

UNIVERSIDADE FEDERAL DE PERNAMBUCO

GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO  
CENTRO DE INFORMÁTICA



2005.1

---



ENVISION: UM MODELO PARA AVALIAÇÃO DE PRODUTO DE  
SOFTWARE BASEADO EM VISÕES EXTENSÍVEIS

---

TRABALHO DE GRADUAÇÃO EM  
ENGENHARIA DE SOFTWARE - QUALIDADE DE SOFTWARE

**Aluno** – Augusto César Pires Spinelli, [acps3@cin.ufpe.br](mailto:acps3@cin.ufpe.br) .

**Orientador** – Hermano Perrelli de Moura, [hermano@cin.ufpe.br](mailto:hermano@cin.ufpe.br).

**Co-orientador** – Alexandre Marcos Lins de Vasconcelos, [amlv@cin.ufpe.br](mailto:amlv@cin.ufpe.br).

22 de Agosto de 2005

*Aos meus pais Marcelo e Madja.  
Aos meus irmãos Marcos e Marcella.  
A todos os meus amigos*

## Agradecimentos

Ao meu orientador Hermano, pelo apoio durante a confecção do trabalho e por ser para mim um exemplo de professor, dentro e fora de sala de aula.

Ao meu co-orientador, Alexandre, pela dedicação a seus alunos e ótimas cadeiras ministradas durante o curso.

A Vânia Sampaio, “tia Vânia” que idealizou comigo este trabalho, meu voto de felicidade para todo o sempre!

A toda a equipe do Laboratório de Avaliação de Produtos de Software, por transformarem as duras horas de trabalho em momentos de alegria e companheirismo.

A meus pais Marcelo e Madja pelo amor, incentivo e por terem me colocado no mundo.

A meus irmãos Marcos e Marcella por tentarem me atrapalhar ao máximo e não conseguirem! E a Marcos, mais uma vez, pelo esforço e trabalho artístico na marca do Envision.

Aos amigos Mano, XFernando, Julio, André Furtado e Thiago Jatobá pela imensa torcida.

A Débora Belleza, por ser uma grande amiga, sempre positiva e incentivadora.

Ao Colégio de Aplicação, Centro de Informática e Universidade Federal de Pernambuco pela formação e por apostar no meu potencial desde o primeiro grau.

A todos os outros que, de alguma forma, contribuíram para o desenvolvimento deste trabalho, meus sinceros agradecimentos.

## **O rio**

*Uma gota de chuva  
A mais, e o ventre grávido  
Estremeceu, da terra.  
Através de antigos  
Sedimentos, rochas  
Ignoradas, ouro  
Carvão, ferro e mármore  
Um fio cristalino  
Distante milênios  
Partiu fragilmente  
Sequioso de espaço  
Em busca de luz.*

*Um rio nasceu.*

*(Vinícius de Moraes)*

## RESUMO

Este trabalho propõe a criação do conceito de “Visões” para avaliação de produtos de software que são, resumidamente, pacotes agregadores de atributos de qualidade de software. O conceito de visão proposto é flexível o bastante para agregar módulos de avaliação já existentes em um laboratório de avaliação e, até mesmo, outras visões. Como estudo de caso, é apresentado como adequar o modelo proposto a um laboratório de avaliação real, o LAPS – Laboratório de Avaliação de Produtos de Software, localizado na cidade brasileira de Recife no estado de Pernambuco. Quando aplicado a uma determinada empresa, o modelo servirá como ferramenta de suporte à decisão e aquisição de serviços de avaliação de produtos de software expondo transparentemente, sob a forma de visões, as avaliações de produtos de software e que atributos de qualidade são explorados ao se adquirir a avaliação de uma determinada visão.

**Palavras-chave:** Qualidade de Produto, Avaliação de Produto, Engenharia de Software, Processos de Desenvolvimento de Software, Qualidade de Software, ISO 12119, ISO 9126, ISO 14598.

## Abstract

This work proposes the creation of a new concept in software product evaluation, called **vision**. Briefly, a vision could be viewed as a package that aggregates software quality attributes. The concept proposed is flexible enough to aggregate the evaluation modules that already exists in a evaluation lab, and include to itself other visions. As a case study it is shown how to suit the proposed model to an existing evaluation lab called LAPS –“Laboratório de Avaliação de Produtos de Software”, located in the brazilian town of Recife, Pernambuco state. When applied to a determined company the model will serve as a tool to support decision and acquisition of product evaluation services, displaying transparently, under the form of visions, the software products evaluations and which quality attributes are explored when a evaluation based on visions is acquired.

**Keywords:** Product Quality, Product Evaluation, Software Engineering, Software Development Processes, Software Quality, ISO 12119, ISO 9126, ISO 14598.

## Résumé

Ce travail propose la création d'un nouveau concept dans l'évaluation des produits de logiciel appelée vision. Une vision pourrait être regardée comme un paquet qui agrège des attributs de qualité de logiciel. Le concept proposé est assez flexible pour agréger des modules d'évaluation déjà existents dans un laboratoire d'évaluation, et pour inclure à lui-même d'autres visions. Comme étude-de-cas, est montré comment il peut convenir le modèle proposé à un existant laboratoire d'évaluation: le LAPS – "Laboratório de Avaliação de Produtos de Software", situé dans la ville brésilienne de Recife, état de Pernambuco. Quand appliqué à une entreprise, le modèle servira d'outil pour soutenir la décision et l'acquisition des services d'évaluation des produits, exposant, sous la forme de visions, les évaluations de produits de logiciel et qui attributs de qualité sont explorés quand une évaluation basée sur visions est achetée.

**Mots-clé:** Qualité des Produits, Evaluation des Produits, Génie Logiciel, Processus pour le développement du logiciel, Qualité du Logiciel, ISO 12119, ISO 9126, ISO 14598.

# ÍNDICE

<b>1. INTRODUÇÃO .....</b>	<b>12</b>
OBJETIVOS DO TRABALHO .....	13
<i>Objetivo Principal</i> .....	13
<i>Objetivos Específicos</i> .....	13
RELEVÂNCIA .....	14
METODOLOGIA DE TRABALHO.....	16
ORGANIZAÇÃO DA MONOGRAFIA.....	17
<b>2. O LAPS.....</b>	<b>19</b>
BREVE HISTÓRICO .....	19
ESTRUTURA .....	19
MODELO DE AVALIAÇÃO .....	20
DIFICULDADES ENFRENTADAS.....	22
<b>3. TRABALHOS SIMILARES.....</b>	<b>23</b>
ATRIBUTOS DE COMPARAÇÃO DOS SISTEMAS.....	23
ANÁLISE DOS MODELOS .....	28
TABELA COMPARATIVA.....	31
CONCLUSÕES DE ANÁLISE .....	33
<b>4. AVALIAÇÃO DE PRODUTO DE SOFTWARE BASEADA EM VISÕES.....</b>	<b>35</b>
O QUE É UMA VISÃO? .....	35
CONFORMIDADE NORMATIVA .....	38
ATRIBUTOS DE QUALIDADE DE SOFTWARE.....	38
MÉTRICAS PARA AVALIAÇÃO .....	39
A AVALIAÇÃO BASEADA EM VISÕES .....	40
COMO INICIAR UMA AVALIAÇÃO BASEADA EM VISÕES?.....	42
A AVALIAÇÃO DE VISÕES: SIMILARIDADES E SINGULARIDADES.....	45
<b>5. VISÕES DE AVALIAÇÃO .....</b>	<b>47</b>
VISÃO DE QUALIDADE INTERNA .....	48
VISÃO DE QUALIDADE EXTERNA.....	49
VISÃO DE QUALIDADE EM USO.....	49
MAPEAMENTO DE VISÕES DE AVALIAÇÃO PARA MÓDULOS LAPS.....	49
<i>Visão de qualidade interna</i> .....	49
<i>Visão de qualidade externa</i> .....	51
<i>Visão de qualidade em uso</i> .....	54
MAPEAMENTO DE VISÕES DE AVALIAÇÃO PARA ATRIBUTOS DE QUALIDADE .....	57
<b>6. CONCLUSÕES E TRABALHOS FUTUROS .....</b>	<b>59</b>
PRINCIPAIS CONTRIBUIÇÕES.....	59
DIFICULDADES ENCONTRADAS .....	60
TRABALHOS FUTUROS.....	60
CONSIDERAÇÕES FINAIS .....	61
<b>REFERÊNCIAS .....</b>	<b>63</b>
<b>APÊNDICE A – MÉTRICAS PARA QUALIDADE EM USO .....</b>	<b>66</b>
MÉTRICAS DE EFICÁCIA .....	66
MÉTRICAS DE PRODUTIVIDADE .....	67
MÉTRICAS DE SEGURANÇA .....	69

MÉTRICAS DE SATISFAÇÃO .....	70
<b>APÊNDICE B – SUBCARACTERÍSTICAS DE QUALIDADE INTERNA E EXTERNA .....</b>	<b>72</b>
<b>APÊNDICE C – MAPEAMENTO DAS CARACTERÍSTICAS DE QUALIDADE DE SOFTWARE DA NORMA 9126-1 NOS MÓDULOS DE AVALIAÇÃO LAPS .....</b>	<b>74</b>
<b>APÊNDICE D - GLOSSÁRIO .....</b>	<b>76</b>
A – E.....	76
F – J.....	77
K – O .....	77
P – T.....	77
U – Z.....	78
<b>APÊNDICE E - ARTE.....</b>	<b>79</b>

## ÍNDICE DE FIGURAS

Figura 1. Gráfico de Conhecimento das Normas ISO.....	14
Figura 2. Conhecimento das normas nas empresas de Software.....	15
Figura 3. 9126 vs. Região .....	15
Figura 4. 12119 vs. Região .....	15
Figura 5. 14598 vs. Região .....	16
Figura 6. Estrutura do LAPS.....	20
Figura 7. Módulos de Avaliação LAPS.....	21
Figura 8. Qualidade e ciclo de vida.....	33
Figura 9. Agrupamento em visões .....	37
Figura 10. Processo de Planejamento e Gerenciamento LAPS .....	44
Figura 11. Visão de Fases do Microsoft Operations Framework .....	48
Figura 12. Logo Envision Básico.....	79
Figura 13. Logo Envision Básico com subtítulo.....	79
Figura 14. Logo Envision Cromado .....	79
Figura 15. Logo Envision Cromado com subtítulo .....	79
Figura 16. Logo Envision Prata.....	80
Figura 17. Logo Envision Prata com subtítulo.....	80
Figura 18. Logo Envision Ouro.....	80
Figura 19. Logo Envision Ouro com subtítulo.....	80
Figura 20. Logo Envision Negativo.....	80
Figura 21. Logo Envision Negativo com subtítulo.....	81
Figura 22. Logo Envision marca d'água.....	81

## ÍNDICE DE TABELAS

Tabela 1. O atributo "Momento de realização" .....	25
Tabela 2. O atributo "Flexibilidade" .....	25
Tabela 3. O atributo "Subjetividade / Objetividade" .....	26
Tabela 4. O atributo "Escopo" .....	26
Tabela 5. O atributo "Utilização" .....	27
Tabela 6. O atributo "Complexidade" .....	27
Tabela 7. O atributo "Extensibilidade" .....	28
Tabela 8. O atributo "Suportada por normas e padrões" .....	28
Tabela 9. O modelo "Guia CenPRA" .....	29
Tabela 10. O modelo "Colombo & Guerra" .....	30
Tabela 11. O modelo "Square" .....	31
Tabela 12 – Análise Comparativa .....	32
Tabela 13. Mapeamento Visões vs. Módulos LAPS .....	56
Tabela 14. Mapeamento Visões vs. Atributos de Qualidade .....	58
Tabela 15. Métricas de Eficácia .....	67
Tabela 16. Métricas de Produtividade .....	68
Tabela 17. Métricas de Segurança .....	70
Tabela 18. Métricas de Satisfação .....	71
Tabela 19. Subcaracterísticas de qualidade interna e externa .....	72
Tabela 20. Módulos LAPS vs. Subcaracterísticas ISO .....	74

## 1. Introdução

---

O ecossistema pernambucano de empresas de software e tecnologia da informação conta atualmente com um grande número de empresas [Porto]. Na grande maioria das áreas de atuação destas empresas, mais de uma concorrente direta existe. Além disso, o mercado de software proporciona situações de livre concorrência entre empresas regionais e de outras localidades, inclusive do exterior.

Ao visar à venda e proteção de seu produto contra a concorrência local, ou buscando penetrar em novos mercados, as empresas de software buscam alternativas que se apresentem como um diferencial de qualidade em seus produtos, objetivando aceitação do usuário final. Da necessidade de agregar valor e qualidade ao software pernambucano surgiram várias iniciativas com foco em qualidade no processo de desenvolvimento de um software e uma em especial com foco em qualidade do produto de software produzido.

O LAPS, **L**aboratório de **A**valiação de **P**rodutos de **S**oftware [LAPS] foi o agente motivador deste trabalho. Ele define um modelo de avaliação de produtos de software baseado em doze módulos de avaliação [Sampaio & Perrelli 2004], cada um deles abordando uma característica distinta de um software: arquitetura, código fonte, segurança, falha e recuperação, documentação do sistema, documentação do usuário, portabilidade, usabilidade, análise especialista, competidores, funcionalidade e desempenho.

O arcabouço teórico em qualidade de produto de software é extenso: existem diversos artigos publicados, livros e material disponibilizado sob outras formas, principalmente via Internet. No entanto, as mais consistentes referências sobre o assunto são encontradas nas normas da ISO, *International Organization for Standardization* [ISO], que definem um modelo de qualidade [ISO 9126-1, 2002], modelos para avaliação de produto de software [ISO 14598-1, 1998] e características de qualidade presentes em um software que são mensuráveis a partir de métricas. As normas ISO tornam possível mapear atributos abstratos e complexos de qualidade em valores numéricos através da aplicação de diversas técnicas. Atualmente, não há nenhum registro formal que ajude a mapear os módulos de avaliação LAPS nos atributos de qualidade de software definidos pela ISO.

## ***Objetivos do trabalho***

O objetivo deste trabalho é a criação de um modelo que mapeie os atributos de qualidade de software definidos em [ISO 9126-1, 2002], nos módulos de avaliação do LAPS, e vice-versa. Indo além, pretende-se desenvolver o conceito de “Visões de Avaliação”, pacotes que agregam diversos módulos de avaliação LAPS e atributos de qualidade de software em uma única perspectiva para, por exemplo, poder medir a qualidade de um produto em uso [Bevan 1999].

O modelo de visões desenvolvido poderá ser aplicado a qualquer laboratório de avaliação de produtos, visto que ele mapeia diretamente as visões propostas aos atributos definidos pela ISO. Ele também ajuda na venda de avaliações de produto de software, expondo para o cliente uma visão de serviços de avaliação de simples compreensão na forma de pacotes, ou “Visões de Avaliação”. Além de atingir diretamente os atributos de qualidade requisitados pelo cliente para uma avaliação de produto, uma visão pode também utilizar um ou mais módulos de avaliação LAPS. O modelo como um todo, uma vez finalizado, servirá como ferramenta de suporte a decisão e aquisição dos serviços de avaliações e ajudará a agregar valor aos produtos de software avaliados utilizando uma ou mais visões propostas pelo modelo.

### **Objetivo Principal**

O principal objetivo do trabalho é propor um modelo de avaliação de produtos de software por visões.

### **Objetivos Específicos**

O trabalho pretende também contemplar alguns objetivos específicos:

- Estar de acordo com a série de normas ISO 9126 e ISO 14598.
- Ser flexível o bastante para se adaptar ao modelo atual do LAPS [LAPS] ou qualquer outro laboratório de avaliação de produtos de software.
- Agregar valor à avaliação de produtos de software e aos softwares avaliados segundo o modelo.

## Relevância

A busca por agregar qualidade ao software produzido no Brasil vem crescendo. No entanto ainda é considerado baixo o número de empresas que utilizam em seus produtos e processos, métodos e metodologias baseadas em normas de qualidade internacional como as da ISO.

Segundo pesquisa do [MCT 2002], realizada em 2001 entre mais de 400 (quatrocentas) empresas que desenvolvem software no Brasil, é baixíssimo o número de empresas que conhece e usa as três principais normas de qualidade para software da ISO, a 12119 [ISO 12119, 1998] e as séries de normas 9126 [ISO 9126-1, 2001] [ISO 9126-2, 2001] [ISO 9126-3, 2001] [ISO 9126-4, 2001] e 14598 [ISO 14598-1, 1998] [ISO 14598-2, 1998] [ISO 14598-5, 1998] [ISO 14598-6, 1998].

Os dados apresentados são vistos na “Figura 1. Gráfico de Conhecimento das Normas ISO” e “Figura 2. Conhecimento das normas nas empresas de Software”, ambas retiradas de [MCT 2002]. A primeira delas mostra de forma gráfica o uso das três normas citadas. Em azul, podemos observar que o uso das normas dentro das empresas ainda é baixo, chegando a um uso de no máximo 11%, na norma 9126. Isso corresponde a uma fatia ínfima das empresas de software no Brasil. Se considerarmos o uso sistemático destas normas, como observado na “Figura 2. Conhecimento das normas nas empresas de Software” extraída de [MCT 2002], veremos que esse número é ainda menor, chegando próximo a 4% de uso.

