizando valores com casas decimais.

Este é um processo subjetivo. A opinião e experiência do tomador de decisão contam para a avaliação (MORAES; SANTALIESTRA, 2008). Os custos de troca devem ser avaliados da seguinte maneira: qual destes é menos custoso na decisão final da escolha de um novo software, se comparado a solução atual? Se o critério A é um pouco mais favorável neste

aspecto que o critério B, uma avaliação possível para este seria 3 (importância moderada).

Escala	Definição	Explicação
1	Ambos elementos são de igual importância.	Ambos elementos contribuem com a propriedade de igual forma.
3	Moderada importância de um elemento sobre o outro.	A experiência e a opinião favorecem um elemento sobre o outro.
5	Forte importância de um elemento sobre o outro.	Um elemento é fortemente favorecido.
7	Importância muito forte de um elemento sobre o outro.	Um elemento é muito fortemente favorecido sobre o outro.
9	Extrema importância de um elemento sobre o outro.	Um elemento é favorecido pelo menos com uma ordem de magnitude de diferença.
2, 4, 6, 8	Valores intermediários.	Usados como valores de consenso entre as opiniões.
1,1 - 1,9	Elementos empatados; graduação fina.	Quando os elementos estão muito próximos, há como atribuir uma leve diferença.

Tabela 2: Escala de relativa importância

Fonte: Moraes e Santalíestra (2008).

As comparações dos critérios podem ser feitas em uma matriz onde os elementos da diagonal são sempre 1 (comparação do critério com ele mesmo). O julgamento é feito da linha atual em relação a sua coluna. Caso o critério da coluna possua um peso maior de acordo com o julgamento, seu valor invertido deve ser utilizado.

Apenas metade da tabela precisa ser preenchida (delimitada pela diagonal). A outra metade deve ser preenchida com seus valores recíprocos, já que a comparação é a mesma, de modo inverso.

A Tabela 3 exemplifica o esquema. Seja Avaliação 1, Avaliação 2 e Avaliação 3 um julgamento que corresponde aos valores da Tabela 2, ou seja, um valor entre 1 e 9. O tomador de decisão julgou que o Custo de troca 1 possui menos impacto que os outros dois custos. Já o Custo de troca 2 possui maior impacto que o custo 3, portanto, sua avaliação é recíproca (invertida). A diagonal inferior é preenchida automaticamente, onde os valores a_{ij} (sendo i linha e j coluna), são preenchidos automaticamente pela fórmula $a_{ji} = \frac{1}{a_{ij}}$. Convém notar que todos os elementos são positivos, isto é, $a_{ij} > 0$.

	Custo de troca 1	Custo de troca 2	Custo de troca 3
Custo de troca 1	1	Avaliação 1	Avaliação 2
Custo de troca 2	$\frac{1}{\text{Avaliação 1}}$	1	$\frac{1}{\text{Avaliação 3}}$
Custo de troca 3	$\frac{1}{\text{Avaliação 2}}$	Avaliação 3	1

Tabela 3: Matriz de comparação com três critérios

Fonte: autoria própria.

3.2.2 NORMALIZAÇÃO DAS COMPARAÇÕES

Com a matriz com as comparações completa, é necessário normalizá-la. O método descrito abaixo é uma aproximação do vetor Eigen. A maioria dos casos práticos utiliza essa aproximação para simplificar o processo de cálculo, pois a diferença entre o valor real e o aproximado é inferior à 10% (KOSTLAN, 1991 apud VARGAS, 2010).

O primeiro passo consiste em somar o peso total de cada coluna e dividir cada valor pelo total. Isso fará com que a soma de cada coluna seja 1. Este processo está representado na Tabela 4, onde os valores da matriz de comparação foram criados simulando uma possível situação real.

	Custo de troca 1	Custo de troca 2	Custo de troca 3
Custo de troca 1	$1 \div \frac{21}{5} = 0,24$	$5 \div 13 = 0,38$	$\frac{1}{3} \div \frac{31}{21} = 0,23$
Custo de troca 2	$\frac{1}{5} \div \frac{21}{5} = 0,05$	$1 \div 13 = 0,08$	$\frac{1}{7} \div \frac{31}{21} = 0,10$
Custo de troca 3	$3 \div \frac{21}{5} = 0,71$	$7 \div 13 = 0,54$	$1 \div \frac{31}{21} = 0,68$
Total	$\frac{21}{5}$	13	$\frac{31}{21}$

Tabela 4: Normalização da matriz de comparação

Fonte: autoria própria.

A aproximação do vetor de prioridade é obtida pela média da soma dos valores obtidos em cada linha. Este passo define o peso dos critérios que estão sendo comparados de acordo com a preferência do tomador de decisão. No exemplo da Tabela 5, que utiliza dados da Tabela 4, o primeiro critério é o que é considerado pelo decisor como o mais benéfico em suas avaliações e, portanto, será o que terá mais peso na decisão final.

O AHP permite que os dados sejam verificados para a eliminação de inconsistências. Se o critério A foi avaliado com um maior peso que o critério B, o critério B com um peso maior que o critério C, é de se esperar que o critério A tenha um maior peso em relação ao critério C. Caso contrário, encontra-se uma inconsistência.

	Vetor de Eigen
Custo de troca 1	$[0,24 + 0,38 + 0,23]/3 = 0,28$
Custo de troca 2	$[0,05 + 0,08 + 0,10]/3 = 0,07$
Custo de troca 3	$[0,71 + 0,54 + 0,68]/3 = 0,64$

Tabela 5: Vetor de prioridade (Eigen)

Fonte: autoria própria.

Como o método permite que o tomador de decisão diga em uma escala o quão preferível é um critério em comparação a outro, é normal que esses tipos de inconsistência ocorram (MORAES; SANTALIESTRA, 2008). Saaty (1980) propôs que o índice de inconsistência deve ser inferior a 10% para ser aceitável.

O primeiro passo consiste na obtenção do valor principal Eigen (λ_{Max}), obtido pela soma dos produtos entre cada elemento do vetor Eigen e a soma total de sua coluna correspondente (VARGAS, 2010). Dando continuidade ao exemplo feito até agora, tem-se o seguinte resultado:

$$\lambda_{Max} = (0,28 \times \frac{21}{5}) + (0,07 \times 13) + (0,64 \times \frac{31}{21}) = 3,1$$

Para o cálculo do índice da consistência utiliza-se a seguinte fórmula, onde λ_{Max} já se tem calculado e n corresponde ao número de critérios:

$$CI = \frac{\lambda_{Max} - n}{n - 1}$$

No exemplo visto até agora, obtemos $CI = 0,05$. Esse índice é usado para calcular a taxa de consistência (CR), que deve satisfazer a seguinte condição comentada previamente:

$$CR = \frac{CI}{RI} < 0,1$$

Onde RI , o índice de consistência aleatória, está representado na Tabela 6. Este índice é fixo e leva em consideração o número de critérios.

Como exemplo:

$$CR = \frac{CI}{RI} = \frac{0,05}{0,52} = 0,09$$

N	3	4	5	6	7	8	9	10	11	12	13	14	15
RI	0,52	0,89	1,11	1,25	1,35	1,40	1,45	1,49	1,51	1,54	1,56	1,57	1,58

Tabela 6: Valores de RI propostos por Saaty

Fonte: Moraes e Santaliestra (2008).

O que satisfaz a condição de *CR* ser menor que 10%. Portanto, os julgamentos subjetivos realizados foram consistentes.

3.2.3 AVALIAÇÃO DAS ALTERNATIVAS

Do mesmo modo que foi realizada comparação em pares com os critérios em relação ao seu precedente - neste caso, com o objetivo -, com as alternativas também se realizam comparações em pares em relação ao critério precedente. Isto significa que cada um dos softwares que se está avaliando para a possível troca, se faz uma comparação com cada um dos custos de troca que está levando em consideração na decisão.

Supondo que existam quatro custos de troca e três softwares, cada um dos softwares será comparado com os quatro custos de troca. Teremos, portanto, uma matriz de comparação dos custos de troca em relação ao objetivo e quatro matrizes para comparação de software com cada um dos custos.

A Tabela 7 exemplifica uma dessas comparações. Os três softwares ponderados são avaliados em relação ao custo de troca 1.

	Software A	Software B	Software C
Software A	$1 \div \frac{23}{15} = 0,65$	$3 \div \frac{13}{3} = 0,69$	$5 \div 9 = 0,56$
Software B	$\frac{1}{3} \div \frac{23}{15} = 0,22$	$1 \div \frac{13}{3} = 0,23$	$3 \div 9 = 0,33$
Software C	$\frac{1}{5} \div \frac{23}{15} = 0,13$	$\frac{1}{3} \div \frac{13}{3} = 0,08$	$1 \div 9 = 0,11$
Total	$\frac{23}{15}$	$\frac{13}{3}$	9

Tabela 7: Matriz de comparação das alternativas em relação a custo de troca 1

Fonte: autoria própria.

Do mesmo modo, obtém-se o vetor de prioridade, que transforma as avaliações efetuadas em números (pesos). Este procedimento se encontra na Tabela 8.

	Vetor de Eigen
Software A	$[0,65 + 0,69 + 0,56]/3 = 0,63$
Software B	$[0,22 + 0,23 + 0,33]/3 = 0,26$
Software C	$[0,13 + 0,08 + 0,11]/3 = 0,11$

Tabela 8: Vetor de prioridade (Eigen) das alternativas em relação a custo de troca 1

Fonte: autoria própria.

3.2.4 ESCOLHA DA MELHOR ALTERNATIVA

Assim como feito com os critérios, o cálculo da taxa de consistência também se aplica para cada uma dessas matrizes. Esse cálculo foi apresentado na subseção 3.2.2.

Com todas as comparações em pares já efetuadas, é possível agora ver qual seria a melhor escolha segundo o AHP. O resultado é um valor corresponde a uma porcentagem de qual seria o software indicado, ou ainda em forma de taxa, como o “Software A é 3 vezes mais recomendado que o Software B”.

Essa porcentagem pode ser obtida pela soma dos produtos de pesos de critérios e alternativas. Para, por exemplo, o Software A:

$$\begin{aligned} \text{Escolha Software A} = & [0,28 \times 0,63] + [\text{Peso (Eigen) critério 2} \times \text{Peso (Eigen) alternativa 2}] \\ & + \dots + [\text{Peso (Eigen) critério } n \times \text{Peso (Eigen) alternativa } n] \end{aligned}$$

Convém ressaltar que a tomada de decisão requer um entendimento mais amplo e complexo do que a simples utilização de uma técnica como o AHP (VARGAS, 2010). Esse tipo de técnica auxilia a decisão final, mas não é necessariamente a última palavra. Além disso, mudar de um software para um novo irá gerar custos que não ocorreriam ao permanecer na solução atual. Portanto, uma análise de benefícios da mudança também é recomendada.

4 MÉTODO DE DESENVOLVIMENTO

Utilizando-se do referencial teórico apresentado nos Capítulos 2 e 3, foi criado um software que facilitasse para os tomadores de decisão a escolha entre sistemas de uma empresa. Esse capítulo busca mostrar as metodologias utilizadas para a produção do sistema, bem como o projeto do mesmo.

4.1 PROCESSO DE SOFTWARE

Segundo Sommerville (2007), um processo de software é um conjunto de atividades que leva à produção de um produto de software. Dentre os principais modelos de processos, estão o modelo em cascata e o desenvolvimento evolucionário.

O modelo em cascata possui fases bem definidas, onde a próxima fase deve se iniciar sempre com o término da anterior. Este tipo de modelo é recomendável para projetos onde os requisitos estão certos desde o início e existe pouca probabilidade de mudança drástica durante o desenvolvimento (SOMMERVILLE, 2007). Como a proposta foi a criação de um software diferenciado e personalizado, focando principalmente o auxílio na mudança de softwares levando em consideração os custos de troca, este modelo foi considerado inadequado, visto que seria importante o *feedback* (dentre eles, do professor orientador), a medida que o software fosse evoluindo.

Um método mais flexível é o de desenvolvimento evolucionário. Este método se baseia na ideia de desenvolvimento de uma implementação inicial, expondo o resultado aos comentários do usuário e refinando esse resultado por meio de várias versões até que seja desenvolvido um sistema adequado. Este é o melhor método para sistemas de pequeno e médio porte (SOMMERVILLE, 2007), portanto, adotado para este trabalho. A possibilidade de adição de diversas funcionalidades à medida da necessidade e *feedback* recebido torna o produto final mais compatível com a necessidade dos usuários.

4.2 LEVANTAMENTO DE REQUISITOS

Os requisitos de um sistema são descrições dos serviços fornecidos pelo sistema e as suas restrições operacionais (SOMMERVILLE, 2007). Estes requisitos estão basicamente divididos entre funcionais e não-funcionais, os quais serão abordados nas subseções seguintes.

