Brechó-ABC: Uma Abordagem Integrada para Avaliação, Busca e Categorização de Componentes de Software

Ronaldo Rodrigues Raposo Junior

Projeto Final de Curso submetido ao Departamento de Ciência da Computação do Instituto de Matemática da Universidade Federal do Rio de Janeiro como parte dos requisitos necessários para obtenção do grau de Bacharel em Informática.

Apresentado por:	
Aprovado por:	Ronaldo Rodrigues Raposo Junior
	Prof ^a Cláudia Maria Lima Werner, D.Sc. (Presidente)
	Marco Alexandre de Macedo Lopes, M.Sc. (Co-orientador)
	Prof ^a Vanessa Braganholo Murta, D.Sc.
	Prof. Leonardo Gresta Paulino Murta, D.Sc.

RIO DE JANEIRO, RJ - BRASIL SETEMBRO DE 2007

Aos meus pais e irmãos.

AGRADECIMENTOS

Agradeço ao Pai Maior pelas condições que tive ao longo da minha vida. Também agradeço a Ele por ter me dado forças e me protegido durante toda esta caminhada.

Agradeço aos meus pais e irmãos. Eles me dão todo o carinho, amor e motivação que preciso para continuar lutando. Agradeço ao meu pai e a minha mãe em especial, pois eles são meus exemplos de dedicação e caridade.

Agradeço à minha orientadora Cláudia Werner, que me deu todo o apoio para poder realizar este projeto. Mesmo numa época em que sua vida estava completamente ocupada, ela não deixou de ter paciência e dedicação com a minha orientação.

Agradeço ao meu orientador Marco Lopes. Ele teve a paciência e a calma para esclarecer todas as dúvidas. Com ele, aprendi a ser mais minucioso e objetivo com as palavras.

Agradeço aos colegas do laboratório de Engenharia de Software, que me ajudaram de alguma forma com o trabalho. Anderson, Danny, Paula, João Gustavo, Rodrigo, Cláudia Susie, obrigado por tudo.

Agradeço a todos os colegas de faculdade pelas noites mal durmidas, pelos trabalhos nos finais de semana e toda a amizade que tenho por estas pessoas. Em especial, agradeço ao Bruno, Fernando, Diogo, Thatiana, Luiz Carlos, Alan, Antônio e Felipe Dias.

Agradeço a todos os professores da UFRJ todo o carinho e compreensão que tive nestes anos. Em especial aos mestres Maria Luiza, Raimundo, João Carlos, Juarez e Vinicius.

Agradeço ao pessoal da Eletrobrás, onde estive por dois anos e meio. Aprendi muito nesta empresa e serei eternamente grato a ela. Em especial ao sr. Eudes, sr. Álvaro, Jairo, Fernando e ao pessoal do Eletrocraques.

Agradeço aos meus amigos e colegas, que ficaram por várias vezes me perguntando por que não saí ou não fui jogar bola, mas compreenderam e aceitaram o motivo. Em especial, a galera do Fominhas, aos meus vizinhos, e uma menina que me faz muito feliz ultimamente. Angel, te adoro muito!

Agradeço a todos aqueles que, de alguma forma, fazem parte da minha vida.

E finalmente, agradeço àqueles que já nos deixaram, mas que fizeram história aqui na Terra. Sinto saudades e lembranças dos tempos bons que passei com estas pessoas. Em especial aos meus avôs, minhas avós e a minha madrinha. Vô Wilson, eu consegui!

RESUMO

Brechó-ABC: Uma Abordagem Integrada para Avaliação, Busca e Categorização de Componentes de Software

Ronaldo Rodrigues Raposo Junior

Orientadores: Cláudia Maria Lima Werner e Marco Alexandre de Macedo Lopes

Este trabalho apresenta mecanismos integrados de avaliação, busca e categorização no contexto de uma biblioteca de componentes de software. As avaliações são feitas de modo simples e objetivo, agregando informações sobre o componente. A flexibilização e o refinamento de mecanismos de pesquisa visam a permitir uma maior precisão nos resultados de busca. Sugestões são dadas para expandir o conjunto de categorias de modo organizado. Estes mecanismos foram implementados de modo a beneficiar produtores e consumidores no processo de seleção e recuperação de componentes. Assim, espera-se que os componentes sejam encontrados e reutilizados com mais facilidade.

ABSTRACT

Brechó-ABC: An Integrated Mechanism for Evaluation, Search and Categorization of Software Components

Ronaldo Rodrigues Raposo Junior

Advisors: Cláudia Maria Lima Werner and Marco Alexandre de

Macedo Lopes

This work presents integrated mechanisms for evaluation, search and categorization in the context of a software component library. Evaluations are made in a simple and objective way, and they aggregate information about the component. Flexibility and refinement of search mechanisms intend to permit a greater level of precision in search results. Suggestions are given to extend the set of categories in an organized way. These mechanisms were developed to improve the selection and retrieval of components. So, we expect that components are easily found and reused.

SUMÁRIO

CAPÍTUI	LO 1 – INTRODUÇÃO	1
1.1.	MOTIVAÇÃO	
1.2.	Objetivo	
1.3.	Organização do Trabalho	
CAPÍTUI	LO 2 – REVISÃO DA LITERATURA	5
2.1.	AVALIAÇÃO	5
2.2.	BUSCA	
2.3.	CATEGORIZAÇÃO	12
2.4.	CONSIDERAÇÕES SOBRE O CAPÍTULO	15
CAPÍTUI	LO 3 – A ABORDAGEM BRECHÓ-ABC	17
3.1.	CENÁRIO DE UTILIZAÇÃO	18
3.2.	DEFINIÇÃO DOS REQUISITOS E PROPOSTA	19
3.2.1	3	
3.2.2		
3.2.3		
3.3.	ARQUITETURA E PRINCIPAIS FUNCIONALIDADES	
3.3.1		22
	3.1.1. Avaliação dos Usuários	
	3.1.2. Visualização das Avaliações	
	2. Mecanismo de Busca	
	3.2.2. Busca Sintática	
	B. Mecanismo de Categorização	
	3.3.1. Sugestão de Categorias	
3.	3.3.2. Organização de Categorias	25
3.4.	CONSIDERAÇÕES SOBRE O CAPÍTULO	26
CAPÍTUI	LO 4 – IMPLEMENTAÇÃO	27
4.1.	O Brechó-ABC e a biblioteca Brechó	
4.1. 4.2.		
4.2. 4.2.1	MECANISMO DE AVALIAÇÃO!!. Detalhes de Implementação!!	
4.2.1	* *	
	2.2.1. Avaliação dos Usuários	
	2.2.2. Visualização das Avaliações	
4.3.	MECANISMO DE BUSCA	
4.3.1		
4.3.2	2. Telas e Utilização	
	3.2.1. Flexibilização e Refinamento das Buscas	
•••	3.2.2. Busca Sintática	
4.4.	MECANISMO DE CATEGORIZAÇÃO	
4.4.1	T	
4.4.2		
	4.2.1. Sugestão de Categorias	
4.5.	EXEMPLOS DE UTILIZAÇÃO	
4.5.1		
4.5.2	2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2	
4.5.3		
4.5.4	1	
4.6.	CONSIDERAÇÕES SOBRE O CAPÍTULO	
CAPÍTIII	LO 5 – CONCLUSÃO	(2)
CALITUL	LO 5 – CONCLUSAO	
5.1.	Considerações Finais	

5.4. IRABA	ALHOS FUTUROS
	AÇÕES

SUMÁRIO DE FIGURAS

FIGURA 1.1: O PROCESSO "LOCALIZAÇAO-COMPREENSAO-MODIFICAÇAO" DE REUTILIZAÇAO DE	
COMPONENTES, ADAPTADO DE (YE, 2001).	
FIGURA 2.1: PROBLEMA DE MAPEAMENTO – NECESSIDADES PARA ARTEFATOS DE SOFTWARE DISPON	
(ADAPTADO DE MITTERMEIR, POZEWAUNIG ET AL. (1998)).	
FIGURA 2.2: O PROCESSO DE RECUPERAÇÃO DE INFORMAÇÃO	
(ADAPTADO DE BAAEZA-YATES E RIBEIRO-NETO, 1999).	14
FIGURA 3.1: ALGUMAS NECESSIDADES DE CONSUMIDORES E PRODUTORES NUMA BIBLIOTECA DE	
COMPONENTES.	
FIGURA 3.2: VISÃO GERAL DA ARQUITETURA DA BRECHÓ-ABC.	
FIGURA 4.1: A PÁGINA INICIAL DA BIBLIOTECA DE COMPONENTES BRECHÓ.	
FIGURA 4.2: O DIAGRAMA DE CLASSES PARA A ABORDAGEM BRECHÓ-ABC.	
FIGURA 4.3: DIAGRAMA CONTENDO A CLASSE EVALUATION E SEUS RELACIONAMENTOS	
FIGURA 4.4: TELA DA BRECHÓ PARA A AVALIAÇÃO DE UMA RELEASE	
FIGURA 4.5: TELA DA BRECHÓ PARA A LISTAGEM DE AVALIAÇÕES DE UMA RELEASE	
FIGURA 4.6: DIAGRAMA CONTENDO A CLASSE RETRIEVALENGINE E O RELACIONAMENTO COM A CL	
COMPONENT.	
FIGURA 4.7: TELA DA BRECHÓ ONDE É MOSTRADA A FILTRAGEM DE COMPONENTES POR CATEGORIA	
FIGURA 4.8: TELA DA BRECHÓ ONDE É MOSTRADA A FILTRAGEM DE COMPONENTES PELOS ELEMENT	
DOCUMENTAÇÃO.	
FIGURA 4.9: TELA DA BRECHÓ ONDE É MOSTRADA A BUSCA SINTÁTICA, COM AS SUGESTÕES DE PAL	AVRAS
GRAFADAS CORRETAMENTE	
FIGURA 4.10: SEQÜÊNCIA DE AÇÕES QUE O MÓDULO DE SUGESTÃO DE PALAVRAS EXECUTA NA BRE	
FIGURA 4.11: DIAGRAMA CONTENDO A CLASSE SUGGESTIONS E SEUS RELACIONAMENTOS	
FIGURA 4.12: TELA DA BRECHÓ ONDE MOSTRA O INÍCIO DO MÓDULO DE SUGESTÃO DE CATEGORIAS	44
FIGURA 4.13: TELA DA BRECHÓ ONDE O USUÁRIO SUGERE AS CATEGORIAS MANUALMENTE OU RECO	ORRE À
Web	45
FIGURA 4.14: TELA DA BRECHÓ ONDE MOSTRA AS CATEGORIAS MAIS RELEVANTES VINDAS DA WEB	45
FIGURA 4.15: TELA INICIAL COM A OPÇÃO DE ADMINISTRAÇÃO DE CATEGORIAS E SUGESTÕES	
SELECIONADA.	46
FIGURA 4.16: TELA DA BRECHÓ ONDE O ADMINISTRADOR ESCOLHE UM NÍVEL DE SIMILARIDADE EN	TRE AS
CATEGORIAS E AS SUGESTÕES.	47
FIGURA 4.17: TELA DA BRECHÓ ONDE SÃO RETORNADAS AS RELAÇÕES DE SIMILARIDADE ENTRE	
SUGESTÕES NESTE CASO	47
FIGURA 4.18: TELA INICIAL DA BRECHÓ DE CADASTRO DE COMPONENTE	48
FIGURA 4.19: TELA COM O PREENCHIMENTO DOS ELEMENTOS DA DOCUMENTAÇÃO ASSOCIADOS À	
CATEGORIA.	49
FIGURA 4.20: TELA QUE PODE LEVAR O USUÁRIO A SUGERIR OU NÃO NOMES DE CATEGORIAS PARA A	A
BIBLIOTECA.	50
FIGURA 4.21: TELA COM O PREENCHIMENTO DAS SUGESTÕES DE CATEGORIAS	50
FIGURA 4.22: TELA COM AS SUGESTÕES PESQUISADAS PELA WEB E RETORNADAS PARA O PRODUTOR	R51
FIGURA 4.23: TELA QUE REPRESENTA O FIM DO CADASTRO DE UM COMPONENTE NA BRECHÓ	51
FIGURA 4.24: TELA QUE REPRESENTA OS COMPONENTES NA BRECHÓ FILTRADOS PELA CATEGORIA	
"CIÊNCIA".	52
FIGURA 4.25: TELA QUE REPRESENTA O REFINAMENTO DE COMPONENTES PELA TECNOLOGIA "EJB"	53
FIGURA 4.26: TELA QUE EXIBE OS DETALHES DO COMPONENTE "EVOLUÇÃO"	53
FIGURA 4.27: TELA QUE EXIBE A PALAVRA-CHAVE "EVOLUÇÃO" SENDO GRAFADA DE MODO INCORI	
FIGURA 4.28: TELA QUE EXIBE SUGESTÕES DE PALAVRAS COM GRAFIA SIMILAR	
FIGURA 4.29: TELA QUE EXIBE AGORA COMPONENTES FILTRADOS PELA PALAVRA-CHAVE "EVOLUÇ.	ÃO"55
FIGURA 4.30: TELA QUE EXIBE COMPONENTES FILTRADOS PELA PALAVRA-CHAVE "LABORATORIU".	56
FIGURA 4.31: TELA QUE LISTA AS RELEASES DE "EVOLUÇÃO", COM OS LINKS PARA AVALIAÇÃO E EX	
DE AVALIAÇÕES PARA A RELEASE 0.1 DO COMPONENTE.	-
FIGURA 4.32: TELA ONDE OS USUÁRIOS EFETUAM AS AVALIAÇÕES.	
FIGURA 4.33: TELA ONDE OS USUÁRIOS VISUALIZAM AS AVALIAÇÕES	
FIGURA 4.34: TELA ONDE O SUPERVISOR SELECIONA O ÍNDICE DE SIMILARIDADE ENTRE CATEGORIA	
SUGESTÕES.	