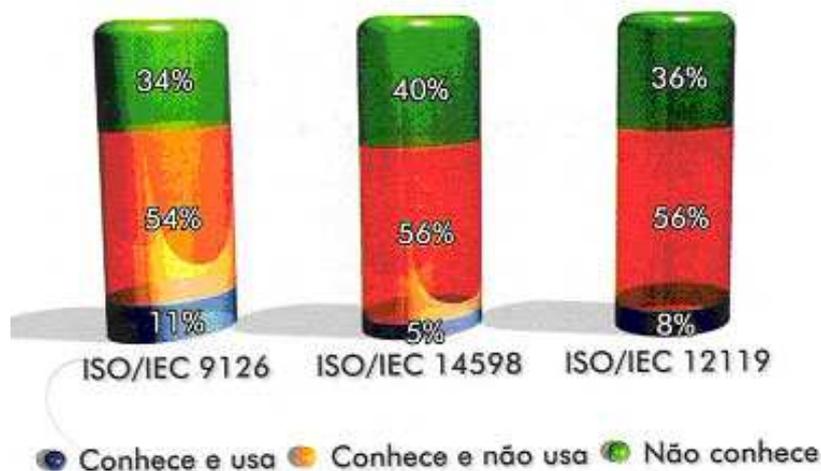


Figura 1. Gráfico de Conhecimento das Normas ISO

Categorias	NBR 13596 (ISO/IEC 9126)		ISO/IEC 14598		ISO/IEC 12119	
	Nº	%	Nº	%	Nº	%
Conhece e usa sistematicamente	16	3,9	5	1,2	10	2,4
Conhece e começa a usar	31	7,5	14	3,4	22	5,4
Conhece, mas não usa	224	54,4	228	55,6	230	56,0
Não conhece	141	34,2	163	39,8	149	36,3
<b>Base</b>	<b>412</b>	<b>100</b>	<b>410</b>	<b>100</b>	<b>411</b>	<b>100</b>

**Figura 2. Conhecimento das normas nas empresas de Software**

Apesar dos números baixos, a região Nordeste ainda é referência em qualidade no Brasil. Esta região possui os números mais altos de conhecimento e adoção das normas supracitadas, como podemos observar na “Figura 3. 9126 vs. Região”, “Figura 4. 12119 vs. Região” e “Figura 5. 14598 vs. Região” também retiradas de [MCT 2002]. Elas mostram que o uso prático das normas 9126 e 14598 é maior na região Nordeste que em outras regiões. Este uso é somente superado na norma 12119, que trata de testes e requisitos de qualidade, mais utilizada no eixo Sul-Sudeste.

Categorias	Total	Sudeste	Sul	Nordeste	Centro-oeste e Norte
Conhece e usa sistematicamente	3,9	3,9	3,8	4,2	3,2
Conhece e começa a usar	7,5	6,1	7,7	8,5	12,9
Conhece, mas não usa	54,4	50,6	60,0	49,3	64,5
Não conhece	34,2	39,4	28,5	38,0	19,4

**Figura 3. 9126 vs. Região**

Categorias	Total	Sudeste	Sul	Nordeste	Centro-oeste e Norte
Conhece e usa sistematicamente	2,4	2,2	3,1	1,4	3,1
Conhece e começa a usar	5,4	5,6	5,4	2,9	9,4
Conhece, mas não usa	56,0	49,4	63,1	53,6	68,8
Não conhece	36,3	42,8	28,5	42,0	18,8

**Figura 4. 12119 vs. Região**

Categorias	Total	Sudeste	Sul	Nordeste	Centro-oeste e Norte
Conhece e usa sistematicamente	1,2	1,1	0,8	1,4	3,2
Conhece e começa a usar	3,4	5,6	2,3	-	3,2
Conhece, mas não usa	55,6	51,1	64,3	48,6	61,3
Não conhece	39,8	42,2	32,6	50,0	32,3

**Figura 5. 14598 vs. Região**

A concorrência no mercado mundial de software faz com que seja cada vez mais difícil vender novos softwares, ou criar novas fatias no mercado (*market share*). Em geral, os softwares produzidos para uma área de negócio específica realizam as mesmas funções se distinguindo por fatores que significam qualidade para o cliente final, ou adquirente, como por exemplo: usabilidade, baixa taxa de erros, reusabilidade de componentes dentre outros fatores. Para atingir este diferencial competitivo que faça com que os softwares de uma empresa consigam ser vendidos se observa a nível mundial uma busca cada vez maior por qualidade, o que justifica a implantação de modelos de qualidade para os produtos de software produzidos por uma empresa, como o proposto neste trabalho.

A predisposição à aplicação e ao uso de modelos que venham a agregar qualidade ao software produzido, foi uma das razões da escolha do LAPS, empresa pernambucana, como base para o desenvolvimento do modelo de visões aqui desenvolvido.

### ***Metodologia de trabalho***

Escrito pelo aluno de graduação em Ciências da Computação na Universidade Federal de Pernambuco Augusto César Pires Spinelli, sob orientação do professor Hermano Perrelli de Moura e co-orientação do professor Alexandre Marcos Lins de Vasconcelos. Esta monografia foi escrita seguindo algumas das boas práticas do [MSF], *framework* de processos e boas práticas para software da Microsoft. A escrita deste trabalho foi realizada através de ciclos incrementais e, em cada ciclo, as atividades de pesquisa de material bibliográfico, análise, escrita e revisão aconteciam. Estas atividades eram realizadas em conjunto pelo autor e seus orientadores, salvo a escrita do trabalho de graduação, realizada somente pelo aluno autor. Em particular, durante o primeiro ciclo, foi também definido o objetivo principal do trabalho e seus

principais capítulos, e a partir disto montada uma proposta para o trabalho de graduação.

### ***Organização da monografia***

O trabalho está organizado em 6 (seis) capítulos:

No capítulo 2 (dois) o leitor irá conhecer o LAPS – Laboratório de Avaliação de Produtos de Software, através de um breve histórico desde sua criação, sua estrutura organizacional e, resumidamente, o modelo de avaliação de produtos de software utilizado no laboratório.

O capítulo 3 (três) analisa trabalhos similares ao desenvolvido, com relação aos modelos de avaliação apresentados, a sua abrangência, *know-how* de seus participantes, situação atual e um comparativo com o modelo proposto neste trabalho.

O capítulo 4 (quatro) desenvolve o conceito de “visão de avaliação” com base nas normas internacionais da *International Organization for Standardization* [ISO]. Este capítulo também discrimina conceitos base para o modelo de avaliação proposto como, por exemplo, atributos de qualidade de software e exemplifica algumas métricas aplicadas na avaliação destas características.

No capítulo 5 (cinco) é descrito o modelo de avaliação proposto, com base nos conceitos apresentados no capítulo anterior. Neste capítulo também são exemplificadas três visões de avaliação e realizado o mapeamento do modelo proposto no atual modelo de avaliação do Laboratório de Avaliação de Produtos de Software [LAPS].

O capítulo 6 (seis) apresenta as conclusões do trabalho desenvolvido, assim como algumas projeções futuras deste trabalho.

Em seguida, também são apresentadas as referências bibliográficas utilizadas e alguns apêndices com material relevante ao trabalho. O Apêndice A apresenta um resumo das métricas para medição da qualidade em uso de um produto. O Apêndice

B discrimina as subcaracterísticas de qualidade interna e externa. O Apêndice C mostra o mapeamento das características de qualidade nos módulos de avaliação LAPS. O Apêndice D apresenta um glossário com termos que serão usados em diversos pontos deste trabalho. É recomendada a leitura deste apêndice, caso o leitor não esteja familiarizado com os termos da Qualidade de Software. Finalmente, o Apêndice E apresenta o trabalho de arte gráfica desenvolvido em conjunto a este trabalho de graduação.

## **2. O LAPS**

---

O LAPS, Laboratório de Avaliação de Produtos de Software, surgiu a partir de uma demanda local pela garantia e aumento da qualidade do software produzido no estado. Ele foi montado visualizando um nicho de mercado até então não explorado na região: avaliação de produtos de software. No momento de sua criação, os maiores expoentes de avaliação de produto se encontravam nas regiões Sul, na cidade de Curitiba, e Sudeste, na cidade de Campinas. Havia também, naquele momento, um laboratório de avaliação de produtos de software na cidade de Fortaleza, estado do Ceará. Ele, no entanto, não atendia a demanda local pernambucana.

### ***Breve Histórico***

Criado em meados de 2004 no Centro de Informática da Universidade Federal de Pernambuco surge o LAPS, através da ação de seus idealizadores, em especial ao professor Hermano Perrelli do Centro de Informática e à aluna de mestrado Vânia Vidal Sampaio. O laboratório também teve o apoio de parceiros como o Recife BEAT e o agente SOFTEX Recife. Em seu primeiro ano de trabalho, foram realizadas seis avaliações de produtos locais e uma avaliação piloto, incluindo todos os módulos de avaliação definidos pelo laboratório.

### ***Estrutura***

O LAPS possui sua estrutura física localizada dentro do Centro de Informática e uma equipe altamente qualificada na área de qualidade de software e engenharia de software em sua grande maioria doutores do Centro de Informática. Cada componente dessa equipe é especialista em uma, ou mais, áreas de domínio dos sistemas de Tecnologia da Informação (TI) e juntamente com os avaliadores LAPS, compostos de alunos de graduação e mestrado, são capazes de realizar avaliações de software com alto valor agregado.

Hierarquicamente o LAPS possui um coordenador geral, um conselho consultivo (composto de parceiros na área de qualidade de software) e uma gerência interna além dos avaliadores e especialistas supracitados. Esta organização hierárquica pode ser melhor visualizada através da “Figura 6. Estrutura do LAPS”.

## ESTRUTURA



**Figura 6. Estrutura do LAPS**

### *Modelo de avaliação*

O LAPS possui um modelo de avaliação baseado em doze módulos dispostos nas seguintes categorias, segundo [Sampaio & Perrelli 2004]:

1. Arquitetura
2. Documentação do sistema
3. Documentação do usuário
4. Funcionalidade
5. Portabilidade
6. Código Fonte
7. Usabilidade
8. Falhas e recuperação
9. Controle de acesso e proteção de dados
10. Desempenho
11. Competidores
12. Especialista

A idéia por trás desta divisão em módulos é segmentar a avaliação de qualidade de produto em pedaços customizáveis, de acordo com a necessidade do usuário e o grau de importância com o qual ele trata cada um desses módulos em sua aplicação. Um cliente LAPS escolhe que módulos de avaliação irá utilizar em seu sistema, atingindo de forma mais dedicada ou mais abrangente o mesmo. Os doze módulos de avaliação LAPS também estão ilustrados neste trabalho na “Figura 7. Módulos de Avaliação LAPS”.

<b>Avaliação da Arquitetura</b>
<b>Avaliação da Documentação do Sistema</b>
<b>Avaliação da Funcionalidade</b>
<b>Avaliação da Portabilidade</b>
<b>Avaliação da Usabilidade</b>
<b>Avaliação do Código Fonte</b>
<b>Avaliação de Desempenho</b>
<b>Avaliação da Documentação do Usuário</b>
<b>Avaliação de Falhas e Recuperação</b>
<b>Avaliação de Controle de Acesso e Proteção de Dados</b>
<b>Avaliação de Competidores</b>
<b>Avaliação de Especialista</b>

**Figura 7. Módulos de Avaliação LAPS**

A “Figura 7. Módulos de Avaliação LAPS” mostra a disposição dos doze módulos de avaliação do laboratório. Não há nenhuma ordem de grandeza ou prioridade envolvida nos módulos de avaliação LAPS. A avaliação dos módulos pode ser realizada de forma independente, através da livre escolha de um, ou mais, módulos em uma avaliação. Existe um módulo adicional no laboratório, não mostrado na figura, chamado de “Módulo de Planejamento e Gerenciamento”. Este módulo é responsável por descrever o processo necessário para iniciar e gerenciar uma avaliação. Todos os módulos LAPS possuem uma atividade em comum chamada “Iniciar Análise”, que é regida pelo Módulo de Planejamento e Gerenciamento, esta atividade elucida a forma de montar o plano de avaliação, importante documento que servirá de base para assinatura do contrato entre o laboratório e o cliente requisitante.

Caso o cliente escolha todos os módulos do laboratório para uma determinada avaliação, todos eles devem estar contidos num mesmo plano de trabalho. Dizemos então que foi requisitada uma avaliação completa de produto.

## *Dificuldades enfrentadas*

Em mais de um ano de atuação do laboratório foi identificada uma grande dificuldade junto aos clientes: a escolha dos módulos de avaliação a serem aplicados ao seu sistema. A maioria dos clientes tinha uma visão mais abstrata dos aspectos de melhoria desejada para seu produto, como, por exemplo, melhorar a performance do sistema, mas não conseguia visualizar inicialmente que módulos LAPS utilizar para a resolução do seu problema.

Objetivando ao mesmo tempo unir a visão mais abstrata dos atributos de qualidade possuída pelos clientes aos módulos de avaliação LAPS, este trabalho de graduação irá propor um modelo de avaliação baseado em visões que irão mapear diretamente os atributos de qualidade a um ou mais módulos LAPS. Ele também irá fazer o caminho inverso, mapeando os módulos LAPS em atributos de qualidade de software. Indo além do descrito acima, este trabalho será um modelo extensível permitindo a montagem de novas visões de avaliação de produto de software (além das três que serão propostas neste trabalho), que terão o papel de agregar em pacotes diversos atributos de qualidade de software e, conseqüentemente, módulos LAPS. Deste modo, a abordagem de escolha de avaliação para o cliente torna-se mais fácil, e o resultado obtido pela avaliação mais transparente, pois desde o início o cliente já sabe quais atributos de qualidade serão abordados na avaliação de seu produto de software.

### **3. Trabalhos Similares**

---

Esta seção tem como objetivo apresentar o resultado da pesquisa realizada para viabilizar o desenvolvimento deste trabalho, que aborda a área de avaliação de produtos de software, propondo uma abordagem de avaliação baseada no conceito de visão que será elaborado. As características de vários modelos de mesmo âmbito, já disponíveis no mercado, são esmiuçadas em relação aos seus pontos fortes e fracos, devassando os aspectos em que os modelos podem melhorar e quais aqueles que podem servir de exemplo para o trabalho proposto. Tal avaliação será fundamentada em critérios, como ausência de um determinado tema ou atributo de qualidade, inconsistências e oportunidades de melhoria. Para avaliarmos a fundo cada um dos modelos apresentados precisaríamos de um tempo maior e de uma pesquisa mais elaborada, no entanto, essa temática não pertence ao objetivo desta seção ou deste trabalho. É importante frisar que a intenção é apenas verificar o que esses modelos concorrentes têm a oferecer, para melhor ambientar e desenvolver o modelo proposto neste trabalho.

O modelo proposto, além de abordar vários tópicos vistos a seguir, tentará propor assuntos que ainda não foram explorados e incrementar os pontos fracos de seus concorrentes, buscando agregar mais valor a uma avaliação de produto de software.

#### ***Atributos de comparação dos sistemas***

A qualidade de um produto de software pode ser descrita em termos das características de software definidas na [ISO 9126-2, 2002]. Entretanto, o estado da arte em medição de qualidade de um software faz com que a abordagem de medição direta dessas características nem sempre seja a mais prática para todas as situações. Uma possibilidade é julgar essas características baseando-se na medição de atributos do produto obtidos em níveis mais baixos de abstração, contexto no qual o avaliador pode usar sua experiência de avaliação e a engenharia de software para fazer o julgamento. Outro aspecto a ser considerado é a possibilidade de utilizar métodos de avaliação não determinísticos que, apesar de bem definidos, podem requerer escolhas que podem não estar predefinidas. [ISO 14598-5, 1998].

Em seguida é elaborado um modelo de atributos de comparação mais leve que o modelo definido pela [ISO 9126-2, 2002], visando simplesmente comparar os modelos de avaliação competidores com o modelo proposto neste trabalho. Este modelo de atributos de comparação será baseado na experiência pessoal do autor deste trabalho e na análise das referências lidas, buscando manter o nível de objetividade da avaliação o mais alto possível em todas as circunstâncias.

“Um exemplo de método de avaliação não determinístico pode ser a tradução de um componente de especificação de produto em um modelo formal, efetuando a avaliação do desempenho ou confiabilidade deste modelo; a fase de tradução envolve muitas opções a serem feitas pelo avaliador [ISO 14598-5, 1998]”.

Atributos foram definidos pelo autor para a análise comparativa dos competidores do modelo proposto, de acordo com a técnica de avaliação heurística. Será atribuído um nome a cada atributo, sua possível lista de valores e o mapeamento destes valores numa escala discreta para posterior análise comparativa. Observações pertinentes aos atributos também serão apresentadas.