4.2.1 REQUISITOS FUNCIONAIS

Os requisitos funcionais descrevem o que o sistema deve fazer. Neste trabalho, estão relacionados a funcionalidades de manipulação de alternativas e critérios, à avaliação em si e o resultado. Nos quadros a seguir, encontram-se os requisitos funcionais levantados para o desenvolvimento deste software.

[RF001] Avaliar critérios
<i>Pré-condições:</i> critérios previamente cadastrados.
<i>Pós-condições:</i> critérios preenchidos.
Descrição: o tomador de decisão avalia os critérios (custos de troca) através de uma comparação em pares.

Quadro 1: Requisito funcional RF001

Fonte: autoria própria.

[RF002] Avaliar alternativas
<i>Pré-condições:</i> alternativas previamente cadastradas.
<i>Pós-condições:</i> alternativas preenchidas.
Descrição: da mesma forma que é realizado com os critérios, as alternativas (softwares) são avaliadas em relação a cada critério.

Quadro 2: Requisito funcional RF002

Fonte: autoria própria.

[RF003] Inserir novos critérios
<i>Pré-condições:</i> objetivo cadastrado.
<i>Pós-condições:</i> um novo critério é cadastrado.
Descrição: novos critérios podem ser inseridos para serem avaliados, caso seja interesse do tomador de decisão.

Quadro 3: Requisito funcional RF003

Fonte: autoria própria.

[RF004] Remover critérios
<i>Pré-condições:</i> critério cadastrado.
<i>Pós-condições:</i> critério removido.
Descrição: critérios que o tomador de decisão considere desnecessários ou que foram inseridos por erro podem ser deletados.

Quadro 4: Requisito funcional RF004

Fonte: autoria própria.

[RF005] Inserir alternativas
<i>Pré-condições:</i> não há.
<i>Pós-condições:</i> uma nova alternativa é cadastrada.
Descrição: alternativas (softwares) podem ser inseridas de acordo com o desejo do usuário.

Quadro 5: Requisito funcional RF005

Fonte: autoria própria.

[RF006] Remover alternativas
<i>Pré-condições:</i> alternativa cadastrada.
<i>Pós-condições:</i> a alternativa é removida.
Descrição: alternativas (softwares) podem ser removidas de acordo com o desejo do usuário.

Quadro 6: Requisito funcional RF006

Fonte: autoria própria.

[RF007] Visualizar resultado
<i>Pré-condições:</i> critérios e alternativas avaliados.
<i>Pós-condições:</i> não há.
Descrição: o usuário visualiza uma tela com informações a respeito da classificação das alternativas, com seus respectivos pesos.

Quadro 7: Requisito funcional RF007

Fonte: autoria própria.

[RF008] Salvar decisão
<i>Pré-condições:</i> avaliação realizada.
<i>Pós-condições:</i> um novo arquivo é criado.
Descrição: a decisão pode ser salva, caso o usuário opte por esta opção.

Quadro 8: Requisito funcional RF008

Fonte: autoria própria.

4.2.2 REQUISITOS NÃO-FUNCIONAIS

Os requisitos não-funcionais não estão relacionados diretamente às funções do sistema, mas sim a questões como confiabilidade e desempenho, além de restrições. Para o desenvolvimento deste software, é importante que as restrições sigam os limites impostos pelo AHP que, por exemplo, aceita um limite máximo de dez critérios. Nos quadros a seguir, encontram-se os requisitos não-funcionais levantados para o desenvolvimento deste software.

[NF001] Validação das avaliações

Descrição: é possível prosseguir com as avaliações se estas tem sido consistentes de acordo com a taxa de consistência do AHP.

Quadro 9: Requisito não-funcional NF001

Fonte: autoria própria.

[NF002] Interface

Descrição: a interface gráfica criada deve utilizar a biblioteca Swing do Java.
--

Quadro 10: Requisito não-funcional NF002

Fonte: autoria própria.

[NF003] Armazenamento

Descrição: dados do projeto e avaliações passadas devem ser armazenadas em arquivos.

Quadro 11: Requisito não-funcional NF003

Fonte: autoria própria.

[NF004] Desempenho

Descrição: um sistema que atinja as configurações necessárias para executar a Java Virtual Machine também deve executar o software.
--

Quadro 12: Requisito não-funcional NF004

Fonte: autoria própria.

[NF005] Número de mínimo de critérios
--

Descrição: o sistema deverá operar com, no mínimo, dois critérios.

Quadro 13: Requisito não-funcional NF005

Fonte: autoria própria.

[NF006] Número de máximo de critérios
--

Descrição: o sistema deverá operar com, no máximo, dez critérios.
--

Quadro 14: Requisito não-funcional NF006

Fonte: autoria própria.

[NF007] Número mínimo de alternativas
--

Descrição: o sistema deverá operar com, no mínimo, duas alternativas.
--

Quadro 15: Requisito não-funcional NF007

Fonte: autoria própria.

[NF008] Número máximo de alternativas
--

Descrição: o sistema deverá operar com, no máximo, dez alternativas.

Quadro 16: Requisito não-funcional NF008

Fonte: autoria própria.

4.3 CASOS DE USO

Com os requisitos levantados, foram elaborados os casos de uso. Os casos de uso são um conjunto de cenários que identificam um roteiro de uso do sistema a ser construído (PRESSMAN, 2011). Um caso de uso narra como um usuário (representado por um ator), usa o sistema para atingir determinado objetivo.

O diagrama de casos de uso está representado na Figura 5, e o detalhamento de cada caso de uso está representado logo em sequência. Existem três casos de usos principais no software, que são manter critérios, manter alternativas e realizar decisão. Tanto em “manter critérios”, como “manter alternativas”, o usuário deve ter total liberdade para edição de acordo com sua preferência, mesmo que o software faça uma pré-sugestão dos custos de troca já levantados. O ator representado em todos eles é o tomador de decisão, único utilizador da ferramenta, que irá manipular critérios e alternativas para poder visualizar o resultado final, que o auxiliará na tomada de decisão entre as diversas alternativas avaliadas.

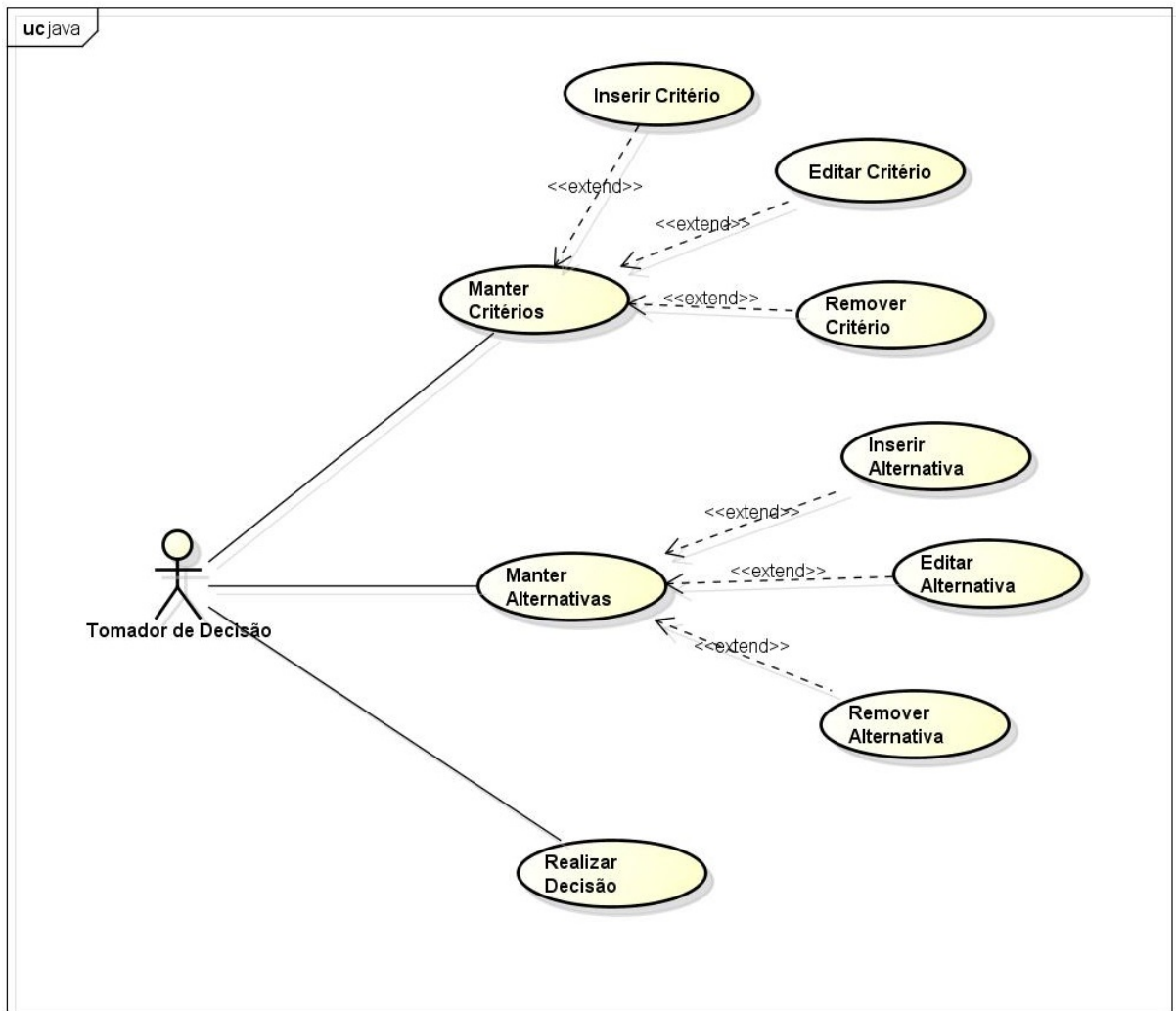


Figura 5: Diagrama de casos de uso

Fonte: autoria própria.

Caso de uso: Manter Critérios
<p>Ator: Tomador de Decisão.</p> <p>Interessado: Tomador de Decisão.</p> <p>Pré-condições: os critérios deverão estar previamente selecionados.</p> <p>Pós-condições: um critério é inserido no sistema.</p> <p>Requisitos correlacionados: RF003 e RF004.</p>
<p>Fluxo Principal</p> <ol style="list-style-type: none"> 1. O tomador de decisão abre um projeto. 2. O tomador de decisão informa o cadastro de um novo critério. 3. O tomador de decisão informa os dados do novo critério. 4. O tomador de decisão confirma o cadastro na janela do sistema. 5. O sistema aguarda pela próxima ação.
<p>Fluxo Alternativo 1</p> <ol style="list-style-type: none"> 2. O tomador de decisão deseja alterar as informações de um critério. <ol style="list-style-type: none"> 2.1 O tomador de decisão busca pelo critério desejado. 2.2 O tomador de decisão informa os novos dados do critério. 2.3 O tomador de decisão confirma a modificação na janela do sistema. 2.4 O fluxo retorna ao passo 5.
<p>Fluxo Alternativo 2</p> <ol style="list-style-type: none"> 2. O tomador de decisão deseja remover um critério. <ol style="list-style-type: none"> 2.1 O tomador de decisão busca pelo critério desejado. 2.2 O tomador de decisão remove o critério desejado. 2.3 O tomador de decisão confirma a remoção do critério. 2.4 O fluxo retorna ao passo 5.

Quadro 17: Caso de uso manter critérios

Fonte: autoria própria.

Caso de uso: Manter Alternativas
<p>Ator: Tomador de Decisão.</p> <p>Interessado: Tomador de Decisão.</p> <p>Pré-condições: as alternativas deverão estar previamente selecionadas.</p> <p>Pós-condições: uma alternativa é inserida no sistema.</p> <p>Requisitos correlacionados: RF005 e RF006.</p>
<p>Fluxo Principal</p> <ol style="list-style-type: none"> 1. O tomador de decisão abre um projeto. 2. O tomador de decisão informa o cadastro de uma nova alternativa. 3. O tomador de decisão informa os dados da nova alternativa. 4. O tomador de decisão confirma o cadastro na janela do sistema. 5. O sistema aguarda pela próxima ação.
<p>Fluxo Alternativo 1</p> <ol style="list-style-type: none"> 2. O tomador de decisão deseja alterar as informações de uma alternativa. <ol style="list-style-type: none"> 2.1 O tomador de decisão busca pela alternativa desejada. 2.2 O tomador de decisão informa os novos dados da alternativa. 2.3 O tomador de decisão confirma a modificação na janela do sistema. 2.4 O fluxo retorna ao passo 5.
<p>Fluxo Alternativo 2</p> <ol style="list-style-type: none"> 2. O tomador de decisão deseja remover uma alternativa. <ol style="list-style-type: none"> 2.1 O tomador de decisão busca pela alternativa desejada. 2.2 O tomador de decisão remove a alternativa desejada. 2.3 O tomador de decisão confirma a remoção da alternativa. 2.4 O fluxo retorna ao passo 5.