FIGURA 4.35: TELA ONDE O SUPERVISOR CONFERE SUGESTÕES E CATEGORIAS COM ALGUM ÍNDICE DE	
SIMILARIDADE.	.59
FIGURA 4.36: TELA ONDE O ADMINISTRADOR CRIA UMA NOVA CATEGORIA A PARTIR DE UMA SUGESTÃO	
DADA PELOS USUÁRIOS.	.60

SUMÁRIO DE TABELAS

TABELA 2.1: CARACTERÍSTICAS DE QUALIDADE NOS MODELOS DE BOEHM, MCCALL, FURPS, DROME	YΕ
DA NORMA ISO 9126. (RETIRADO DE RAWASHDEH E MATALKAH (2006))	7
TABELA 2.2: COMPARATIVO DAS TÉCNICAS DE BUSCA. (ADAPTADO DE (ZHANG, 2000))	10
TABELA 5.1: COMPARATIVO DAS ABORDAGENS DE CATEGORIZAÇÃO CITADAS NA LITERATURA COM O	
Brechó-ABC.	63

Capítulo 1 Introdução

1.1. Motivação

A Reutilização de Software é o uso de software previamente desenvolvido ou a utilização de conhecimento referente a este software no desenvolvimento de novas aplicações (FRAKES, 2007). A reutilização possibilita o aumento da produtividade, reduz custos de desenvolvimento e ajuda no desenvolvimento de sistemas mais confiáveis, baratos e complexos, possibilitando minimizar os atrasos na entrega destes sistemas (KIM e STOHR, 1997) (WERNER e BRAGA, 2000) (FRAKES e KANG, 2005).

Conforme pesquisas relacionadas à reutilização de software (FRAKES e KANG, 2005), a técnica de Desenvolvimento Baseado em Componentes (DBC) vem crescendo como um novo paradigma de desenvolvimento de software. O DBC envolve a construção de sistemas usando componentes pré-empacotados, que são, geralmente, unidades de código executáveis que provêm um conjunto de funcionalidades através de uma interface específica. Ele também envolve a reutilização de *frameworks* de aplicação que provêm o arcabouço necessário para agrupar os componentes, formando a aplicação. Os componentes podem ser desenvolvidos no contexto de uma organização ou encontrados em bibliotecas (RAVICHANDRAN e ROTHENBERGER, 2003).

Neste sentido, a complexidade no DBC pode ser reduzida, assim como os custos, através da reutilização de componentes previamente avaliados. O desenvolvimento de novas soluções com a combinação de componentes pode aumentar a qualidade das aplicações e a velocidade de desenvolvimento, fazendo com que o componente chegue rapidamente ao mercado (BRAGA *et al.*, 2006). No entanto, existem algumas dificuldades relacionadas ao DBC. O desenvolvimento de um mercado de componentes de software ainda não é uma realidade amadurecida, sobretudo no Brasil.

Para (YE, 2001), o processo de reutilização consiste em três passos: (i) localização, (ii) compreensão e (iii) modificação, como mostra a Figura 1.1. Os desenvolvedores precisam localizar os componentes que sejam potencialmente reutilizáveis, compreender suas funcionalidades e utilização e fazer as modificações

necessárias, caso os componentes não satisfaçam por completo suas necessidades (FISCHER et al., 1991).

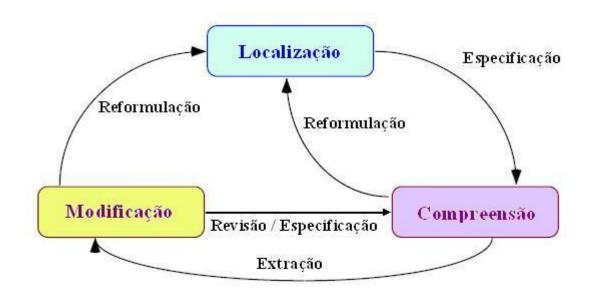


Figura 1.1: O processo "localização-compreensão-modificação" de reutilização de componentes, adaptado de (YE, 2001).

Na ausência de um mercado consumidor amadurecido, as diferentes tecnologias de desenvolvimento de componentes têm pouca utilidade (LUCRÉDIO *et al.*, 2004). Dentre os principais inibidores a este amadurecimento, podemos destacar como cruciais a baixa disponibilidade de componentes de software e a imaturidade dos canais de distribuição (BASS *et al.*, 2000). Neste cenário, um elemento fundamental é o local onde os componentes podem ser encontrados (por exemplo, uma biblioteca de componentes). Uma biblioteca de componentes deve prover um local adequado para publicação, armazenamento, busca e recuperação de componentes, de forma que possam ser aplicados no desenvolvimento de novos sistemas (WERNER e BRAGA, 2000) (YE, 2001).

Desta forma, para que o mercado de componentes amadureça, alguns fatores de fundamental importância incluem, a partir de bibliotecas disponibilizadas na Web, o aumento da oferta de componentes de boa qualidade e que sejam facilmente encontrados, compreendidos, adquiridos e reutilizados.

De maneira a atender as necessidades expostas, as bibliotecas de componentes necessitam prover mecanismos eficientes para: (i) publicação; (ii) documentação/categorização; (iii) armazenamento/empacotamento; (iv) pesquisa; (v)

recuperação; (vi) aquisição e (vii) avaliação/qualidade de componentes de software, entre outras atividades.

Neste contexto, introduzimos o termo "produtor" para representar o usuário que desenvolve ou faz modificações em um componente já produzido por terceiros. O produtor publica o componente em uma biblioteca e espera que outros usuários o reutilizem. Já o termo "consumidor" representa o usuário que procura por um componente que esteja de acordo com as suas necessidades. Caso encontre o componente na biblioteca, o consumidor o adquirirá e, assim, o reutilizará.

1.2. Objetivo

Este trabalho destaca algumas questões sobre três atividades específicas citadas anteriormente: a (i) avaliação, a (ii) busca e a (iii) categorização de componentes de software. Assim, será apresentada uma abordagem integrada que atenda cada uma destas atividades no contexto de uma biblioteca de componentes. Cada mecanismo atuará de modo a ajudar os produtores a classificar os componentes e fazer com que eles tenham maiores possibilidades de serem encontrados. Além disso, os mecanismos devem auxiliar os comsumidores a buscar e avaliar o componente de modo eficiente. Deste modo, tanto os produtores quanto consumidores seriam beneficiados por esta proposta de solução integrada.

O mecanismo de avaliação deve permitir aos usuários fazer os devidos comentários e críticas sobre o componente de forma simples e, também, permitir a visualização destas avaliações.

O mecanismo de busca deve auxiliar os consumidores a procurar um componente de forma rápida e fornecendo a eles vários modos de pesquisa. Além disso, deve ainda alertá-los sobre possíveis erros cometidos nas pesquisas.

O mecanismo de categorização deve apoiar os produtores a fazer sugestões de categorias, na tentativa de classificar corretamente os componentes na biblioteca, e também ajudar o administrador da biblioteca a manter o conjunto de categorias coerente e organizado.

Além das propostas citadas, este trabalho apresenta a implementação destes mecanismos no contexto de uma biblioteca de componentes e serviços de software, mostrando como se utiliza cada um destes mecanismos.

1.3. Organização do Trabalho

Este trabalho está dividido em cinco capítulos. Após esta breve introdução, que faz parte do Capítulo 1, serão expostos no Capítulo 2 os principais conceitos relativos à avaliação, busca e categorização que foram encontrados e pesquisados em outros trabalhos na literatura. No Capítulo 3, será apresentada a abordagem proposta para o desenvolvimento de mecanismos de avaliação, busca e categorização de componentes no contexto de uma biblioteca de componentes. No Capítulo 4, serão apresentadas a implementação e a utilização destes mecanismos na biblioteca de componentes e serviços Brechó (WERNER *et al.*, 2007). Finalmente, no Capítulo 5, serão apresentadas as considerações finais deste trabalho, incluindo contribuições, limitações e sugestões de trabalhos futuros.

Capítulo 2 Revisão da Literatura

Este capítulo apresenta algumas abordagens encontradas na literatura, relacionadas à avaliação, busca e categorização de componentes. Estes trabalhos reunidos formam a fundamentação teórica desse projeto final.

O capítulo está organizado da seguinte forma: a avaliação de componentes é tratada na Seção 2.1, descrevendo sua relação com a área de qualidade de software, os modelos e as dificuldades existentes nesta área. A Seção 2.2 está relacionada à busca de componentes, onde o termo é definido, além de descrever as diferentes técnicas, desafios e problemas encontrados em trabalhos da literatura. A Seção 2.3 apresenta os conceitos relacionados à categorização de componentes, assim como algumas abordagens referentes ao assunto. Finalmente, a Seção 2.4 apresenta algumas considerações sobre os tópicos abordados.

2.1. Avaliação

De acordo com PRESSMAN (2002), a avaliação de componentes é uma das atividades associadas ao DBC que tem como responsabilidade garantir a qualificação de um artefato, ou seja, que este realizará as funções que se propõe, irá se adequar ao estilo de arquitetura especificado para o sistema e apresentará as características de qualidade (por exemplo, confiabilidade, eficiência e maturidade) necessárias para a aplicação. A norma ISO-9126 (1991) define a qualidade de um software como um conjunto de atributos de um produto cujas características estejam bem descritas e avaliadas. Segundo RAWASHDEH e MATALKAH (2006), qualidade é uma medida subjetiva e funcional usada para especificar a satisfação com um produto, ou quão bem ele realiza suas operações, em comparação com outros similares.

NYMAN e NÅLS (2004) defendem que um processo eficiente de reutilização acontece com uma adequada e precisa descrição dos componentes. Quando estes componentes passam a ser comercializados, o consumidor destes quer mais informações do que as contidas na descrição para verificar se o artefato preenche os requisitos que ele necessita. Os autores apresentam três abordagens para assegurar a qualidade neste

caso: (i) certificação do produto, (ii) processo de auditagem e (iii) satisfação dos usuários.

PRESSMAN (2002) afirma ainda que somente a descrição da interface de um componente não oferece toda a informação necessária para determinar se de fato pode haver uma reutilização efetiva. O autor cita que fatores como características de segurança, requisitos de execução e manipulação de exceções devem ser considerados durante o processo de qualificação. Desta forma, uma análise do componente poderia ser feita para determinar se este funciona sob essas condições. No entanto, existe uma dificuldade adicional para definir a qualidade de um componente de terceiros, já que a única informação disponível pode ser somente a especificação da interface.

Existem na literatura vários modelos de qualidade de software. RAWASHDEH e MATALKAH (2006) apresentam alguns destes: o proposto por McCALL em 1976 (1994), o de BOEHM (1978), o do FURPS, proposto por GRADY (1987), a norma da ISO-9126 (1991) e o de DROMEY (1995). Os autores lembram que estes avaliam software em geral, e que nenhum deles é totalmente dedicado a componentes. A Tabela 2.1 apresenta as características de qualidade de software presentes em cada um dos modelos citados acima.

Atributo de Qualidade	McCall	Boehm	FURPS	ISO 9126	Dromey
Testabilidade	X	X		X	
Corretude	X				
Eficiência	X	X	X	X	X
Inteligibilidade		X		X	
Confiança	X	X	X	X	X
Flexibilidade	X		X		
Funcionalidade			X	X	X
Engenharia Humana		X			
Integridade	X			X	
Interoperabilidade	X			X	
Maturidade					Х
Importância	X	X	X	X	х
Manutenibilidade		X			
Portabilidade	X	X		X	Х
Reusabilidade	X				х
Usabilidade	X		X	X	х

Tabela 2.1: Características de qualidade nos modelos de Boehm, McCall, FURPS, Dromey e da norma ISO 9126. (Retirado de RAWASHDEH e MATALKAH (2006)).

Cada um dos modelos apresentados acima possui um conjunto diferente de métricas para qualificar um software. Pesquisas em modelos de qualidade especificamente desenvolvidos para componentes estão sendo realizadas. Um exemplo de modelo para o processo de componentes de software é encontrado na norma ISO-15504 (1999). Algumas das práticas base em um dos processos (denominado "processo de aceitação do cliente) incluem a avaliação do componente de acordo com os requisitos especificados e a aceitação do componente caso todas as condições estiverem satisfeitas. Outro exemplo é encontrado em GOULÃO *et al.* (2004), onde algumas medidas para adaptar a norma ISO-9126 para o caso de avaliação de componentes são citadas. ALVARO *et al.* (2005), a partir do trabalho de GOULÃO *et al.* (2004), mostram um modelo de qualidade já adaptado da ISO-9126. Os autores dos dois últimos trabalhos concluem que ainda é necessária uma evolução maior destes modelos.

2.2. Busca

O termo buscar pode ser definido como o ato de examinar, efetuar uma procura, realizar uma investigação a partir de algum critério pré-estabelecido para se encontrar determinado tipo de informação (AURÉLIO, 2005). Numa biblioteca de componentes, a busca é o ato de procurar por um componente cujos resultados formam uma lista dos componentes de maior relevância e que satisfazem os critérios estabelecidos pelos usuários. A busca é uma das tarefas do processo de recuperação. Este processo tem como objetivo transformar as necessidades do usuário em uma pergunta; ranquear os componentes de acordo com algum nível de relevância; fazer a busca para obter os componentes; acessar as informações sobre o componente de modo mais detalhado; e iniciar o feedback com o usuário, coletando informações para serem utilizadas em novas buscas. Além disso, a recuperação pode resultar na reutilização do mesmo (ZHANG, 2000).