No total foram definidos 8 (oito) atributos que são apresentados nas tabelas: Tabela 1. O atributo “Momento de realização”, Tabela 2. O atributo “Flexibilidade”, Tabela 3. O atributo “Subjetividade / Objetividade”, Tabela 4. O atributo “Escopo”, Tabela 5. O atributo “Utilização”, Tabela 6. O atributo “Complexidade”, Tabela 7. O atributo “Extensibilidade” e Tabela 8. O atributo "Suportada por normas e padrões".

Tabela 1. O atributo “Momento de realização”

<b>Nome do Atributo</b>	Momento de realização
<b>Valoração</b>	<ul style="list-style-type: none"> <li>• Somente com o produto finalizado e posto em ambiente de produção no cliente (**)</li> <li>• Com o produto finalizado (***)</li> <li>• Qualquer momento após atingir a fase de testes [RUP] / estabilização [MSF] no desenvolvimento (****)</li> <li>• Em qualquer fase: durante o desenvolvimento, com o produto finalizado ou após posto em ambiente de produção (*****)</li> </ul>
<b>Classificação</b>	<ul style="list-style-type: none"> <li>• <b>Inexistente (*)</b></li> <li>• <b>Fraco (**)</b></li> <li>• <b>Regular (***)</b></li> <li>• <b>Bom (****)</b></li> <li>• <b>Excelente (*****)</b></li> </ul>
<b>Observações</b>	A classificação inexistente não se aplica a este atributo, pois isso significaria que a avaliação não pode ser realizada.

Tabela 2. O atributo “Flexibilidade”

<b>Nome do Atributo</b>	Flexibilidade
<b>Valoração</b>	<ul style="list-style-type: none"> <li>• Nula (*)</li> <li>• Pouca flexibilidade (**)</li> <li>• Razoável (***)</li> <li>• Boa flexibilidade (****)</li> <li>• Totalmente flexível (*****)</li> </ul>
<b>Classificação</b>	<ul style="list-style-type: none"> <li>• <b>Inexistente (*)</b></li> <li>• <b>Fraco (**)</b></li> <li>• <b>Regular (***)</b></li> <li>• <b>Bom (****)</b></li> <li>• <b>Excelente (*****)</b></li> </ul>
<b>Observações</b>	O atributo flexibilidade neste contexto trata da possibilidade da avaliação se adaptar a diferentes cenários, por exemplo, o mesmo modelo pode ser adaptado para avaliação de um produto de software desenvolvido para a indústria espacial e para um sistema de informação simples, desenvolvido por alunos de uma universidade?

Tabela 3. O atributo “Subjetividade / Objetividade”

<b>Nome do Atributo</b>	Subjetividade / Objetividade
<b>Valoração</b>	<ul style="list-style-type: none"> <li>• Totalmente subjetiva (**)</li> <li>• Muito subjetiva (***)</li> <li>• Pouco subjetiva (****)</li> <li>• Totalmente objetiva (*****)</li> </ul>
<b>Classificação</b>	<ul style="list-style-type: none"> <li>• <b>Inexistente (*)</b></li> <li>• <b>Fraco (**)</b></li> <li>• <b>Regular (***)</b></li> <li>• <b>Bom (****)</b></li> <li>• <b>Excelente (*****)</b></li> </ul>
<b>Observações</b>	Quanto mais objetiva for uma avaliação melhor, pois erros humanos são evitados e a imparcialidade da avaliação é maior. A classificação inexistente não se aplica a este atributo.

Tabela 4. O atributo “Escopo”

<b>Nome do Atributo</b>	Escopo
<b>Valoração</b>	<ul style="list-style-type: none"> <li>• Somente o software (**)</li> <li>• Software e documentação (***)</li> <li>• Software como um todo (caixa, documentação do sistema e do usuário, código fonte, comportamento...) (****)</li> <li>• Software por completo e sua qualidade em uso. (*****)</li> </ul>
<b>Classificação</b>	<ul style="list-style-type: none"> <li>• <b>Inexistente (*)</b></li> <li>• <b>Fraco (**)</b></li> <li>• <b>Regular (***)</b></li> <li>• <b>Bom (****)</b></li> <li>• <b>Excelente (*****)</b></li> </ul>
<b>Observações</b>	O escopo da avaliação refere-se, neste contexto, a até que ponto o modelo analisado avalia o produto de software. A classificação inexistente não se aplica a este atributo, pois isso significaria que a avaliação não pode ser realizada.

Tabela 5. O atributo “Utilização”

<b>Nome do Atributo</b>	Utilização
<b>Valoração</b>	<ul style="list-style-type: none"> <li>• Difícil, somente com ajuda de terceiros. (**)</li> <li>• Difícil (***)</li> <li>• Fácil com necessidade de treinamento (****)</li> <li>• Fácil sem necessidade de treinamento (*****)</li> </ul>
<b>Classificação</b>	<ul style="list-style-type: none"> <li>• <b>Inexistente (*)</b></li> <li>• <b>Fraco (**)</b></li> <li>• <b>Regular (***)</b></li> <li>• <b>Bom (****)</b></li> <li>• <b>Excelente (*****)</b></li> </ul>
<b>Observações</b>	A utilização refere-se à facilidade de aplicação do modelo analisado. Em um cenário ideal, somente a leitura dos passos do modelo bastaria para que o mesmo fosse utilizado. A classificação inexistente não se aplica a este atributo, pois isso significaria que a avaliação não pode ser realizada.

Tabela 6. O atributo “Complexidade”

<b>Nome do Atributo</b>	Complexidade
<b>Valoração</b>	<ul style="list-style-type: none"> <li>• Muito simples (insuficiente) (**)</li> <li>• Complexa (***)</li> <li>• Razoavelmente complexo (****)</li> <li>• Simples (*****)</li> </ul>
<b>Classificação</b>	<ul style="list-style-type: none"> <li>• <b>Inexistente (*)</b></li> <li>• <b>Fraco (**)</b></li> <li>• <b>Regular (***)</b></li> <li>• <b>Bom (****)</b></li> <li>• <b>Excelente (*****)</b></li> </ul>
<b>Observações</b>	A complexidade, neste contexto, refere-se à quantidade de passos e atividades necessários para a execução da avaliação, de acordo com o modelo analisado. Quanto mais esmiuçada for a avaliação, dividindo-se num maior número de passos e atividades, mais complexa ela se torna. Em contrapartida, um modelo de complexidade muito baixa torna a avaliação insuficiente para uma boa análise da qualidade do produto de software. A classificação inexistente não se aplica a este atributo.

Tabela 7. O atributo “Extensibilidade”

<b>Nome do Atributo</b>	Extensibilidade
<b>Valoração</b>	<ul style="list-style-type: none"> <li>• Nula (*)</li> <li>• Existem poucos pontos de extensão e esta extensão é restrita aos idealizadores do modelo (**)</li> <li>• Existem poucos pontos de extensão, mas estes são abertos à comunidade (***)</li> <li>• Possui pontos de extensão, mas estes são restritos aos idealizadores do modelo (****)</li> <li>• Possui pontos de extensão e estes são abertos à comunidade (*****)</li> </ul>
<b>Classificação</b>	<ul style="list-style-type: none"> <li>• <b>Inexistente (*)</b></li> <li>• <b>Fraco (**)</b></li> <li>• <b>Regular (***)</b></li> <li>• <b>Bom (****)</b></li> <li>• <b>Excelente (*****)</b></li> </ul>
<b>Observações</b>	A extensibilidade, neste contexto, analisa se o modelo em questão disponibiliza pontos de extensão, se adaptando a novas práticas que venham a surgir em qualidade de software. Ela também analisa se a extensão do modelo é restrita aos seus idealizadores, ou aberta à comunidade de avaliação de produto de software.

Tabela 8. O atributo "Suportada por normas e padrões"

<b>Nome do Atributo</b>	Suportada por normas e padrões
<b>Valoração</b>	<input checked="" type="checkbox"/> Sim <input type="checkbox"/> Não
<b>Classificação</b>	<input checked="" type="checkbox"/> Sim <input type="checkbox"/> Não
<b>Observações</b>	Um modelo que é suportado por normas e padrões aceitos internacionalmente tem uma maior credibilidade.

### Análise dos modelos

Em virtude do tema abordado neste trabalho possuir um grande número de modelos similares ao Envision, foi dada preferência de análise àqueles modelos que abordam a avaliação de produto de software como um todo e não somente uma parte do software (por exemplo, modelos que abordam somente avaliação do código fonte). Os modelos selecionados são um subconjunto do estado da arte de avaliação de produtos, e têm projeção dentro do âmbito acadêmico nacional ou internacional.

Para cada um dos três modelos analisados foi montada uma tabela onde é apresentado o nome do modelo, seu(s) autor(es), algumas características do modelo

e pontos positivos e negativos do mesmo. Para os modelos “Guia CenPRA”, “Colombo & Guerra” e “Square” são apresentadas, respectivamente, as tabelas: “Tabela 9. O modelo "Guia CenPRA"”, “Tabela 10. O modelo "Colombo & Guerra"” e “Tabela 11. O modelo "Square"”.

**Tabela 9. O modelo "Guia CenPRA"**

<b>Projeto</b>	Guia de Avaliação da Qualidade de Produto de Software [Guia CenPRA]
<b>Autor(es)</b>	CenPRA – Centro de Pesquisas Renato Archer
<b>Características</b>	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Modelo maduro, resultado de cerca de 10 anos de pesquisa no Centro de Pesquisas Renato Archer.</li> <li><input checked="" type="checkbox"/> Modelo de avaliação de produto baseado no MEDE-PROS, Método de Avaliação da Qualidade de Produto de Software, do próprio CenPRA.</li> </ul>
<b>Pontos fortes</b>	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Apresentado sob a forma de um guia passo-a-passo.</li> <li><input checked="" type="checkbox"/> Presença mínima de subjetividade.</li> <li><input checked="" type="checkbox"/> Abrangente, avaliando além do produto de software, sua caixa, instalação e documentação do usuário.</li> </ul>
<b>Pontos fracos</b>	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Necessidade de Treinamento no método (MEDE-PROS) e no guia de avaliação.</li> <li><input checked="" type="checkbox"/> Proprietário do CenPRA. O uso do guia envolve pagamento de um certo valor ao CenPRA por cada avaliação executada seguindo o modelo.</li> <li><input checked="" type="checkbox"/> Modelo sem flexibilidade.</li> <li><input checked="" type="checkbox"/> Extensão do modelo só é permitida dentro do CenPRA.</li> <li><input checked="" type="checkbox"/> Focada somente em softwares de prateleira.</li> </ul>
<b>Observações</b>	Elaborado com base nas normas de avaliação de produto de software da ISO (séries ISO/IEC 9126 e 14598 e norma 12119)

**Tabela 10. O modelo "Colombo & Guerra"**

<b>Projeto</b>	The Evaluation Method for Software Product [Colombo & Guerra 2002]
<b>Autor(es)</b>	Regina Colombo e Ana Guerra
<b>Características</b>	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Toda a avaliação está baseada em três conceitos: <i>measurement</i> (medição), <i>rating</i> (avaliação), <i>judgement</i> (julgamento).</li> <li><input checked="" type="checkbox"/> O modelo foi idealizado para ser utilizado em conjunto com o MEDE-PROS método de avaliação do CenPRA.</li> <li><input checked="" type="checkbox"/> Ele se apresenta como um <i>framework</i>, propondo boas práticas para a melhoria da qualidade na empresa.</li> </ul>
<b>Pontos fortes</b>	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Simplicidade, quanto à facilidade de aprendizado e aplicação.</li> <li><input checked="" type="checkbox"/> Aberto a comunidade científica, na forma de artigo.</li> <li><input checked="" type="checkbox"/> Proposta de melhorias e avanços para o MEDE-PROS.</li> <li><input checked="" type="checkbox"/> Pode ser adaptado de acordo com as necessidades da empresa que o utiliza.</li> </ul>
<b>Pontos fracos</b>	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Modelo bastante novo, ainda imaturo.</li> <li><input checked="" type="checkbox"/> Dependência do MEDE-PROS</li> </ul>
<b>Observações</b>	Elaborado com base nas normas de avaliação de produto de software da ISO (séries ISO/IEC 9126 e 14598 e norma 12119)

**Tabela 11. O modelo "Square"**

<b>Projeto</b>	Software product <b>Quality Requirements and Evaluation</b> [Square 2004]
<b>Autor(es)</b>	ISO/IEC
<b>Características</b>	<input checked="" type="checkbox"/> Dividido em quatro vertentes: requisitos de qualidade, modelo de qualidade, medição de qualidade e avaliação de qualidade de um produto. <input checked="" type="checkbox"/> Os quatros vertentes são geridos por um módulo central de gerenciamento de qualidade.
<b>Pontos fortes</b>	<input checked="" type="checkbox"/> O passo a passo para executar uma avaliação é bem definido no modelo. <input checked="" type="checkbox"/> Explora qualidade como um todo (qualidade em uso, qualidade externa e qualidade interna). <input checked="" type="checkbox"/> Propõe métricas para medir cada característica definida no modelo de qualidade.
<b>Pontos fracos</b>	<input checked="" type="checkbox"/> Modelo ainda em finalização. <input checked="" type="checkbox"/> Extensibilidade do modelo limitada à ISO e comunidade envolvida com o desenvolvimento do modelo.
<b>Observações</b>	Modelo baseado na nova geração das séries ISO/IEC 9126 e 14598, proposto inicialmente em 1999 na reunião de Kanazawa e aprovado pelo SC7 (ISO Software and System Enginnering) em 2000 na reunião de Madri. Será a futura ISO/IEC 25000.

### *Tabela comparativa*

Buscando obter uma visão comparativa dos competidores, a “Tabela 12 – Análise Comparativa” os apresenta de acordo com os atributos de comparação definidos na seção anterior. Uma análise geral dos dados apresentados na tabela mostra que os modelos comparados apresentam resultados bons.

Dos competidores analisados o mais experiente é o [Guia CenPRA], com cerca de dez anos no mercado e um vasto histórico de avaliações realizadas. No entanto, o guia do CenPRA peca em alguns pontos como sua flexibilidade, que é baixa e a extensibilidade, que é muito reduzida. O momento de realização também é um ponto fraco deste modelo, visto que uma avaliação segundo o guia só pode ser realizada com o software pronto para a venda, além disso, a avaliação é restrita a softwares de prateleira apenas.

O modelo definido por [Colombo & Guerra 2002] é similar ao guia do CenPRA, com algumas peculiaridades. Como este modelo foi elaborado para apresentação à comunidade científica, ele se torna mais flexível que o guia do CenPRA, podendo ser utilizado em empresas e laboratórios de avaliação, sem custo financeiro para os utilizadores. Entretanto, o modelo ainda é bastante dependente do método de avaliação MEDE-PROS que, por ser pago, faz com que o método seja apresentado sob a forma de um artigo ainda insuficiente para que ele seja utilizado de imediato. No entanto, esta forma de apresentação abre precedentes para que qualquer pessoa interessada em utilizar os *guidelines* e boas práticas definidas neste *framework* faça de uma maneira que se adapte às suas necessidades, abrindo então, inúmeros pontos de extensão.

Dos competidores analisados o que tem mais perspectivas de futuro é o [Square 2004]. Ele é um modelo que vem sendo desenvolvido por gigantes, como a [ISO], e já foi aprovado em diversos eventos internacionais de renome sobre qualidade, como por exemplo, a reunião de Kanazawa em 1998 e a de Madri em 2000. Os dois grandes problemas identificados no Square são seu estado atual de incompletude e a dúvida com relação a sua utilização. Por não estar finalizado, o Square ainda não tem um modelo de negócios definido a ponto de dissertarmos sobre os custos envolvidos em realizar uma avaliação, mas provavelmente ela terá custos envolvidos, como por exemplo, a compra da futura série de normas ISO/IEC 25000 onde está definido o Square.

Tabela 12 – Análise Comparativa

Modelo	MR	FL	SO	ES	UT	CO	EX	NP
[Guia CenPRA]	***	**	*****	*****	*****	*****	*	✓
[Colombo & Guerra 2002]	***	***	*****	*****	*****	**	*****	✓
[Square 2004]	*****	*****	*****	*****	*****	*****	**	✓

**LEGENDA:**

- ☑ MR: Momento de realização
- ☑ FL: Flexibilidade
- ☑ SO: Subjetividade / Objetividade
- ☑ ES: Escopo
- ☑ UT: Utilização
- ☑ CO: Complexidade
- ☑ EX: Extensibilidade
- ☑ NP: Suportada por normas e padrões

**CLASSIFICAÇÃO:**

Inexistente (\*) / Fraco (\*\*) / Regular (\*\*\*) / Bom (\*\*\*\*) / Excelente (\*\*\*\*\*)

## Conclusões de análise

Buscando identificar pontos positivos e oportunidades de melhoria, foi realizada uma análise dos competidores do modelo que será proposto neste trabalho, o Envision. O objetivo principal desta análise é saber em que pontos o Envision pode se espelhar e em quais ele deve apresentar um diferencial competitivo com relação a seus concorrentes.

Um importante ponto identificado nesta análise foi a necessidade do modelo proposto possuir um momento de realização mais flexível, que possa se adaptar tanto a produtos ainda em finalização, quanto a produtos que já estão em uso. Para tal, o autor pretende elaborar nas visões propostas um modelo temporal, no qual cada visão irá se encaixar em um determinado momento do ciclo de vida de um software, para avaliação.