Quadro 18: Caso de uso manter alternativas

Fonte: autoria própria.

Caso de uso: Realizar Decisão
<p>Ator: Tomador de Decisão.</p> <p>Interessado: Tomador de Decisão.</p> <p>Pré-condições: alternativas deverão estar previamente cadastradas.</p> <p>Pós-condições: não há.</p> <p>Requisitos correlacionados: RF001, RF002 e RF007.</p>
<p>Fluxo Principal</p> <ol style="list-style-type: none"> 1. O tomador de decisão abre um projeto. 2. O tomador de decisão seleciona o critério ou objetivo que deseja avaliar. 3. O tomador de decisão informa, na comparação em pares, qual será menos custoso. 4. O tomador de decisão confirma o resultado na janela do sistema. 5. O sistema aguarda pela próxima ação.
<p>Fluxo Alternativo</p> <ol style="list-style-type: none"> 3. Os dados fornecidos na comparação não foram válidos. <ol style="list-style-type: none"> 3.1 O programa informa com uma janela de informação que os dados da comparação estavam inconsistentes. 3.2 O fluxo retorna ao passo 3.

Quadro 19: Caso de uso realizar decisão

Fonte: autoria própria.

4.4 DIAGRAMA DE PACOTES

O diagrama de pacotes representa as dependências entre pacotes do sistema. Este diagrama também simplifica diagramas UML complexos (LUJAN-MORA et al., 2002), fornecendo uma visualização de como o software está estruturado.

A Figura 6 representa os pacotes do software e seus relacionamentos. O pacote *il8n* possui classes e arquivos referentes à internacionalização do projeto, que está disponível nos idiomas português, inglês e espanhol. O pacote *visualizacao* possui classes referentes às interfaces gráficas. Já em *modelo* existem classes que representam dados, lógica e funções. Este pacote conta ainda com subdivisões, tendo dentro os pacotes *ahp*, *bean*, *dao* e *ferramentas*.

No pacote *ahp*, estão as classes responsáveis pela lógica do método AHP. Em *bean*, temos classes que armazenam e retornam valores. O pacote *dao* é uma referência ao padrão de

projeto *Data Access Object*, que originalmente separa as regras de negócio do acesso ao banco de dados, mas neste contexto o acesso é feito à arquivos de texto. Já o pacote *ferramentas* abrange classes de outras funcionalidades disponíveis no sistema, apresentado no capítulo 5.

Por fim, o pacote *recursos* possui recursos utilizados no projeto. Neste caso, apenas um tipo de recurso é utilizado, que são imagens que estão no pacote de mesmo nome. Essas imagens são utilizadas nas mais diversas interfaces do sistema, como em menus e ao lado de critérios e alternativas.

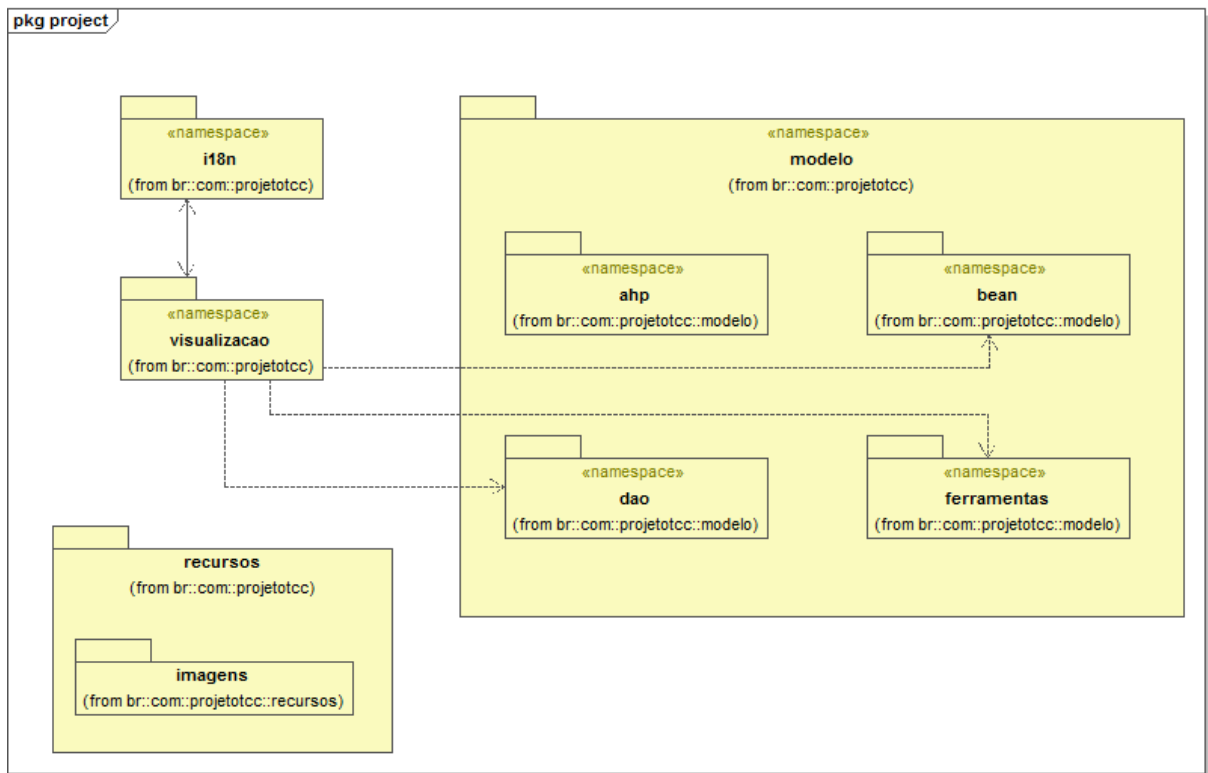


Figura 6: Diagrama de pacotes

Fonte: autoria própria.

4.5 DIAGRAMA DE CLASSES

O diagrama de classes, como o próprio nome sugere, modela classes, incluindo seus atributos, operações e relações e associações com outras classes (PRESSMAN, 2011).

Na Figura 7, está representado o diagrama de classes geral do sistema. Atributos e métodos foram ocultados para permitir todas as classes caberem em um único diagrama. O programa se inicia na classe *Main*, do pacote *principal*. Esta é responsável por instanciar a janela principal que dará início ao software, passando por isso pelo *Controlador*.

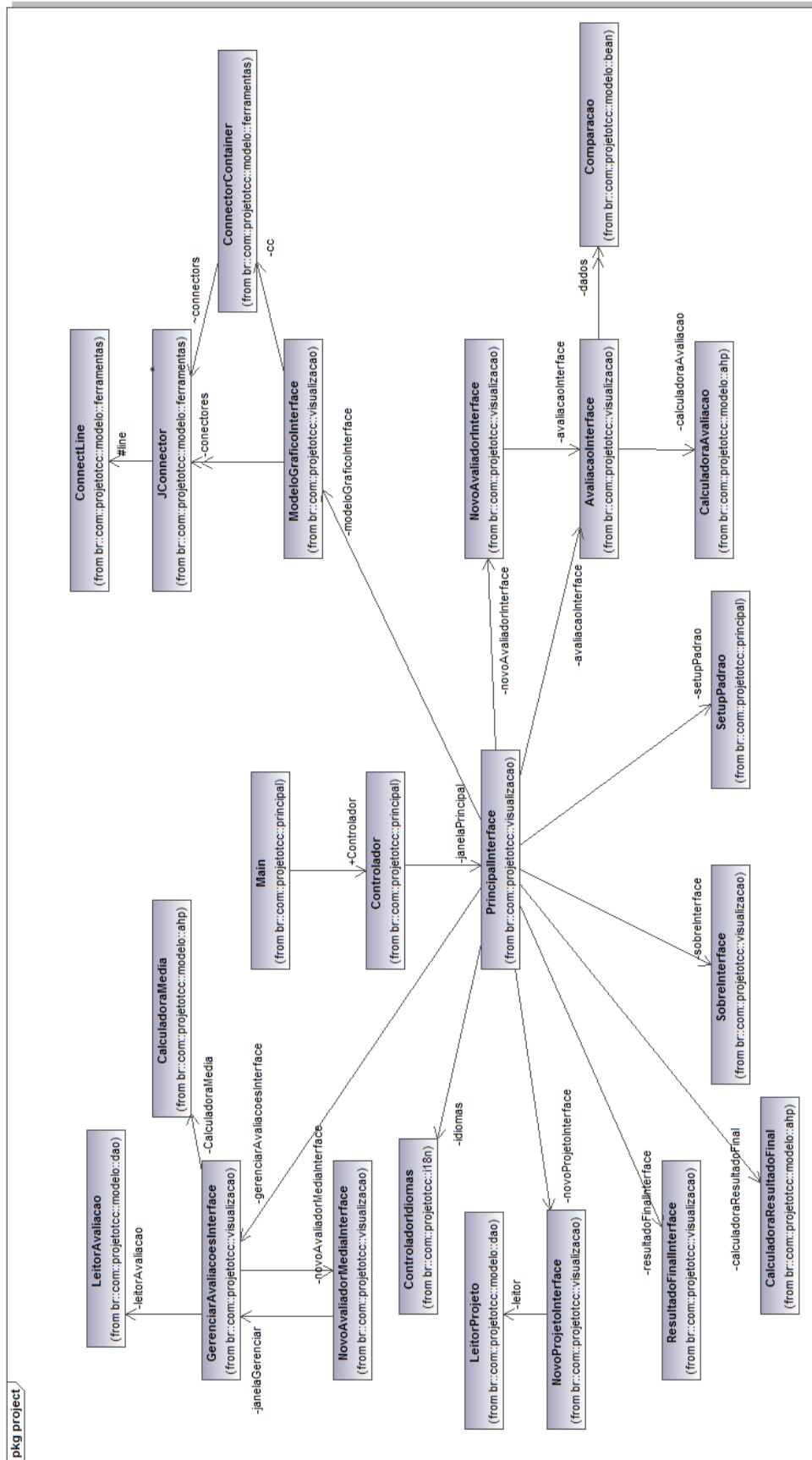


Figura 7: Diagrama de classes: visão geral

Fonte: autoria própria.

O pacote *principal* está representado na Figura 8. O *Controlador* é uma classe que segue o padrão de projeto *Singleton*, que garante que o objeto de uma classe seja único. Neste caso, a janela principal do programa será sempre única. Esta solução foi utilizada para que todas as interfaces que desejassem ter acesso à janela principal para executar alguma ação (como por exemplo, atualizar a árvore de critérios) não precisassem ter como parâmetro a instância de *PrincipalInterface*, além de garantir uma única janela principal.

A última classe desse pacote é o *SetupPadrao*, responsável por preencher os custos de troca e suas descrições ao criar um novo projeto, caso o tomador de decisão opte por esta opção.

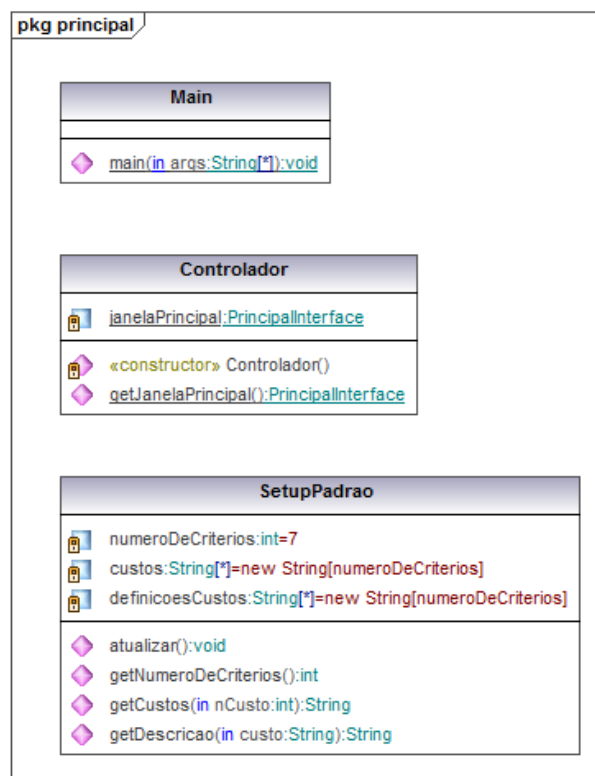


Figura 8: Diagrama de classes: pacote principal

Fonte: autoria própria.

O pacote *ahp* (Figura 9) possui classes cujo prefixo é *Calculadora*, e o que fazem é realizar cálculos do método. A classe *CalculadoraAvaliacao* é a principal delas. Realiza cálculos como taxa e índice de consistência, além do vetor de prioridade (Eigen). Seus cálculos já foram abordados no presente trabalho.