Os mecanismos de busca e recuperação de componentes de software desempenham uma função importante para repositórios voltados para o mercado consumidor de componentes. Sem estes, todo o empenho e investimento realizado em ferramentas, métodos, modelos e tecnologias para o desenvolvimento de componentes reutilizáveis não resultará em nenhum benefício (LUCRÉDIO *et al.* 2004).

Conforme ZHANG (2000), ainda que as estratégias para a busca de informação sejam diferentes umas das outras, muitas delas seguem as seguintes técnicas: (i) por palavras-chave, (ii) por documentos, (iii) baseada em esquemas de classificação e (iv) baseada em hipertexto. Segue uma breve descrição destes tipos de busca citados pelo autor, além de outros comentários.

A busca por palavras-chave é uma técnica simples e utilizada em diversos mecanismos de busca. Consiste em digitar uma ou um conjunto de palavras, que servirá de base para a procura de componentes que possuem estes termos em suas descrições. Um dos problemas desta abordagem é o fato de que os produtores do componente devem igualmente fornecer palavras-chave apropriadas para serem indexadas para o momento da procura. De acordo com FANCHAO et al. (2006) e SUGUMARAN e STOREY (2003), esta técnica é bem definida e simples de entender. No entanto, a busca por palavras-chave pode trazer resultados não esperados e deixa de fora outros componentes que seriam importantes, pois somente palavras-chave são utilizadas na pesquisa.

Já a busca por documentos tem como finalidade, após indexar as palavras, ordenar os documentos de forma automática. Este tipo de busca possui a vantagem de que documentos relevantes têm maiores chances de aparecer nos resultados. Por outro lado, se há alguma palavra relevante fora de contexto, ou se palavras (como por exemplo, preposições) que aparecem em muitos documentos não forem removidas ou não tiverem um peso menor no cálculo da relevância, os resultados podem ser menos precisos.

O esquema de busca baseado em classificação utiliza categorias para a busca de componentes. Ele possui a vantagem de prover uma clareza maior ao atribuir sistematicamente os componentes a categorias de acordo com um conjunto de princípios. Assim, quando o usuário realizar a busca, os artefatos podem ser mais fáceis de serem encontrados, bastando somente navegar entre as categorias. Esta técnica é utilizada por diversas bibliotecas. Um dos problemas desta técnica é a ambigüidade ou má interpretação de algum nome de categoria existente, já que existem palavras que podem assumir diversos significados. FANCHAO et al. (2006) destacam que uma grande quantidade de categorias pode resultar em problemas de controle e de organização das mesmas, tornando esta técnica ineficiente. De acordo com SUGUMARAN e STOREY (2003), a classificação dos componentes em categorias é útil se o esquema de taxonomia for coerente e explícito. No entanto, os autores ressaltam que apenas esta técnica não possibilita a modificação da busca inicial com outras informações para potencializar a busca.

Por fim, a busca baseada em hipertexto permite aos usuários navegar através de nós e *links*, que contêm dados associados aos componentes. Com relação às outras três técnicas, este esquema de recuperação é não linear, pois o usuário navega através dos *links* sem a obrigação de seguir uma ordem definida. Ele também é flexível, porque esta técnica permite ao usuário especificar os requisitos do componente em conjunto com outras informações. No entanto, dada a estrutura de hipertexto, os usuários podem se perder nos detalhes da informação que acessam.

ZHANG (2000) destaca que as técnicas de busca de componentes estão em processo de evolução. Cada técnica possui características que podem ser combinadas para enriquecer o processo de busca. A Tabela 2.2 sumariza as técnicas de busca abordadas, classificando-as de acordo com quatro características (organização da informação, indexação, tipos de coleções de dados e flexibilidade). Vale destacar que as técnicas citadas podem ser utilizadas em conjunto, com o objetivo de aumentar a

precisão dos resultados retornados. A técnica de hipertexto, por exemplo, pode ser complementada com uma das outras três técnicas, já que ela organiza as informações de forma não-linear, enquanto as outras organizam de forma linear. A técnica por palavrachave e a baseada em classificação em conjunto introduzem informação semântica ao processo, pois as categorias e as palavras-chave agregam informações relevantes para a busca.

Técnicas de	Busca por	Busca em	Busca Baseada em	Busca Baseada	
Busca	Palavra-Chave	Documentos	Classificação	em Hipertexto	
Propriedades					
Organização da	Linear	Linear	Linear	Não-Linear	
Informação					
	Manual	Automática	Manual /	-	
			Automática		
Indexação					
j	Vocabulário	Vocabulário	Informação	-	
	Controlado	Não	Semântica		
		Controlado			
Tipos de	Coleção	Documentos	Qualquer Coleção	Qualquer Coleção	
Coleções de	Especificada		Específica para um		
Dados	por Palavras-		Domínio		
	Chave				
Flexibilidade	As técnicas de busca são flexíveis e, quando a coleção de dados se expande,				
	elas conseguem se acomodar sem problemas.				

Tabela 2.2: Comparativo das técnicas de busca. (Adaptado de (ZHANG, 2000)).

Existem outras técnicas de busca, como *signature matching* e *behavioral matching*. A técnica *signature matching* leva em consideração os argumentos definidos nos componentes. O usuário especifica o tipo e o número destes argumentos na busca. Esta busca funciona no nível de método ou no nível de classe. A técnica *behavioral matching* leva em conta o comportamento dos objetos e das classes de um componente. Estas classes são testadas e aquelas que tiverem comportamento semelhante ao esperado pelo usuário são recuperadas. As duas técnicas são ineficientes, pois elas não

consideram informações sobre o domínio. Além disso, os usuários têm dificuldades em mapear os requisitos tanto para os argumentos quanto para os comportamentos. (SUGUMARAM e STOREY, 2003).

MITTERMEIR, POZEWAUNIG *et al.* (1998) e WERNER e BRAGA (2000) ressaltam que, em geral, um usuário que deseja encontrar um componente não está interessado na sua representação ou como determinada funcionalidade foi implementada, mas sim o quê esta funcionalidade provê e qual a sua utilidade. O usuário não se interessa pelo algoritmo utilizado e nem pelo número de classes que o componente possui. Ele deseja saber quais funcionalidades o artefato possui, além de características de segurança, portabilidade, entre outras. Por conta disto, esta diferença pode acarretar em resultados insatisfatórios de busca. A Figura 2.1 mostra o problema das visões diferentes que o consumidor e o produtor de componentes têm. O usuário possui a seguinte visão: ele tem um problema, faz a conceituação da solução e a representa, baseando-se nas funcionalidades que irão resolver o problema. Já o produtor descreve como se comportam as funcionalidades do componente. Como as visões são diferentes, os resultados da busca podem não corresponder aos requisitos do usuário.

De acordo com GUO e LUQI (2000), para um bom funcionamento de um repositório, dentre outros fatores, é importante que o mecanismo de busca implementado seja automatizado. Os autores ainda afirmam que, qualquer que seja a técnica de busca a ser utilizada, a biblioteca deve oferecer ao usuário uma maneira rápida de encontrar o componente de acordo com o que ele precisa, evitando atrasos e frustrações.

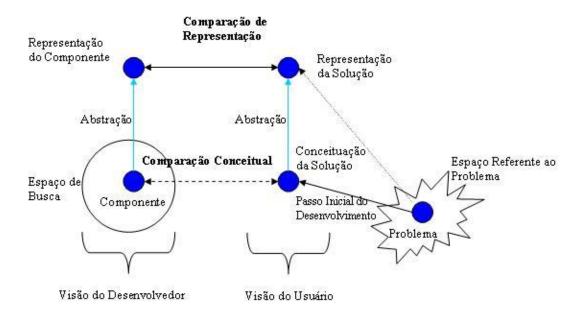


Figura 2.1: Problema de mapeamento – necessidades para artefatos de software disponíveis (Adaptado de MITTERMEIR, POZEWAUNIG *et al.* (1998)).

2.3. Categorização

Desde a Grécia antiga, o termo categorização, também conhecido como classificação, começou a ser explorado. A idéia clássica criada por Platão em seus diálogos e, mais tarde, empregada por Aristóteles num contexto filosófico, era de analisar as diferenças entre classes e objetos e, dessa forma, reunir objetos que possuíam certa semelhança entre si em grupos que fossem nitidamente definidos (WIKIPEDIA, 2007).

Categorizar consiste em organizar os objetos de um dado universo em grupos ou categorias, com um propósito específico (WIKIPEDIA, 2007). Dado que estes objetos estão agrupados, é possível que fique mais fácil recuperar informação e que sejam mais bem compreendidos estes objetos. Por exemplo, os catálogos de lojas de departamentos são divididos em setores, tais como eletrodomésticos, papelaria, vestuário, entre outros. Se uma televisão fosse anunciada neste catálogo, esta estaria situada no setor de eletrodomésticos, assim como um par de meias estaria na parte de vestuário. É natural classificar objetos para que estes sejam melhor identificados e localizados.

No campo dos sistemas de informação, as tarefas de categorização, sobretudo as de textos, ganharam importância nas últimas décadas (SEBASTIANI, 2002). O autor

afirma que o volume de documentos digitalizados cresceu vertiginosamente e, por esta razão, a necessidade de acesso a estes textos através de modos flexíveis e intuitivos aumentou na mesma proporção. Ele ainda afirma que, devido à disponibilidade de máquinas com maior poder de processamento, aliado ao interesse das pessoas em encontrar soluções para este problema, este tópico se tornou importante na área de Recuperação de Informação.

De acordo com BAAEZA-YATES e RIBEIRO-NETO (1999), a Recuperação de Informação é uma disciplina que lida com a representação, armazenamento, organização e acesso aos itens de informação, com o objetivo de prover ao usuário facilidade de acesso à informação no qual está interessado.

Para os autores, os termos "recuperar informação" e "recuperar dados" não são equivalentes. No caso de se recuperar dados, é retornado um conjunto de objetos a partir de palavras-chave contidas numa pergunta. A recuperação de informação leva também em conta a extração de conteúdo sintático-semântico e a ordenação de documentos de acordo com um nível de relevância previamente definido de maneira a satisfazer os requisitos do usuário. A arquitetura referente ao processo de recuperação de informação é apresentada na Figura 2.2.

A caracterização dos requisitos do usuário não é um problema simples de ser solucionado. Consideremos, por exemplo, a hipótese de um indivíduo ter a necessidade de encontrar todas as páginas da Web que contenham informações sobre organizações não-governamentais no Brasil e que têm por objetivo combater o preconceito. Além disso, para serem relevantes, estes documentos devem incluir informações sobre o número de adesões à associação nos últimos três anos e terem e-mail ou telefone da diretoria. Da forma que estão caracterizadas as necessidades do usuário, o mecanismo que recupera as informações não consegue processá-las diretamente. Por isso, é necessário que o usuário traduza seus requisitos através de uma pergunta para que o sistema possa interpretá-las.

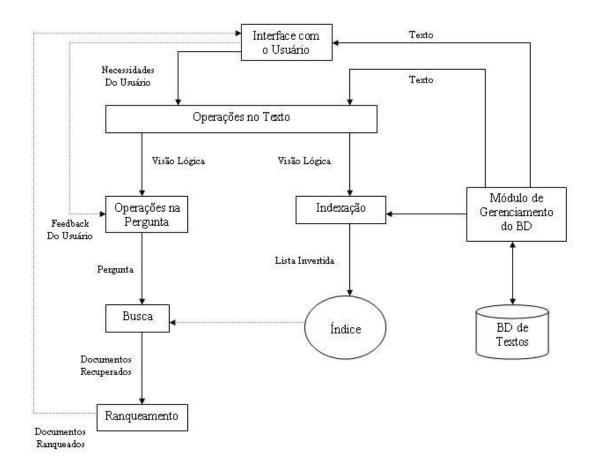


Figura 2.2: O processo de Recuperação de Informação (Adaptado de BAAEZA-YATES e RIBEIRO-NETO, 1999).

Em diversos tópicos de Recuperação de Informação, a categorização de texto é aplicada. Alguns exemplos seriam: filtragem de documentos, geração automática de metadados, indexação de documentos baseados num vocabulário controlado, remoção de ambigüidade de palavras com vários sentidos, categorização hierárquica de páginas da Internet, categorização de componentes, entre outros. Estudos comparativos sobre os métodos de classificação e como estas técnicas são utilizadas em alguns destes temas podem ser encontrados em SEBASTIANI (2002) e YANG e PEDERSEN (1997).

Com relação à classificação de componentes, é possível encontrar na literatura algumas abordagens relacionadas ao tema. Como exemplos, temos o sistema DesCOTS (GRAU *et al.*, 2004), que engloba diversas ferramentas para o apoio à reutilização de componentes. Na ferramenta de taxonomia, um conjunto inicial de categorias é ofertado. Quando uma nova categoria (já previamente definida) é inserida numa árvore hierárquica, são preenchidas algumas especificações, tais como o sistema operacional, o tipo de componente, entre outros, e herdam-se os atributos das categorias "pais". Já a

ferramenta de seleção de componentes utiliza a ferramenta de taxonomia para apoiar os usuários a navegarem pela árvore, a fim de encontrar um componente que satisfaça às suas necessidades.

