

Medição da Qualidade de Aplicações Web na Fase de Requisitos

Tiago P. C. Reis Jaelson F. B. Castro
Universidade Federal de Pernambuco – Centro de Informática
Caixa Postal 7851, CEP 50732-970, Recife-PE, Brasil
{tpcr,jbc}@cin.ufpe.br

Luis A. Olsina
Departamento de Informática, Facultad de Ingeniería, UNLPam
Calle 9 esq. 110 - (6360) General Pico - LP - Argentina
olsinal@ing.unlpam.edu.ar

Resumo

Este trabalho propõe uma metodologia de avaliação de qualidade de aplicações Web ainda na sua fase de requisitos. A partir de um documento de requisitos bem elaborado algumas características do sistema em desenvolvimento já podem ser previamente identificadas e avaliadas. Tendo em vista o conhecimento do domínio da aplicação, algumas funcionalidades e atributos ausentes no projeto inicial podem ser recomendados, contribuindo para a melhoria do produto. Ao término da medição será apresentado um conceito, que representa a qualidade do projeto Web seguindo a metodologia proposta. Obviamente que os pontos fracos levantados ainda na fase de requisitos, podem ser corrigidos, acarretando o desenvolvimento de um produto de maior qualidade.

Abstract

This work presents a methodology for the evaluation of the quality of software application during the requirements phase. From a well prepared requirements document some characteristics of the system under development can be identified and evaluated. Taking in consideration the application domain knowledge, some missing attributes and functionality from the initial design can be suggested, contributing towards the product improvement. At the end of the measurement a grade representing the quality of the Web project is presented. Obviously the weak points detected during the requirements phase can still be corrected, generating the development of a product of better quality.

1 Introdução

O desenvolvimento Web tem sido cada vez mais difundido na comunidade de tecnologia da informação. Existem os mais variados tipos de aplicações, com níveis de complexidade distintos, variando desde sites estáticos orientados a documentos até aplicações de softwares dinâmicos. Infelizmente, os modelos de processo que promovem um ciclo de vida de desenvolvimento eficiente muitas vezes não são utilizados. De forma análoga, os modelos de

processo e metodologias de avaliação que promovem a compreensão e a melhora na qualidade desses artefatos não são adotados. Isso acarreta práticas de desenvolvimento *ad hoc*, prejudicando tanto as previsões de custo e tempo, quanto à qualidade do produto final [5].

Uma das metas principais de avaliação e comparação da qualidade de artefatos Web é medir, analisar e compreender o grau de cumprimento de um conjunto de características e atributos com respeito aos requisitos de qualidade estabelecidos, para um determinado perfil de usuário, em um certo domínio de aplicação. O nosso trabalho baseia-se em [13, 14], que avalia os produtos na sua fase final, operativa. Esta metodologia foi desenvolvida com o propósito de ser uma estratégia eficaz para avaliar e analisar a qualidade de sites e aplicações Web em geral. Está baseada em um modelo hierárquico das características de mais alto nível previstas na norma ISO 9126-1 [9], a saber: Usabilidade, Funcionalidade, Confiabilidade, Eficiência, Manutenibilidade e Portabilidade. A partir dessas características, se derivam sub-características e, a partir dessas, seguindo um processo de decomposição hierárquico, se especificam atributos. O resultado do processo de avaliação (e eventualmente de comparação) pode ser interpretado como o grau de satisfação dos requisitos de qualidade. O que estamos propondo antecipa essa avaliação, para que sejam beneficiados pela descoberta de erros numa fase mais inicial do processo de desenvolvimento.

Os métodos existentes para análise da qualidade de produtos Web não levam em consideração o princípio da engenharia de software de que o custo para modificar um requisito na fase de manutenção chega a ser cem vezes maior do que na fase de definição de requisitos. Então por que esperar para a última fase de desenvolvimento o que se quer medir, sabendo-se que quanto mais se distancia das fases iniciais mais caro ficará sua posterior correção? O leitor pode estar questionando que não há como medir as características na fase inicial uma vez que ainda não existe o produto. No entanto, existe um documento que pode representá-lo, o documento de requisitos. Vale salientar que este cobre tanto os requisitos funcionais quanto os não funcionais [7].

O documento de requisitos pode ser considerado um espelho do produto final. Estamos levando em consideração que o desenvolvimento vai realmente se apoiar no documento de requisitos. Assim é possível fazer essa pré-avaliação. Os requisitos que forem surgindo durante o processo podem ser incorporados à metodologia.

Partimos do princípio da existência de um documento de requisitos bem elaborado, para que se tenha algo a ser analisado. Sabe-se que existem certas propriedades, chamadas de emergentes, pertinentes ao sistema como um todo que apenas surgem quando todos os subsistemas estão integrados, isto é, na existência do produto final. Essas propriedades realmente têm que esperar um pouco mais para serem investigadas, mas isso não invalida a nossa proposta, uma vez que muitas outras podem ser descobertas e corrigidas nas fases iniciais do processo de desenvolvimento.

Este artigo é organizado em cinco seções. A seção 2 trata das bases e motivações para os processos de medição. Em seguida, na seção 3, falaremos um pouco do documento de requisitos. Mostramos a metodologia proposta na seção 4, e por fim, na seção 5 apresentamos as conclusões e os trabalhos futuros.