O vetor de prioridade pode ser calculado de duas maneiras. A primeira é através da biblioteca *jblas*, que executa o cálculo exato. A segunda é através de uma aproximação, através de uma média (também já explicado na seção 3.2). A utilização da biblioteca externa

faz com que se ganhe precisão no cálculo, em troca de desempenho, principalmente ao ser executado pela primeira vez. Já o método de média sacrifica a precisão para uma execução mais veloz. Em avaliações com poucos critérios ou alternativas, a diferença de resultados obtidos é imperceptível.

A classe *CalculadoraMedia* é responsável por gerar uma média entre avaliações caso o tomador de decisão deseje¹. E a classe *CalculadoraResultadoFinal* realiza cálculos para obter o software - ou qualquer outro problema analisado -, que proporciona o maior benefício, com base na análise dos custos de troca envolvidos.

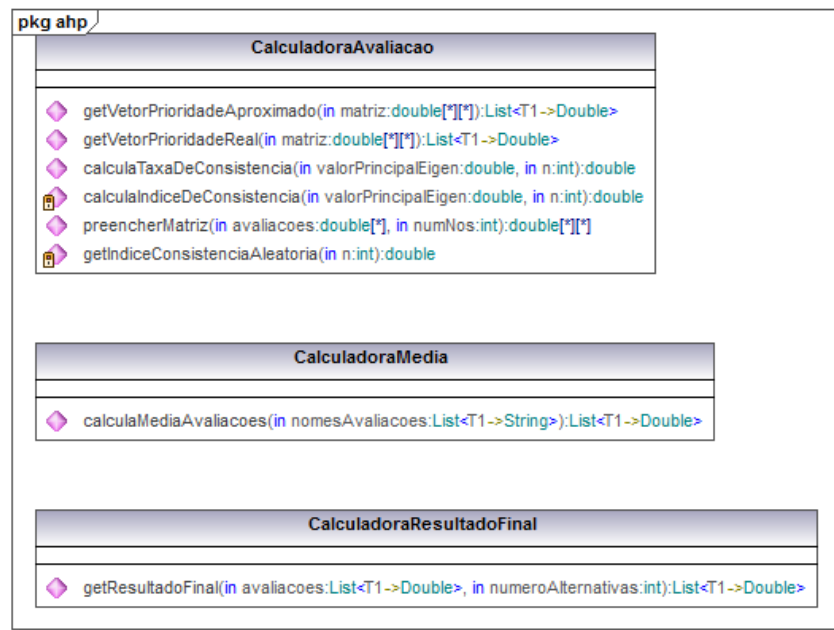


Figura 9: Diagrama de classes: pacote ahp

Fonte: autoria própria.

O software trabalha com três tipos de arquivos, que são de extensão *.projeto*, *.avaliacao* e *.modelo*. Os arquivos *projeto* são responsáveis por armazenar informações referentes ao projeto, que são nome do projeto, status, objetivo geral, critérios, descrições dos critérios e alternativas. Os arquivos *avaliacao* possuem cálculos de avaliações de um usuário de cada nó da árvore, armazenados em um arquivo de texto. Para cada novo usuário, um novo arquivo deste tipo é criado. Os arquivos *modelo* são utilizados na criação de um novo projeto, no qual estão salvos objetivo, critérios e descrições personalizados, caso o usuário opte por um projeto que não se refira à custos de troca ou não deseje um projeto em branco. Esta funcionalidade é

¹A explicação do funcionamento do software, como também de multi-avaliadores, está detalhado no capítulo 5.

uma adição ao projeto original, permitindo um software com mais opções e flexibilidade para o tomador de decisão, que poderá também utilizá-lo em diversos outros contextos.

A classe que trabalha com os arquivos de projeto e modelo é o *LeitorProjeto*, e os arquivos de avaliação é o *LeitorAvaliacao*. Estas classes estão representadas na Figura 10.

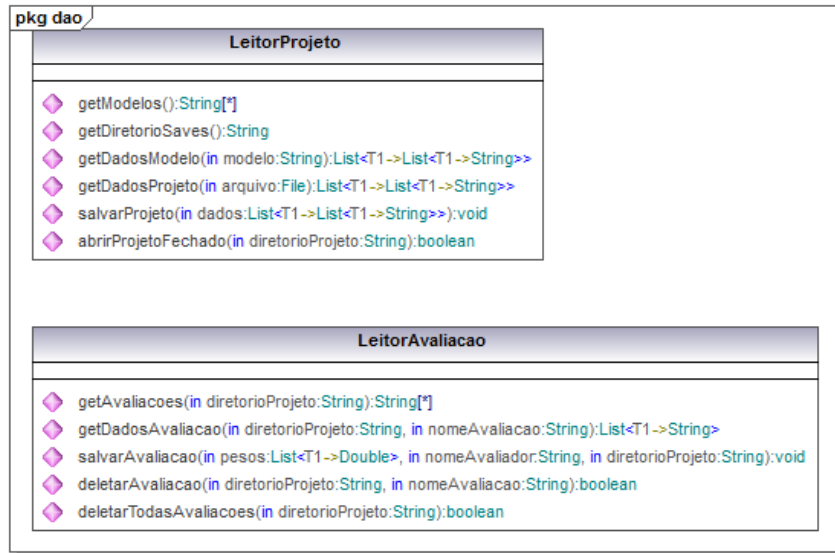


Figura 10: Diagrama de classes: pacote dao

Fonte: autoria própria.

O pacote *visualizacao* possui classes referentes à interface gráfica. Por ser um pacote grande, foi quebrado em dois diagramas, correspondentes às Figuras 11 e 12. Em ambos, alguns atributos e métodos foram ocultados, como por exemplo, a declaração de botões e seus *listeners*.

A primeira parte desse pacote, Figura 11, possui a principal classe desse pacote, a janela principal cujo nome é *PrincipalInterface*. Esta classe possui a árvore de critérios e alternativas, descrições, menus e acesso às funcionalidades do software. Ela também instancia a maior parte das demais interfaces, e possui métodos que manipulam principalmente a árvore de critérios e a lista de descrições.

A classe *ModeloGraficoInterface* demonstra graficamente a estrutura do problema em análise, inserindo o objetivo, critérios e alternativas. *SobreInterface* é uma janela simples que revela os autores do software, bem como sua versão. A classe *ResultadoFinalInterface* exibe o resultado final em forma de um gráfico de barras, e *NovoProjetoInterface* é a janela para criação de um novo projeto.



Figura 11: Diagrama de classes: pacote visualização (parte 1)

Fonte: autoria própria.

Na Figura 12 está representada a continuação desse pacote. *AvaliacaoInterface* é a interface da comparação paritária entre critérios e alternativas. *NovoAvaliadorMediaInterface* é para geração de médias entre avaliações passadas. Possui a referência para *GerenciarAvaliacoesInterface* (responsável pelo gerenciamento de avaliações), para poder usar alguns de seus métodos. A última janela é a classe *NovoAvaliadorInterface*, responsável por receber o nome do novo avaliador e então dar prosseguimento ao processo de avaliação.

O pacote *i18n* é composto apenas pela classe *ControladorIdiomas* junto a arquivos de tradução. Outro pacote, *bean*, também possui uma única classe (*Comparacao*), utilizada em

AvaliacaoInterface para manipular comparações paritárias. Portanto, seus diagramas seriam compostos por apenas uma classe cada e foram omitidos. Por fim, o pacote *ferramentas* é um código externo utilizado para desenho, e também teve seu diagrama omitido.

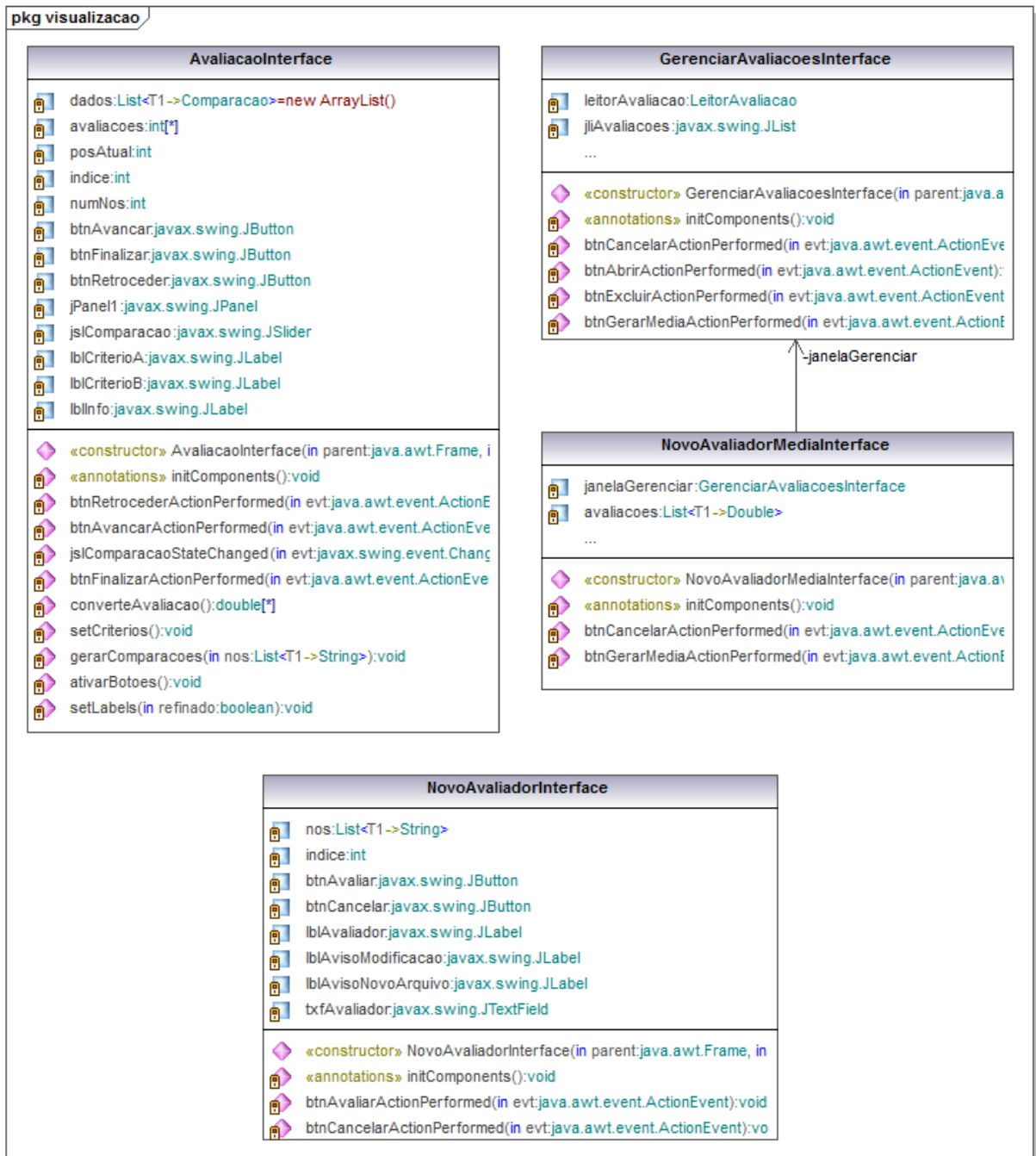


Figura 12: Diagrama de classes: pacote visualização (parte 2)

Fonte: autoria própria.

5 RESULTADOS OBTIDOS

Neste capítulo será apresentado o software desenvolvido, incluindo as interfaces com a indicação dos seus componentes e das suas funcionalidades disponíveis para o usuário, juntamente com os testes realizados.

5.1 INTERFACE INICIAL

Ao abrir o sistema a interface a ser mostrada está representada pela Figura 13. Essa janela possui basicamente dois componentes principais; o primeiro, à esquerda, trata-se do local onde será apresentada a relação entre o objetivo, os critérios avaliados para que esse objetivo seja alcançado e as possíveis alternativas de softwares disponíveis; o segundo componente principal, à direita, traz a descrição sobre os objetos presentes à esquerda.

Na parte superior dessa janela há os botões relacionados ao gerenciamento dos projetos e das avaliações desses projetos, além da possibilidade de troca de idioma através dos ícones à direita.

5.1.1 BARRA DE MENU

A barra de menu da janela principal traz o acesso para o usuário de todas as possíveis funcionalidades do software. Abaixo temos o mapeamento de todos os botões de acesso do usuário através da barra de menu, as operações básicas estão explicadas e as que levam a outras interfaces fazem referência à seção da interface específica.

- Arquivo: Menu relacionado a modificações de arquivos do projeto.
 - Novo: Criação de um novo projeto, explicado na seção 5.2.
 - Abrir: Abertura de um projeto já existente.
 - Salvar: Salva o projeto que está aberto no momento.
 - Sair: Sai do programa.