O método semi-automático de identificação e classificação de componentes SemaCS (SJACHYN e BEUS-DUKIC, 2006) utiliza um esquema de ontologias para definir uma taxonomia de classes num momento inicial. Posteriormente, o mecanismo recebe as palavras-chave resultantes da interação com os usuários, que sugerem novos nomes de categorias para refinar a hierarquia. Além dos usuários, ele procura a partir dos sites do Google e da Wikipédia. Estas informações recebidas são divididas para serem categorizadas em: sistema operacional, infra-estrutura do componente e tipo de componente.

Outra abordagem a ser citada é o repositório digital de componentes educacionais proposto por LALEUF e SPALTER (2001). Os autores definem a estratégia de categorizar os componentes educacionais em três grandes grupos: tecnologias de aplicação, tecnologias de suporte e tecnologias centrais, cada um com diferentes tipos de granularidade, características e público-alvo.

Finalmente, temos a iniciativa de SAMETINGER e KELLER (2002) no que diz respeito à reutilização de componentes de projeto, que auxiliariam no desenvolvimento de grandes sistemas orientados a objeto. Os autores discutem sobre a classificação dos componentes, dividindo-os em alguns grupos, como, por exemplo, componentes de interface gráfica, de modelagem, de padrões de projeto, de aspectos e aqueles que não se enquadram em nenhuma das opções anteriores.

2.4. Considerações sobre o Capítulo

Diante do que foi pesquisado em trabalhos relacionados à avaliação, busca e categorização de componentes, é possível considerar alguns pontos importantes.

Com relação à categorização de componentes, pode ser observado que, na maioria das abordagens existentes, as categorias em que os componentes serão classificados foram previamente definidas pelos administradores das bibliotecas. Na abordagem que utiliza o método semi-automático de inclusão de novas categorias, os usuários não têm a chance de ter algum recurso que o ajude a sugerir novos nomes.

Além disso, os nomes das categorias são elementos já previamente definidos. Algumas páginas da Internet que têm como uma das funções servir de repositório de componentes, tanto de código aberto, como o SOURCE FORGE (2007), quanto os que são pagos, como o COMPONENT SOURCE (2007), têm sua árvore de categorias baseada neste modo. Nesta situação, se um produtor, ao publicar um componente, não encontrar uma categoria que o atenda, como isto se resolverá?

Com relação à busca de componentes, um ponto importante é a adoção de métodos de flexibilização e de refinamento de mecanismos de recuperação de componentes, visando uma maior precisão nos resultados. Uma alternativa interessante é combinar técnicas diferentes de busca. Deste modo, os consumidores passam a ter formas automatizadas de pesquisar os componentes e isto se torna uma tarefa que aumenta a precisão nos resultados de busca. Além do mais, um componente categorizado de maneira coerente e clara pode gerar resultados de busca mais precisos. Portanto, esta relação entre busca e categorização é importante para uma biblioteca.

Quanto à avaliação, observamos que os estudos para modelos de qualidade específicos para componentes de software estão se desenvolvendo. Logo, as práticas retiradas dos modelos servirão como base para uma abordagem que permita aos consumidores efetuarem suas críticas e aos produtores usar estas opiniões como uma ferramenta para melhorar a qualidade dos componentes presentes na biblioteca.

Desta maneira, o objetivo principal deste projeto final é oferecer uma alternativa na qual: (i) os desenvolvedores possam categorizar seus componentes de acordo com suas expectativas; (ii) os usuários possam buscar estes componentes de variadas formas e de maneira precisa; e (iii) os usuários consigam efetuar avaliações sobre a qualidade do componente adquirido. Neste sentido, na próxima seção, será apresentada a abordagem proposta para a avaliação, busca e categorização no contexto de uma biblioteca de componentes de software.

Capítulo 3 A Abordagem Brechó-ABC

Produtores e consumidores de componentes devem estabelecer uma relação de confiança para que o processo de seleção de componentes, no contexto de uma biblioteca, possa ser mais eficiente e preciso (WANG e MEHANDJISKA-STAVREVA, 2004). Estes autores propõem um processo de seleção de componentes baseado na cooperação entre os produtores e os consumidores, chamado de CBCS (*Collaboration-Based Component Selection*). Para eles, este tipo de processo traria algumas vantagens em relação à colaboração; ao se estabelecer esta relação de cooperação, o tempo e o esforço para ambos poderiam diminuir, e, além disso, os sistemas e os componentes comercializados poderiam se tornar mais confiáveis.

Os administradores e desenvolvedores da biblioteca de componentes também devem ter um papel importante nesta relação, pois, se estes desenvolverem ferramentas que facilitem o processo de seleção de componentes, é possível promover uma maior interação entre produtores e consumidores, na tentativa de aumentar a oferta de componentes de qualidade dentro da biblioteca e também de reutilizá-los e encontrá-los com mais facilidade.

Sendo assim, ferramentas de avaliação, busca e categorização de componentes podem auxiliar o processo de seleção a partir de uma biblioteca de componentes. Os produtores publicam componentes e necessitam categorizá-los. Caso o componente esteja classificado de forma incoerente ou se não existir categorias que o satisfaçam, os consumidores terão problemas na hora da busca, pois o componente não será encontrado. Os consumidores precisam buscar componentes que satisfaçam seus requisitos. Um componente mal classificado faz com que o consumidor não encontre o componente que poderia atender às suas necessidades. Os consumidores também avaliam os componentes que obtêm da biblioteca de componentes.

Neste capítulo, é apresentada a Brechó-ABC, uma proposta de abordagem integrada de avaliação, busca e categorização de componentes. Esta abordagem se propõe a apoiar os consumidores e produtores de componentes na interação em uma biblioteca de componentes.

Inicialmente, será exposto um cenário de utilização de uma biblioteca de componentes na Seção 3.1. O objetivo desta seção é mostrar que os mecanismos de avaliação, busca e categorização podem atuar de forma integrada, apoiando o processo

de seleção de componentes. Na Seção 3.2 é apresentada uma proposta de abordagem integrada de avaliação, busca e categorização com os principais requisitos de cada um dos mecanismos descritos. A descrição da arquitetura, bem como as principais funcionalidades esperadas para cada um dos três mecanismos são expostas na Seção 3.3. Finalmente, a Seção 3.4 apresenta algumas considerações sobre o capítulo.

3.1. Cenário de Utilização

A Figura 3.1 ilustra um cenário de utilização de mecanismos de avaliação, busca e categorização sendo utilizados em conjunto. As setas indicam as atividades que produtores e / ou consumidores realizam na biblioteca e o que eles esperam no final. Considere o seguinte exemplo: um produtor desenvolveu um componente chamado "Dollar Quotations". Este componente guarda um histórico de todas as cotações do dólar em relação ao real feitas nos últimos 365 dias. Este produtor queria publicar o componente na categoria "Indicadores Financeiros", mas esta categoria não existe na biblioteca. Já um consumidor deseja encontrar um componente que tenha as cotações do dólar nos últimos meses. A partir disto, este consumidor utilizará os dados para fazer gráficos que mostram a evolução destas cotações ao longo dos meses. Há alguns dias, ele adquiriu outro componente que guarda o histórico de cotações do euro, chamado "Euro Quotat". Ele gostaria de avaliá-lo e comentar que a apresentação dos valores e que o mecanismo de busca das cotações são confusos.

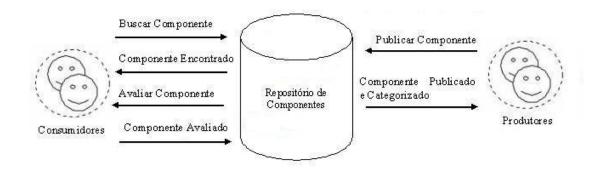


Figura 3.1: Algumas necessidades de consumidores e produtores numa biblioteca de componentes.

Se a biblioteca de componentes dispuser de um mecanismo que apóie a tarefa de categorização, os produtores poderão classificar seus componentes de maneira coerente.

Este tipo de mecanismo auxiliará os consumidores no momento em que eles buscarem os componentes, pois estes estarão organizados de forma clara. Os produtores serão favorecidos, pois seus componentes poderão ter maiores chances de aproveitamento na hora da busca. Quando o consumidor adquirir o componente, o mecanismo de avaliação o ajudará a efetuar uma avaliação, deixando suas impressões sobre o artefato. Esta avaliação ajudará outros consumidores, que poderão usar este depoimento como ajuda para determinar se compram ou não o componente. Além do mais, os produtores usarão estas críticas para corrigir erros e aperfeiçoar o componente em próximas versões.

No exemplo acima, o produtor poderá classificar o componente em "Indicadores Financeiros" ou em alguma outra categoria semelhante, como "Valores Financeiros". O consumidor encontrará as categorias bem organizadas. Assim, na hora da busca, este consumidor terá maiores chances de achar "Dollar Quotations", o componente que satisfaz suas necessidades. Ele ainda pode deixar as devidas críticas sobre o "Euro Quotat". Assim, os produtores deste componente se aproveitarão da avaliação para melhorar os pontos fracos citados pelo consumidor.

Por isso, a abordagem Brechó-ABC foi concebida para ser aplicada em cenários similares ao da Figura 3.1, que integra os mecanismos de avaliação, busca e categorização de componentes. A proposta visa a auxiliar o processo de recuperação de componentes, podendo também beneficiar tanto produtores quanto consumidores.

3.2. Definição dos Requisitos e Proposta

A partir dos diferentes trabalhos abordados no capítulo anterior e do cenário exposto na seção 3.1, foram identificados os requisitos para o estabelecimento de uma proposta que integra todos os mecanismos citados, enumerando as principais características previstas para cada um deles.

3.2.1. Avaliação

Vislumbra-se um mecanismo para a avaliação de componentes que contemple os seguintes requisitos:

✓ Avaliação de componentes pelos consumidores feita de forma simples e objetiva;

- ✓ Os indicadores precisam ter uma definição clara, sem ambigüidades, ou que requeiram uma sólida base de conhecimento, como, por exemplo, fórmulas matemáticas;
- ✓ Todas as avaliações existentes de um componente devem estar disponíveis para todos os usuários;
- ✓ Informações relativas à avaliação podem estar também disponíveis de maneira gráfica.

Com base nestes requisitos, deseja-se que o consumidor de componentes possa emitir sua opinião sobre um componente e também visualizar as outras opiniões. Com isso, os produtores terão a chance de usar as avaliações como instrumento para melhorar a qualidade dos componentes. Em contrapartida, os consumidores poderão utilizar as avaliações para verificar se outros usuários aprovam ou não o componente, facilitando sua escolha.

3.2.2. Busca

Abaixo, seguem os requisitos identificados para a proposta do mecanismo de busca de componentes:

- ✓ Limitar o espaço de busca durante a pesquisa por componentes;
- ✓ Permitir o uso de técnicas de pesquisa para potencializar a busca;
- ✓ Buscas textuais devem tratar possíveis casos de erros de digitação cometidos pelo usuário;
- ✓ Todas as fontes de documentação associadas ao componente, como versão e
 plataforma do componente, entre outros, devem ser utilizadas como fontes de
 pesquisa.

O propósito deste mecanismo é oferecer opções de busca mais flexíveis e automatizadas, visando facilitar a pesquisa de componentes na biblioteca, aumentando a precisão dos resultados na busca.

3.2.3. Categorização

A proposta de um mecanismo de categorização de componentes deve ter o caráter semi-automático, tanto para os mecanismos de sugestão de categorias quanto para a organização das categorias e das sugestões, inspirado no método SemaCS (SJACHYN e BEUS-DUKIC, 2006).

Os requisitos identificados para o mecanismo de categorização de componentes são:

- ✓ O mecanismo de categorização deve permitir sugestões de novas categorias para a biblioteca;
- ✓ Deve ser definido um mecanismo de busca de sugestões na Web baseado em características do componente a ser cadastrado;
- ✓ O conjunto de mecanismos externos de busca de sugestões de categorias oferecido deve ser extensível;
- ✓ O mecanismo de categorização deve viabilizar a organização, de modo semiautomático, de categorias e sugestões;
- ✓ Os usuários da biblioteca de componentes devem ser notificados, de maneira automática, quando houver certos tipos de alterações na lista de categorias do repositório.

3.3. Arquitetura e Principais Funcionalidades

A partir dos requisitos explicitados na seção anterior, é proposta uma arquitetura referente à Brechó-ABC. A Figura 3.2 apresenta uma visão geral da solução, explicitando os mecanismos da proposta e seus módulos. As setas e traços pretos mostram a ordem dos eventos que ocorrem em cada um dos três mecanismos (avaliação, busca e categorização de componentes) que fazem parte do Brechó-ABC. Já as tracejadas mostram a interação dos módulos com diferentes atores envolvidos no processo, além da WEB. Cada mecanismo é composto de dois módulos.

O mecanismo de categorização possui os módulos de sugestão de categorias e de organização de categorias e sugestões. A seta (1) mostra que a organização das categorias deve ser realizada após os produtores sugerirem categorias para a biblioteca. A seta (2) exibe a transição entre o evento de categorização, feita pelo produtor, e o evento de busca por um componente, feito pelo consumidor posteriormente.

O mecanismo de busca possui os módulos de refinamento das buscas e o de busca sintática. O traço (3) significa que, no momento da pesquisa, os dois módulos são utilizados ao mesmo tempo. Além disso, a seta (4) exibe a transição entre o evento de busca, feita por um consumidor, e o evento de avaliação de um componente, feito mais tarde pelo mesmo consumidor.