2 Bases para o processo de medição

A motivação do nosso processo de medição pode ser validada por duas definições de uma reconhecida organização:

“Definir a qualidade de um software para um sistema é equivalente a definir uma lista e atributos de qualidade de software requeridos por este sistema” [6].

“Para medir atributos de qualidade de software se deve identificar um conjunto apropriado de métricas” [6].

Também nos baseamos na Organização de Padrões Internacionais ISO, que em um trabalho conjunto com IEC definiram seis características de alto nível que descrevem a qualidade de um produto: usabilidade, funcionalidade, confiabilidade, eficiência, portabilidade, e manutenibilidade. *Essas características provêm uma baseline para posteriores refinamentos e descrições da qualidade de um software [8].*

Essa mesma situação é revelada na publicação [8], que declara: *“A qualidade dos produtos de software podem ser descritas em termos de características de qualidade como as definidas em ISO/IEC 9126. Não obstante, o estado da arte em medição de software é tal que, em geral, não são práticas de medições diretas dessas características. O que é possível é valorar essas características sobre a base medidas de atributos de menor nível de abstração”.*

O padrão de qualidade de produtos propõe características de alto nível de abstração. A partir dessas, sub-características devem ser propostas, e assim sucessivamente, sendo possível trabalhar com atributos de qualidade e suas respectivas métricas [2, 10] cada vez mais concretos. De fato, o uso de métricas de software reduz a subjetividade da avaliação de qualidade de software ao prover uma base quantitativa para tomar decisões acerca da qualidade do software [6].

Muitas características de qualidade de artefatos Web devem ser levadas em consideração durante o processo de desenvolvimento. Entre essas características podemos citar: usabilidade, navegabilidade, segurança, características de busca e recuperação de informação, manutenibilidade, performance, acessibilidade, e aspectos de funcionalidades específicas do domínio da aplicação.

Devem ser assegurados mecanismos por meio do quais possam ser construídos produtos que cumpram tais características de forma organizada. Por meio da avaliação da qualidade, pode-se estimar o grau de cumprimento das características existentes com o que foi proposto.

3 Documento de Requisitos

A metodologia proposta recebe como entrada um documento de requisitos e produz como saída a avaliação do futuro produto, espelho daquele documento. Deste modo, resolvemos nos estender sobre a elaboração deste documento, já que é de fundamental importância para aplicação da metodologia.

A elaboração de um documento de requisitos pressupõe a existência da definição do domínio da aplicação, para que possam ser levantadas características importantes que possuem relação direta com o produto a ser desenvolvido. No caso, por exemplo, do domínio de sites acadêmicos, alguns atributos como mecanismos de busca por professores e índice de disciplinas acadêmicas devem estar presentes. Muitos desses atributos não estariam presentes caso o domínio da aplicação fosse de comércio eletrônico [13].

A avaliação também depende do perfil do usuário. Em termos gerais, num alto nível de abstração para domínios WEB, são considerados três perfis de usuário, a saber: visitantes, desenvolvedores e gerentes. Podemos dividir a classe de visitantes em classe mais específicas, como veremos posteriormente.

Contudo, a importância das características de qualidade (como usabilidade, funcionalidade, confiabilidade, eficiência, portabilidade e manutenibilidade) varia de acordo com o ponto de vista considerado [8]. Por exemplo, a visão do usuário (perfil do visitante, para domínios WEB), consiste no interesse dos mesmos em usar os artefatos de software, como também a sua performance, eficiência, facilidade de uso, entre outros. Os usuários não estão interessados nas características internas ou no desenvolvimento dos artefatos. É importante salientar que os atributos do produto contribuem para a qualidade em uso. As métricas de qualidade em uso medem o grau com que o produto cumpre as necessidades dos usuários em cenários específicos.

Por outro lado, a visão de qualidade dos desenvolvedores deve considerar não apenas os requisitos do produto para a visão dos usuários, como também a qualidade dos produtos intermediários resultantes das atividades das fases de desenvolvimento. Para isso devem considerar as métricas apropriadas. Características de qualidade como manutenibilidade e portabilidade são de especial importância para os desenvolvedores.

Finalmente, a visão de qualidade dos gerentes é uma visão integradora que necessita incorporar requisitos de negócios às qualidades individuais. Por exemplo, um gerente pode estar interessado em um ótimo equilíbrio entre a melhora de um produto dentro dos custos e tempo estabelecidos.

Como foi dito, a categoria de visitante pode ser dividida: visitantes gerais e visitantes especialistas. Os visitantes gerais ainda podem ser classificados como: visitantes casuais ou intencionais. Segundo [4], um visitante casual se define como um usuário que ingressa no site sem intenção ou vontade própria, mas devido a algum acontecimento que não é de seu controle, a indexação errada de um mecanismo de busca, por exemplo. Esse tipo de visitante permanece provavelmente por um curto intervalo de tempo, e dificilmente retorna. Já um visitante intencional se define como uma audiência que tem ao menos algum conhecimento ou manifesta algum interesse no domínio da aplicação em questão, para provavelmente informar-se ou aprender um pouco mais acerca do mesmo. Sua permanência no site é geralmente maior do que na audiência casual. Observe que caso o visitante casual volte ao site de forma espontânea, este passa a fazer parte da categoria visitante intencional. Por último, um visitante especialista se define como uma audiência bem determinada, como pesquisadores. Sua permanência no site é geralmente maior do que a audiência intencional. Segundo [3], um visitante especialista é aquele que se encontra interessado geralmente por todo o conteúdo do site, e em todos os links existentes, tanto na página principal quanto nas restantes.