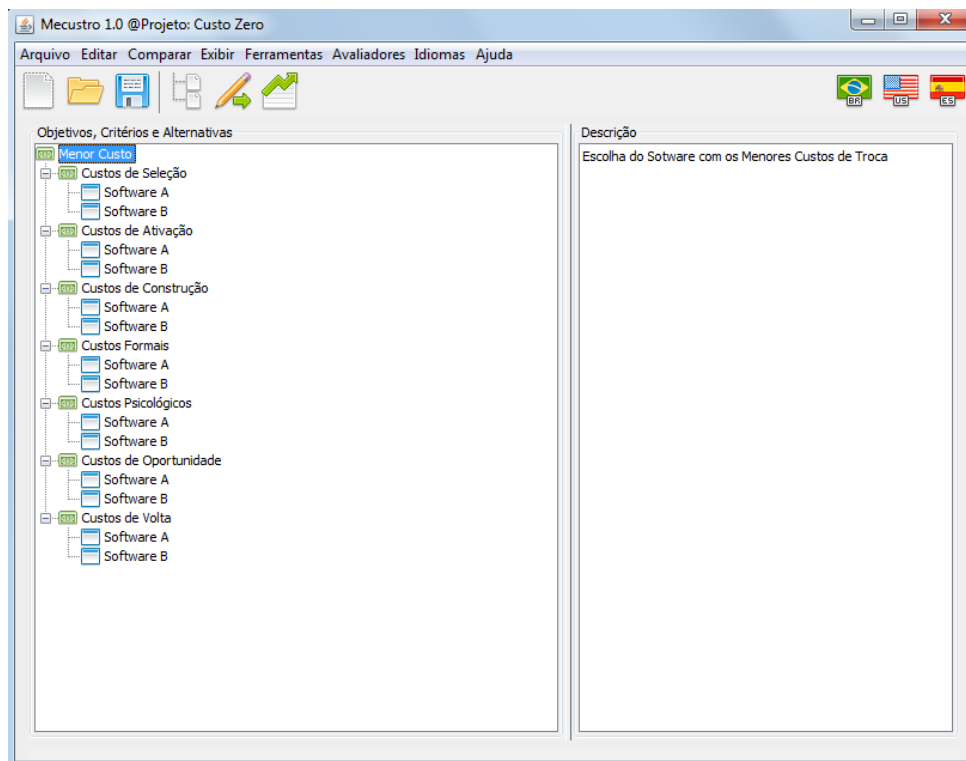


Figura 13: Interface inicial com os critérios e as alternativas já estabelecidos

Fonte: Autores.

- Editar: Menu relacionado à edição do projeto atual.
 - Inserir Critério: Adiciona um critério a ser avaliado.
 - Inserir Alternativa: Adiciona uma alternativa a ser avaliada.
 - Editar Item Selecionado: Edita o item selecionado pelo usuário.
 - Excluir Item Selecionado: Exclui o item selecionado pelo usuário.
- Comparar: Menu relacionado às comparações do software.
 - Avaliar: Vai para a interface de avaliação, explicada na seção 5.4.
 - Resetar Comparações do Critério: Reseta as comparações de um critério específico.
 - Resetar Todas as Comparações: Reseta todas as comparações já realizadas.
 - Resultado Final: Vai para a interface que mostra o resultado final, explicada na seção 5.6.
- Exibir: Menu relacionado a exibição dos componentes do sistema e do projeto.
 - Expandir Tudo: Expande todos os componentes da “árvore” do projeto.

- Contrair Tudo: Contrai todos os componentes da “árvore” do projeto.
- Barra de Ferramentas: Mostra/Oculta a barra de ferramentas.
- Barra de Progresso: Mostra/Oculta a barra de progresso.
- Ferramentas: Menu relacionado a diferentes funcionalidades para o projeto.
 - Modelo Gráfico-Hierárquico: Gera o modelo gráfico-hierárquico do projeto, explicado na seção 5.7.
 - Análise Refinada: Habilita/Desabilita a análise refinada do projeto.
 - Bloquear a edição de critérios: Bloqueia/Desbloqueia a edição de critérios do projeto.
 - Vetor de Prioridade: Menu para balancear a precisão com a velocidade do cálculo.
 - * Aproximado: Aumenta a rapidez do cálculo.
 - * Real: Aumenta a precisão dos resultados.
- Avaliadores: Menu relacionado às avaliações do projeto.
 - Nova Avaliação: Inicia uma nova avaliação para o projeto, explicada na seção 5.3.
 - Gerenciar Avaliações: Gerencia as avaliações do projeto atual, explicada na seção 5.5.
 - Apagar Tudo e Reabrir Projeto: Apaga todas as avaliações e reabre o projeto, habilitando a edição de seus critérios e alternativas.
- Idiomas: Menu relacionado aos idiomas suportados pelo sistema.
 - Português: Muda o idioma do sistema para português.
 - English: Muda o idioma do sistema para inglês.
 - Español: Muda o idioma do sistema para espanhol.
- Ajuda: Menu relacionado a informações sobre o sistema.
 - Conteúdo: Traz informações sobre as funcionalidades do sistema.
 - Sobre: Traz informações sobre a criação do sistema.

5.1.2 BARRA DE FERRAMENTAS

Diferente da barra de menu, a barra de ferramentas traz apenas as funcionalidades mais utilizadas pelo usuário. Assim, cada botão desta barra representa uma funcionalidade já presente na barra de menu, só que de uma maneira mais intuitiva. A Figura 14 mostra a barra de ferramenta. Abaixo há o mapeamento das funcionalidades disponíveis.

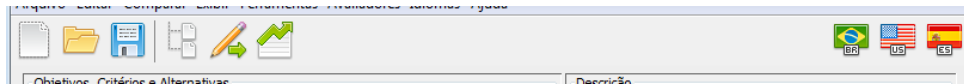


Figura 14: Barra de ferramentas

Fonte: Autores.

Da esquerda para a direita há as seguintes funcionalidades:

- Novo: Criação de um novo projeto, explicado na seção 5.2.
- Abrir: Abertura de um projeto já existente.
- Salvar: Salva o projeto que está aberto no momento.
- Inserir Alternativa: Adiciona uma alternativa a ser avaliada.
- Avaliar: Vai para a interface de avaliação, explicada na seção 5.4.
- Resultado Final: Vai para a interface que mostra o resultado final, explicada na seção 5.6.
- Português: Muda o idioma de toda a interface do sistema para português.
- English: Muda o idioma de toda a interface do sistema para inglês.
- Español: Muda o idioma de toda a interface do sistema para espanhol.

5.2 INTERFACE NOVO PROJETO

Ao selecionar a opção de criação de um novo projeto, tanto pela barra de menu quanto pela barra de ferramentas, a interface representada pela Figura 15 será aberta, mostrando várias opções para a criação de um novo projeto.

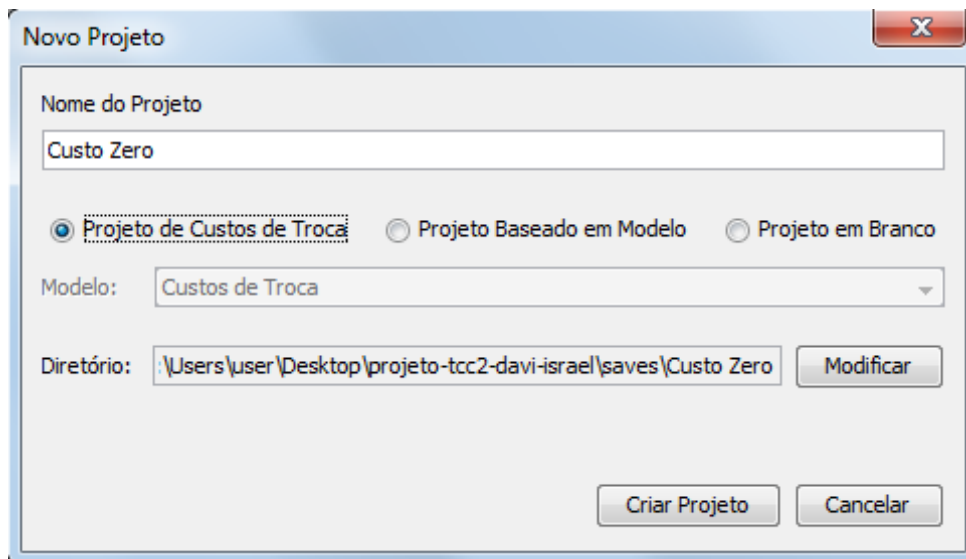


Figura 15: Interface novo projeto

Fonte: Autores.

Primeiramente, o usuário deverá entrar com o nome do seu respectivo projeto e depois ele poderá selecionar uma dentre as três opções de projeto. A primeira cria um projeto padrão para calcular os custos de troca, onde já estão determinados como critérios os custos de troca que foram analisados neste trabalho. O segundo também trabalha com modelos pré-definidos, mas estes são definidos pelo usuário, sendo que ele poderá escolher qual irá usar, pois o “modelo” ficará habilitado. Por fim, a criação de um projeto em branco resultará em um projeto onde todos os critérios de avaliação deverão ser informados pelo usuário.

Definido o tipo de projeto a ser criado, o usuário pode então escolher o diretório para salvar os seu projeto caso não goste do diretório padrão definido pelo sistema. Com tudo definido o programa voltará para a interface inicial e caso algum modelo tenha sido escolhido este já será preenchido automaticamente.

5.3 INTERFACE NOVA AVALIAÇÃO

Com todos os critérios e alternativas definidos, o usuário poderá então escolher realizar a avaliação do seu projeto. Para isso, primeiramente ele passará pela interface de nova avaliação (representada na Figura 16), onde colocará seu nome para identificar aquela avaliação em específico, pois um projeto poderá possuir diversas avaliações relacionadas a ele de diferentes usuários.

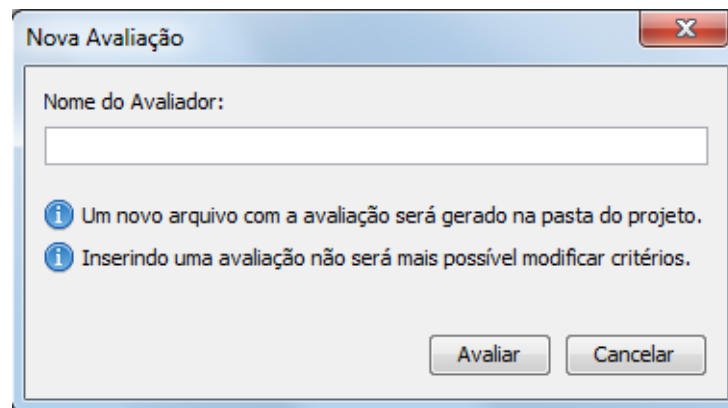


Figura 16: Interface nova avaliação

Fonte: Autores.

5.4 INTERFACE AVALIAÇÃO

Com a referência ao avaliador já definida, será então aberta uma interface para que o usuário avalie os critérios e as alternativas dois a dois. A Figura 16 mostra um exemplo da interface, o tomador de decisão deverá então arrastar a barra para determinar a importância de um critério/alternativa sobre o outro.

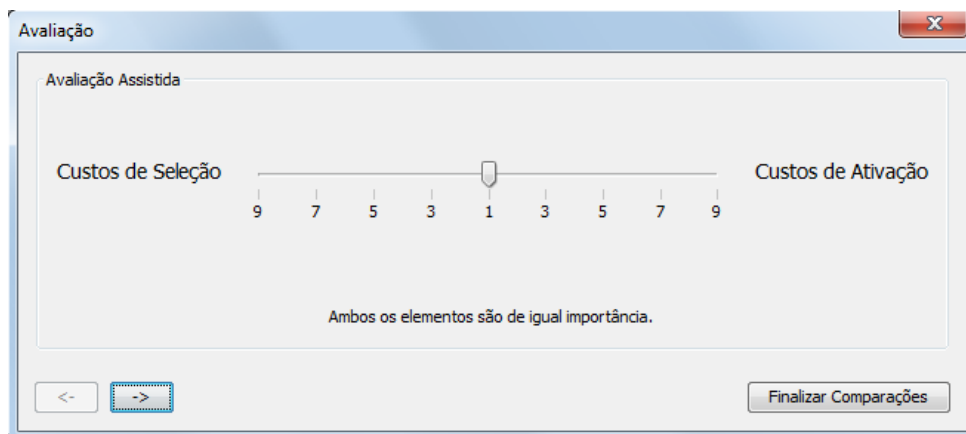


Figura 17: Interface avaliação

Fonte: Autores.

Após todas as comparações necessárias terem sido realizadas o programa calculará, nos moldes do AHP estudado neste trabalho, e indicará a melhor solução a ser utilizada, considerando os custos de troca e baseando-se nas escolhas entre os critérios e entre as alternativas.

5.5 INTERFACE GERENCIAR AVALIAÇÕES

Com o suporte a diversas avaliações é necessário o seu gerenciamento, para isso ao abrir a janela gerenciar avaliações, representada pela Figura 18, o usuário verá uma lista das avaliações daquele projeto e poderá excluir ou abrir uma avaliação existente.