O mecanismo de avaliação possui os módulos de avaliação dos usuários e o de visualização das avaliações. A seta (5) mostra que o módulo de visualização acontece depois da avaliação de um componente. Isso se torna verdadeiro porque é necessário que existam avaliações para que, mais tarde, elas sejam exibidas.

Nas próximas subseções, serão detalhados os módulos e suas principais funcionalidades associadas.

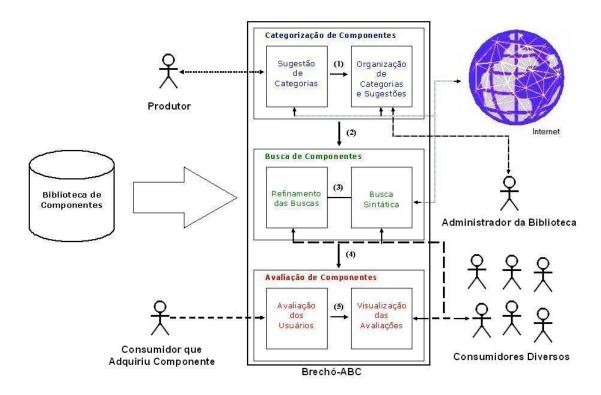


Figura 3.2: Visão geral da arquitetura da Brechó-ABC.

3.3.1. Mecanismo de Avaliação

O mecanismo de avaliação de componentes é dividido em dois módulos. Um deles é o módulo de avaliação dos usuários. Este módulo permite ao consumidor emitir sua opinião a respeito do componente que ele recuperou da biblioteca.

O módulo de visualização das avaliações permite, tanto ao consumidor, quanto ao produtor do componente, observar graficamente as avaliações feitas pelos consumidores. Além disso, é disponibilizado para os usuários, na forma de um gráfico, o percentual de avaliações conforme cada uma das qualificações dadas pelos consumidores sobre o componente.

3.3.1.1. Avaliação dos Usuários

Uma premissa para o uso do mecanismo de avaliação é que os consumidores devem recuperar e adquirir uma versão de um componente antes de avaliá-lo. Este procedimento é realizado como uma tentativa de se evitar avaliações tendenciosas, interferindo negativamente no principal objetivo do mecanismo. Quando o consumidor faz o *download* de uma versão do componente da biblioteca, ele ganha o direito de responder duas questões: a primeira é relativa à atribuição de um grau de satisfação para o componente, através de uma qualificação. O consumidor escolhe um dos diferentes valores para representar o quão satisfeito está com o componente. A segunda questão é referente a comentários diversos, onde consumidores podem, por exemplo, relatar elogios, opiniões, sugestões e críticas sobre o componente.

3.3.1.2. Visualização das Avaliações

Este módulo tem a função de exibir as avaliações já feitas por quem utiliza determinado componente. Existem duas maneiras de visualizar as avaliações: (i) os produtores do componente podem visualizar diretamente as avaliações dos componentes publicados por eles próprios, e (ii) todos os usuários podem visualizar as avaliações de um determinado componente. Também se encontra à disposição dos usuários da biblioteca um gráfico que exibe o percentual de avaliações, de acordo com os diferentes níveis de satisfação existentes para um componente.

3.3.2. Mecanismo de Busca

O mecanismo de busca de componentes na Brechó-ABC é composto por dois módulos: (i) refinamento e flexibilização dos mecanismos das buscas; e (ii) busca

sintática. O primeiro módulo reúne as seguintes características: utilização de filtros para restringir o espaço de busca e a utilização da especificação do componente para pesquisa.

O módulo de busca sintática tem como propósito alertar aos usuários sobre prováveis erros de digitação de palavras que podem ocorrer no momento da busca. Por exemplo, caso a string de busca fornecida por um usuário seja a palavra "verssão", o mecanismo deve alertar o usuário, caso não identifique como válida a palavra, sobre o possível erro de digitação, sugerindo palavras com grafias similares.

3.3.2.1. Refinamento e Flexibilização

A busca de componentes pela Brechó-ABC pode ser realizada de duas maneiras: (i) busca por palavra-chave; e (ii) através das categorias existentes na biblioteca. Foram também concebidas outras formas de busca, com a intenção de reduzir o espaço de busca.

Um exemplo destas formas de busca é a filtragem de componentes por categorias em conjunto com a busca por palavra-chave. O objetivo desta combinação é ter a chance de realizar uma melhor pesquisa, já que foram utilizadas técnicas de busca por palavra-chave e por hiperlinks. Por exemplo, o usuário pode fazer a pesquisa pela palavra "DNA" e filtrar pela categoria "Medicina" ou "Bioinformática".

Além do mais, é possível filtrar os componentes por elementos presentes em sua documentação (informações preenchidas pelo produtor no momento da publicação do componente) em conjunto com a categoria e / ou palavra-chave. Assim, pode-se buscar, por exemplo, um componente com a palavra "DNA", filtrar pela categoria "Bioinformática" e, finalmente, pesquisar somente pelos componentes que estejam sob a plataforma CORBA.

3.3.2.2. Busca Sintática

Esta funcionalidade permite que, caso ocorra um possível erro de digitação na *string* de busca por palavra-chave, isto é, se não for encontrado um componente com este nome, o mecanismo deve apresentar ao usuário o fato. Além disso, o mecanismo sugere ao usuário da biblioteca palavras com grafias similares. Por exemplo, se fosse

digitada a palavra "kabra", o mecanismo alertaria sobre o fato (caso não encontre um componente chamado "kabra") e sugeriria palavras como "abra", "cabra", "magra" e "lavra".

3.3.3. Mecanismo de Categorização

Existem dois módulos para o mecanismo de categorização: (i) sugestão e (ii) organização de categorias. O módulo de sugestão de categorias permite que o produtor, ao publicar o componente, sugira uma categoria, podendo ainda contar com a ajuda de um mecanismo de busca de sugestões na Web.

Dada uma configuração de categorias existentes numa biblioteca, deseja-se incorporar novas sugestões de categorias no conjunto de categorias existente. Sendo assim, o módulo de organização de categorias deve apoiar o administrador na tarefa de organizar as categorias da biblioteca, levando em consideração as sugestões propostas. Além disso, os usuários devem ser notificados se suas sugestões foram aceitas pelo administrador.

3.3.3.1. Sugestão de Categorias

Os produtores do componente podem, ao publicar o componente na biblioteca, fazer sugestões de novas categorias. Estas sugestões podem ser feitas manualmente e / ou de forma semi-automática, com a procura de sugestões em mecanismos de busca na Web. Neste caso, o produtor escolhe os mecanismos, como o Yahoo (YAHOO, 2007), o Google (GOOGLE, 2007), entre outros, onde a pesquisa por sugestões poderá ser feita. A partir das informações do componente, estes mecanismos encontram palavras que possam ser candidatas a possíveis categorias. Algumas destas palavras são retornadas ao produtor, que então escolhe as palavras que deseja que se tornem sugestões.

3.3.3.2. Organização de Categorias

Levando em consideração que categorias podem ser sugeridas pelos produtores de componentes, o módulo de organização de categorias tem o propósito de apoiar o

administrador na tarefa de manter o conjunto de categorias existentes. Este módulo realiza pesquisa em mecanismos na Web que têm como objetivo identificar similaridades entre palavras, para que seja possível relacionar sugestões e categorias. Esta relação entre termos (sugestões e categorias existentes) é útil para reorganizar o conjunto de categorias da biblioteca de componentes, visando diminuir a ambigüidade e inconsistência entre categorias.

Este módulo contempla, após a organização do conjunto de categorias, a notificação dos produtores de componentes cuja sugestão dada foi aceita. Desta forma, estes produtores terão a possibilidade de publicar o artefato nesta nova categoria. A notificação de produtores de componentes ocorre ainda quando categorias relacionadas a seus componentes estão para ser excluídas, emitindo uma mensagem para que estes componentes afetados sejam publicados em outra categoria.

3.4. Considerações sobre o Capítulo

A Brechó-ABC serve como uma abordagem composta por mecanismos de avaliação, busca e categorização que, de forma integrada, têm como finalidade ajudar os produtores e consumidores no processo de seleção de componentes. Foi exposto o cenário de uma biblioteca onde os componentes são: (i) publicados e categorizados; (ii) pesquisados e, após serem recuperados, (iii) avaliados.

Neste cenário, foram identificados alguns problemas, como por exemplo, a dificuldade de se categorizar os componentes em um conjunto limitado de categorias. Desta forma, foram definidos os requisitos para solucionar os problemas encontrados para cada um dos mecanismos da abordagem. Foi ainda apresentada a arquitetura referente à abordagem, detalhando os módulos e descrevendo suas principais funcionalidades.

No capítulo seguinte, será apresentada a implementação de cada uma das partes da abordagem Brechó-ABC, mostrando os principais aspectos e a utilização de cada um dos mecanismos no contexto da biblioteca Brechó.

Capítulo 4 Implementação

Neste capítulo, serão apresentados os principais aspectos da implementação da abordagem Brechó-ABC. Na Seção 4.1, será apresentada a biblioteca Brechó e o que o protótipo Brechó-ABC vem acrescentar a esta biblioteca, assim como as tecnologias que foram utilizadas para a sua implementação. Na Seção 4.2, serão detalhados os aspectos de implementação do mecanismo de avaliação, descrevendo a utilização de cada um dos módulos. Os aspectos de implementação do mecanismo de busca serão apresentados na Seção 4.3, expondo como estas funcionalidades podem ser usadas no contexto da biblioteca. Na Seção 4.4 serão discutidas as funcionalidades do mecanismo de categorização e como estas funcionalidades poderão ser empregadas através da interface gráfica do Brechó. Na Seção 4.5, são apresentados alguns exemplos de utilização da biblioteca Brechó e como os mecanismos da Brechó-ABC atuam em cada situação. Finalmente, na Seção 4.6, serão feitas as considerações sobre este capítulo.

4.1. O Brechó-ABC e a biblioteca Brechó

O conjunto de mecanismos de avaliação, busca e categorização de componentes Brechó-ABC está integrado à Brechó, uma biblioteca de componentes e serviços de software. A bliblioteca Brechó é um sistema de informação Web que oferece ferramentas para apoiar o relacionamento entre os fornecedores e os consumidores de componentes e serviços. Podemos citar como ferramentas criadas no núcleo inicial as de publicação, armazenamento, documentação, pesquisa e recuperação de componentes.

O projeto Brechó e o Brechó-ABC são implementados na tecnologia Java EE (SUN, 2007), utilizando o *framework* Struts (APACHE, 2007). Para a parte de persistência de dados, foram utilizados o *framework* Hibernate (REDHAT, 2007) e o banco de dados MySQL (MYSQL AB, 2007).

Na versão atual do protótipo, existem os papéis de (i) usuário e (ii) administrador. O usuário tem a responsabilidade de publicar os componentes, fornecendo as especificações e classificando em uma ou mais categorias. Ele também pode pesquisar e recuperar componentes cadastrados. O usuário, assim, assume o papel tanto do produtor quanto do consumidor na Brechó. Já o administrador tem o controle sobre os novos

componentes incluídos no repositório e sobre as categorias existentes. Além disso, ele também tem o poder sobre o cadastro de novos usuários na biblioteca.

Os mecanismos de publicação e documentação na Brechó consideram um conceito flexível de componente, além do binário, visando uma representação que inclua os possíveis artefatos produzidos durante o desenvolvimento do componente (como especificações, código fonte, manuais, etc.). Desta forma, a biblioteca torna-se capaz de permitir a aquisição de diferentes combinações de conjuntos de artefatos empacotados, a partir de licenças personalizadas e configuráveis.

A estrutura de documentação é fundamentada em categorias e formulários dinâmicos e configuráveis ao invés de um formato prefixado. Desta forma, é possível classificar um componente em várias categorias, além de criar e associar diferentes formulários de documentação a cada categoria, permitindo a construção da documentação do componente como um mosaico.

Outro ponto importante a se destacar é a organização interna da Brechó. Ela é dividida em níveis para representação de um componente. Os níveis definidos são: Componente, Distribuição, Release, Pacote, Licença.

O nível Componente representa conceitualmente as entidades armazenadas na Brechó, sem as informações concretas sobre as implementações dessas entidades. O nível Distribuição representa um corte funcional sobre as entidades, fornecendo conjuntos de funcionalidades que são desejadas por grupos específicos de usuários. O nível Release representa um corte temporal sobre as distribuições, no qual define versões dos artefatos que implementam as entidades em um determinado instante no tempo. A partir desse nível, as entidades passam a ter informações concretas sobre suas implementações. Essas implementações concretas das entidades usualmente existem em diferentes níveis de abstração (e.g.: documentação do usuário, análise, projeto, código, binário, etc.), e diferentes empacotamentos podem ser definidos para possibilitar a reutilização de parte dos níveis de abstração disponíveis (e.g. documentação do usuário e binário). Desta forma, o nível **Pacote** permite que seja feito um corte em níveis de abstração, possibilitando que sejam agrupados artefatos de acordo com o público alvo de reutilização. O nível Licença possibilita a definição de níveis de serviço sobre os pacotes. Para cada pacote podem ser estabelecidas licenças específicas, que garantem regras entre os produtores e os consumidores dos componentes.

A Figura 4.1 mostra a página inicial da Brechó. A biblioteca está disponível no *site* http://reuse.cos.ufrj.br/brecho. Estão disponíveis em WERNER *et al.* (2007) maiores informações sobre o projeto.



Figura 4.1: A página inicial da biblioteca de componentes Brechó.