Dentre os visitantes de uma mesma categoria, ainda é possível achar divergências. Isso ocorre principalmente entre usuários de diferentes culturas. Isso abre margem à seguinte questão: deve-se desenvolver uma única interface para todas as culturas ou várias interfaces distintas, dependendo da localização geográfica, da cultura do usuário? Com isso nota-se claramente que o projeto de uma interface depende também de elementos culturais.

O documento de requisitos descreve os requisitos do sistema utilizado para prover comunicação entre usuários, desenvolvedores e gerentes. Não existe um nome padrão único para representá-lo, em diferentes organizações são utilizados diferentes denominações, como especificação funcional, especificação dos requisitos do sistema etc. Seguem alguns tópicos que o documento de requisitos poderá abordar:

1. Serviços e funções que o sistema deve prover;
2. Limitações sobre as quais o sistema deve operar;
3. Propriedades gerais do sistema, isto é, comportamento do sistema diante das propriedades emergentes;

4. Definições de outros sistemas que irão interagir com o sistema em questão;
5. Informações a respeito do domínio de aplicação do sistema;
6. Limitações no processo utilizado para desenvolver o sistema;
7. Definições sobre o hardware no qual o sistema executará.

Adicionalmente, o documento de requisitos deverá conter um capítulo introdutório que provê um resumo do sistema, necessidades de negócio suportadas pelo sistema e um glossário que explica a terminologia utilizada [11]. Esse glossário é particularmente importante, pois pessoas de diferentes áreas terão contato com o documento de requisitos, facilitando assim sua compreensão.

Seguem exemplos de usuários do documento de requisitos, com seus respectivos papéis:

1. Clientes do Sistema: especificam os requisitos e o lêem para avaliar se os mesmos satisfazem suas necessidades;
2. Gerentes de Projeto: usam os documentos de requisitos para planejarem uma proposta para o sistema e o processo de desenvolvimento do sistema;
3. Engenheiros de Sistema: usam os requisitos para entenderem o sistema em construção;
4. Engenheiros de teste do sistema: usam os requisitos para desenvolverem testes de validação do sistema;
5. Engenheiros de manutenção do sistema: usam os requisitos para entenderem o sistema.

Muitas organizações propuseram uma estrutura para o documento de requisitos. Entre elas podemos citar o padrão IEEE 830 [7], que sugere uma estrutura para o documento de requisitos, apresentada na Figura 1:

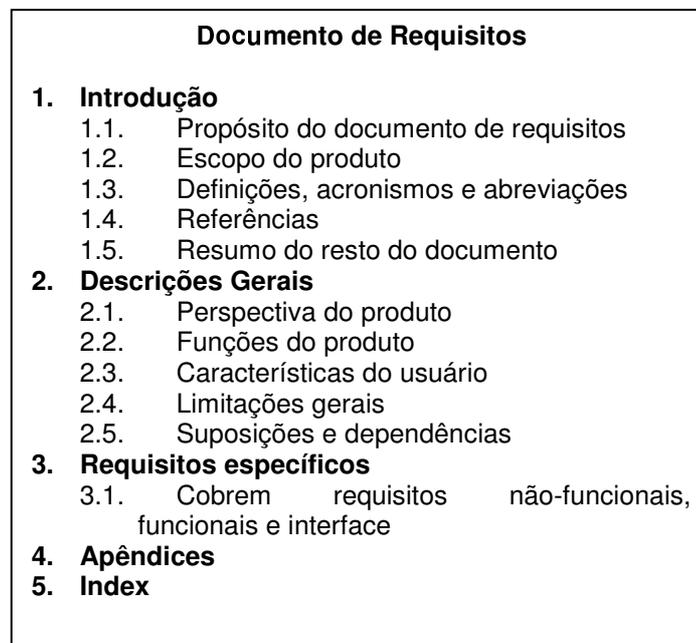


Figura 1 – Estrutura de um documento de requisitos segundo IEEE 830.

É possível omitir partes, e adicionar novas seções, se necessário. Também podemos definir uma lista de partes estáveis e variáveis. Partes estáveis devem aparecer em todos os documentos de requisitos, sejam quais forem os seus leitores. Partes variáveis devem ser incluídas dependendo do grupo alvo. É como se existisse uma versão executiva, mais resumida, e uma versão técnica, como todos os detalhes.

Uma vez confeccionado o documento de requisitos será possível aplicar a metodologia proposta, que será detalhada na seção a seguir.