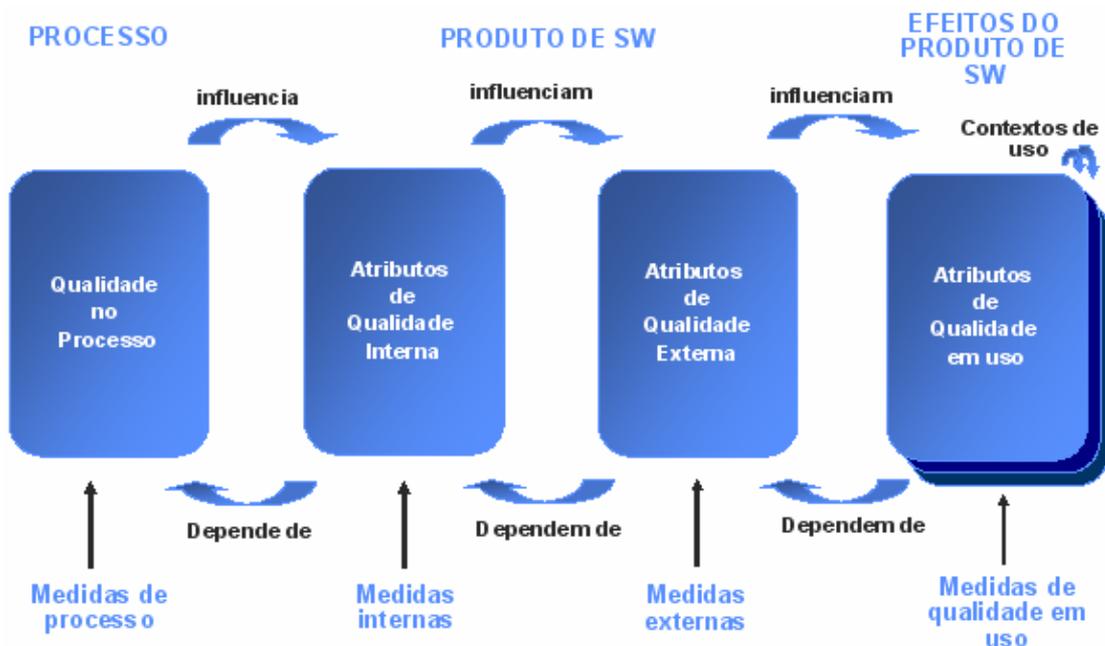


Figura 8. Qualidade e ciclo de vida

A “Figura 8. Qualidade e ciclo de vida” exemplifica a idéia supracitada. A qualidade de um software pode ser vista sob três perspectivas: a qualidade no processo de produção de um software, a qualidade do produto de software em si e o efeito do produto de software que é percebido pelo seu usuário. A avaliação de qualidade no processo de desenvolvimento de um software foge do escopo deste

trabalho, que irá propor um modelo para avaliação de produto, no entanto, a qualidade no processo influencia a qualidade interna do produto, que será o foco de uma das visões elaboradas (visão interna). Ter um produto com boa qualidade interna é o primeiro passo para se obter uma boa qualidade externa de produto, que por sua vez influencia os diversos contextos de uso em que o usuário estará interagindo com o sistema. As normas de produto da ISO, as quais este trabalho se propõe estar em conformidade, atuam no âmbito da qualidade interna, externa e em uso de um produto. Para a qualidade no processo de desenvolvimento de um software diversas metodologias já são sucesso de mercado, como é o caso do Rational Unified Process [RUP] e o Microsoft Solutions Framework [MSF].

A análise também mostrou que o modelo Envision terá que ser relativamente simples de entender e usar, e estar em conformidade com as normas de produto da ISO, seguindo a tendência dos modelos de avaliação de produtos avaliados, todos eles baseados nas normas e convenções da organização.

Finalmente, a análise dos modelos concorrentes mostrou que a cultura de avaliação de produto ainda é muito recente (dos modelos avaliados, o mais experiente tem apenas dez anos) e ainda tem muito a evoluir nos próximos anos. O modelo a ser elaborado neste trabalho deve contribuir para a melhoria e evolução dos modelos de avaliação de produto, e no aumento da qualidade dos produtos de software avaliados segundo o modelo.

## **4. Avaliação de Produto de Software baseada em visões**

---

Esta seção apresenta o conceito de visão, elaborado neste trabalho. Ela também apresenta a conformidade do conceito proposto com as normas internacionais da ISO, assim como introduz o mesmo a partir de atributos de qualidade de software e métricas para medi-lo. Desta maneira, a avaliação de produtos de software baseada em visões torna-se viável.

### *O que é uma visão?*

Segundo [ISO 9126-1, 2002], para avaliar-se a qualidade de um produto de software por algum método quantitativo é necessário um conjunto de características de qualidade que descreva o produto e forme a base para a avaliação. A partir desta declaração, a norma propõe um modelo de qualidade baseado em uma série de características que, por si sós, não são suficientes para a avaliação de um produto de software. A avaliação requer, além das características propostas, métricas para cada uma delas, as quais são exemplificadas em [ISO 9126-2, 2002] (para a qualidade interna) [ISO 9126-3, 2002] (para a qualidade externa) e [ISO 9126-4 2002] (para a qualidade em uso de um produto de software).

Este trabalho propõe um conceito, que foi chamado de visão, que está em um nível de abstração acima das características apresentadas pelo modelo da ISO. Uma determinada visão engloba uma série de características de qualidade e recebe um nome específico. Como “empacotadora” de características, uma visão automaticamente agrega métricas para tais características, o que permite que uma avaliação de produto de software seja realizada a partir de visões.

A definição de uma visão pode ser criada funcionalmente, temporalmente ou não seguir um padrão predefinido. Funcionalmente, agregam-se características de qualidade para uma área funcional específica, como por exemplo, a criação de uma visão para avaliação de softwares agrícolas. Temporalmente, agregam-se características específicas para um determinado momento do ciclo de vida de um produto de software, como na criação de uma visão de avaliação para a fase de testes de um software. Finalmente, a definição pode não seguir um padrão estrito, como na criação de uma visão de avaliação específica para uma empresa de software.

Este trabalho irá apresentar três propostas de visões de avaliação: visão externa, visão interna e visão em uso. Elas agregam características de qualidade específicas para aspectos do produto visíveis ao cliente/usuário final (visão externa), não visíveis ao cliente/usuário final (visão interna) e aspectos visíveis ao cliente/usuário final quando o produto é colocado em ambiente de produção (visão em uso).

Mais uma vez retomando o conceito de agente responsável por “empacotar características”, o modelo de visões de avaliação apresentado irá agrupar módulos de avaliação LAPS em uma determinada visão, módulos estes que na sua definição também agregam algumas características de qualidade em si, o que, por conseguinte mantém a conformidade do modelo de visões proposto, suportado pelas normas ISO. Do mesmo modo, o conceito de visão de avaliação permite a criação de uma visão que englobe uma ou mais visões anteriormente criadas, “visões dentro de visões”.

O conceito apresentado pode ser melhor assimilado através da observação da “Figura 9. Agrupamento em visões”. Nela podemos observar uma macro-visão de avaliação, chamada de VA2, que engloba dentro de si outra visão, VA1. Ambas as visões agregam módulos de avaliação LAPS (ML1 e ML2), que irão atingir determinadas características e subcaracterísticas de qualidade de software. A partir das subcaracterísticas, podemos aplicar métricas para atribuir uma nota discreta à avaliação. É interessante observar que uma outra característica de qualidade, C3, foi abordada dentro da VA1. Esta característica, por sua vez, possui apenas uma subcaracterística de qualidade de software, S1, que pode ser mensurada no âmbito de uma avaliação de produto de software a partir de uma ou mais métricas definidas para esta subcaracterística. Estas métricas estão representadas pelos atributos “A”, vistos na expansão através da lupa. Deste modo conseguimos também atribuir uma nota discreta à avaliação desta subcaracterística de qualidade somente.

Uma vez dadas às notas para características e subcaracterísticas de qualidade, através da aplicação de métricas sobre elas, a nota final de uma avaliação realizada sob o modelo de visões se dará pela composição das notas a partir das subcaracterísticas. Por exemplo, se uma característica de qualidade possui três subcaracterísticas, a sua nota será a composição da nota das três subcaracterísticas, de acordo com seus respectivos pesos. Já a nota de um módulo LAPS, se dará pela

composição das notas das características de qualidade definidas para o módulo. Este processo se repete até chegarmos ao mais alto nível de visão.

Pode-se observar, então, que a “Figura 9. Agrupamento em visões” aborda dois importantes conceitos do modelo proposto neste trabalho: o primeiro é a independência dos módulos de avaliação LAPS e o segundo é o modo como se dá a quantificação das características para a atribuição de uma nota discreta para a avaliação.

Deste modo, o modelo de visões pode ser facilmente adaptado para uma corporação que trabalhe com avaliação de produtos de software, através da agregação dos módulos de avaliação já criados pela empresa em visões, que podem se relacionar com os módulos já existentes em uma relação de um módulo para uma visão, ou não. Conceitualmente, os módulos de avaliação existentes já constituem visões, pelo fato de agregarem características de qualidade de software específicas que são mensuradas a partir de métricas de software.

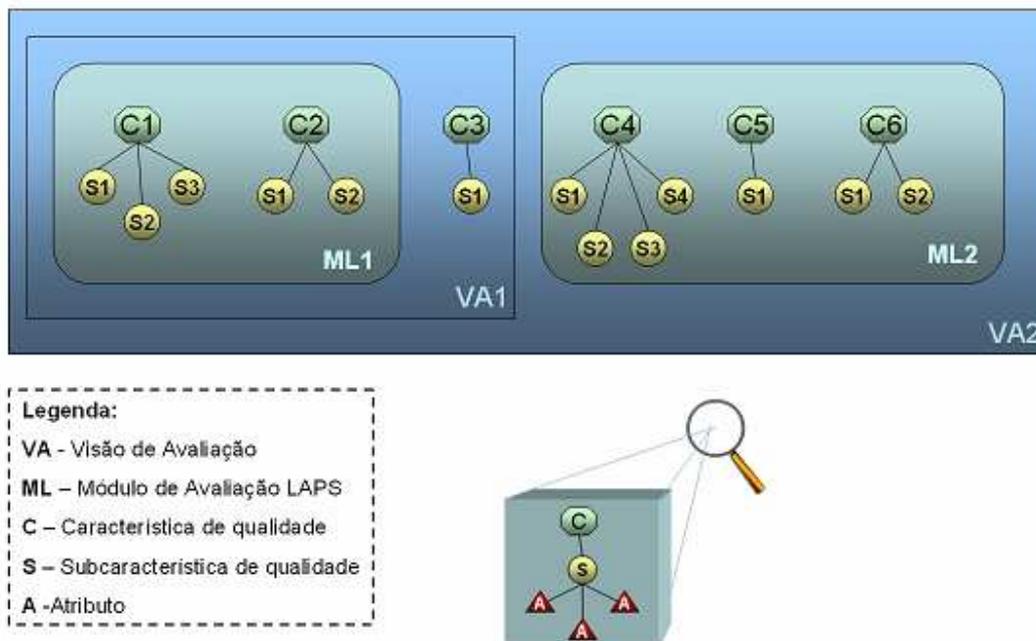


Figura 9. Agrupamento em visões

## *Conformidade normativa*

A elaboração de um modelo de avaliação dividido em visão externa, visão interna e visão em uso está previsto na norma NBR ISO/IEC 14598-1 [ISO 14598-1, 1998], seção 7.2: Identificar tipos de produto(s) a serem avaliados.

Tipo de produto de software é um conceito intrinsecamente relacionado com o estágio de ciclo de vida de um software, e a criação das visões está relacionada com o propósito buscado pela avaliação.

A visão externa é aplicada durante a execução do sistema, o que faz com que a medida de qualidade externa de um software esteja relacionada não só com o sistema em si como também com aspectos extra-software como hardware e ambiente de execução. Uma avaliação externa considera o software como parte de um sistema em operação.

A visão de uso de um software avalia a qualidade percebida para um usuário particular do software ao utilizá-lo. Um usuário é classificado por [Vasconcelos et al., 2003] como todos os seres humanos que irão, de fato, interagir com o sistema. Através da percepção sensorial um usuário percebe, armazena e em seguida processa as informações recebidas. Para que o software tenha qualidade em uso ele deve atender às necessidades e aos requisitos do usuário, no ambiente específico de hardware e software utilizado pelo mesmo. Enquanto a visão externa de um software simula sua execução em um ambiente similar ao de produção, a avaliação de uso é realizada no ambiente real do sistema.

A visão interna é aquela que está preocupada com propriedades internas do software (como a sua especificação ou código fonte) que são mais bem observadas e mais facilmente alteradas enquanto o software ainda está em fase de desenvolvimento. Isso possibilita a avaliação de um subproduto ainda em desenvolvimento, usando atributos internos como modularidade e rastreabilidade que contribuem em tempo de desenvolvimento para atingir os requisitos de qualidade externa e de uso do produto antes que esse entre em ambiente de teste ou produção.

## *Atributos de qualidade de software*

A antiga norma “ISO/NBR 13596 – Tecnologia de Informação – Avaliação de Produtos de Software” foi o primeiro esforço da [ISO] na definição de um modelo para

a avaliação de produtos de software e características de qualidade necessárias a esta avaliação. Esta norma foi substituída por duas séries de normas relacionadas: a 9126, que trata da qualidade de produto [ISO 9126-1, 2002], [ISO 9126-2, 2002], [ISO 9126-3, 2002], [ISO 9126-4, 2002] e a 14598 que cobre a avaliação de produtos de software [ISO 14598-1, 1998], [ISO 14598-2, 1998], [ISO 14598-5, 1998] e [ISO 14598-6, 1998].

A [ISO 9126-1, 2002] descreve um modelo de qualidade dividido em duas partes: 1- Qualidade interna e externa e 2- Qualidade em uso. Para a primeira parte, a norma define seis características de qualidade: funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade. Cada uma destas características é subdividida em subcaracterísticas, que têm seu comportamento manifestado externamente quando o software é utilizado como parte de um sistema computacional, o que torna possível sua medição a partir da definição de uma série de métricas para cada uma destas subcaracterísticas. Já para a segunda parte, qualidade em uso, a norma define quatro características: eficácia, produtividade, segurança e satisfação. No entanto, o modelo para qualidade em uso não é apresentado na norma no nível de subcaracterísticas o que torna um pouco mais difícil a medição de qualidade em uso de um produto. A [ISO 9126-1, 2002] define para o usuário qualidade em uso como o efeito combinado das seis características de qualidade de produto de software interna e externa.

### *Métricas para avaliação*

Uma visão de avaliação é suportada por uma série de características de qualidade de software, que são subdivididas em subcaracterísticas. Para medir quantitativamente se o software possui um bom nível de aceitação para estas características e subcaracterísticas, métricas de software são definidas. O objetivo dessas métricas é gerar medidas discretas e mensuráveis, que serão utilizadas para atribuir uma nota ou um conceito na avaliação de um determinado software. Em seguida, serão apresentadas três diferentes classificações para métricas de software: internas, externas e métricas em uso, para que a avaliação se torne possível. Uma descrição mais detalhada de algumas dessas métricas é apresentada nos anexos deste trabalho.

As métricas internas geralmente são aplicadas ao produto durante sua fase de desenvolvimento, devido a seu poder de prever a qualidade final do produto. A aplicação destas métricas e sua verificação de resultados permitem a tomada de ações preventivas contra problemas de qualidade desde os primórdios do ciclo de desenvolvimento de um software

As métricas externas geralmente são aplicadas ao produto em sua fase de testes [RUP] ou estabilização [MSF]. A medição é realizada, dentro do laboratório de avaliação, em ambiente que simula o real de uso do produto.

As métricas em uso são aplicadas geralmente em ambiente de produção, dentro de um contexto de uso específico. Um contexto de uso seria, por exemplo, a execução por completo de uma atividade ou processo rotineiro. Métricas em uso objetivam verificar se um produto atinge as necessidades do usuário nas características de eficiência, produtividade, segurança e satisfação.