A Figura 4.2 mostra o diagrama de classes do Brechó-ABC, com as entidades e relacionamentos pertencentes. A partir da seção 4.2, será mostrado, para cada mecanismo, a parte do diagrama de classe com a entidade principal, os atributos desta entidade e os relacionamentos mais próximos a esta classe.

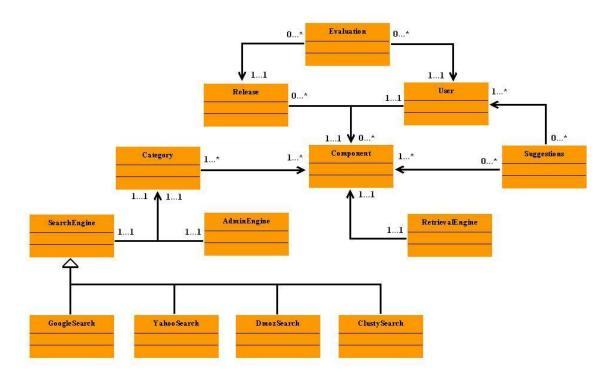


Figura 4.2: O diagrama de classes para a abordagem Brechó-ABC.

4.2. Mecanismo de Avaliação

O mecanismo de avaliação de componentes do Brechó-ABC é o módulo responsável por coletar os depoimentos dos usuários que adquiriram um componente. Este módulo também tem o papel de apresentar as avaliações já feitas, com um gráfico, mostrando o percentual de avaliações, de acordo com os diferentes graus de satisfação do artefato.

4.2.1. Detalhes de Implementação

A classe *Evaluation* é a responsável por conter as informações sobre uma avaliação. A Figura 4.3 mostra a parte do diagrama de classes contendo a classe *Evaluation* e seus relacionamentos com as classes *User* e *Release*.

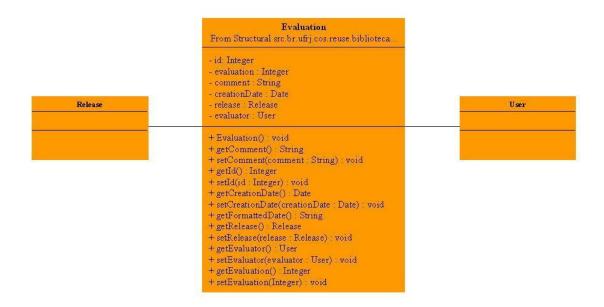


Figura 4.3: Diagrama contendo a classe Evaluation e seus relacionamentos.

A classe *Evaluation* agrega os atributos básicos para caracterizar uma avaliação feita pelo consumidor do componente. Seus atributos são:

- ✓ id: Identificador numérico da avaliação. No momento que o usuário faz a avaliação, este número é gerado, respeitando-se a numeração das avaliações anteriores:
- ✓ evaluation: Representa o nível de satisfação que o usuário atribui para o componente. Esta qualificação aparece na forma de cinco diferentes adjetivos, ordenados de forma crescente de satisfação: Péssimo, Ruim, Razoável, Bom e Excelente.
- ✓ comment: Refere-se aos comentários sobre o componente que determinado usuário adquiriu.
- ✓ creationDate: Data e hora em que a avaliação foi submetida pelo usuário na biblioteca.
- ✓ release: Na organização interna da Brechó, existem vários níveis de representação de um componente. Um desses níveis, chamado Release, representa um corte no qual se definem as versões dos artefatos que implementam os componentes em um determinado momento no tempo. É a

partir deste nível que os artefatos passam a ter informações concretas sobre suas implementações. Por esta razão, as avaliações estão relacionadas a uma *release* de um componente. Portanto, este atributo contém as informações sobre a *release* associada à avaliação.

 ✓ evaluator: As avaliações dos componentes estão relacionadas, além das releases destes, com o usuário que as realiza.

A partir dos atributos mostrados na classe *Evaluation*, existe uma ação que verifica se o usuário possui os requisitos para efetuar a avaliação, além de preparar a página para o usuário fazer a avaliação. Quando o usuário termina de fazer a avaliação, existe outra ação que trata de extrair os dados do formulário da página, criar uma nova instância da entidade *Evaluation* e armazenar as informações da classe na base de dados.

Para a parte de visualização das avaliações, temos dois tipos de apresentação. Um destes tipos mostra ao usuário todas as avaliações relacionadas aos componentes que ele publicou. Outro modo é visualizar todas as avaliações de um determinado componente. Um gráfico em forma de setores, mostrando a distribuição das avaliações conforme os cinco níveis de satisfação, é disponibilizado para visualizar as avaliações. A biblioteca JFreeChart (OBJECT REFINERY, 2007) foi utilizada na implementação desta funcionalidade.

4.2.2. Telas e Utilização

Uma vez vistos os aspectos de implementação para o mecanismo de avaliação de componentes na biblioteca, vamos mostrar nesta seção como estas funcionalidades estão acessíveis ao usuário através da interface gráfica da Brechó.

4.2.2.1. Avaliação dos Usuários

Para efetuar uma avaliação, o usuário deve antes adquirir uma versão de um componente, ou seja, uma release. Quando o download desta release estiver completo, o usuário pode, na tela de listagem das releases do componente, escolher a opção

"Avaliar". Na tela de avaliação da release, mostrada na Figura 4.4, o usuário avalia a release, deixando seus comentários sobre a mesma.

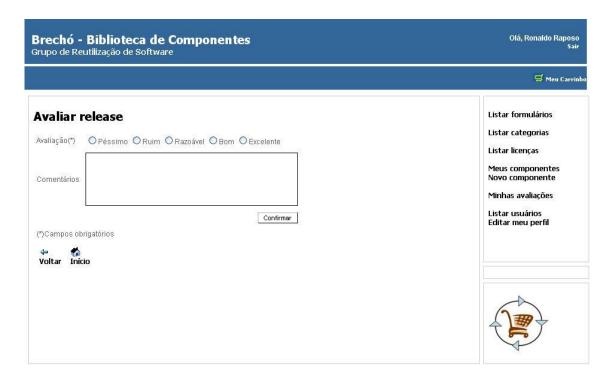


Figura 4.4: Tela da Brechó para a avaliação de uma release.

4.2.2.2. Visualização das Avaliações

Para visualizar as avaliações feitas por um usuário, é necessário clicar no link "Minhas Avaliações" no menu à direita da tela. A Figura 4.5 mostra os links disponíveis para o usuário na biblioteca e, entre eles, o link citado.

A visualização de todas as avaliações de um componente, encontra-se na tela de listagem de *releases* do componente. Escolhendo a opção "Mostrar Avaliações", é possível visualizar todos os depoimentos feitos pelos usuários sobre uma *release*. A Figura 4.4 mostra, a título de exemplo, as avaliações do componente Odyssey, release 1.6.0, e o gráfico de distribuição das avaliações.

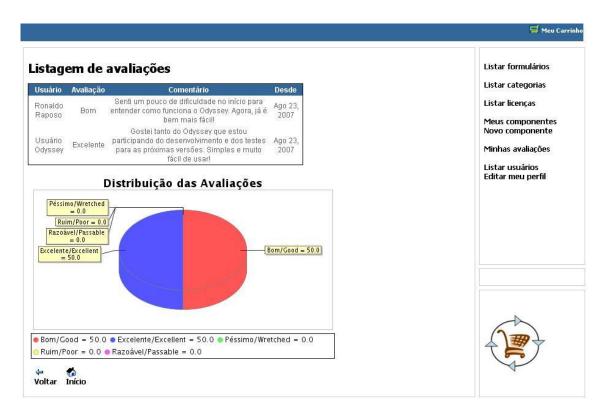


Figura 4.5: Tela da Brechó para a listagem de avaliações de uma release.

4.3. Mecanismo de Busca

Como visto no capítulo 3, o mecanismo de busca de componentes da abordagem Brechó-ABC é o módulo responsável por refinar as técnicas de busca já existentes no núcleo inicial da Brechó. Além disso, este módulo é responsável por apresentar ao usuário a ocorrência de um erro na grafia da palavra-chave na hora da busca, sugerindo algumas palavras similares.

4.3.1. Detalhes de Implementação

A classe *RetrievalEngine* possui os métodos que implementam os módulos de flexibilização e refinamento das buscas e de busca sintática. A Figura 4.6 mostra uma representação desta classe e seu relacionamento com a classe Component.

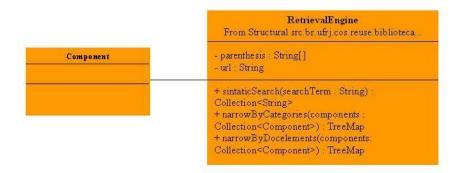


Figura 4.6: Diagrama contendo a classe RetrievalEngine e o relacionamento com a classe Component.

Assim, foram implementados os seguintes tipos de busca no núcleo inicial: (i) a busca por categorias e (ii) a busca por palavra-chave. Estas duas opções de busca eram executadas de forma isolada. Não se pesquisava por categoria e por palavra-chave de maneira simultânea.

Com a implementação do mecanismo de busca pela Brechó-ABC, é adicionada à busca por palavra-chave a função de refinar a pesquisa realizada pelas categorias existentes. Estas categorias estão organizadas em *hiperlinks* que se encontram abaixo dos resultados da busca. Em cada *hiperlink*, aparece o nome da categoria relacionada à palavra-chave e, entre parênteses, ao lado, o número de componentes que pertencem à categoria. O método **narrowByCategories** executa a tarefa de refinar os componentes pela categoria.

Outro refinamento de busca se dá quando se busca um componente por categoria ou por categoria e palavra-chave. É possível filtrar os resultados da busca por elementos presentes na documentação do componente. Um exemplo destes elementos seria o tipo de licença do componente. O componente pode ter a licença GPL, LGPL, X11, BSD, entre outras. Desta forma, neste exemplo, existirão *hiperlinks* abaixo dos resultados da busca com o nome da licença e o número de componentes que possuem uma

determinada licença. O método **narrowByDocelements** executa a tarefa de refinar os componentes pelos elementos da documentação.

Para o módulo de busca sintática, foi implementado o método **sintaticSearch** da classe *RetrievalEngine*. Neste método, alguns dicionários de sinônimos da Web são pesquisados com a finalidade de encontrar sugestões de palavras com grafias idênticas ou similares àquelas informadas pelo usuário. Por exemplo, procura-se o termo informado pelo usuário no Dicionário da Língua Portuguesa On-Line (PRIBERAM, 2007). Neste dicionário, quando se busca o significado de uma palavra que não exista nele, o corretor ortográfico deste dicionário sugere algumas palavras. Assim, o método **sintaticSearch** se utiliza dessas sugestões, apresentando-as para o usuário. Outros dicionários, além deste, podem ser adicionados, bastando apenas configurar a comunicação entre o mecanismo e o dicionário.

4.3.2. Telas e Utilização

Após apresentar os aspectos de implementação para o mecanismo de busca de componentes na Brechó, será tratada nesta seção como estas funcionalidades estão disponíveis para o usuário através da interface gráfica da biblioteca.

4.3.2.1. Flexibilização e Refinamento das Buscas

O usuário faz a pesquisa de componentes de acordo com suas necessidades. No início, podem ser escolhidas as buscas por palavra-chave ou por categorias. Após buscar pela palavra-chave, ele tem a possibilidade de refinar pelas categorias. Basta clicar no link onde está a categoria na qual deseja-se fazer o filtro. A Figura 4.7 mostra a tela onde o usuário fez a busca pela palavra-chave "componente". Neste momento, ele pode refinar a pesquisa procurando componentes que sejam ou da "Categoria 1", que possui duas ocorrências, ou da "Categoria 2", que possui apenas uma ocorrência.

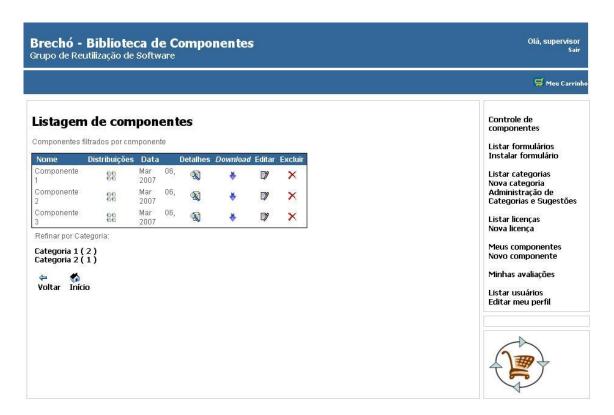


Figura 4.7: Tela da Brechó onde é mostrada a filtragem de componentes por categorias.

Quando o usuário escolhe a categoria e a palavra-chave, ele pode ainda filtrar os componentes pelos elementos da documentação. A Figura 4.8 mostra a tela onde, a partir da Figura 4.6, o usuário escolheu refinar a pesquisa por componentes que estejam na categoria "Categoria 2". Neste momento, o usuário poderá, em seguida, refinar por componentes que são da tecnologia DotNet ou EJB. Além disso, poderá filtrar a busca pela versão do componente.

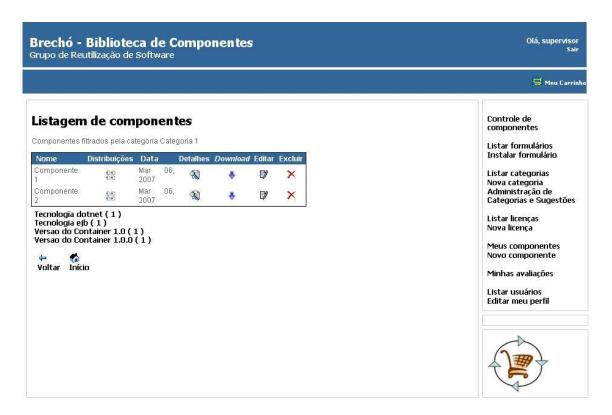


Figura 4.8: Tela da Brechó onde é mostrada a filtragem de componentes pelos elementos da documentação.