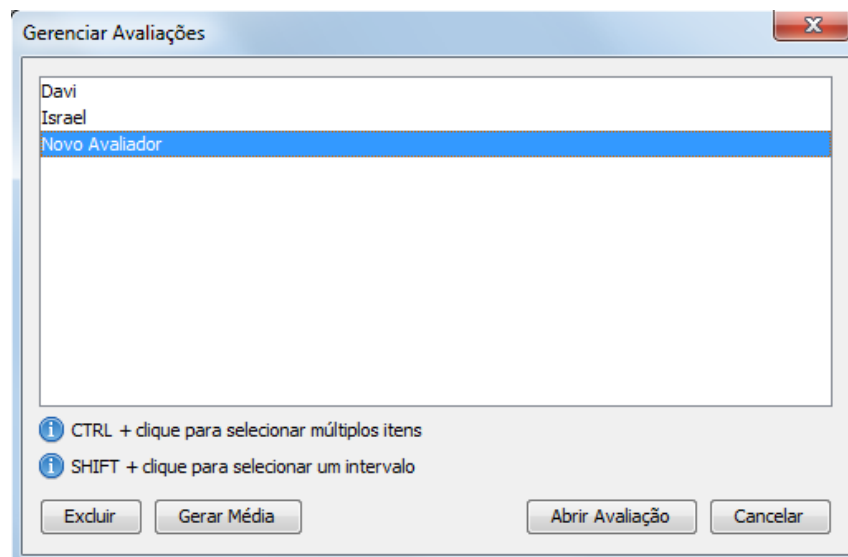


Figura 18: Interface gerenciar avaliações

Fonte: Autores.

Também é possível gerar uma média com um grupo selecionado de avaliações, facilitando a utilização do sistema ao gerar um único resultado após alternativas terem sido avaliadas por diversos avaliadores diferentes. Como para gerar essa média é necessário que as avaliações estejam realizadas completamente, o sistema verifica a integridade das avaliações selecionadas e caso alguma não esteja completa aparece uma tela de aviso para o usuário que está representada pela Figura 19.

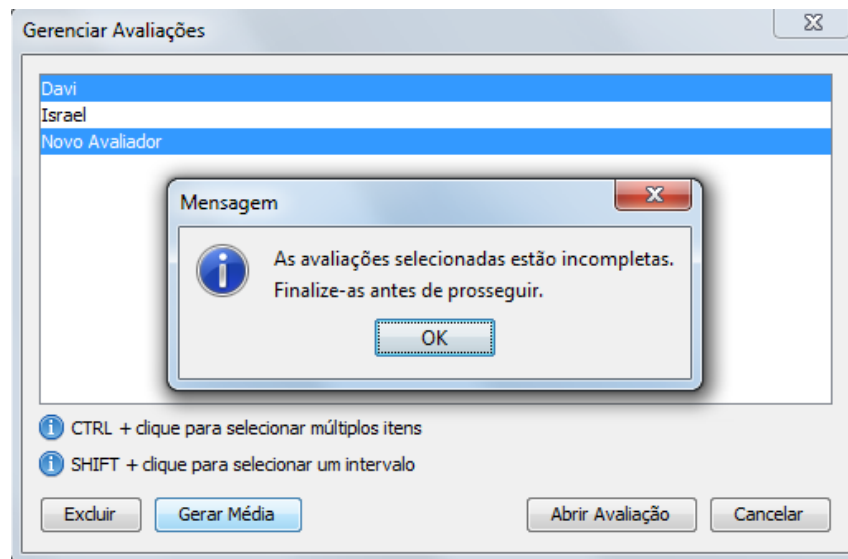


Figura 19: Aviso para avaliações incompletas.

Fonte: Autores.

5.6 INTERFACE RESULTADO FINAL

Com a avaliação finalizada, o usuário poderá abrir a interface de resultado final, representada na Figura 20. Nela são apresentados os diversos softwares que foram avaliados e a suas respectivas notas onde o software com a melhor nota será o que possuiu o maior benefício para a empresa. Lembrando que esse resultado serve como apoio a tomada de decisão, seu resultado deve ser avaliado juntamente com os benefícios de cada sistema para que a melhor escolha seja realizada.

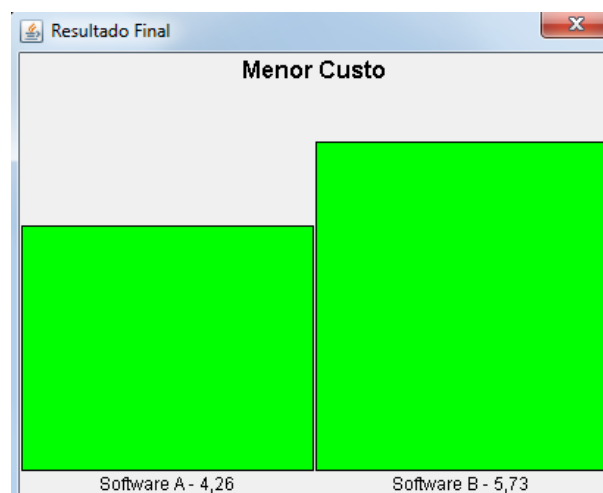


Figura 20: Interface resultado final

Fonte: Autores.

5.7 INTERFACE MODELO GRÁFICO HIERÁRQUICO

A relação entre o objetivo, os critérios e as alternativas também pode ser visualizado de acordo com estrutura do AHP, como mostrado na seção 3.2 (Figura 4). Facilitando a visualização e o entendimento da relação entre os componentes, a Figura 21 retrata essa interface.

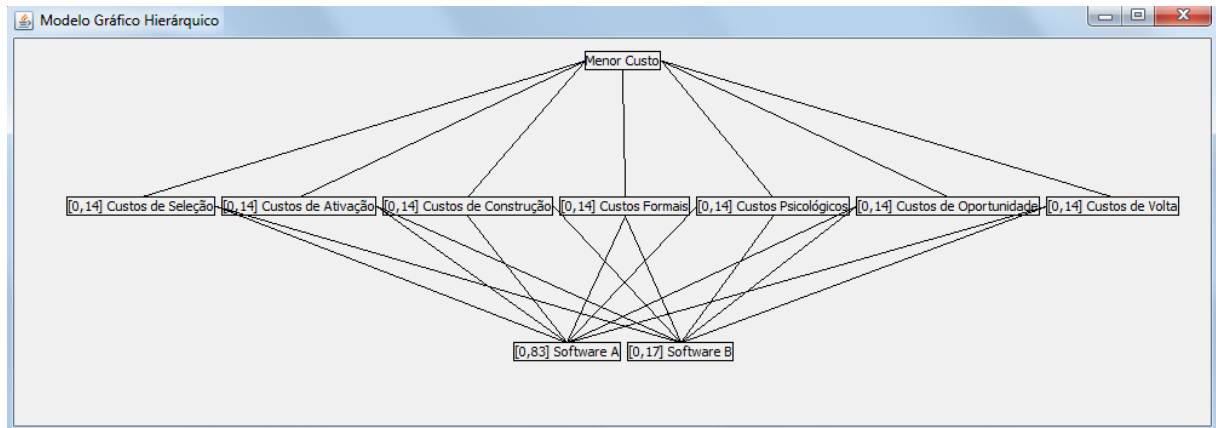


Figura 21: Interface modelo hierárquico

Fonte: Autores.

5.8 INTERFACE CONTEÚDO

A interface conteúdo, representada pela Figura 22, traz um *preview* do manual do software e ao clicar no botão "Abrir PDF" o sistema irá abrir o manual no formato PDF.

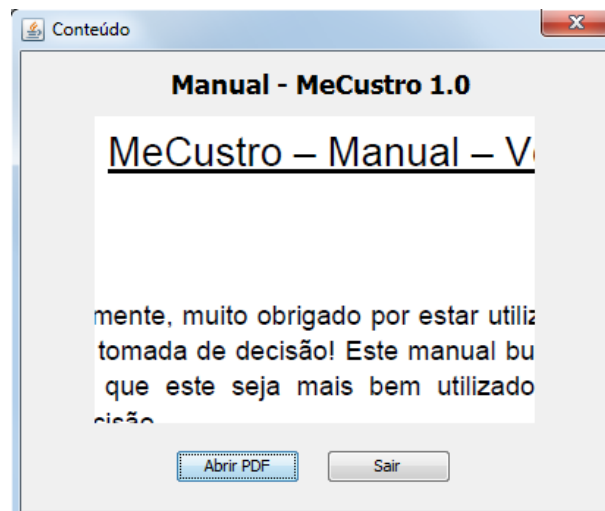


Figura 22: Interface conteúdo

Fonte: Autores.

5.9 INTERFACE SOBRE

Esta interface, representada na Figura 23, traz informações sobre o sistema, como por quem foi desenvolvido e qual professor orientou este trabalho.

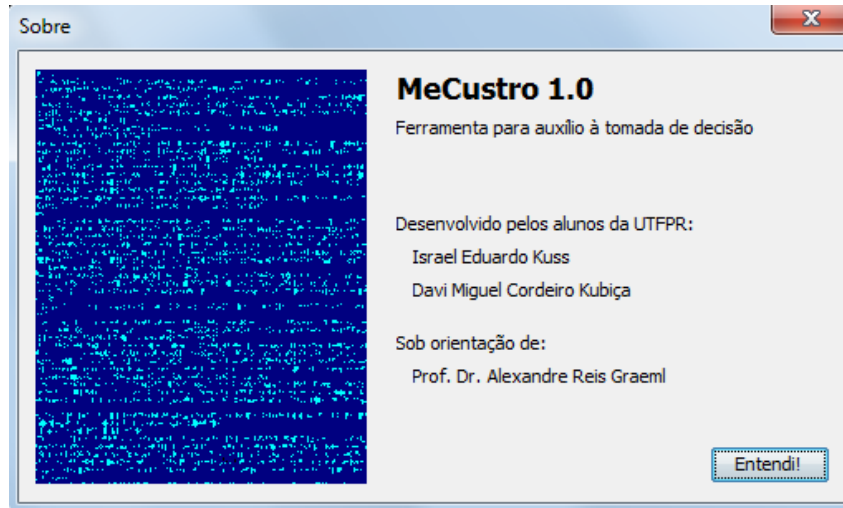


Figura 23: Interface com as características dos desenvolvedores.

Fonte: Autores.

5.10 TESTES

O objetivo de se testar um software é encontrar erros. Um bom teste é aquele com alta probabilidade de encontrar um erro (PRESSMAN, 2011). Também é nesse estágio que deve-se verificar se o sistema atende aos requisitos funcionais e não funcionais (SOMMERVILLE, 2007). Com a abordagem evolucionária, cada nova unidade do software implementada é testada para garantir o funcionamento correto até aquele ponto, mas ao pensar no sistema final como um todo comportamentos inesperados podem surgir.

Ainda segundo Pressman (2011), um bom teste também não é redundante. Ficar testando um determinado componente de maneira que tenha a mesma finalidade de um teste anterior é apenas consumo de tempo e recurso. Um bom teste também não deve ser muito simples nem muito complexo. Isso significa que combinar diversos casos de testes em um único pacote pode mascarar problemas. Por isso, é importante elaborar casos de testes com cautela.

O conjunto de casos de teste da lógica interna do software é chamado de caixa branca, enquanto requisitos são exercitados por técnicas chamadas de caixa preta. Embora testes caixa

branca pareçam gerar programas sem defeito algum, estes podem gerar caminhos lógicos muito grandes, mesmo em programas simples (PRESSMAN, 2011).

O teste caixa preta procura por funções incorretas ou faltando, erros de interface, erros em estrutura de dados, erros de desempenho e de inicialização ou término. Este, portanto, será o foco dos testes a serem executados nas próximas subseções. Em um primeiro momento, testes com relação à usabilidade do software foram feitos, e em seguida, com relação aos cálculos do AHP, comparando com um software já existente que utilize o método para validação dos resultados.

5.10.1 TESTES DE INTERFACE

No Quadro 20 estão representados os casos de testes escolhidos, elaborados com base nos requisitos levantados e nas funcionalidades do sistema.