### *A avaliação baseada em visões*

A avaliação baseada em visões sempre se comporta como uma camada superior ao modelo de avaliação utilizado pelo laboratório em questão. Caso o laboratório já utilize um modelo baseado em visões, agregar uma avaliação de visões a este laboratório significa utilizar uma ou mais visões existentes em uma determinada avaliação, ou então criar uma nova visão que agrega, ou não, uma ou mais visões existentes. Para o caso de um laboratório que utilize um modelo de módulos de avaliação, como é o caso do [LAPS], a aplicação do modelo de visões significa utilizar em uma determinada avaliação uma, ou mais, visões que agregam os módulos de avaliação escolhidos para avaliar o produto do cliente. Em ambos os casos, o resultado final da avaliação será dado através da medição de métricas de avaliação, a serem medidas através de características ou subcaracterísticas de qualidade de software, em conformidade com as séries de normas de avaliação de produto da ISO. Uma instância do processo descrito acima foi criada e é explanada na hierarquia que segue, exemplificando um laboratório que já utiliza o modelo de visões:

- Visão para avaliação do produto do cliente
  - Visão “Funcionalidade vs. Satisfação” (Existente no laboratório)
    - Característica de qualidade Satisfação
      - Métrica A para medição de satisfação
      - Métrica B para medição de satisfação
    - Subcaracterística de qualidade Maturidade
      - Métrica para medição de Maturidade
    - Subcaracterística de qualidade Tolerância a Falhas
      - Métrica para medição de Tolerância a Falhas
    - Subcaracterística de qualidade Recuperabilidade
      - Métrica A para medição de Recuperabilidade
      - Métrica B para medição de Recuperabilidade
      - Métrica C para medição de Recuperabilidade
  - Visão dois (Criada especialmente para esta avaliação baseada em um módulo cedido pelo LAPS)
    - Módulo de Avaliação LAPS
      - Característica(s) de qualidade
        - Métrica(s) para medição

O mapeamento das características de qualidade de software definidas pela ISO nos módulos de avaliação LAPS pode ser visto no “Apêndice C – Mapeamento das características de qualidade de software da norma 9126-1 nos módulos de avaliação LAPS”. As subcaracterísticas de qualidade interna e externa discriminadas pela ISO podem ser vistas no “Apêndice B – Subcaracterísticas de qualidade interna e externa” e as métricas para sua medição se encontram em [ISO 9126-2, 2002] e [ISO 9126-3, 2002]. Para a qualidade em uso, que não apresenta qualidade em nível de subcaracterísticas (para qualidade em uso são definidas apenas quatro características: satisfação, eficiência, produtividade e segurança), um resumo das métricas de medição, apresentadas em [ISO 9126-4, 2002], encontra-se no “Apêndice A – métricas para qualidade em uso”.

## *Como iniciar uma avaliação baseada em visões?*

A avaliação baseada em visões segue o mesmo processo de gerenciamento definido pelo “Processo de Planejamento e Gerenciamento” [LAPS]. O processo pode ser visto na Figura 10. Processo de Planejamento e Gerenciamento LAPS.

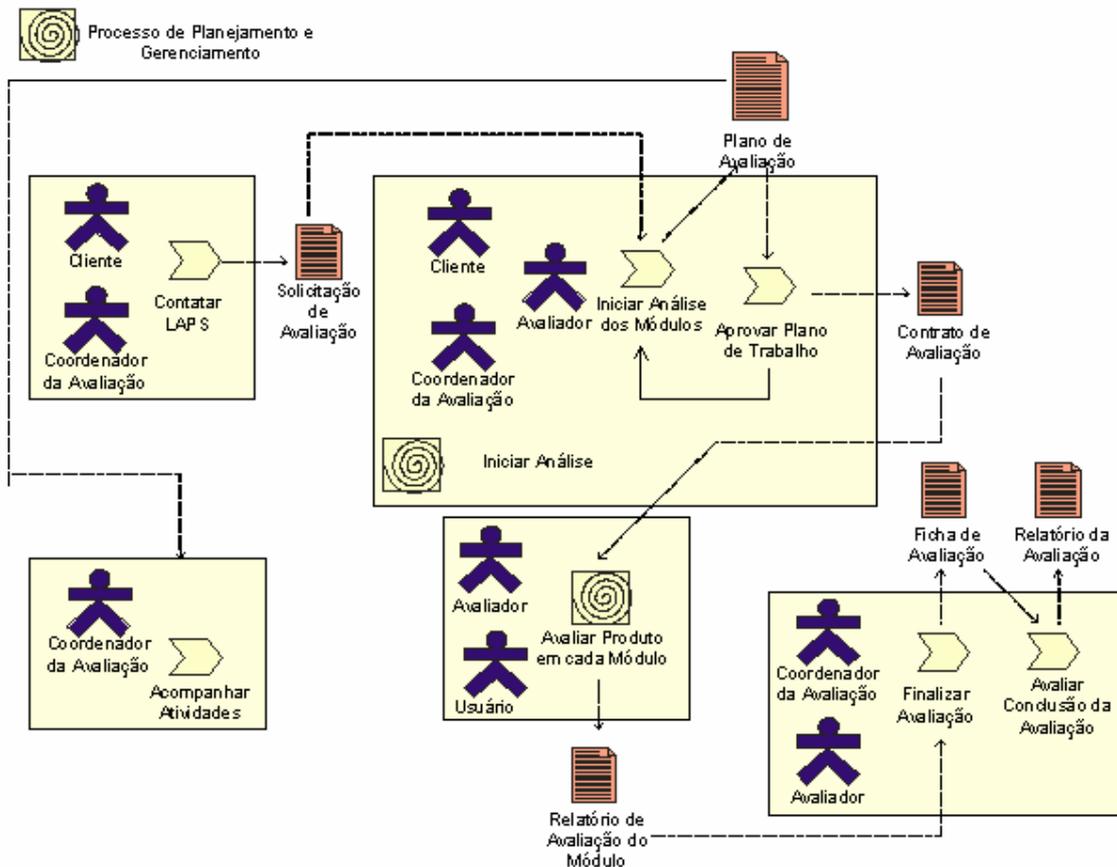
Segundo o processo, uma avaliação inicia quando o cliente contacta o laboratório de avaliação para solicitar uma avaliação. Neste momento o cliente geralmente tem uma visão bastante abstrata do que deseja que seja avaliado em seu software, sabendo apenas que seu software tem que ser avaliado. É papel também do laboratório esclarecer para o cliente como a avaliação trará benefícios para seu software. Após o contato inicial entre cliente e laboratório um artefato é produzido: a solicitação de avaliação. Na solicitação constam informações de contato do cliente e do laboratório, informações iniciais do produto (como ambiente e plataforma de software e hardware para execução, finalidade do produto, infra-estrutura de rede necessária, etc.), principais funcionalidades do produto a ser avaliado e um esboço do que o cliente inicialmente deseja melhorar em seu software (desempenho, performance, qualidade do código, etc.). A solicitação de avaliação funciona como contrato informal entre o cliente e o laboratório, revelando o desejo de realizar a avaliação.

A partir da solicitação de avaliação o cliente irá sentar junto com o laboratório (representado pelo papel do coordenador da avaliação e um ou mais avaliadores designados para a avaliação) para montar a visão de avaliação para este cliente. Esta visão pode ser uma das visões já existentes no laboratório, ou uma visão criada especialmente para este cliente, de acordo com as indicações para criação de visão apresentadas neste trabalho. O processo de montagem de visões funciona de acordo com os seguintes passos:

1. Leitura das necessidades iniciais do produto do cliente apresentadas no documento de Solicitação de Avaliação
2. Apresentação do modelo de qualidade (características e subcaracterísticas de qualidade de software) para o cliente sob responsabilidade do laboratório de avaliação.
3. Apresentação do produto de software a ser avaliado para o laboratório, sob responsabilidade do cliente.

4. Reuniões de *Brainstorm* para identificação das características de qualidade a serem aplicadas na avaliação do software do cliente.
5. Identificação de módulos do laboratório que já satisfaçam um conjunto de características escolhidas para avaliação (opcional).
6. Montagem de uma ou mais visões de avaliação para as características escolhidas. Caso o item 5 da lista seja selecionado, montam-se somente visões para as características restantes. Neste caso, o Envision trata cada módulo de avaliação do laboratório escolhido como uma visão independente.
7. Dar continuidade ao Processo de Planejamento e Gerenciamento [LAPS] a partir do passo “montar plano de avaliação” da atividade “Iniciar Análise dos Módulos”.

Ao término do processo de início de análise teremos um documento de Plano de Avaliação o Contrato de Avaliação, que é assinado pelos representantes legais do laboratório e da empresa contratante.



**Figura 10. Processo de Planejamento e Gerenciamento LAPS**

Cada visão identificada é então avaliada segundo o modelo de avaliação para a visão. Para cada visão de avaliação temos uma série de características de qualidade de software e para cada característica uma ou mais métricas associadas. A pontuação de cada visão se dará pela composição das notas atribuídas à aplicação de cada uma das métricas, como já descrito neste trabalho.

Uma vez avaliada cada visão definida para a avaliação e montados relatórios de avaliação para cada uma destas visões, a avaliação entra na fase de finalização, como definido no Processo de Planejamento e Gerenciamento [LAPS].

## *A avaliação de visões: similaridades e singularidades*

A avaliação de visões, proposta no modelo Envision, foi inspirada na experiência do LAPS [LAPS] e nas normas de qualidade ISO [ISO]. No entanto, é importante deixar claro para o leitor do trabalho em que pontos o modelo proposto se diferencia das atuais abordagens de avaliação de produto.

O modelo Envision propõe a realização de avaliações customizadas para o cliente diferenciando-se, por exemplo, do modelo estático de módulos de avaliação praticado pelo LAPS. Numa avaliação LAPS, quando o cliente requisita a avaliação de um módulo, todas as atividades previstas para o módulo são realizadas e, conseqüentemente, diversas características de qualidade são abordadas. Este fato pode ser observado no “Apêndice C – Mapeamento das características de qualidade de software da norma 9126-1 nos módulos de avaliação LAPS”. No entanto, algumas das características de qualidade abordadas podem não ser de interesse do cliente, e, por conseguinte, pode haver alguma característica de qualidade abordada que lhe interesse e esteja presente em outro módulo de avaliação LAPS. Isso o obrigaria a também adotar aquele módulo para a avaliação, aumentando os custos financeiros e o esforço para sua realização. O modelo envision propõe uma abordagem mais flexível, na qual somente as características de qualidade que interessam para o cliente em uma avaliação sejam abordadas (para tal é criada uma nova visão com tais características, ou usada uma visão já existente). A montagem da visão é realizada nos primórdios da negociação entre cliente e laboratório de avaliação: o cliente expõe suas necessidades (por exemplo, quero melhorar desempenho, quero portabilidade, quero verificar erros no meu código fonte, etc.) e, juntamente com um especialista do laboratório, decidem que características irão ser trabalhadas na avaliação. Deste modo, o custo associado de uma avaliação para o cliente é reduzido e as expectativas acordadas entre cliente e laboratório para a avaliação se tornam explícitas, chegando ao nível de características de qualidade abordadas na avaliação. Isso garante uma maior satisfação ao contratar uma avaliação, a atração de novos clientes com preços reduzidos e avaliações mais “enxutas” abordando somente os aspectos necessários para o contratante.

A [ISO 9126-1, 2002] define um modelo de qualidade baseado em características de qualidade interna e externa, e qualidade em uso. No entanto, a

norma serve apenas como guia, apresentando quais são as características de qualidade do modelo e propondo algumas métricas para a medição de cada uma delas. O Envision também se baseia neste modelo de qualidade para suas avaliações, mas se diferencia por apresentar o processo de criação de uma visão para avaliação, propor uma maneira de atribuir uma nota para a avaliação realizada e permitir sua extensão.

Nas três visões inicialmente propostas pelo Envision, um novo conceito é adicionado para a avaliação de software: o caráter temporal da avaliação de acordo com a fase do produto no ciclo de vida de software (vide “Figura 8. Qualidade e ciclo de vida”). As avaliações tradicionais de produto, realizadas comumente por laboratórios de avaliação, acontecem com o produto de software já pronto. O Envision permite que a avaliação seja realizada em vários momentos da vida de um software (em desenvolvimento, depois de finalizado, quando em fase de testes [RUP], etc.) e, devido à redução de custos e tempo trazida pela possibilidade de requisitar a avaliação de somente uma característica de qualidade, se este for o caso, o cliente pode realizar não só uma, mas diversas avaliações com o mesmo software, em diversos momentos. Um exemplo prático seria a compra de uma avaliação para servir como prova de conceito, projeto de avaliação de software de menor escopo e duração, que é tipicamente realizado com um protótipo do sistema. O cliente, juntamente com laboratório de avaliação, define o problema a ser resolvido, que tipicamente consiste na aplicação de uma ou mais (poucas) características de qualidade de software em um contexto específico, por exemplo, o cliente pode requisitar uma avaliação da tolerância à falhas apresentada por seu sistema.

## 5. Visões de Avaliação

---

Esta seção apresenta, a título de exemplo, a proposta de três diferentes visões de avaliação: a visão de qualidade externa, a visão de qualidade interna, e a visão de qualidade em uso. Nesta seção iremos relacionar as três visões propostas ao nosso estudo de caso, o Laboratório de Avaliação de Produtos de Software [LAPS]. Adicionalmente, iremos realizar um mapeamento bidirecional das visões propostas até o nível de atributos de qualidade de software passando, durante este processo, pelos doze módulos de avaliação do laboratório.

As três visões propostas neste trabalho são visões temporais com relação às etapas pela qual o software passa durante seu ciclo de existência. A primeira visão, a de qualidade interna, trata dos aspectos internos de um produto de software, como seu código fonte, e uma avaliação baseada nesta visão geralmente é aplicada ao produto enquanto o mesmo se encontra ainda em fase de desenvolvimento [RUP] [MSF].

A segunda visão apresentada, qualidade externa, trata de aspectos externos do produto de software, ou seja, aqueles que são visíveis ao cliente ou usuário final do sistema; um exemplo desse tipo de aspecto é o desempenho do sistema, falhas e erros apresentados e até mesmo a caixa do produto. Ela pode ser aplicada na fase de testes [RUP] ou estabilização [MSF] de um produto de software, e também como estratégia de pré-venda, antes de lançar o produto.

A nossa última visão proposta, a de qualidade em uso, trata de avaliar o nível de uso do produto segundo as quatro características de qualidade apresentadas por [ISO 9126-1, 2002]: eficiência, produtividade, segurança e satisfação. Ela deve ser aplicada quando o produto já esta sendo vendido e utilizado na residência do usuário final, visando detectar possíveis melhorias das quatro características recém apresentadas. Esta visão também pode ser aplicada com o produto em ambiente de produção em uma empresa, ou seja, quando o mesmo já faz parte de algum processo operacional da corporação, buscando identificar falhas no produto de software e propor melhorias que irão otimizar os processos operacionais da empresa [MOF].

Essa abordagem pode ser vista na “Figura 11. Visão de Fases do Microsoft Operations Framework” [MOF]. Ela mostra as quatro fases que compõem o Microsoft

Operations Framework: *Changing, Operating, Supporting e Optimizing*. O [MOF] descreve estruturas de times, processos operacionais e aplica as melhores práticas de tecnologia de informação para melhorar a eficiência e qualidade das operações de uma empresa. Utilizar o modelo de visões, proposto neste trabalho, na fase de otimização (*Optimizing*) de uma empresa que aplica [MOF], ajudaria a aumentar a qualidade dos processos e atingir os objetivos dessa fase do framework.

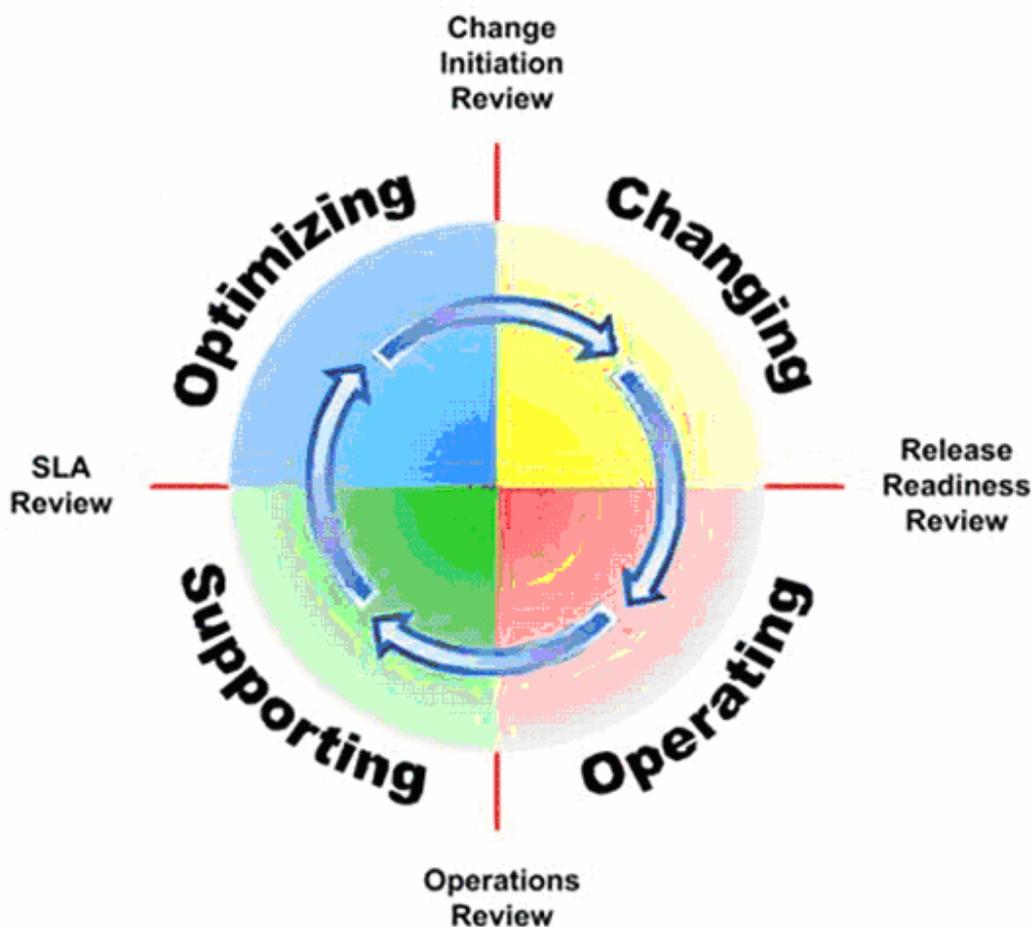


Figura 11. Visão de Fases do Microsoft Operations Framework

### *Visão de qualidade interna*

A visão interna do produto é aquela que verifica o nível de qualidade do produto com relação aos seus aspectos internos, não visíveis ao usuário/cliente final do produto.