4.3.2.2. Busca Sintática

O usuário, quando busca um termo que não existe no dicionário de termos, receberá o alerta de que nenhum componente foi encontrado. Além disso, o mecanismo de sugestões de termos irá sugerir palavras com grafias similares. A Figura 4.9 mostra um exemplo do uso da busca sintática. Para reproduzir esta situação, foi realizada uma pesquisa procurando-se pela palavra-chave "verssão". O mecanismo alerta o fato de ter ocorrido o erro ortográfico e sugere algumas palavras, como "versão" e "verão". Se, por exemplo, o usuário clicar em cima da palavra "versão", será realizada uma nova pesquisa com esta palavra-chave.

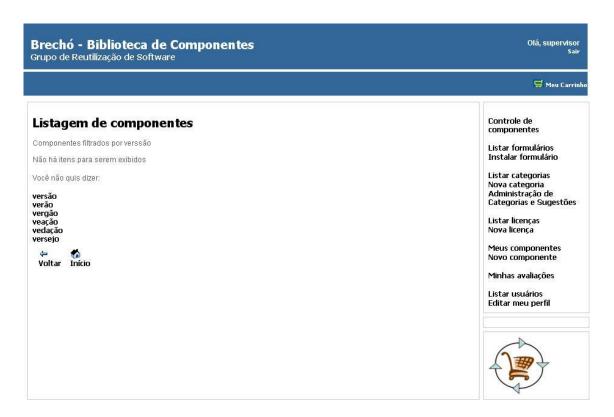


Figura 4.9: Tela da Brechó onde é mostrada a busca sintática, com as sugestões de palavras grafadas corretamente.

4.4. Mecanismo de Categorização

O mecanismo de categorização de componentes do Brechó-ABC é o módulo responsável por apoiar os usuários na tarefa de sugerir novas categorias na Brechó. Este mecanismo também é responsável por auxiliar o administrador da biblioteca na organização das categorias, levando-se em conta as sugestões dos usuários, e notificar aos produtores dos componentes que a sugestão dada foi aceita.

4.4.1. Detalhes de Implementação

Quando o produtor publica o componente na Brechó, ele pode sugerir novas categorias. Estas sugestões podem ser feitas preenchendo os campos na tela de sugestões. Desta maneira, é possível fornecer algumas sugestões.

Além desta forma, o mecanismo de sugestões de categorias pode auxiliar os usuários nesta tarefa, sugerindo categorias. A Figura 4.10 mostra as etapas para que o módulo de sugestão de categorias possa realizar as sugestões.

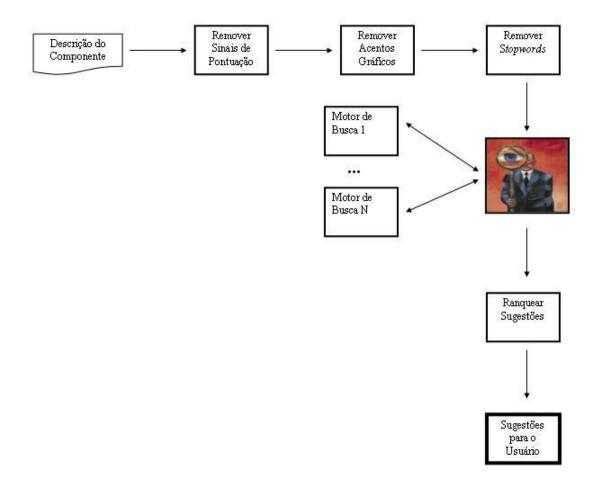


Figura 4.10: Seqüência de ações que o módulo de sugestão de palavras executa na Brechó.

Inicialmente, o mecanismo quebra a descrição do componente em palavras e faz algumas alterações na estrutura da descrição do componente. São removidos os sinais de pontuação e os acentos gráficos de todas as palavras. Em seguida, são filtradas algumas palavras conhecidas como *stopwords*. Exemplos de *stopwords* são preposições ("sob", "sobre"), advérbios ("bastante", "sempre") e números. Estas palavras aparecem com freqüência (entre quarenta a sessenta por cento do total de palavras de um texto), mas não possuem grande relevância para a busca, pois agregam pouca informação útil (SHNEIDERMAN *et al.*, 1997).

Sendo assim, o mecanismo toma cada palavra resultante e realiza pesquisas em diversos motores de busca na Web. No momento, o usuário pode escolher dentre os seguintes motores: (i) Google Directory (GOOGLE, 2007), (ii) Yahoo Directory (YAHOO, 2007), (iii) Clusty (VIVÍSIMO, 2007) e (iv) Dmoz (NETSCAPE, 2007). A comunicação entre o Brechó-ABC e os mecanismos de busca é feito através de um

parser que coleta as palavras que são candidatas a sugestões a partir da página HTML com o resultado da busca.

A seguir, o resultado consolidado das buscas passa por um processo de classificação por relevância, que faz uma contagem de sugestões levando-se em consideração o número de resultados retornados por um motor. Desta forma, se uma sugestão aparecer nos resultados do Google e Yahoo, ou se forem encontradas duas ocorrências no Google, maiores serão as chances desta sugestão estar melhor classificada. Finalmente, as mais relevantes são retornadas para o usuário e ele, então, escolhe as que serão armazenadas na base.

A classe que representa a entidade composta por uma sugestão e seus atributos se chama *Suggestions*. A Figura 4.11 mostra, de forma simplificada, um diagrama contendo a classe *Suggestions* e os relacionamentos com as classes *User* e *Component*.



Figura 4.11: Diagrama contendo a classe Suggestions e seus relacionamentos.

A classe *Suggestions* agrega os atributos básicos para caracterizar uma sugestão feita pelo produtor do componente. Seus atributos são:

- ✓ id: Identificador numérico e único da sugestão. No momento que o produtor faz a sugestão ou escolhe uma das opções vindas dos motores da Web, este número é gerado, respeitando-se a numeração das sugestões anteriores;
- ✓ name: O nome da sugestão dada ou escolhida pelo usuário.
- ✓ **source**: Este campo mostra se a sugestão de categoria foi realizada pelo usuário ou foi pelos motores de busca da Web.
- ✓ **creationDate**: Data e hora em que a sugestão foi submetida pelo usuário na biblioteca.
- ✓ user: Uma sugestão está relacionada a um usuário que a fez. Por isso, este atributo contém as informações sobre o usuário que fez uma determinada sugestão.
- ✓ componentId: Uma sugestão também está associada ao componente publicado pelo produtor. Por esta razão, este atributo contém o número identificador do componente cuja sugestão foi feita por este usuário.

O administrador do Brechó pode aproveitar as sugestões dadas para organizar a lista de categorias existentes. O módulo de organização das categorias e sugestões o apóia ao tentar identificar relações de similaridade entre sugestões e categorias, visando aumentar a coerência e a clareza entre as categorias.

Esta relação de similaridade entre sugestões e categorias se dá através de uma variável encontrada nos estudos de CILIBRASI e VITANYI (2007). Os autores pesquisaram um método que calculasse o nível de similaridade entre palavras utilizando o Google. Esta variável é chamada distância normalizada do Google (NGD). Ela é definida em função do número de resultados de busca do Google. A medida de NGD foi escolhida a partir do mecanismo que pesquisa na Web sugestões de categorias. Este mecanismo foi aproveitado e configurado para a pesquisa no Google a fim de se calcular o índice de similaridade entre palavras. A NGD é definida por:

$$NGD (termo1, termo2) = \frac{\max \{ \log f(termo1), \log f(termo2) \} - \log f(termo1, termo2) }{\log N - \min \{ \log f(termo1), \log f(termo2) \}}$$

Um exemplo pode ajudar a entender esta variável. Considere o cálculo da NGD entre as palavras "cavalo", que será o "termo1" da fórmula, e "jóquei", que será o "termo2" da fórmula. A função f(termo) é dada pela frequência de resultados de busca

no Google para a variável "termo". Desta forma, f(cavalo) = 4.060.000, f(jóquei) = 193.000 e f(cavalo, jóquei) = 26.300. "N" significa o total de páginas indexadas pelo Google. Na época que os autores pesquisaram este método, existiam um pouco mais que oito bilhões de páginas.

Com estas informações, calcula-se que o valor de NGD (cavalo, jóquei) seja aproximadamente igual a 0,475. Isto significa que as palavras cavalo e jóquei possuem relação de similaridade entre média e alta. O valor de NGD vai de zero a infinito. Quanto mais próximo de zero, mais similares as palavras serão. No entanto, grande parte dos NGD's varia entre zero e um. Se o valor de NGD for maior que 1 (um), significa que há uma correlação negativa entre as palavras: sozinhas, as palavras possuem uma freqüência alta de resultados, porém juntas, elas possuem uma freqüência baixa.

Logo, o administrador da biblioteca, para estabelecer as relações entre sugestões e categorias, deve estabelecer um índice de similaridade. Este índice pode ser escolhido entre cinco diferentes níveis: Muito Baixo (NGD varia entre 0,8 e 1,0); Baixo (NGD varia entre 0,6 e 0,8); Médio (NGD varia entre 0,4 e 0,6); Alto (NGD varia entre 0,2 e 0,4) ou Muito Alto (NGD varia entre 0 e 0,2). Após isto, o módulo de organização lança robôs que procuram na Web, para cada categoria e para cada sugestão, definições no Google ou palavras relacionadas no KartOO (KARTOO, 2007). Se existirem nestes dados alguma relação entre categorias e sugestões, será calculado o NGD para cada um destes pares (categoria – sugestão ou sugestão – sugestão). Se o NGD do par estiver dentro do nível estabelecido pelo administrador, ele será retornado.

Deste modo, aparecerão para o administrador duas tabelas: uma com a categoria e a sugestão similar, e outra com um par de sugestões similares. Ele escolhe então se deseja fazer alguma alteração na lista de categorias. Se o administrador quiser tornar como nova categoria uma sugestão, basta dar um clique em cima da palavra. Após a criação, serão notificados os usuários cuja sugestão dada foi aceita. Se o administrador quiser excluir uma categoria, os usuários cujos componentes estão relacionados a esta categoria serão também notificados.

4.4.2. Telas e Utilização

Depois de expor os aspectos de implementação para o mecanismo de categorização de componentes na Brechó, será apresentado nesta seção como estas funcionalidades estão disponíveis para o usuário e para o administrador na biblioteca.

4.4.2.1. Sugestão de Categorias

O produtor, depois de publicar o componente, escolhe se está ou não satisfeito com a lista de categorias da Brechó. A Figura 4.12 mostra esta tela. Se o produtor escolher não, ele poderá então sugerir alguma categoria.



Figura 4.12: Tela da Brechó onde mostra o início do módulo de sugestão de categorias.

Na tela abaixo, o usuário escreve as sugestões que quiser e também escolhe os sites nos quais o módulo lançará os robôs com o intuito de procurar as sugestões para ele na Web. A Figura 4.13 mostra que foi dada uma sugestão (pode-se dar até três sugestões) e que o usuário escolheu alguns sites de busca.

	🛒 Meu Carr
Escreva aqui as categorias que melhor se encaixam na descrição do seu componente: Sugestão 1 Imposto de Renda Sugestão 2 Sugestão 3 Você também pode recorrer à Web para ajudá-lo(a) a sugerir alguma categoria. Quer executar esta opção? Se sim, escolha os sites da Web que procurarão as sugestões:	Listar formulários Listar categorias Listar licenças Meus componentes Novo componente Minhas avaliações Listar usuários Editar meu perfil
☑ Yahoo (http://br.search.yahoo.com/dir) ☑ Clusty (http://clusty.com/) □ Dmoz (http://dmoz.org/World/Português/)	
Sin Não Voltar Início	\

Figura 4.13: Tela da Brechó onde o usuário sugere as categorias manualmente ou recorre à Web.

Em seguida, são retornadas as cinco sugestões mais relevantes para que o produtor faça sua opção. Assim, ele clica no *checkbox* e confirma para que o módulo armazene as sugestões na base. A Figura 4.14 mostra que o usuário escolheu como sugestões as palavras "Contabilidade" e "Cálculo".

Brechó - Biblioteca de Componentes Grupo de Reutilização de Software	Olá, Ronaldo Raposo Sair
	₩ Meu Carrinho
Sugestão de Categorias dada pela Web As categorias que melhor se encaixam na descrição do componente são as seguintes: Imposto de Renda - IR Negócios e Economia Jornais Contabilidade Cálculo Confirmar Voltar Início	Listar formulários Listar categorias Listar licenças Meus componentes Novo componente Minhas avaliações Listar usuários Editar meu perfil

Figura 4.14: Tela da Brechó onde mostra as categorias mais relevantes vindas da Web.

4.4.2.2. Organização das Categorias e Sugestões

O administrador tem, na tela inicial, a opção Administração de Categorias e Sugestões, onde ele pode utilizar o mecanismo a qualquer momento. A Figura 4.15 mostra a tela inicial, com a opção selecionada na cor cinza.



Figura 4.15: Tela inicial com a opção de administração de categorias e sugestões selecionada.