4 Metodologia para avaliação da qualidade de aplicações Web na fase de requisitos

Esta metodologia antecipa a avaliação da qualidade de um projeto Web para a sua fase de requisitos. Logo, com o seu uso, temos o benefício de descobrir erros numa fase inicial do processo de desenvolvimento.

Para sua aplicação, é necessário seguir cinco passos: representação das características e atributos de qualidade; especificação descritiva da árvore de características e atributos de qualidade; associação de pesos aos atributos; associação de notas aos atributos; cálculo geral (média ponderada). Os passos serão detalhados a seguir:

Passo 1 – Representação das características e atributos de qualidade: modelos de qualidade

A metodologia é flexível, permitindo a agregação de novas características, sub-características e atributos de uma forma modular, conforme as necessidades específicas do perfil do usuário, do domínio da aplicação Web e do contexto organizacional.

É preciso confeccionar a árvore de características e atributos de qualidade conforme o domínio e o perfil do usuário selecionado. Partimos de seis características, que podem ser mais ou menos relevantes dependendo do domínio e do perfil do usuário. Usaremos como exemplo um domínio de sites acadêmicos, e o perfil do usuário sendo o de visitantes. Levamos em consideração visões do perfil visitantes: visão dos estudantes (ou visitantes como seus pais), dos professores, e pesquisadores acadêmicos e externos.

Visitantes observam o site de forma dinâmica, em geral em pouco tempo. Estão interessados nas suas funcionalidades. Além de utilizar os serviços oferecidos, o visitante está preocupado com a confiabilidade e a performance (ou eficiência), como também o quão fácil será utilizar os serviços oferecidos (usabilidade). No entanto, manutenibilidade e portabilidade não estão ligadas aos conceitos dos visitantes (usuários finais).

A especificação da árvore de características e atributos de qualidade determinará os critérios de medição. O cruzamento dos dados dessa árvore proposta com o documento de requisitos apontará deficiências que podem ser corrigidas.

Algumas dessas características possuem uma dificuldade adicional de serem medidas na ausência de um produto final, outras podem ser mais facilmente avaliadas. Com relação à funcionalidade, como prever quais funcionalidade são importantes, tendo em mão o domínio da aplicação. Consulta a especialistas? Questionários aos usuários? Muitos desenvolvedores de aplicações e sistemas Web podem questionar que algumas funções notificadas como ausentes na nossa avaliação (lembre-se que temos domínio da aplicação e o perfil do usuário selecionado) não são interessantes, e que isso não deveria contribuir para o site ter uma avaliação mais baixa.

Para solucionar esse problema, pedimos auxílio à uma área chamada engenharia do domínio. Esta é baseada na teoria do domínio, onde os domínios são especificados em detalhes.

Partimos do pressuposto que existe uma teoria do domínio para o domínio da aplicação que estamos querendo avaliar. Assim, o que estaria em questionamento seria a própria teoria do domínio, e não a nossa metodologia de avaliação.

Outros requisitos não funcionais poderiam ser definidos por especialistas. É o caso, por exemplo, da confiabilidade. Pesquisas na área de segurança mostrariam o que deve ser levado em consideração para que uma aplicação seja considerada confiável. Assim seria possível construir uma árvore de características e atributos de segurança.

Apesar da árvore a ser apresentada estar baseada em métricas de produtos, já é possível efetuar uma medição, uma vez que o documento de requisitos deve prevê-las e especificá-las. Como já foi dito, existem outras métricas que necessariamente apenas podem ser avaliadas com a existência do produto final.

Segue parte da árvore em questão, adaptada de [12]:

1. Usabilidade

- 1.1 Entendimento global do site
 - 1.1.1 Esquema de organização global
 - 1.1.1.1 Mapa do site
 - 1.1.1.2 Tabela de conteúdos
 - 1.1.1.3 Índice alfabético
 - 1.1.2 Qualidade dos labels
 - 1.1.3 Visita guiada para estudantes
 - 1.1.4 Mapa de imagens
- 1.2 Tarefas de retorno e ajuda
 - 1.2.1 Qualidade das tarefas de ajuda
 - 1.2.1.1 Ajuda geral direcionada para o estudante
 - 1.2.1.2 Ajuda na busca
 - 1.2.2 Último indicador de atualização do Web-site
 - 1.2.2.1 Global
 - 1.2.2.2 Por sub-site ou página
 - 1.2.3 Acesso aos endereços
 - 1.2.3.1 Endereço eletrônico
 - 1.2.3.2 Telefone/fax
 - 1.2.3.3 Endereço para correspondência
 - 1.2.4 Tarefas de FAQ
 - 1.2.5 Formulário de retorno
 - 1.2.5.1 Questionários
 - 1.2.5.2 Livro de visitas
 - 1.2.5.3 Comentários
- 1.3 Tarefas diversas
 - 1.3.1 Suporte a outros idiomas
 - 1.3.2 Mecanismos para destacar novidade
 - 1.3.3 Indicação da resolução do monitor