Os principais pontos abordados em uma avaliação interna do produto são: a qualidade de seu código fonte, documentação de desenvolvimento e a arquitetura utilizada no desenvolvimento do produto.

### ***Visão de qualidade externa***

A visão externa do produto é aquela que verifica o nível de qualidade do produto com relação aos aspectos visíveis do mesmo para o usuário / cliente final, desde a prateleira (caixa do produto), até a aquisição do mesmo e sua execução. Esses aspectos são importantes na promoção do produto, visto que um produto com boa visibilidade externa, refletida em alta qualidade, tem mais chances de prosperar no mercado.

Os principais pontos abordados na avaliação externa do produto são: sua funcionalidade, a documentação do usuário (na forma de manuais, sites...), a usabilidade, a documentação do produto (incluindo caixa, especificações, requisitos de instalação...) e a posição do produto com relação aos concorrentes.

### ***Visão de qualidade em uso***

A visão de uso do produto é aquela que verifica o nível de uso do produto a partir do momento em que o usuário já o adquiriu e colocou em ambiente de produção. Pode ser requisitada uma avaliação utilizando o pacote de visão de uso do produto para verificar a satisfação e a facilidade de uso de um produto em produção, antecipando as mudanças necessárias e garantindo seu sucesso de mercado.

Os principais pontos abordados na avaliação de uso do produto são a usabilidade de suas funcionalidades, o desempenho do sistema em uso, a segurança na utilização do mesmo e se o produto realmente atende o problema de negócio ao qual ele se propõe.

### ***Mapeamento de visões de avaliação para módulos LAPS***

A proposta desta seção é saber no modelo de três visões proposto: visão de qualidade externa, interna e em uso, que módulos de avaliação LAPS atingimos.

### **Visão de qualidade interna**

#### **Código Fonte**

O módulo LAPS de análise de código fonte é utilizado completamente na visão interna do produto. Verificamos se o código fonte está de acordo com o modelo de análise e projeto do sistema e com o padrão de codificação proposto para o projeto. Depois disso, são colhidas métricas de qualidade de software como grau de aferência e eferência, número de linhas de código não comentadas, entre outras métricas. Todas elas são essenciais na verificação e manutenção da reusabilidade, extensibilidade e qualidade geral do código fonte do produto.

A principal justificativa para a análise do código fonte LAPS estar contida na avaliação de visão interna é, que considerando o caráter temporal da visão proposta, a análise de código fonte se dá durante o desenvolvimento do produto de software, buscando minimizar erros e aumentar a qualidade do produto final.

### **Arquitetura**

O módulo de arquitetura é utilizado totalmente na visão interna do produto. É verificado se a arquitetura atual, apresentada nos documentos de arquitetura, foi realmente implementada no produto e, a partir daí, são montados cenários de testes para ver se esta arquitetura é a mais adequada às necessidades da aplicação.

Este módulo LAPS faz também parte da visão interna proposta, pois a análise e projeto da arquitetura de um software se dão durante seu desenvolvimento, tipicamente em sua fase de planejamento [RUP] [MSF].

### **Portabilidade**

O módulo de portabilidade é utilizado na avaliação interna do produto em todas as suas atividades, com exceção da atividade *Executar Avaliação Black-Box (Caixa preta)*, que é tratada no pacote de visão externa do produto. Nesta avaliação serão montados ambientes para teste de portabilidade, identificados os módulos dependentes de cada ambiente e executados testes que verificam a adoção do sistema em questão com relação a padrões de linguagem de programação e ao grau de portabilidade de cada um de seus componentes.

A visão interna proposta se alinha a este módulo de avaliação LAPS principalmente em sua atividade *Avaliação White-Box (Caixa branca)*. O teste "caixa-branca" procura exercitar todas as partes do código de um sistema. Dessa forma, é necessário que tenhamos o código fonte de todos os programas disponíveis, para podermos controlar o que foi e o que não foi testado. Essa atividade geralmente é

realizada durante o desenvolvimento de um software, o que se encaixa perfeitamente ao caráter temporal da visão interna de avaliação.

### **Documentação do Sistema**

O módulo de documentação do sistema é usado na visão interna do produto somente em sua atividade de verificação de conformidade com a metodologia. Nesta atividade é verificado se os artefatos, aos quais a aplicação propôs desenvolver, foram realmente confeccionados e se eles estão de acordo com o determinado na metodologia de desenvolvimento. Claramente esta atividade é executada durante o desenvolvimento do produto de software, o que a classifica dentro da visão interna proposta.

## **Visão de qualidade externa**

### **Usabilidade**

O módulo de usabilidade é usado no pacote de avaliação da visão externa do produto em todas as suas atividades, exceto a avaliação com usuários que é realizada na visão de uso do produto. Um especialista em usabilidade e outro especialista no domínio da aplicação irão contribuir nesta avaliação tratando de aspectos comuns de usabilidade (como facilidade de uso, disposição de janelas e botões, etc.) e redigir relatórios que sugerem modificações para melhorar a usabilidade do produto.

Usabilidade, conceito fundamental na visão em uso, também pode ser vista de acordo com a visão externa de avaliação, como uma proposta de melhoria dos aspectos de usabilidade durante os testes do produto, procurando identificar pontos de melhoria que poderão se tornar um diferencial de qualidade no momento em que o produto passe a ser vendido ou posto em produção.

### **Funcionalidade**

O módulo de funcionalidade é usado na avaliação de visão externa do produto em todas as suas atividades. São projetados casos de teste para verificar se os resultados funcionais esperados da aplicação realmente são atingidos, ou seja, se a funcionalidade do produto está de acordo com o esperado e definido na especificação funcional. Este módulo LAPS é o que mais se assemelha com o conceito de visão de avaliação externa proposto, por envolver fortemente atividades de testes.

## **Falhas e Recuperação**

O módulo de Falhas e Recuperação é utilizado na avaliação da visão externa do produto em todas as suas atividades exceto a análise ponderada do produto de software, realizada pela visão de uso do produto. Na visão externa será realizada uma análise ponderada da documentação do sistema, que inclui uma análise ponderada do sistema de acordo com os padrões de manipulação de exceções. No fim, será gerado um relatório da avaliação que irá ajudar em melhorias no processo de falhas e recuperação do sistema antes que o mesmo entre em produção, justificando sua participação em uma visão externa de avaliação.

## **Desempenho**

O módulo de desempenho é utilizado na avaliação da visão externa do produto em todas as suas atividades. Inicialmente, é definido o escopo dos testes de desempenho, identificando o que será testado no produto durante a avaliação. Por conseguinte, são definidas métricas de desempenho relativas a tempo de resposta, vazão de informações em recursos de entrada e saída (*Input / Output*), memória, processamento e transmissão. Depois, são definidos os critérios de aceitação ou resultados esperados para as métricas definidas e são realizados os testes de desempenho, finalizando a avaliação com um relatório listando os resultados obtidos.

O desempenho geral de um sistema é um aspecto externo de um produto de software que pode ser observado enquanto o mesmo ainda se encontra em fase de testes, caracterizando sua participação na avaliação de visão externa.

## **Portabilidade**

O módulo de portabilidade é utilizado na avaliação de visão externa do produto em todas as suas atividades, com exceção da atividade de *Executar Avaliação White-Box (Caixa branca)*, que é tratada no pacote de visão interna do produto. Nesta avaliação serão montados ambientes para teste de portabilidade, identificados os módulos dependentes de cada ambiente e executados testes que captarão meta informações e o grau de independência de cada componente. Também é verificado o uso do instalador do produto, pois instaladores geralmente suportam a parametrização do sistema, o que influencia diretamente sua portabilidade.

Para a avaliação de portabilidade utilizamos bastante o teste "caixa-preta", também chamado *teste funcional*, parte de uma visão externa do sistema e testa o mesmo do ponto de vista do usuário, isto é, não considera a estrutura interna ou a

forma de implementação do sistema [Beizer 1995]. Este é o único tipo de teste possível quando não se dispõe do código-fonte, por exemplo.

### **Controle de Acesso e Proteção de Dados**

O módulo de Controle de Acesso e Proteção de dados é utilizado por completo na avaliação do pacote de visão externa do produto. Várias técnicas de invasão e testes são utilizadas para verificar quão seguro está o sistema com relação a sua acessibilidade e proteção de dados.

Geralmente, o controle e a segurança dos dados são realizados para satisfazer um ou mais requisitos elicitados para um projeto de software, e a segurança geral de um sistema é testada antes mesmo dele entrar em produção, ainda na fase de testes, como previsto na visão externa de avaliação.

### **Documentação do Usuário**

O módulo de documentação do usuário é utilizado em todas as suas atividades na avaliação do pacote de visão externa do produto. Toda a documentação, voltada para o usuário, como manuais, ajuda do sistema, ajuda via site e caixa do produto de software serão avaliadas com relação a sua completude e correteude.

### **Documentação do Sistema**

O módulo de documentação do sistema é usado na avaliação de visão externa do produto em sua atividade de Testar Pacote de Software. Nesta atividade aspectos externos do produto como a aparência de sua caixa, descrição dada ao produto, requisitos de instalação e outros serão avaliados. Essa avaliação se dá por meio de uma verificação simples: confere-se se o que o produto “promete” conter, realmente está presente. Esta avaliação é realizada antes de pôr o produto em ambiente de produção, e o sucesso de sua aplicação durante a avaliação de visão externa pode vir a refletir em aspectos de qualidade em uso, como por exemplo, a satisfação do usuário.

### **Competidores**

O módulo de competidores é um módulo particular de avaliação no qual o produto do cliente é comparado com seus competidores (rivais) de mercado. Essa avaliação se encaixa nas visões externa e de uso do produto, pois se pode comparar o produto do cliente com os produtos rivais antes do seu lançamento, estando ele na fase de estabilização (visão externa) ou quando o produto já está no mercado (visão de uso). Em ambos os casos, o principal propósito é executar testes de *benchmark*

nos produtos competidores, para identificar pontos fortes e fracos de ambos, dando suporte a melhorias no produto do cliente para futuros *releases*.

## **Visão de qualidade em uso**

### **Usabilidade**

O módulo de usabilidade é usado no pacote de avaliação da visão de uso do produto na sua atividade Avaliar com Usuários. Serão realizados testes com as classes de usuários que irão utilizar o sistema e identificadas oportunidades de melhoria e pontos de sucesso do produto em uso. A aplicação do módulo de usabilidade durante a visão de uso do produto vai permitir atingir, em diferentes escalas, as quatro características de qualidade em uso: segurança, satisfação, eficiência e produtividade.

### **Funcionalidade**

O módulo de funcionalidade é usado na avaliação de uso do produto em todas as suas atividades. São projetados casos de teste para verificar se os resultados funcionais esperados da aplicação realmente são atingidos, ou seja, se a funcionalidade do produto está de acordo com o esperado e definido na especificação funcional. Aplicar o módulo de funcionalidade durante a visão de uso do produto objetiva, principalmente, identificar pontos de melhoria não identificados na fase de desenvolvimento e testes, para correção no lançamento de futuros *releases*.

### **Falhas e Recuperação**

O módulo de Falhas e Recuperação é utilizado na avaliação da visão de uso do produto em todas as suas atividades exceto a análise ponderada da documentação do sistema, realizada pela visão externa do produto. Na visão de uso será realizada uma análise ponderada do produto de software que inclui testes de tratamento de exceções no sistema e testes de stress físico que dizem respeito não ao modo como o produto foi implementado, mas a fatores relevantes a elementos associados ao funcionamento correto do sistema e que podem levar à falhas graves do mesmo durante sua utilização.

No fim será gerado um relatório da avaliação que identificará falhas a serem corrigidas em futuras versões do software.

## **Desempenho**

O módulo de desempenho é utilizado na avaliação da visão de uso do produto em todas as suas atividades. Inicialmente é definido o escopo dos testes de desempenho, identificando o que será testado no produto durante a avaliação. Em seguida são definidas métricas de desempenho relativas a tempo de resposta, vazão de informações, recursos de entrada e saída, memória, processamento e transmissão. Depois são definidos os critérios de aceitação ou resultados esperados para as métricas definidas e são realizados os testes de desempenho, finalizando a avaliação com um relatório listando os resultados obtidos.

Este módulo fornece uma visão da eficiência do produto do ponto de vista do usuário, sendo útil para informar o comportamento do sistema, tanto em relação a tempo quanto em relação a recursos físicos, em um contexto de uso específico. São analisados, por exemplo, o tempo de resposta das operações do sistema e a utilização de recursos de memória e processamento durante estas operações.

## **Controle de Acesso e Proteção de Dados**

O módulo de Controle de Acesso e Proteção de dados é utilizado por completo na avaliação do pacote de visão de uso do produto. Várias técnicas de invasão e testes são utilizadas para verificar quão seguro está o sistema com relação a sua acessibilidade e proteção de dados.

A Segurança de um sistema é um conceito volátil temporalmente, pois a cada dia surgem novas técnicas de invasão e proteção. Outros aspectos de segurança, não tratados durante a fase de testes de um produto, podem vir a aparecer durante o uso do sistema em ambiente de produção, junto ao seu cliente final, o que justifica colocar este módulo de avaliação também na visão em uso do sistema.

## **Competidores**

O módulo de competidores é um módulo particular de avaliação, no qual o produto do cliente é comparado com seus competidores (rivais) de mercado. Essa avaliação se encaixa nas visões externa e de uso do produto, pois se pode comparar o produto do cliente com os produtos rivais antes do lançamento do produto do cliente, estando esse na fase de estabilização (visão externa) ou quando o produto já está no mercado (visão de uso). Em ambos os casos o principal propósito é executar testes de *benchmark* nos produtos competidores, para identificar pontos fortes e

fracos de ambos, dando suporte a melhorias no produto do cliente para futuros releases.

### Especialistas

O módulo de especialistas é realizado, em sua totalidade, na visão de uso do produto. Um especialista no negócio modelado pela aplicação irá avaliar se o produto realmente atende às necessidades da área, dentro do que o software se propôs a implementar. A análise do especialista agrega bastante valor ao software do cliente, pois dá ao produto um aval de aprovação por quem mais entende da área de negócio do sistema.

A “Tabela 13. Mapeamento Visões vs. Módulos LAPS” apresenta o nível de utilização de cada módulo de avaliação do LAPS em cada uma das visões propostas neste trabalho. O uso de um processo LAPS em uma visão é considerado total se todas as atividades e passos previstos para o módulo são utilizados, ao avaliar um produto, seguindo a visão proposta. Se pelo menos uma atividade ou um passo de um módulo LAPS não estiver contemplado na visão proposta, a sua utilização na visão é considerada parcial. Os campos em branco na tabela significam que o módulo não se encaixa na visão proposta.

Tabela 13. Mapeamento Visões vs. Módulos LAPS

Processos LAPS	Visão Interna	Visão Externa	Visão em Uso
Código Fonte	Total		
Usabilidade		Parcial	Parcial
Funcionalidade		Total	Total
Falhas e Recuperação		Parcial	Parcial
Desempenho		Total	Total
Arquitetura	Total		
Portabilidade	Parcial	Parcial	
Controle de Acesso e Proteção de Dados		Total	Total
Documentação do Usuário		Total	
Documentação do Sistema	Parcial	Parcial	
Competidores		Total	Total
Especialista			Total

### *Mapeamento de visões de avaliação para atributos de qualidade*

Utilizando o modelo de visão proposto, é importante saber que atributos de qualidade de software atingimos, quando utilizamos uma determinada visão. Esta seção irá apresentar o mapeamento das três visões de avaliação propostas nos atributos definidos pela ISO em [ISO 9126-1, 2002].

Uma vez criadas novas visões, o autor sugere que seja realizado um mapeamento similar ao da “Tabela 14. Mapeamento Visões vs. Atributos de Qualidade”. Para realizar tal mapeamento, se faz necessária a presença de um especialista em qualidade de software, ou alguém bastante familiarizado com a série de normas ISO 9126. Este mapeamento permite uma identificação clara de que atributos de qualidade estamos atingindo ao utilizar uma visão. No caso da “Tabela 14. Mapeamento Visões vs. Atributos de Qualidade”, que considera as visões elaboradas para este trabalho, podemos notar algumas particularidades.