Casos de teste			
Ação	Resumo	Resultado esperado	Resultado obtido
Adicionar, editar e remover critérios	A manipulação de critérios pode ser feita pelo menu <i>Editar</i>	Critérios adicionados ou modificados na árvore. Alternativas existentes devem ser copiadas para o novo critério	Painel de descrição não é zerado ao remover critério. Objetivo possível de ser deletado
Adicionar, editar e remover alternativas	Adicionar e remover alternativas podem ser feitas pelo menu <i>Editar</i> e com o menu do botão direito do mouse. A adição também é possível pela barra de ferramentas	Alternativa modificada na árvore. A edição de uma alternativa deve impactar os demais critérios	Nenhum problema detectado
Adicionar e remover avaliador do projeto	Deve ser feito pelo menu <i>Nova Avaliação</i> . Também possível pelo menu <i>Avaliar</i> , caso não haja avaliador atual	Projeto entra no modo avaliação. Novo arquivo é gerado na pasta do projeto	Nenhum problema detectado

Avaliar projeto	Esta função pode ser executada pelo menu principal, botão direito ou barra de ferramentas	Árvore é atualizada com o valor da avaliação e a barra de progresso também	Nenhum problema detectado
Exibir modelo gráfico hierárquico	Disponível no menu <i>Ferramentas</i> , exibe a árvore de critérios em um modo gráfico	Uma janela com a exibição gráfica é aberta	Nenhum problema detectado
Visualizar resultado final	Disponível no menu <i>Comparar</i> e barra de ferramentas. Exibe as alternativas e suas porcentagens	Exibir gráficos de barra correspondendo às alternativas (softwares)	Nenhum problema detectado
Salvar projeto	Feito pelo menu <i>Arquivo</i> , salva o projeto atual	Se o projeto está em edição, ele é salvo. Se está em modo avaliação, a avaliação é salva	Nenhum problema detectado
Reabrir projeto	Projeto em modo avaliação torna-se editável novamente. Disponível no menu <i>Avaliadores</i>	Avaliações existentes são deletadas e o projeto volta a ser editável	Nenhum problema detectado
Desfazer comparações	Comparações do nó ou da árvore são removidas. Estas opções estão no menu <i>Comparar</i>	O vetor de avaliação tem a avaliação removida e o critério deixa de exibir a avaliação ao seu lado	Não desfaz todas as comparações dos critérios com o objetivo
Mudar idioma	Disponível no menu <i>Idiomas</i> e na barra de ferramentas, muda para o idioma desejado	Clicar em um novo idioma deve mudá-lo instantaneamente	Nenhum problema detectado

Quadro 20: Casos de teste da interface

Fonte: autoria própria.

Como pode ser observado pela Tabela 20, durante a execução foram detectados *bugs* na remoção de critérios e *reset* de comparações em um caso específico. Estes foram corrigidos em sequência, e o processo de testes desses casos de teste foram repetidos, não tendo nenhum problema detectado.

5.10.2 TESTES DE CÁLCULOS

Para verificação dos resultados apresentados pelo software, os custos de troca já levantados e embutidos neste software foram criados no software *Expert Choice*. Esta ferramenta existe há mais de 30 anos, e também faz uso do AHP para tomada de decisão. Alguns de seus clientes incluem *NASA*, *3M*, *General Motors* e *Boeing*¹.

O número de critérios para o teste será sete (número de custos de troca levantados), e o número de alternativas será duas, quatro e dez. Estes três testes foram selecionados com base no teste chamado análise de valor limite, que seleciona testes de “borda”, onde o maior número de erros tendem a ocorrer (PRESSMAN, 2011). Duas alternativas é o mínimo aceitável para uma avaliação, dez é o máximo, e quatro é um valor mais central.

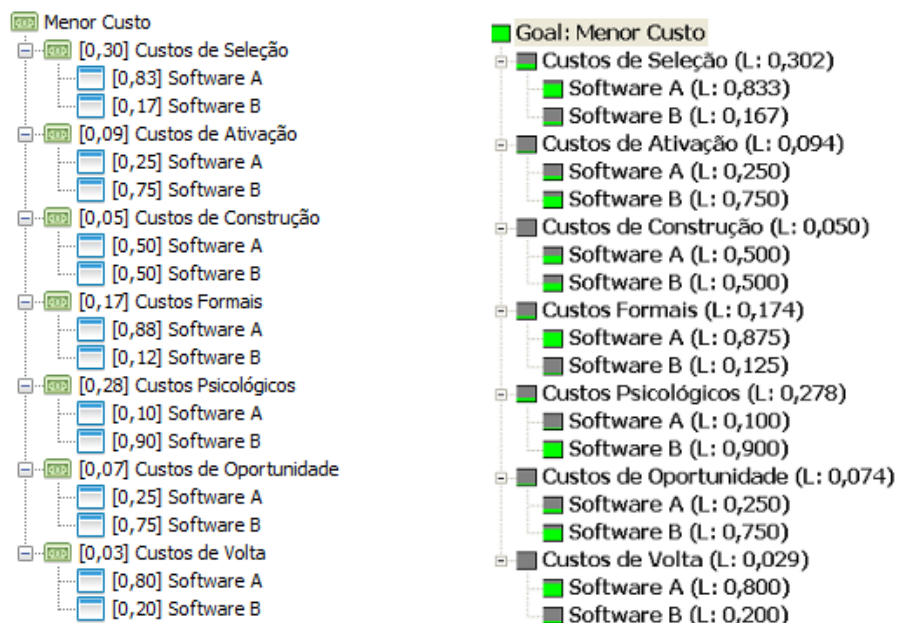


Figura 24: Resultado dos cálculos das avaliações com duas alternativas

Fonte: autoria própria.

Na Figura 24, está a avaliação feita nas duas ferramentas. À esquerda, o software desenvolvido utilizando a avaliação precisa (com auxílio da biblioteca *jblas*), e a direita, *Expert*

¹Mais detalhes em <http://expertchoice.com/clients-successes/>

Choice. Todas as opções de avaliação foram exploradas, isto é, variações na escala de 1 a 9 para um software e para outro, de maneira que pudesse ser verificado os cálculos realizados. Como pode se observar, os cálculos saíram iguais nas duas ferramentas.

Já na Figura 25, encontra-se o resultado final apresentado também nas duas ferramentas. O restante dos testes (com quatro e dez alternativas) estão disponíveis para visualização no Apêndice A. Em todos os testes, os resultados foram os mesmos nos dois softwares, o que demonstra o correto funcionamento da ferramenta desenvolvida nesses casos.

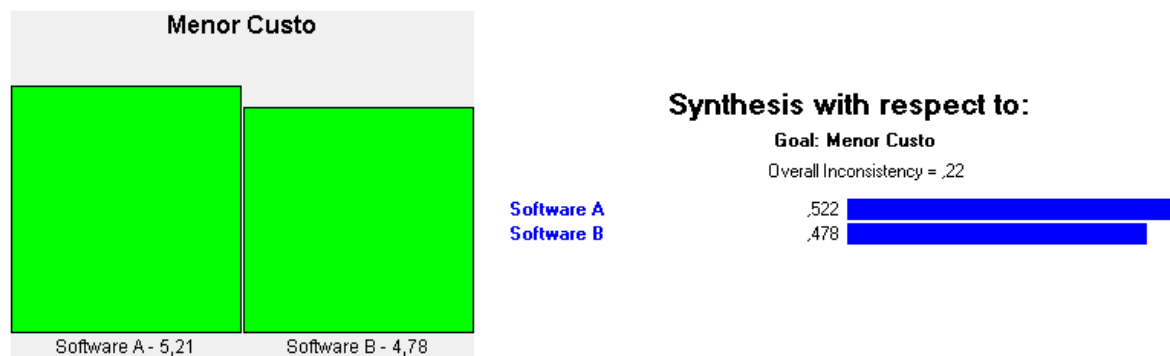


Figura 25: Resultado final com duas alternativas

Fonte: autoria própria.

6 CONCLUSÃO

Existe uma grande quantidade de diferentes soluções para os mesmos problemas relacionados a software. A escolha entre a solução mais benéfica para uma empresa é uma decisão estratégica essencial para que a companhia tenha um diferencial ao diminuir custos relacionados a diversos critérios relacionados ao sistema.

Com a finalização deste trabalho pode-se concluir que os custos de troca possuem uma grande importância para os tomadores de decisão de uma empresa. Visto que são utilizados para determinar se alguma outra tecnologia possui mais vantagem na sua utilização em relação à que já está implantada.

Com o estudo dos diversos custos de troca associados a software foi possível observar que o processo de escolha de um novo produto implica na análise de diversos fatores que estão relacionados àquele produto, desde a busca de novos fornecedores e os benefícios de cada sistema até as possíveis despesas caso seja necessário voltar para o sistema utilizado anteriormente.

Isso mostrou o quão impactante e relevante para uma organização a troca de um software pode ser, modificando possivelmente toda uma estrutura já consolidada dentro da companhia. Fazendo-se necessário realizar novos treinamentos para preparar e capacitar os funcionários na utilização deste novo sistema para que ele traga melhores resultados que o antigo para a empresa.

Através da análise de diversos métodos de tomada de decisão com múltiplos critérios, chegou-se a conclusão de que o melhor método a ser utilizado para a realização deste TCC fosse o processo analítico hierárquico, possibilitando criar uma melhor co-relação entre os critérios (custos de troca) e alternativas (softwares) além de fácil usabilidade para o tomador de decisão.

A criação do sistema possibilitou a consolidação dos estudos realizados sobre os custos de troca e sobre o processo analítico hierárquico, resultando em uma ferramenta que auxiliasse os tomadores de decisão de maneira fácil e simples.

Características específicas de cada parte analisada também são apresentadas no sistema, fazendo com que os usuários consigam ter um maior entendimento sobre o que é cada custo de troca (com a descrição e explicação de cada um) e com diversas customizações (explicadas no manual para o usuário) no cálculo do AHP para melhor atender as suas necessidades.

A especificação inicial envolveu a criação de um software não muito complexo (tanto para os desenvolvedores como para os tomadores de decisão), que fosse capaz de realizar cálculos do AHP, de maneira a não comprometer o prazo final. Mas, à medida que foi evoluindo, diversas funcionalidades adicionais foram sendo acrescentadas, como a disponibilização do sistema em três idiomas e a capacidade de vários tomadores de decisão avaliarem o mesmo projeto separadamente e através disso ser obtida uma média. Desta forma, os requisitos funcionais e não-funcionais do software foram atendidos e ampliados.

Comparado ao *Expert Choice* e as demais ferramentas de apoio a decisão que utilizam o AHP, este software fornece uma solução gratuita e especializada na escolha de um software considerando os custos de troca, mas também pode ser aplicado a diversas outras áreas devido à possibilidade da adição de modelos e a criação de projetos em branco. Também oferece recursos próprios, como descrição dos critérios e objetivo e a multiavaliação com opção de média. A opção de salvar avaliações em arquivos separados também permite a utilização simultânea por vários tomadores de decisão.

A criação deste trabalho permitiu aplicar conhecimentos adquiridos de diversas disciplinas ao longo do curso, como Fundamentos de Programação, Análise e Projeto de Sistemas, Engenharia de Software e Gestão da Informação e de Sistemas de Informação.

Como trabalhos futuros para este projeto podem ser citadas as seguintes melhorias/aperfeiçoamentos:

- Adicionar opção de análise com casas decimais para critérios empatados;
- Incluir análise de sensibilidade e mais opções de gráficos para os resultados;
- Aumentar a quantidade de idiomas suportados pelo sistema, salvos em arquivos separados;
- Criar uma versão *web* do programa, visando facilitar a multiavaliação de projetos.

REFERÊNCIAS

- BACIC, N. M. **O software livre como alternativa ao aprisionamento tecnologico imposto pelo software proprietario.** 2003. Disponível em: <<http://www.bibliotecadigital.unicamp.br/document/?code=000294736>>. Acesso em: 22 de junho de 2014.
- BALMAT, J.-F. et al. **A Decision-Making System to Maritime Risk Assessment.** 2011. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0029801810002271>>. Acesso em: 26 de agosto de 2014.
- BROCKMAN, M. S.; RUSSELL, S. T. **Decision Making and Reasoning Skills.** 2009. Disponível em: <http://calscf.calsnet.arizona.edu/fcs/bpy/content.cfm?content=decision_making>. Acesso em: 25 de maio de 2014.
- BURNHAM, T. A.; FRELS, J. K.; MANHAJAN, V. **Consumer Switching Costs: A Typology, Antecedents, and Consequences.** 2003. Disponível em: <link.springer.com/article/10.1177%2F0092070302250897>. Acesso em: 10 de maio de 2014.
- ENGELBERT, R. **Os Custos de Troca em Tecnologia da Informação e o Aprisionamento Tecnológico das Organizações.** 2010. Disponível em: <http://www.up.com.br/painelgpa/uploads/imagens/files/Mestrado%20ADM/DISSERTA%C3%87%C3%83O_RICARDO%20ENGELBERT.pdf>. Acesso em: 25 de maio de 2014.
- ENGELBERT, R.; GRAEML, A. **Custos de troca em tecnologia da informação: a proposição de um modelo taxonômico integrado a partir da literatura.** 2011. Disponível em: <<http://www.spell.org.br/documentos/download/5956>>. Acesso em: 25 de maio de 2014.
- FIGUEIRA, J.; GRECO, S.; EHRGOTT, M. **Multiple Criteria Analysis: State of the Art Surveys.** [S.l.]: Springer, 2005.
- FORMAN, C.; CHEN, P. yu. **Network Effects and Switching Costs in the Market for Routers and Switches.** 2003. Disponível em: <<http://www.netinst.org/FormanChen.pdf>>. Acesso em: 20 de junho de 2014.
- HARRIS, R. G. **The Role of Switching Costs in the Markets for PC Operating Systems, Online Search, Internet Access and Mobile Service: Implications for Australian Competition and Consumer Protection Policy.** 2012. Disponível em: <http://www.ipria.org/events/conf/Competition_Conference/Switching_Costs.pdf>. Acesso em: 15 de maio de 2014.
- HESS, M.; RICART, J. E. **Managing Customer Switching Costs: A Framework For Competing in the Networked Environment.** 2002. Disponível em: <http://papers.ssrn.com/sol3/papers.cfm?abstract_id=443320>. Acesso em: 20 de junho de 2014.