2. Funcionalidade

- 2.1 Tarefas de busca
 - 2.1.1 Mecanismos de busca no Web-site
 - 2.1.1.1 Busca por escopo
 - 2.1.1.1.1 Busca de pessoas
 - 2.1.1.1.2 Busca por cursos
 - 2.1.1.1.3 Busca por disciplinas acadêmicas
 - 2.1.1.2 Busca global
- 2.2 Tarefas de navegação
 - 2.2.1 Navegabilidade
 - 2.2.1.1 Orientação
 - 2.2.1.1.1 Indicador do caminho
 - 2.2.1.1.2 Marcador da posição atual
 - 2.2.1.2 Média de links por página
 - 2.2.2 Objetos de controle de navegação
 - 2.2.2.1 Apresentação da permanência e estabilidade dos controles contextuais
 - 2.2.2.1.1 Permanência dos controles
 - 2.2.2.1.2 Estabilidade dos controles
 - 2.2.2.2 Nível de "scrolling"
 - 2.2.2.2.1 "scrolling" vertical
 - 2.2.2.2.2 "scrolling" horizontal
 - 2.2.3 Intuitividade da navegação

- 2.2.3.1 Título do link
- 2.2.3.2 Qualidade da frase do link
- 2.3 Tarefas do domínio relativo ao estudante
 - 2.3.1 Conteúdo relevante
 - 2.3.1.1 Informações sobre unidade acadêmicas
 - 2.3.1.1.1 Índice de disciplinas acadêmicas
 - 2.3.1.1.2 Subsite de disciplinas acadêmicas
 - 2.3.1.2 Informações diversas
 - 2.3.1.2.1 Entrada de requerimentos
 - 2.3.1.2.2 Download de requerimentos
 - 2.3.1.3 Qualidade do conteúdo
 - 2.3.1.3.1 Qualidade do índice
 - 2.3.1.3.2 Qualidade das descrições
 - 2.3.1.3.3 Qualidade do curso oferecido
 - 2.3.1.3.3.1 Ementas
 - 2.3.1.3.3.2 Formas de avaliação
 - 2.3.1.3.4 Descrição dos curso
 - 2.3.1.3.4.1 Comentários
 - 2.3.1.3.4.2 Calendário
 - 2.3.1.4 Informação dos serviços dos estudantes
 - 2.3.1.4.1 Índice de serviços
 - 2.3.1.4.2 Informações do calendário escolar
 - 2.3.1.4.3 Informações dos alojamentos
 - 2.3.1.4.4 Informações culturais
 - 2.3.1.4.5 Informações esportivas
 - 2.3.1.5 Informação da infraestrutura acadêmica
 - 2.3.1.5.1 Informação da biblioteca
 - 2.3.1.5.2 Informação dos laboratórios
 - 2.3.1.5.3 Informação dos resultados de pesquisa
 - 2.3.2 Serviços on-line
 - 2.3.2.1 Informação da grade curricular
 - 2.3.2.2 Serviços Web
 - 2.3.2.3 Serviços FTP
 - 2.3.2.4 Serviços de newsgroup

Passo 2 – Especificação descritiva da árvore de características e atributos de qualidade

A título de documentação, para cada característica, sub-característica ou atributo, deve ser preenchida planilhas de informações (vide planilha ampliada em [14]). A seguir mostraremos alguns exemplos.

<p>Título: _____ Código: _____</p> <p>Tipo: Característica</p> <p>Sub características (códigos): _____</p> <p>Definição / Comentários: _____</p> <p>Peso: _____</p> <p>Nota(calculada através das notas das sub-características): _____</p> <p>Exemplo/s(opcional): _____</p>	<p>Título: _____ Código: _____</p> <p>Tipo: Atributo</p> <p>Característica de maior nível(código): _____</p> <p>Supercaracterística (código): _____</p> <p>Definição / Comentários: _____</p> <p>Peso: _____</p> <p>Nota(dada pelo avaliador através do cruzamento com a árvore): _____</p> <p>Exemplo/s: _____</p>
<p>Título: _____ Código: _____</p> <p>Tipo: Sub-característica</p> <p>Supercaracterística (código): _____ Sub-características (códigos): _____ Atributo/s _____</p> <p>(códigos/s): _____</p> <p>Definição / Comentários _____ Peso: _____</p> <p>Nota(calculada através das notas das sub-características e atributos): _____</p> <p>Exemplo/s(opcional): _____</p>	

Figura 2 - Templates para especificar características, sub-características e atributos.

Nas Figuras 3 e 4 apresentamos exemplos para a característica Usabilidade e para o atributo Mapa do Site.

Nome: Usabilidade Código: 1 Tipo: **Característica**
Sub-características (códigos): Entendimento global do site(1.1); Tarefas de retorno e ajuda(1.2); Tarefas diversas(1.3)
Definição/Comentários: Capacidade do produto de software de ser entendível, aprendido e usado de forma atrativa pelos usuários, quando utilizado sob as condições de especificação.
Peso: 0.3
Nota: (calculada através das notas das sub características):7.5
Exemplo: vide [11]

Figura 3 – Exemplo de característica

Nome: Mapa do site Código: 1.1.1.1 Tipo: **Atributo**
Maior característica (código): Usabilidade(1)
Sub-característica pai(código): Esquema de organização global (1.1.1)
Definição/Comentários: Um mapa do site é uma representação gráfica que mostra a estrutura ou arquitetura de todo o website. Qualquer página do site deve poder ser acessada através do mapa do site.
Peso: 0.5
Nota: 10
Exemplo: <http://www.upc.es/catala/index.htm>