A primeira delas é a presença dos quatro atributos de qualidade em uso: satisfação, produtividade, eficiência e segurança. Estes atributos são cobertos somente na visão de avaliação em uso, pois, devido ao caráter temporal escolhido para a definição das três visões, a visão em uso é a única que é realizada quando o produto já está em contato direto com o usuário final. Outra particularidade, facilmente percebida, é o não mapeamento das visões propostas na subcaracterística “conformidade” de cada característica de qualidade interna e externa. Conformidade é “a capacidade de aderir a padrões, convenções, leis e prescrições similares” (vide Apêndice B – Subcaracterísticas de qualidade interna e externa) relativa à característica em questão, e a razão da falta do seu mapeamento reside no fato de que as três visões propostas foram montadas em direto paralelismo com o modelo de avaliação do LAPS, cujos módulos ainda não estão avaliando esta subcaracterística.

Um aprimoramento dos módulos do LAPS deve vir seguido da atualização das tabelas presentes neste trabalho.

**Tabela 14. Mapeamento Visões vs. Atributos de Qualidade**

	Interna	Externa	Uso
Adequação		x	x
Acurácia		x	x
Interoperabilidade	x	x	x
Segurança de Acesso		x	x
Conformidade			
Maturidade	x	x	x
Tolerância a Falhas		x	x
Recuperabilidade		x	x
Conformidade		x	x
Inteligibilidade	x	x	x
Apreensibilidade		x	x
Operacionalidade		x	x
Atratividade		x	x
Conformidade			
Comportamento com relação ao tempo	x	x	x
Comportamento com relação aos recursos	x	x	x
Conformidade			
Analisabilidade	x		
Modificabilidade	x	x	
Estabilidade	x		x
Testabilidade		x	
Conformidade			
Adaptabilidade	x	x	
Capacidade Instalação	x	x	
Coexistência	x	x	
Capacidade Substituição	x	x	
Conformidade			
Satisfação			x
Produtividade			x
Eficiência			x
Segurança			x

Características de Qualidade	
Portabilidade	
Manutenibilidade	
Eficiência	
Usabilidade	
Confiabilidade	
Funcionalidade	

\*As quatro características definidas pela [ISO] para qualidade em uso estão em roxo

## 6. Conclusões e Trabalhos Futuros

---

O trabalho apresentou o conceito de visões de avaliação e também um modelo de avaliação de produtos de software, baseado neste conceito, chamado aqui de Envision. Além disso, foi mostrado um mapeamento do modelo proposto nos módulos do Laboratório de Avaliação de Produtos de Software, LAPS, o que serviu como pontapé inicial para que outros laboratórios, existentes ou não, venham a se adaptar ao Envision.

Foi realizada, de início, uma apresentação do LAPS, seu histórico, estrutura organizacional e módulos de avaliação. Em seguida, uma análise de trabalhos similares ao proposto foi realizada, visando identificar oportunidades de melhorias que poderiam ser incorporadas ao modelo proposto neste trabalho que, naquele momento, ainda seria montado. Dando seqüência ao desenvolvimento do trabalho foi proposto o conceito de visão e sua conformidade com as normas da série ISO 9126 e ISO 14598. Foram também apresentados alguns atributos e métricas para a avaliação baseada em visões e como utilizá-las na avaliação. Finalmente, foi especificado como realizar uma avaliação baseada em visões, e propostas três visões de avaliação (interna, externa e em uso) e seu mapeamento para os já existentes módulos de avaliação do LAPS.

Durante todo o decorrer do trabalho houve uma grande preocupação em desenvolvê-lo em conformidade com a série de normas ISO 9126 e ISO 14598. Esta preocupação se justifica no fato de que a busca por uma padronização de processos dentro de uma empresa é uma tendência mundial, e a série de normas apresentada é referência internacional no campo de qualidade e avaliação de produtos de software.

### *Principais Contribuições*

A contribuição majoritária do trabalho foi a própria definição do Envision, modelo de avaliação de produtos de software baseado em visões. Este trabalho fornece o arcabouço necessário para a criação de novas visões de avaliação de produtos de software e a aplicação do modelo proposto em laboratórios de avaliação ou empresas que possuam uma área dedicada à qualidade de seus produtos.

O trabalho também contribuiu para disseminar ainda mais a cultura de Qualidade de Software dentro do Centro de Informática da Universidade Federal de Pernambuco e em médio prazo, espera-se também, no ecossistema pernambucano de empresas de software.

Um dos grandes ganhadores do trabalho foi o Laboratório de Avaliação de Produtos de Software, LAPS, pois o trabalho apresenta a modelagem e o mapeamento dos atuais módulos do laboratório no modelo proposto.

### ***Dificuldades Encontradas***

Durante a escrita do trabalho, algumas dificuldades foram encontradas. A maior parte delas não teve impacto no trabalho e foram facilmente superadas. O tempo curto para a leitura do extenso material bibliográfico de qualidade de software, voltada para avaliação de produtos, e posterior desenvolvimento deste trabalho foi uma das dificuldades encontradas, superada com a escolha criteriosa do material a ser estudado dentro do arsenal de artigos, livros e sites encontrados.

No entanto, existiram barreiras que não conseguiram ser transpostas. A principal delas foi o acesso por completo a outros modelos de avaliação de produtos de software, na maioria das vezes proprietários (como por exemplo, o modelo de avaliação de produtos de software do CenPRA, o MEDE-PROS), o que acarretou uma análise resumida dos competidores do modelo proposto. Fica como sugestão para trabalhos futuros um estudo mais detalhado destes modelos e a viabilização do acesso à suas fontes de informação.

### ***Trabalhos Futuros***

Este trabalho abre precedentes para a elaboração de diversas outras visões de avaliação, que podem ser futuramente apresentadas em outros trabalhos de graduação, teses de mestrado e doutorado e *papers* científicos. Não somente no âmbito acadêmico podem (e devem) ficar a elaboração dessas novas visões. O objetivo de elaborá-las é sua incorporação no processo de avaliação de produtos de software de uma empresa, agregando, assim, valor aos softwares produzidos.

Com o objetivo de sempre aperfeiçoar o Envision e as visões apresentadas neste trabalho, o autor propõe a execução prática de uma avaliação de visão em uso. Essa visão pode ser estendida, por exemplo, com uma série de processos e passos

(feitos em UML, SPEM ou outra técnica de modelagem de processos), que vão desde questionários de satisfação até testes de usabilidade, falhas e recuperação de sistemas e segurança.

Também é proposta a prática de uma avaliação de visão interna, temporal, durante o desenvolvimento de um software ainda não finalizado. A série de processos e passos descritos anteriormente, neste caso, iria desde o código fonte do produto até sua documentação de desenvolvimento (modelo de análise e projeto, documento de requisitos etc.).

Outra proposta de trabalho futuro é a aplicação prática de uma avaliação de visão externa, assim como a elaboração de uma série de processos e passos (feitos em UML, SPEM ou outra técnica de modelagem de processos), que vão desde a caixa do produto até sua funcionalidade externa.

Fica também como sugestão para trabalhos futuros, um estudo mais detalhado dos modelos concorrentes do Envision, e a extensão desse estudo para outros modelos, assim como a viabilização do acesso às suas fontes de informação.

Finalmente, o Processo de Planejamento e Gerenciamento [LAPS] pode ser modificado, tirando as especificidades existentes no documento (ainda dependente do modelo de módulos de avaliação LAPS) e incorporando novos conceitos, relativos ao modelo de avaliação baseado em visões.

### *Considerações Finais*

É sempre importante lembrar que a implantação de um modelo de qualidade em uma empresa deve ocorrer junto a uma mudança cultural que, algumas vezes, significa a quebra de alguns paradigmas empresariais.

“Líderes de negócio às vezes vêem qualidade de software como luxo, algo que pode ser sacrificado, se necessário, para uma maior funcionalidade, rápido desenvolvimento ou redução de custos. No entanto, na prática, organizações que possuem compromisso com a qualidade podem atualmente desenvolver com velocidade, com custos reduzidos e adicionando novas funcionalidades facilmente. Organizações que desenvolvem software de baixa qualidade, para uso interno ou venda, estão sempre olhando pra trás, gastando tempo e dinheiro consertando defeitos em produtos teoricamente finalizados. Em

contraste, uma organização que produz e garante a qualidade desde o início pode olhar pra frente e inovar, gastando tempo e dinheiro na busca de novas oportunidades. Produtos de qualidade são também um poderoso diferenciador em um mercado onde software de alta qualidade é a exceção, em vez da regra” [Bessin 2004].

Este trabalho apresentou a proposta de um modelo de avaliação de produto a ser implantado em uma empresa de desenvolvimento de softwares ou laboratório de avaliação de produto de software. A implantação e a execução do modelo, no dia-a-dia empresarial, envolvem custos financeiros e tempo em agenda para a execução das atividades de avaliação. De nada adiantará a aplicação deste ou de outro modelo de qualidade, em uma empresa ou laboratório de avaliação, uma vez que não seja entendida, pelos executivos e funcionários de uma empresa de software, a importância de produzir produtos de qualidade como estratégia de crescimento empresarial.

## Referências

---

- [ISO 9126-1, 2002]** ISO 9126-1. *Software engineering – Product Quality – Part 1 Quality model* 2002.
- [ISO 9126-2, 2002]** ISO 9126-2. *Software engineering – Product Quality – Part 2 External metrics* 2002.
- [ISO 9126-3, 2002]** ISO 9126-3. *Software engineering – Product Quality – Part 3 Internal metrics* 2002.
- [ISO 9126-4, 2002]** ISO 9126-4. *Software engineering – Product Quality – Part 4 Quality in use metrics* 2002.
- [ISO 14598-1, 1998]** ISO/IEC 14598-1. *Information Technology – Evaluation of Software Products – Part 1 General Guide*, 1998.
- [ISO 14598-2, 1998]** ISO/IEC 14598-2. *Information Technology – Evaluation of Software Products – Part 2 Planning and management*, 1998.
- [ISO 14598-5, 1998]** ISO/IEC 14598-5. *Information Technology – Evaluation of Software Products – Part 5 Process for evaluators*, 1998.
- [ISO 14598-6, 1998]** ISO/IEC 14598-6. *Information Technology – Evaluation of Software Products – Part 6 Documentation of evaluation modules*, 1998.
- [ISO 12207, 1998]** NBR ISO/IEC 12207 - *Tecnologia da Informação - Processos de Ciclo de Vida de Software*, 1998.
- [ISO 12119, 1998]** ISO/IEC 12119 - *Information technology – Software packages – Quality requirements and testing*, 1998.
- [ISO 2382-1, 1993]** ISO/IEC 2382-1 - *Information technology - Vocabulary -- Part 1: Fundamental terms*, 1993.
- [Bevan 1999]** Bevan, N. *Quality in Use: Meeting User Needs for Quality*. Journal of System and Software, 1999 (in press)

- [Sampaio & Perrelli 2004]** Sampaio, Vânia e Perrelli, Hermano. *LAPS: Um Modelo Estruturado para Avaliação de Produtos de Software*.
- [Vasconcelos et al., 2003]** Vasconcelos, C.; Garcia, F.; Turnell, M. *Integrando Usabilidade e Engenharia de Software: um modelo para o desenvolvimento de sistemas centrado no usuário*. WIHC-ES 2003: Integrating Human-Computer Interaction and Software Engineering Models and Processes, Rio de Janeiro – RJ, Brazil, August 2003.
- [Anjos & Perrelli 2004]** Anjos, Luis André Mendonça dos e Perrelli, Hermano. *Um Modelo para Avaliação de Produtos de Software*.
- [Colombo & Guerra 2002]** Colombo, Regina e Guerra, Ana. *The Evaluation Method for Software Products*. ICSSEA '2002 - International Conference "Software & Systems Engineering and their Applications" Paris – França
- [Square 2004]** Scarlet, Danilo. *SQUARE - Software product Quality Requirements and Evaluation*. Palestra ministrada no I Fórum Brasileiro de Qualidade de Software, Porto Digital – Recife – Pernambuco.
- [MCT 2002]** *Qualidade e Produtividade no Setor de Software Brasileiro N.4*. Ministério da Ciência e Tecnologia - Secretaria de Política de Informática, Brasília, 2002. ISSN 1518-112X.
- [Bessin 2004]** Bessin, Geoffrey. *The business value of software quality*. Whitepaper da IBM, Publicado em 15 de Dezembro de 2004,  
<http://www-128.ibm.com/developerworks/rational/library/dec04/bessin/>, último acesso em 11/jun/2005.
- [Beizer 1995]** - Beizer, B. *Black-Box Testing*. Wiley, USA, 1995.
- [Guia CenPRA]**. *Guia de Avaliação da qualidade de Produto de Software*, Divisão da Qualidade de Software –DQS – CenPRA, [www.cenpra.gov.br](http://www.cenpra.gov.br), último acesso em 11/jun/2005
- [LAPS]** Laboratório de Avaliação de Produtos de Software, [www.cin.ufpe.br/laps](http://www.cin.ufpe.br/laps), último acesso em 20/jul/2005.
- [Porto]** Porto Digital, <http://www.portodigital.com.br>, último acesso em 20/jul/2005..

**[ISO]** ISO – International Organization for Standardization, <http://www.iso.org>, último acesso em 20/jul/2005..

**[RUP]** RUP – Rational Unified Process, <http://www-306.ibm.com/software/awdtools/rup/>, último acesso em 20/jul/2005..

**[MSF]** MSF – Microsoft Solutions Framework  
<http://msdn.microsoft.com/vstudio/enterprise/msf/>, último acesso em 20/jul/2005..

**[MOF]** MSF – Microsoft Operations Framework  
<http://www.microsoft.com/technet/itsolutions/cits/mo/mof/default.mspx>, último acesso em 20/jul/2005.

**[Webster]** Merriam Webster Online, <http://www.webster.com/>, último acesso em 3/ago/2005.

**[WikiPedia]** WikiPedia, the Free Encyclopedia, <http://www.wikipedia.org/>, último acesso em 3/ago/2005.

## **Apêndice A – métricas para qualidade em uso**

---

As métricas de qualidade apresentadas neste apêndice refletem um resumo das métricas propostas para avaliação de qualidade em uso da [ISO 9126-4, 2001]. A justificativa para elas estarem presentes neste trabalho reside no fato de que métricas de qualidade em uso são as mais difíceis de mensurar, já que a série de normas ISO 9126 não apresenta para elas um detalhamento ao nível de subcaracterísticas, como é feito com as normas de qualidade externa [ISO 9126-2 2001] e interna [ISO 9126-3 2001].

### ***Métricas de Eficácia***

As métricas de Eficácia medem o quanto as tarefas realizadas pelos usuários atingem objetivos específicos de acurácia e completude, em um contexto de uso específico. Estas métricas não levam em conta como, mas somente a extensão de quanto elas atingem. A Tabela 15. Métricas de Eficácia apresenta algumas métricas definidas para a característica Eficácia.

**Tabela 15. Métricas de Eficácia**

<b>Métrica</b>	<b>Propósito</b>	<b>Método sugerido</b>	<b>Forma de Medição</b>	<b>Interpretação da Medição</b>	<b>Entrada sugerida</b>
<i>Eficácia da Tarefa</i>	Que proporção dos objetivos das tarefas são atingidos corretamente	Teste com o usuário	$M1 =  1 - \sum A_i $ A <sub>i</sub> é o valor proporcional de cada componente errado ou incorreto na saída da tarefa	$0 \leq M1 \leq 1$ Quanto mais próximo de 1 melhor	Relato do teste com o usuário  Gravação do teste com o usuário
<i>Compleitude da Tarefa</i>	Que proporção das tarefas são completadas	Teste com o usuário	$X = A/B$ A é o num das tarefas completadas B é o num total de tarefas	$0 \leq X \leq 1$ Quanto mais próximo de 1 melhor	Relato do teste com o usuário  Gravação do teste com o usuário
<i>Frequência de erros</i>	Qual é a frequência de erros	Teste com o usuário	$X = A/T$ A é o num de erros cometidos pelo usuário T é o tempo ou número de tarefas	$0 \leq X$ Quanto mais próximo de 0 melhor	Relato do teste com o usuário  Gravação do teste com o usuário

### ***Métricas de Produtividade***

As métricas de Produtividade medem o consumo de recursos em relação à eficácia atingida em determinado contexto de uso. O recurso mais comum de medição é o tempo levado para completar a tarefa, mas outros podem ser levados em conta, como o esforço dos usuários, custos materiais e financeiros do uso. A Tabela 16. Métricas de Produtividade apresenta algumas métricas definidas para a característica Produtividade.