JUNIOR, L. F. N. **Tomada de Decisão com Múltiplos Critérios: Pesquisa-Ação sobre o Método AHP em Pequenas Empresas.** 2006. Disponível em: <http://www.ppga.com.br/mestrado/2006/nunesjr-luis_fernando.pdf>. Acesso em: 23 de julho de 2014.

KONIDARI, P.; MAVRAKIS, D. **A multi-criteria evaluation method for climate change mitigation policy instruments.** 2007. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0301421507003229>>. Acesso em: 27 de junho de 2014.

KOSTLAN, E. **Statistical Complexity of Dominant Eigenvector Calculation.** 1991. Disponível em: <<http://www.developmentserver.com/randompolynomials/scdec/paper.pdf>>. Acesso em: 18 de julho de 2014.

LUJAN-MORA, S.; TRUJILLO, J.; SONG, I.-Y. **Multidimensional Modeling with UML Package Diagrams.** 2002. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.3289>>. Acesso em: 20 de janeiro de 2015.

MANTARAS, R. L. de et al. **Retrieval, reuse, revision, and retention in case-based reasoning.** 2005. Disponível em: <<http://www.iiia.csic.es/mantaras/RRRR.pdf>>. Acesso em: 27 de junho de 2014.

MARINS, C. S.; SOUZA, D. de O.; BARROS, M. da S. **O Uso do Método de Análise Hierárquica (AHP) na Tomada de Decisões Gerenciais - Um Estudo de Caso.** 2009. Disponível em: <<http://www.din.uem.br/sbpo/sbpo2009/artigos/55993.pdf>>. Acesso em: 17 de julho de 2014.

MORAES, E. A. de; SANTALIESTRA, R. **Modelo de Decisão com Múltiplos Critérios para Escolha de Software de Código Aberto e Software de Código Fechado.** 2008. Disponível em: <<http://www.spell.org.br/documentos/download/5923>>. Acesso em: 17 de julho de 2014.

OGUZTIMUR, S. **Why Fuzzy Analytic Hierarchy Process Approach For Transport Problems?** 2010. Disponível em: <<http://www.sre.wu.ac.at/ersa/ersaconfs/ersa11/e110830aFinal00438.pdf>>. Acesso em: 30 de junho de 2014.

PRESSMAN, R. S. **Engenharia de software: uma abordagem profissional.** [S.l.]: McGraw-Hill, 2011.

SAATY, T. L. **A scaling method for priorities in hierarchical structures.** 1977. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0022249677900335>>. Acesso em: 7 de julho de 2014.

SAATY, T. L. **The Analytic Hierarchy Process.** [S.l.]: McGraw-Hill, 1980.

SAATY, T. L. **Decision Making for Leaders - The Analytic Hierarchy Process for Decisions in a Complex World.** [S.l.]: RWS Publications, 1995.

SAATY, T. L. **Decision making with the analytic hierarchy process.** 2008. Disponível em: <<http://inderscience.metapress.com/content/02t637305v6g65n8/>>. Acesso em: 27 de junho de 2014.

SHAPIRO, C.; VARIAN, H. R. **Information Rules: A Strategic Guide to the Network Economy**. [S.l.]: Harvard Business Review Press, 1999.

SOMMERVILLE, I. **Engenharia de software**. [S.l.]: Pearson Addison-Wesley, 2007.

THANASSOULIS, E.; KORTELAJINEN, M.; ALLEN, R. **Improving envelopment in Data Envelopment Analysis under variable returns to scale**. 2012. Disponível em: <<http://inderscience.metapress.com/content/02t637305v6g65n8/>>. Acesso em: 27 de junho de 2014.

VARGAS, R. V. **Utilizando a Programação Multicritério (Analytic Hierarchy Process - AHP) para Selecionar e Priorizar Projetos na Gestão de Portfólio**. 2010. Disponível em: <http://www.ricardo-vargas.com/wp-content/uploads/downloads/articles/ricardo_vargas_ahp_project_selection_pt.pdf>. Acesso em: 17 de julho de 2014.

VELASQUEZ, M.; HESTER, P. T. **An Analysis of Multi-Criteria Decision Making Methods**. 2013. Disponível em: <http://www.orstw.org.tw/ijor/vol10no2/ijor_vol10_no2_p56_p66.pdf>. Acesso em: 27 de junho de 2014.

WEISTROFFER, H. R.; NARULA, S. C. **Implementation of MCDM methodology in decision support systems**. 1989. Disponível em: <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=71447>. Acesso em: 27 de agosto de 2014.

ZAHIR, S. **Clusters in a group: Decision making in the vector space formulation of the analytic hierarchy process**. 1999. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221798000216>>. Acesso em: 30 de junho de 2014.

APÊNDICE A – TESTES DOS CÁLCULOS DO AHP

Nas figuras deste apêndice se encontram comparações da ferramenta desenvolvida com o software *Expert Choice*, utilizando quatro e dez alternativas. A árvore gerada e o resultado final são apresentados.

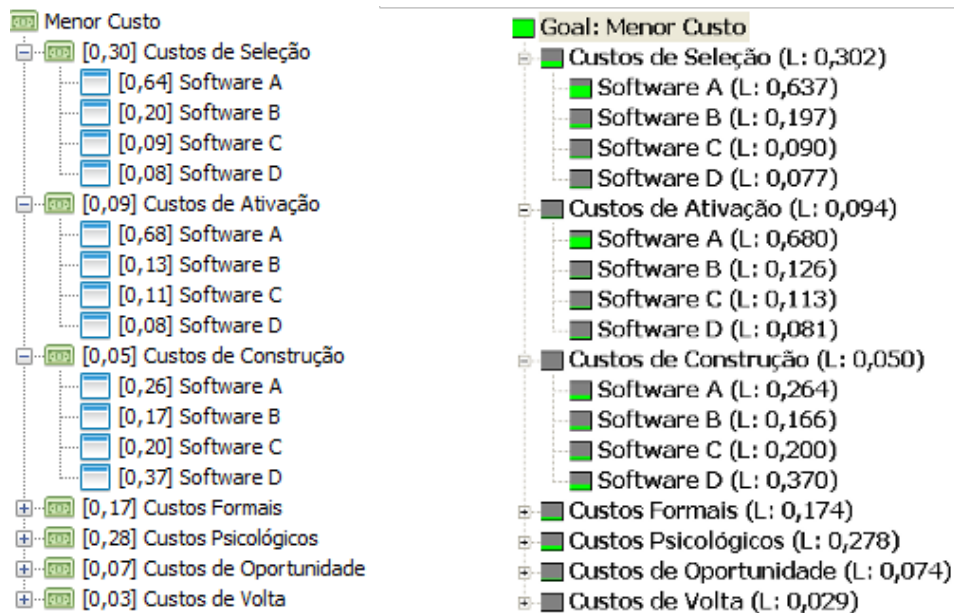


Figura 26: Resultado dos cálculos das avaliações com quatro alternativas

Fonte: autoria própria.

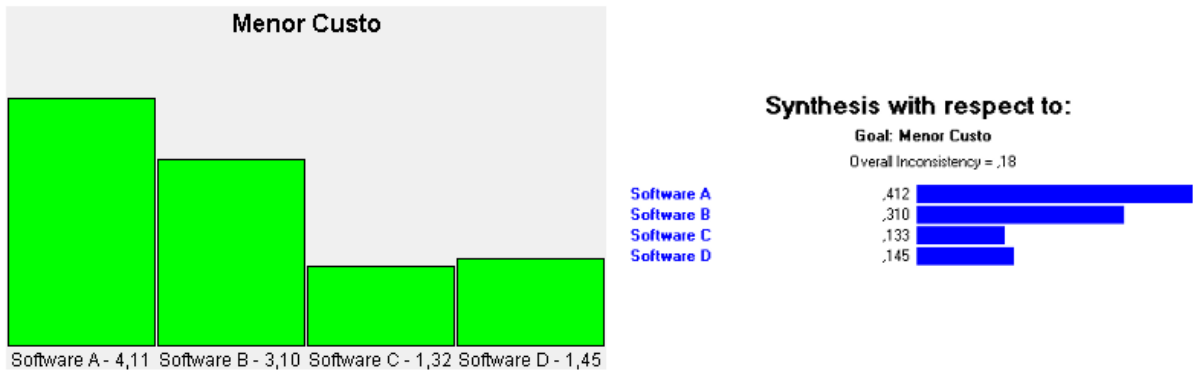


Figura 27: Resultado final com quatro alternativas

Fonte: autoria própria.

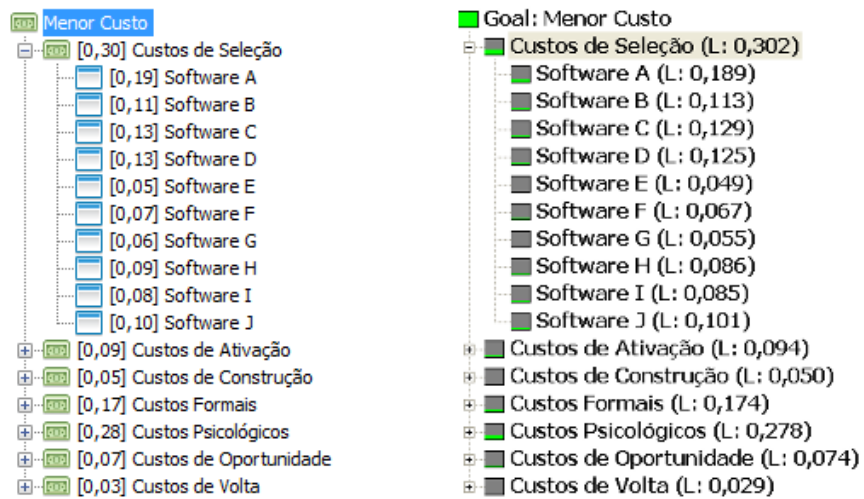
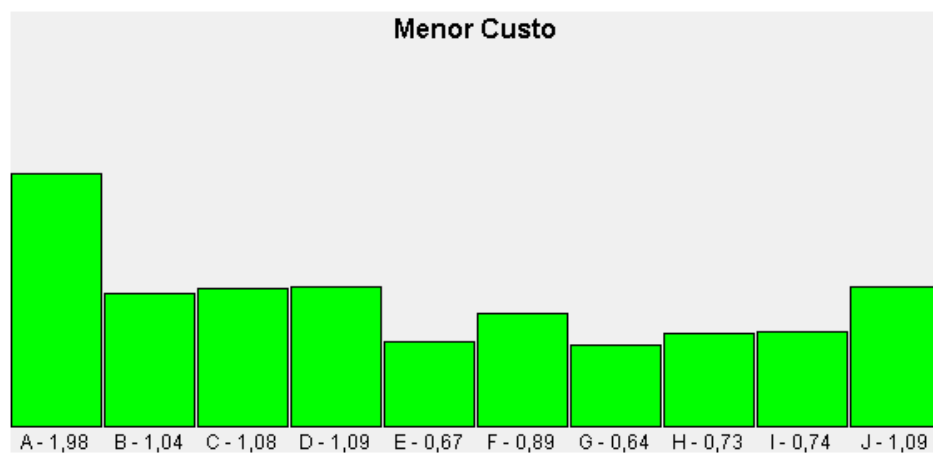


Figura 28: Resultado dos cálculos das avaliações com dez alternativas

Fonte: autoria própria.



Synthesis with respect to:

Goal: Menor Custo

Overall Inconsistency = ,32

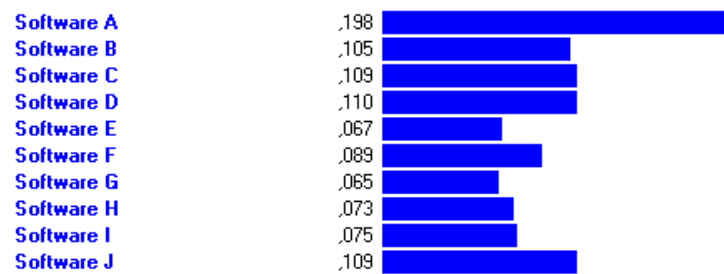


Figura 29: Resultado final com dez alternativas

Fonte: autoria própria.