Figura 4 - Exemplo de atributo

Passo 3 – Associar pesos aos atributos

Obviamente que os atributos possuem importâncias variadas, onde uns são mais relevantes do que outros. Deve-se associar a cada atributo um peso entre 0(zero) e 10(dez). Quanto mais importante for o atributo para a qualidade do sistema, maior deverá ser seu peso. Se uma característica é decomposta em sub-características, a soma de todos os pesos dessas sub-características deve ser 10(dez). Da mesma forma, se uma sub-característica é decomposta em atributos, o somatório dos pesos desses atributos deve ser igual a 10(dez). Veja um exemplo:

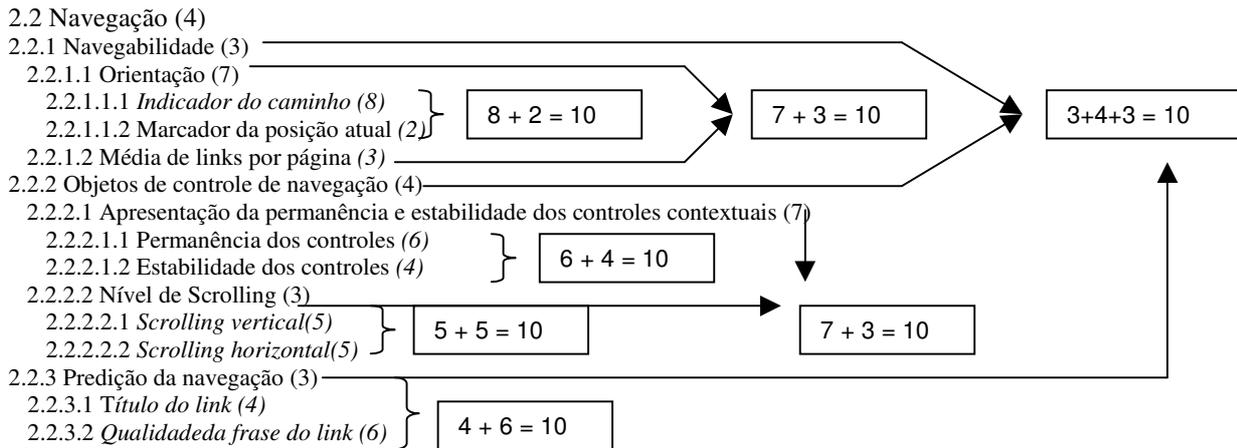


Figura 5 – Atribuição de pesos aos elementos da árvore

Cabe ressaltar que atualmente estamos estudando métodos científicos capazes de associar pesos aos atributos de uma forma objetiva.

Passo 4 - Associar notas aos atributos

Associar nota 0 (zero), 5 (cinco) ou 10 (dez) a cada folha da árvore, dependendo do seu grau de cumprimento com o documento de requisitos. Caso não esteja previsto deverá ser atribuída nota zero. Neste caso, o atributo ausente deverá ser anotado na seção de recomendações, para que seja analisado posteriormente o por quê da sua falta. Se o documento de requisitos abordar de forma parcial a nota daquele atributo deverá ser cinco, e se estiver completamente especificado dez. As notas das sub-características serão calculadas através das notas dos respectivos atributos, associadas a seus pesos. De modo análogo serão calculadas as notas das características. Vale salientar que se uma sub-característica que não possui atributos entrar no cálculo da média ponderada (ver passo 5) com outra(s) sub-característica(s) que possui(em), sua nota deve ser multiplicada por dez (10), para que a média calculada seja coerente. Caso não fosse feito esse ajuste a nota máxima de 2.2.1.1 seria 100 e a de sua irmã 2.2.1.2 seria 10. Veja um exemplo na Figura 6:

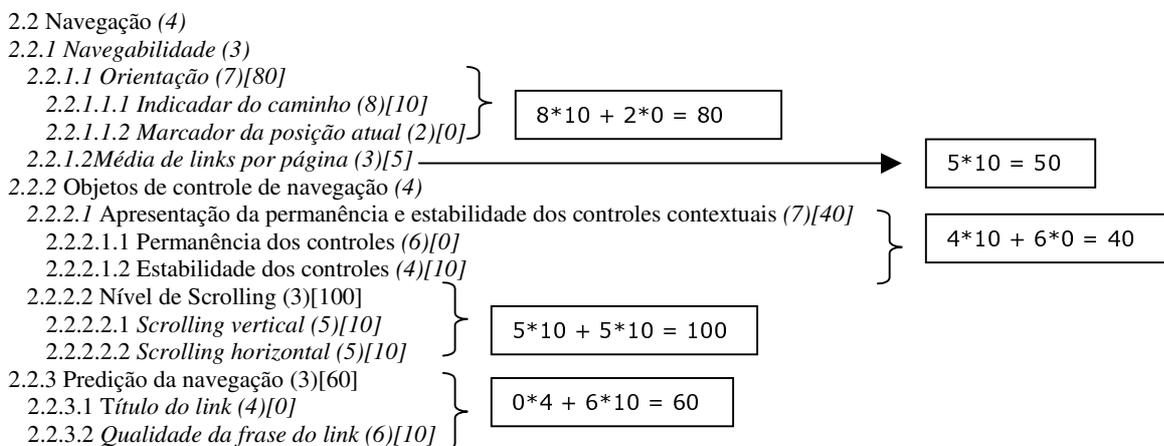


Figura 6 - Cálculo das notas das sub-características a partir dos atributos.