**Tabela 16. Métricas de Produtividade**

<b>Métrica</b>	<b>Propósito</b>	<b>Método sugerido</b>	<b>Forma de Medição</b>	<b>Interpretação da Medição</b>	<b>Entrada sugerida</b>
<i>Tempo da Tarefa</i>	Quanto tempo leva para completar uma tarefa	Teste com o usuário	$X=Tt$ Tt é tempo da tarefa	$0 \leq X$ Quanto menor, melhor	Relato do teste com o usuário  Gravação do teste com o usuário
<i>Eficiência da Tarefa</i>	O quão eficiente são os usuários	Teste com o usuário	$X=M1/T$ M1 é a eficácia da tarefa Tt é tempo da tarefa	$0 \leq X$ Quanto maior, melhor	Relato do teste com o usuário  Gravação do teste com o usuário
<i>Produtividade Econômica</i>	Qual a relação custo-eficácia do usuário	Teste com o usuário	$X=M1/C$ M1 é a eficácia da tarefa C é o custo total da tarefa	$0 \leq X$ Quanto maior, melhor	Relato do teste com o usuário  Gravação do teste com o usuário
<i>Proporção Produtiva</i>	Qual proporção do tempo o usuário gasta realizando ações produtivas	Teste com o usuário	$X=Ta/Tb$ Ta é o tempo produtivo (Tb – tempo gasto em atividades paralelas) Tb é o tempo da tarefa	$0 \leq X \leq 1$ Quanto mais próximo de 1 melhor	Relato do teste com o usuário  Gravação do teste com o usuário

Continuação da “Tabela 16. Métricas de Produtividade”

Métrica	Propósito	Método sugerido	Forma de Medição	Interpretação da Medição	Entrada sugerida
<i>Eficiência de Uso Relativa</i>	O quão eficiente é um usuário comparado a um expert	Teste com o usuário	X=A/B A é a eficiência do usuário comum B é a eficiência do expert	$0 \leq X \leq 1$ Quanto mais próximo de 1 melhor	Relato do teste com o usuário  Gravação do teste com o usuário

### *Métricas de Segurança*

As métricas de Segurança medem o risco de prejuízos a pessoas, negócios, software, propriedade ou ambiente, em um determinado contexto de uso. Inclui a saúde e a segurança dos usuários e daqueles afetados pelo uso, assim como conseqüências físicas e econômicas não-previstas. A Tabela 17. Métricas de Segurança apresenta algumas métricas definidas para a característica Segurança.

**Tabela 17. Métricas de Segurança**

<b>Métrica</b>	<b>Propósito</b>	<b>Método sugerido</b>	<b>Forma de Medição</b>	<b>Interpretação da Medição</b>	<b>Entrada sugerida</b>
<i>Segurança e Saúde dos Usuários</i>	Qual a incidência de problemas de saúde entre os usuários do produto	Estatísticas de uso	$X=1-A/B$ A é o nº de usuários que reportaram problemas B número total de usuários	$0 \leq X \leq 1$ Quanto mais próximo de 1 melhor	Gravação do teste com o usuário
<i>Segurança das Pessoas Afetadas pelo Uso do Sistema</i>	Qual a incidência de riscos em pessoas afetadas pelo uso do sistema	Estatísticas de uso	$X=1-A/B$ A é o nº de pessoas submetidas a riscos B número total de usuários potencialmente afetados pelo sistema	$0 \leq X \leq 1$ Quanto mais próximo de 1 melhor	Gravação do teste com o usuário
<i>Danos Econômicos</i>	Qual a incidência de danos econômicos	Estatísticas de uso	$X=1-A/B$ A é o nº de ocorrências de danos econômicos B número total situações de uso do sistema	$0 \leq X \leq 1$ Quanto mais próximo de 1 melhor	Gravação do teste com o usuário
<i>Danos ao Software</i>	Qual a incidência de danos à integridade do software	Estatísticas de uso	$X=1-A/B$ A é o nº de ocorrências de danos à integridade do software B número total situações de uso do sistema	$0 \leq X \leq 1$ Quanto mais próximo de 1 melhor	Gravação do teste com o usuário

### ***Métricas de Satisfação***

As métricas de Satisfação medem as atitudes dos usuários enquanto utilizam o sistema, em determinado contexto de uso. A Tabela 18. Métricas de Satisfação apresenta algumas métricas definidas para a característica Satisfação.

Tabela 18. Métricas de Satisfação

Métrica	Propósito	Método sugerido	Forma de Medição	Interpretação da Medição	Entrada sugerida
<i>Escala de Satisfação</i>	O quão satisfeito está o usuário	Teste com o usuário	$X=A/B$ A é um questionário produzindo escalas psicossométricas B é a média da população	$0 < X$ Quanto maior, melhor	Relato do teste com o usuário  Gravação do teste com o usuário
<i>Questionário de Satisfação</i>	O quão satisfeito está o usuário com funcionalidades específicas do software	Teste com o usuário	$X = \sum(A_i)/N$ $A_i$ é a resposta a uma questão N é o número total de respostas	Comparação com valores prévios, ou com a média	Relato do teste com o usuário  Gravação do teste com o usuário
<i>Uso Arbitrário</i>	Qual proporção de potenciais usuários escolhem utilizar o sistema	Observação de uso	$X=A/B$ A é o número de vezes que funções/aplicações/sistemas de software específicos são utilizados B é o número de vezes que eles deveriam ser utilizados	$0 \leq X \leq 1$ Quanto mais próximo de 1 melhor	Relato do teste com o usuário  Gravação do teste com o usuário

## Apêndice B – Subcaracterísticas de qualidade interna e externa

Este apêndice apresenta a Tabela 19. Subcaracterísticas de qualidade interna e externa. Ela traz as características de qualidade interna e externa definidas pela série ISO 9126 (verticalmente à esquerda) e a definição de cada uma de suas subcaracterísticas de qualidade.

Tabela 19. Subcaracterísticas de qualidade interna e externa

funcionalidade	<b>Adequação</b>	Capacidade de fornecer um conjunto apropriado de funções para tarefas específicas e objetivos do usuário
	<b>Acurácia</b>	Capacidade de fornecer o resultado com o grau de precisão desejado
	<b>Interoperabilidade</b>	Capacidade de interagir com um ou mais sistemas
	<b>Segurança de Acesso</b>	Capacidade de proteger dados e informações de pessoas ou sistemas não autorizados
	<b>Conformidade</b>	Capacidade de aderir a padrões, convenções, leis e prescrições similares, relativas à funcionalidade
confiabilidade	<b>Maturidade</b>	Capacidade de evitar defeitos no software
	<b>Tolerância a Falhas</b>	Capacidade de manter um nível de desempenho estabelecido em caso de defeito no software
	<b>Recuperabilidade</b>	Capacidade de recuperar dados diretamente afetados no caso de falhas
	<b>Conformidade</b>	Capacidade de aderir a padrões, convenções, leis e prescrições similares, relativas à confiabilidade
usabilidade	<b>Inteligibilidade</b>	Capacidade do produto de fazer o usuário entender se o software é adequado, e como ele pode ser usado para tarefas particulares
	<b>Apreensibilidade</b>	Capacidade que o produto deve ter de fazer o usuário entendê-lo
	<b>Operacionalidade</b>	Capacidade que o produto deve ter para que o usuário possa aprendê-lo e controlá-lo
	<b>Atratividade</b>	Capacidade do produto em ser atraente para o usuário
	<b>Conformidade</b>	Capacidade de aderir a padrões, convenções, leis e prescrições similares, relativas à usabilidade

Continuação da “Tabela 19 Subcaracterísticas de qualidade interna e externa”

eficiência	<b>Comportamento em relação ao tempo</b>	Capacidade de fornecer tempos de resposta e processamento adequados, bem como taxas de transferência
	<b>Comportamento em relação aos recursos</b>	Capacidade de usar quantidade e tipos de recursos adequados
	<b>Conformidade</b>	Capacidade de aderir a padrões e convenções relativas à eficiência
manutenibilidade	<b>Analisabilidade</b>	É a dificuldade que se tem ao detectar um erro, localizá-lo no código fonte ou encontrar a causa da ocorrência do mesmo (caso seja este de hardware). Em alusão ao código fonte é preciso que este seja modular, legível, pouco complexo e bem documentado, a fim de que o erro seja detectado rapidamente.
	<b>Modificabilidade</b>	Capacidade que o produto tem de receber modificações
	<b>Estabilidade</b>	Capacidade de evitar efeitos inesperados a partir de modificações
	<b>Testabilidade</b>	Capacidade de validar as modificações efetuadas no produto
	<b>Conformidade</b>	Capacidade de aderir a padrões e convenções relativas à manutenibilidade
	<b>Adaptabilidade</b>	Capacidade de ser adaptado em diferentes ambientes sem intervenção
portabilidade	<b>Capacidade Instalação</b>	Capacidade de ser instalado em um ambiente específico
	<b>Coexistência</b>	Capacidade que o produto tem de coexistir com outro software independente em um ambiente comum, compartilhando recursos comuns.
	<b>Capacidade Substituição</b>	Capacidade que o produto de software deve ter de ser usado no lugar de outro produto de software com o mesmo propósito, no mesmo ambiente
	<b>Conformidade</b>	Capacidade de aderir a padrões e convenções relativas à portabilidade
	<b>Extensibilidade</b>	Capacidade do produto de permitir acoplamentos de novos módulos

## Apêndice C – Mapeamento das características de qualidade de software da norma 9126-1 nos módulos de avaliação LAPS

---

Este apêndice traz o mapeamento das características de qualidade interna e externa da norma 9126-1 para os módulos de avaliação do LAPS. Este mapeamento pode ser observado na “Tabela 20. Módulos LAPS vs. Subcaracterísticas ISO”. Esta tabela apresenta, na primeira coluna, todas as subcaracterísticas de qualidade interna e externa da norma. Cada classe de características esta representada por uma cor, que é refletida nas subcaracterísticas da característica em questão. A tabela também apresenta os módulos LAPS abreviados, e a legenda correspondente a cada um dos módulos. É importante observar que o módulo de competidores do LAPS não possui, previamente, nenhum mapeamento para as características de qualidade interna e externa. A avaliação de competidores, como já citado neste trabalho, se encaixa bem com avaliações de visão externa e de visão em uso, para tal, ela utiliza outros módulos LAPS para realizar testes de *benchmark* que servirão como base de comparação dos competidores em questão. Já que a avaliação de competidores, conseqüentemente também pode envolver diferentes características de qualidade, um mapeamento prévio não pode ser realizado.

Tabela 20. Módulos LAPS vs. Subcaracterísticas ISO

Características de Qualidade	
Portabilidade	
Manutenibilidade	
Eficiência	
Usabilidade	
Confiabilidade	
Funcionalidade	

Legenda de Módulos LAPS	
CF	Código Fonte
US	Usabilidade
FU	Funcionalidade
FR	Falhas e Recuperação
DE	Desempenho
AR	Arquitetura
PO	Portabilidade
SC	Segurança e Controle de Acesso
DU	Doc. Usuário
DS	Doc. Sistema
CO	Competidores
ES	Especialista

Continuação da “Tabela 20. Módulos LAPS vs. Subcaracterísticas ISO”

	CF	US	FU	FR	DE	AR	PO	SC	DU	DS	CO	ES
Adequação			X									X
Acurácia			X									
Interoperabilidade			X			X	X					
Segurança de Acesso								X				
Conformidade												
Maturidade			X			X						
Tolerância a Falhas				X								
Recuperabilidade				X								
Conformidade				X								
Inteligibilidade		X							X	X		
Apreensibilidade		X							X			
Operacionalidade		X							X			
Atratividade		X							X			
Conformidade												
Compto relação ao tempo					X	X						
Compto relação aos recursos					X	X						
Conformidade												
Analisabilidade	X					X						
Modificabilidade	X					X	X					
Estabilidade						X						
Testabilidade			X									
Conformidade												
Adaptabilidade							X					
Capacidade Instalação							X		X			
Coexistência							X					
Capacidade Substituição							X					
Conformidade												
<b>Extensibilidade</b>												

## Apêndice D - glossário

---

A definição dos termos técnicos apresentados neste apêndice são todos retirados de [ISO 14598-1, 1998], salvo referenciados explicitamente. Estes termos, bastante usados neste trabalho, são aqui explanados.

### *A – E*

**Atributo** – propriedade mensurável, física ou abstrata, de uma entidade.

**Avaliação Heurística** - Esta avaliação é feita a partir da inspeção de tabelas de critérios nas quais aspectos como consistência da representação, usabilidade, performance, qualidade do feedback e outros atributos são considerados segundo uma escala ordinal de três, quatro ou mais níveis (ex: regular, bom, muito bom e ótimo) [WikiPedia].

**Benchmark** – A: um ponto de referência do qual medidas podem ser tiradas. B: algo que serve como padrão para os outros poderem medir ou julgar. C: um problema padrão ou este que serve como base para avaliação ou comparação (como por exemplo, comparação de performance de sistemas computacionais) [Webster].

**Brainstorm** – técnica que pode ser feita individualmente ou em grupo. Nas sessões de brainstorm os participantes são encorajados a compartilhar suas idéias com os outros componentes da reunião sem interrupção do processo cognitivo. Esta é a chave para o brainstorm: respeitar e não interromper as idéias de cada componente. Assim que as idéias vêm em mente, elas são capturadas e é estimulado o desenvolvimento de idéias melhores baseadas nas idéias apresentadas. O processo ocorre em vários ciclos e se repete até que as opiniões dos componentes da reunião de brainstorm convirjam para um ideal comum [WikiPedia].

**Defeito:** passo, processo ou definição de dados incorretos em um programa de computador.

**Desenvolvedor:** organização que executa atividades de desenvolvimento (incluindo análise de requisitos, projeto, testes até a aceitação) durante o processo de ciclo de vida de software. [ISO 12207, 1998]

**Framework:** 1a: Uma estrutura conceitual básica para algo (por exemplo, idéias). b: um esqueleto, uma estrutura [Webster].

## *F – J*

**Falha:** término da capacidade de um produto de executar uma função requerida ou a sua incapacidade de executá-la dentro de limites previamente especificados.

**Guideline:** conjunto de indicações ou guias de conduta para uma determinada atividade ou ação [Webster].

## *K – O*

**Medida (substantivo):** número ou categoria atribuído a um atributo de uma entidade através de uma medição.

**Medição:** uso de uma métrica para atribuir um valor (o qual pode ser um número ou categoria), obtido a partir de uma escala, a um atributo de uma entidade.

**Métrica:** método e escala de medição definidos

**Modelo de qualidade:** conjunto de características e os relacionamentos entre elas, que fornecem a base para a especificação dos requisitos de qualidade e para a avaliação de qualidade.

**Módulo de avaliação:** pacote de tecnologia de avaliação para uma característica ou subcaracterística de qualidade de software específica.

## *P – T*

**Produto de software:** conjunto de programas de computador, procedimentos e possível documentação e dados associados. Produtos incluem produtos intermediários e produtos destinados a usuários como desenvolvedores ou mantenedores [ISO 12207, 1998]

**Qualidade externa:** o quanto um produto satisfaz necessidades explícitas e implícitas quando utilizado em condições especificadas.

**Qualidade interna:** totalidade dos atributos de um produto que determinam sua capacidade para satisfazer necessidades explícitas e implícitas quando utilizado em condições especificadas.

**Qualidade em uso**<sup>1</sup>: o quanto um produto, utilizado por usuários especificados, atende às necessidades desses usuários para que eles atinjam as metas

---

<sup>1</sup> NOTA DA NORMA 14598-1 – A característica “segurança” foi adotada após a publicação do texto internacional da ISO/IEC 14598-1, por isso a característica segurança não faz parte do conjunto de características de qualidade em uso aqui descritas (eficácia, produtividade e satisfação).

especificadas com eficácia, produtividade e satisfação, em contextos de uso especificados.

**Release:** dar permissão para publicação, performance, exibição ou venda. Tornar algo disponível para algum público [Webster].

**Software:** conjunto completo ou apenas uma parte, dos programas, procedimentos, regras e documentação associada de um sistema de processamento de informação. [ISO 2382-1, 1993]

## ***U – Z***

**Usuário:** um indivíduo que usa o produto de *software* para executar uma função específica. Podem ser operadores, pessoas que recebem o resultado do *software*, ou desenvolvedores e mantenedores, equipes de desenvolvimento e manutenção de *software*. Um indivíduo ou organização que utiliza um sistema em operação para executar uma função específica. O usuário pode executar outros papéis, tais como adquirente, desenvolvedor ou mantenedor.

## Apêndice E - Arte

---

O trabalho artístico do Envision foi feito por Marcos Vinícius Pires Spinelli, coordenado pelo autor.



**Figura 12. Logo Envision Básico**



**Figura 13. Logo Envision Básico com subtítulo**



**Figura 14. Logo Envision Cromado**



**Figura 15. Logo Envision Cromado com subtítulo**

ENVISION

Figura 16. Logo Envision Prata

ENVISION  
UM MODELO PARA AVALIAÇÃO DE PRODUTO DE SOFTWARE BASEADO EM VISÕES EXTENSÍVEIS

Figura 17. Logo Envision Prata com subtítulo

ENVISION

Figura 18. Logo Envision Ouro

ENVISION  
UM MODELO PARA AVALIAÇÃO DE PRODUTO DE SOFTWARE BASEADO EM VISÕES EXTENSÍVEIS

Figura 19. Logo Envision Ouro com subtítulo

ENVISION

Figura 20. Logo Envision Negativo



**Figura 21. Logo Envision Negativo com subtítulo**



**Figura 22. Logo Envision marca d'água**

Augusto César Pires Spinelli  
Orientando

Hermano Perrelli de Moura  
Orientador

Alexandre Marcos Lins de Vasconcelos  
Co-orientador