É importante salientar que a associação das notas aos atributos está sendo revista, onde estamos procurando nos basear em técnicas mais robustas de agregação e cálculo de notas [1, 14]. Uma das técnicas em avaliação é a WSM (Weighted Score Method), bastante utilizada nas situações de tomada de decisão.

Passo 5 – Cálculo geral (média ponderada)

Após o cálculo da nota de cada atributo, esta deve ser multiplicada pelo seu peso e adicionada entre os atributos do mesmo nível que resultará na nota da sub-característica. Essa por sua vez deverá ser multiplicada pelo seu peso e adicionada com o resultado das sub-características do mesmo nível, resultando na nota da característica em questão. Por exemplo, segue uma parte da árvore de características e atributos de qualidade com os pesos entre parêntese e as notas entre colchetes:

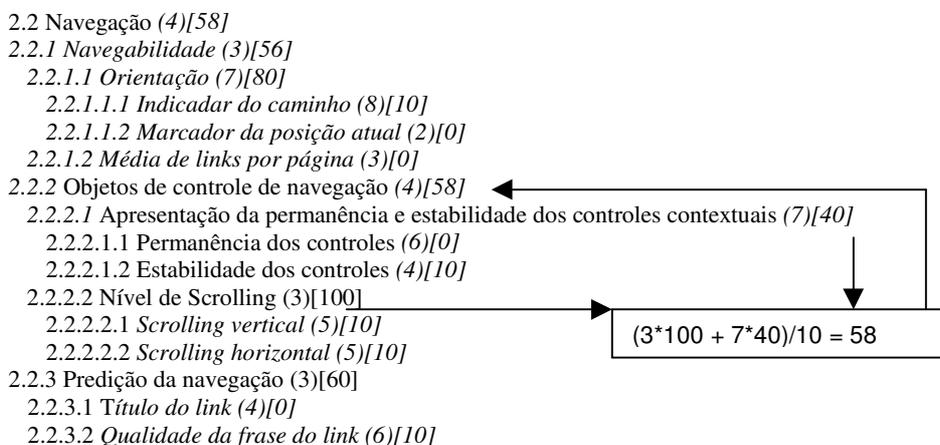


Figura 7 - Processo do cálculo geral segundo o modelo aditivo.

Note que a nota de 2.2 foi calculada da seguinte forma:

$$(3 \times 56 + 4 \times 58 + 3 \times 60) / 10 = 58$$

E assim deve ser feito com toda a árvore de características e atributos de qualidade, resultando numa nota única, que é a nota do documento de requisitos que representa o produto final. Vale salientar que possuímos esta nota antes da existência do produto.

Por motivo de espaço não estamos propondo uma estrutura de um documento de requisitos. Temos o interesse de apresentar esse template, definindo os recursos para a elaboração de um documento de requisitos de qualidade. No entanto, iremos mostrar como a metodologia se aplica tomando por base um trecho de um documento de requisitos escrito em linguagem natural:

‘O site deve possuir um mapa, capaz de permitir a visualização de todos os caminhos do site, de forma centralizada. A partir desse mapa do site deve ser possível clicar nos links e acessar as páginas. Este mapa tem o objetivo de orientar o usuário, facilitando a busca de informações, logo deve ser de fácil acesso. Os links do mapa devem coincidir com o título das páginas que serão carregadas. Associado ao mapa do site também deve existir um mapa de imagens’.

Vamos fazer o cruzamento deste trecho de código com a árvore apresentada na Figura 8:

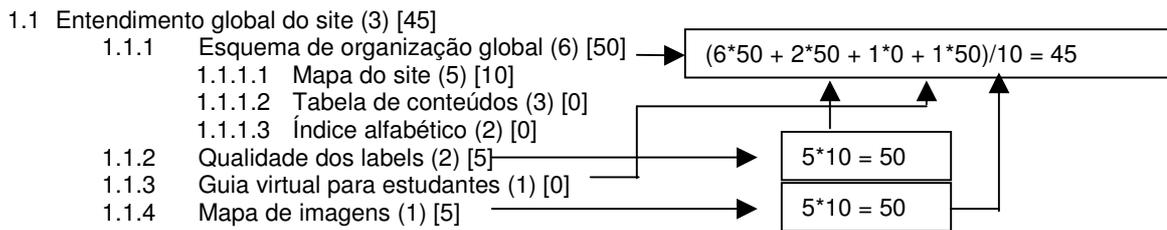


Figura 8 - Cruzamento dos requisitos iniciais com a árvore.

Foram associados os pesos entre parênteses, e as notas dos atributos entre colchetes. A metodologia apenas recebe as notas dos atributos, calculando a das sub-características e das características. A nota 1.1.3 foi cinco porque o trecho do documento não define os labels a serem utilizados no mapa, apenas é feita uma alusão que os mesmos devem coincidir com o título das páginas. Logo, o respectivo mapa pode não ser intuitivo, dificultando a localização de funcionalidades por parte dos usuários. O mesmo ocorre com 1.1.4, onde se fala da existência do mapa de imagens, mas não se entra em detalhes. As notas zero dizem respeito aos atributos de qualidade não especificados.

5 Conclusões e trabalhos futuros

A proposta aqui apresentada pode ser muito útil se empregada em empresas que desenvolvem sistema para Web. É certo que a metodologia apenas se aplica nos casos em que existe um documento de requisitos bem definido. Caso isso não faça parte do processo de desenvolvimento, esse tempo adicional para elaborá-lo será compensado por uma maior qualidade final do produto, além de uma documentação mais completa do software como um todo.

Esse trabalho está centrado nas características de qualidade de artefatos WEB. No entanto, outros indicadores como custo, ou relação custo-benefício podem ser de interesse para avaliação e comparação de sistemas em geral. Esse levantamento do custo obviamente que apenas será viável devido a existência de um documento de requisitos detalhado.

Um trabalho a ser desenvolvido é a especificação de uma árvore de características e atributos de qualidade genérica, pois esta seria apropriada para todo e qualquer domínio de aplicação, bem como para diferentes perfis de usuários. Seria necessário apenas estender essa árvore genérica, com atributos que fossem específicos ao domínio em questão.

Estamos desenvolvendo uma segunda versão de um ambiente integrado, chamado WebQEM_Tool, capaz de suportar a automatização de métricas Web. A idéia final é desenvolver uma aplicação colaborativa, centrada na Web, onde seja possível integrar o documento de requisitos em um formato pré-determinado e a árvore de características e atributos de qualidade daquele domínio da aplicação em questão. Assim, o resultado da avaliação seria apresentado de forma eficiente e automática, antes mesmo da existência de qualquer produto intermediário.

6 Agradecimentos

Este trabalho é parcialmente suportado pelo Programa CYTED, no projeto WEST (Web-Oriented Software Technology).

7 Referências Bibliográficas

- [1] Clemen, P. T. Making Hard Decisions: An Introduction to Decision Analysis. Hardcover 2nd edition, 1997.
- [2] Fenton, N.E., Pfleeger, S.L., 1997, Software Metrics: a Rigorous and Practical Approach, 2nd Ed., PWS Publishing Company.
- [3] Furano, F.; Orsini, R.; Celentano, A., 1997, Museum-on-demand: dynamic management of resources in World Wide Web museums, Hypertext and Hypermedia: Products, Tools, Methods (H2PTM'97), Hermes Editorial, Paris, Fr, V. 1, N° 2-3-4/97 pp. 115,124.
- [4] Garzotto, F.; Mainetti, L.; Paolini, P., 1997, *Designing Modal Hypermedia Applications*, The Eighth ACM Conference on Hypertext, Southampton, England, pp 38-47.
- [5] Ginige, A., Murugesan, S., 2001, *Web Engineering: An Introduction*, IEEE Multimedia, V8, N°1, pp.14-18.
- [6] IEEE Std 1061-1992, *IEEE Standard for a Software Quality Metrics Methodology*, IEEE Computer Society Press.
- [7] IEEE Std 830-1998, *IEEE Recommended Practice for Software Requirements Specifications*.
- [8] ISO/IEC 14598-5:1998 *International Standard, Information technology -- Software product evaluation -- Part 5: Process for evaluators*.
- [9] ISO/IEC 9126-1:2001, *Software engineering -- Product quality -- Part 1: Quality model*
- [10] Lafuente, G.; González Rodríguez, J.; Olsina, L.; 2001, *Automatizando Métricas Web*, Anais 4to Encuentro para la Calidad en las Tecnologías de la Información y las Comunicaciones, (QUATIC), Lisbon, Portugal, pp. 17-24
- [11] Leite, J, Gustavo Rossi, Vanesa Maiorana, Federico Balaguer, Gladys Kaplan, Graciela Hadad and Alejandro Oliveros, 1997, *Enhancing a Requirements Baseline with Scenarios*, Requirements Engineering Journal: Vol. 2, N.4, pp. 184 - 198, Springer-Verlag, London.
- [12] Olsina, L.; Lafuente, G.J.; Godoy, D; Rossi, G.; 1999, *Assessing the Quality of Academic Websites: a Case Study*, New Review of Hypermedia and Multimedia (NRHM) Journal, Taylor Graham Publishers, UK, Vol. 5, pp. 81-103.
- [13] Olsina, L.; Lafuente, G.J.; Rossi, G., 2000, *E-commerce Site Evaluation: a Case Study*, LNCS 1875 of Springer, 1st International Conference on Electronic Commerce and Web Technologies, EC-Web 2000, London, UK, pp. 239-252.
- [14] Olsina, L., 2000, *Metodologia Quantitativa para a Avaliação e Comparação de Qualidade de Sites Web*, tese de doutorado defendida em abril, Facultad de Ciências Exactas, UNLP, La Plata, Argentina.