Em seguida, ele escolhe o índice de similaridade entre as sugestões e categorias da biblioteca. A Figura 4.16 mostra que o administrador optou pelo nível médio de similaridade (onde o NGD será menor ou igual a 0,6) para estabelecer as relações entre os termos.

Finalmente, na próxima tela, o administrador pode ver se existe alguma relação entre categorias e sugestões, ou entre duas sugestões similares. Na Figura 4.17, o administrador não encontrou relações de similaridade entre categorias e sugestões. Contudo, ele achou algumas sugestões com potencial para tornar futuras categorias. Por exemplo, o administrador clicou em Ciência a partir destas relações e será levado para a tela de cadastro de novas categorias automaticamente.



Figura 4.16: Tela da Brechó onde o administrador escolhe um nível de similaridade entre as categorias e as sugestões.



Figura 4.17: Tela da Brechó onde são retornadas as relações de similaridade entre sugestões neste caso.

4.5. Exemplos de Utilização

Para os exemplos de utilização da biblioteca Brechó apresentados a seguir, considere que o administrador tenha criado quatro categorias: (i) Ciência, (ii) Geografia, (iii) Matemática e (iv) Informática.

Além disso, existem três usuários: (i) Ronaldo's Corporation, que fará o papel do produtor de um componente; (ii) Ronaldo Raposo, que será o consumidor de um componente publicado pela Ronaldo's Corporation; e (iii) Supervisor, que fará o papel do administrador da biblioteca.

4.5.1. Exemplo 1: Publicando e categorizando o componente

O usuário Ronaldo's Corporation acessa a Brechó e irá publicar o componente desenvolvido por esta empresa. A Figura 4.18 mostra a tela inicial de cadastro de um novo componente. Ele será chamado "Evolução" e será criado na categoria "Ciência".



Figura 4.18: Tela inicial da Brechó de cadastro de componente.

Além destes dados, existem também informações relativas à categoria que serão preenchidas na próxima tela. Estas informações são os elementos da documentação

associados pelo administrador à categoria "Ciência". O preenchimento destes dados é exibido na Figura 4.19.



Figura 4.19: Tela com o preenchimento dos elementos da documentação associados à categoria.

Após esta etapa, os dados do componente "Evolução" já estarão salvos na biblioteca. Porém, o produtor pode sugerir categorias para melhor classificar o componente, além de pedir ajuda aos mecanismos da Web. Nas Figuras 4.20 e 4.21, o produtor sugere a categoria "Homem" e escolhe todas as opções de *sites* para se fazer a pesquisa por categorias.

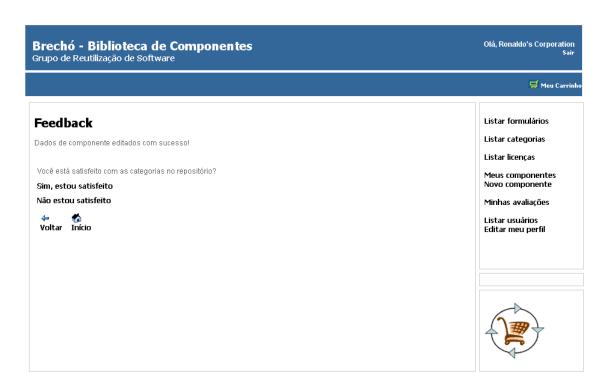


Figura 4.20: Tela que pode levar o usuário a sugerir ou não nomes de categorias para a biblioteca.



Figura 4.21: Tela com o preenchimento das sugestões de categorias.

Depois da pesquisa, são retornados alguns nomes de categorias. Na Figura 4.22, o produtor escolhe o termo "História" como nova sugestão de categoria. O produtor

confirma sua escolha e o cadastro de componente termina. A Figura 4.23 apresenta o final do processo.



Figura 4.22: Tela com as sugestões pesquisadas pela Web e retornadas para o produtor.



Figura 4.23: Tela que representa o fim do cadastro de um componente na Brechó.

4.5.2. Exemplo 2: Pesquisando o componente

Considere que outros componentes tenham sido cadastrados e outras palavras foram sugeridas. Num dado instante, o consumidor Ronaldo Raposo necessita de um componente Java para mostrar aos seus alunos de História a linha do tempo com as eras que o ser humano passou e continua passando. Ele recorre à Brechó para encontrar este componente. Na Figura 4.24, ele pesquisa os componentes escolhendo a categoria "Ciência" e encontra três tipos diferentes de componente.

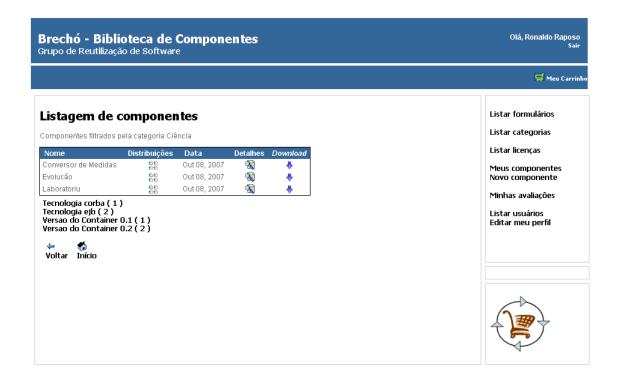


Figura 4.24: Tela que representa os componentes na Brechó filtrados pela categoria "Ciência".

O consumidor então clica no link "Tecnologia ejb" e filtra os componentes pela tecnologia de desenvolvimento do componente, escolhendo somente componentes EJB. A Figura 4.25 mostra o resultado deste passo.

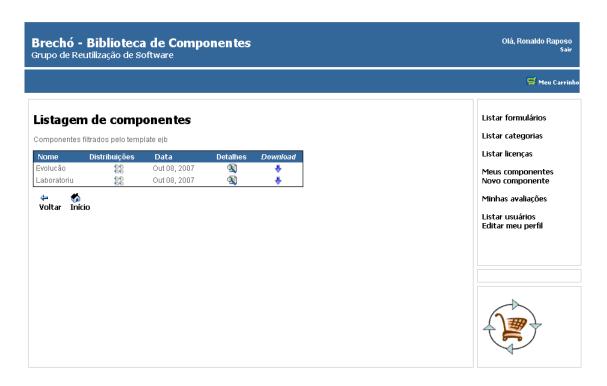


Figura 4.25: Tela que representa o refinamento de componentes pela tecnologia "EJB".

Na Figura 4.26, Ronaldo clicou nos detalhes do componente e "Evolução" e percebeu que é este o componente que estava procurando.



Figura 4.26: Tela que exibe os detalhes do componente "Evolução".

Considere que o consumidor viu outros componentes e se decidiu por adquirir o componente "Evolução". Fez a pesquisa por palavra-chave e, inadvertidamente, escreveu a palavra "Evolussão". As três figuras abaixo mostram a seqüência de ações que acontecerão na biblioteca. A Figura 4.27 mostra a palavra sendo digitada de forma errada pelo consumidor; a Figura 4.28 mostra que o mecanismo detecta que não há componentes identificados com esta palavra e sugere palavras com grafias similares; e a Figura 4.29 mostra a tela após o usuário ter clicado na palavra "Evolução", desta vez grafada de modo correto.

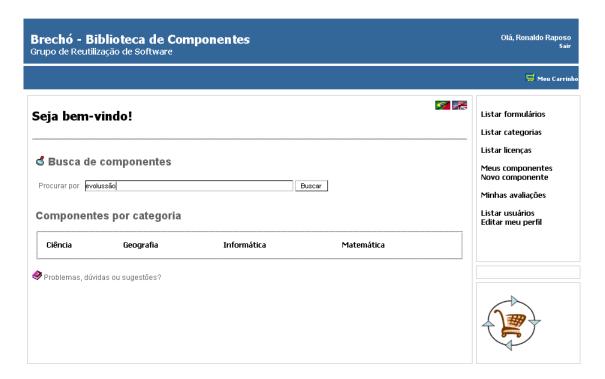


Figura 4.27: Tela que exibe a palavra-chave "Evolução" sendo grafada de modo incorreto.

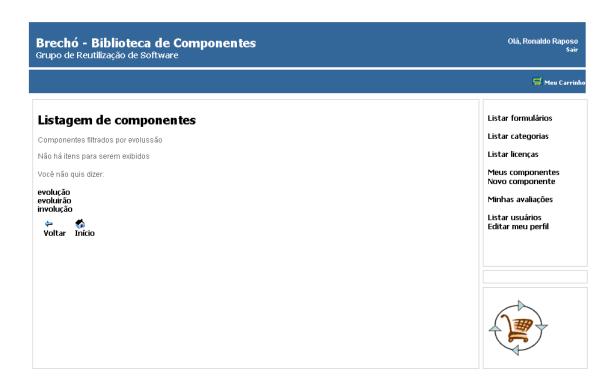


Figura 4.28: Tela que exibe sugestões de palavras com grafia similar.



Figura 4.29: Tela que exibe agora componentes filtrados pela palavra-chave "Evolução".

Se existir um componente grafado de modo diferente, como "Laboratoriu", por exemplo, a biblioteca irá encontrar o componente. A Figura 4.30 mostra o resultado da filtragem de componentes por essa palavra-chave.



Figura 4.30: Tela que exibe componentes filtrados pela palavra-chave "Laboratoriu".

4.5.3. Exemplo 3: Avaliando o componente

Suponha que Ronaldo tenha adquirido "Evolução" após a pesquisa por componente ter sido realizada. O consumidor fez os testes e deseja fazer uma avaliação sobre a release 0.1 do componente. Além disso, outros usuários podem ver as avaliações sobre esta release. A Figura 4.31 mostra a tela onde tem o link tanto para avaliar quanto para mostrar as avaliações.

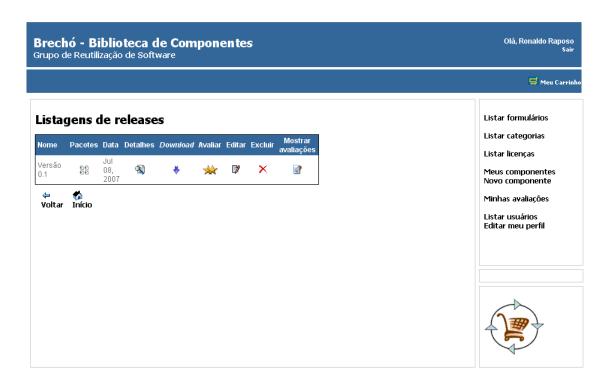


Figura 4.31: Tela que lista as releases de "Evolução", com os links para avaliação e exibição de avaliações para a release 0.1 do componente.

O consumidor faz a avaliação se clicar em Avaliar. A Figura 4.32 mostra a tela onde foi feita uma avaliação sobre esta release.



Figura 4.32: Tela onde os usuários efetuam as avaliações.

Outros usuários visualizam as avaliações se clicarem em Mostrar Avaliações. A Figura 4.33 mostra a tela onde as avaliações são exibidas.

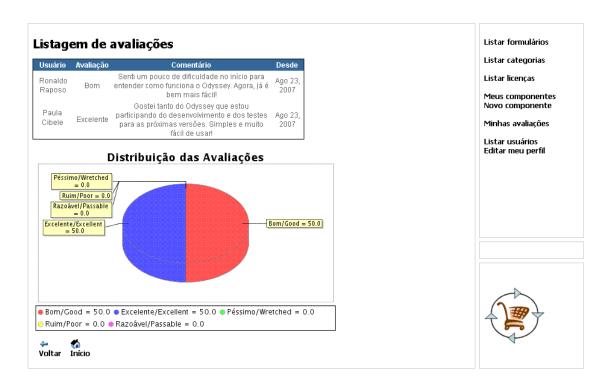


Figura 4.33: Tela onde os usuários visualizam as avaliações.

4.5.4. Exemplo 4: Administrando as Categorias

Depois de os produtores publicarem seus componentes e fazerem sugestões de categorias, o administrador deseja manter organizado o conjunto de categorias. A Figura 4.34 exibe a tela inicial da administração de categorias e sugestões, onde é escolhido um índice de similaridade entre categorias e sugestões.



Figura 4.34: Tela onde o supervisor seleciona o índice de similaridade entre categorias e sugestões.

A Figura 4.35 mostra similaridades entre sugestões e categorias, mostrando sugestões que podem se tornar categorias no futuro.

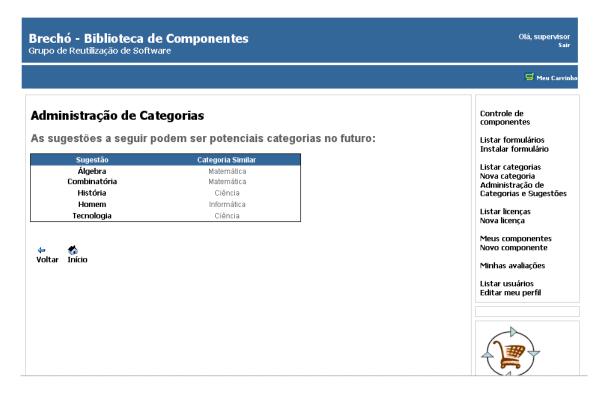


Figura 4.35: Tela onde o supervisor confere sugestões e categorias com algum índice de similaridade.

Se o administrador clicar sobre alguma das sugestões da tabela, ele irá direto para a tela de criação de categorias. Lá, ele pode criar a nova categoria baseada na sugestão dada. Todos os produtores que sugeriram aquele nome serão notificados da criação da nova categoria. A Figura 4.36 mostra a tela de criação de categorias quando o administrador clicar, por exemplo, na sugestão Tecnologia.

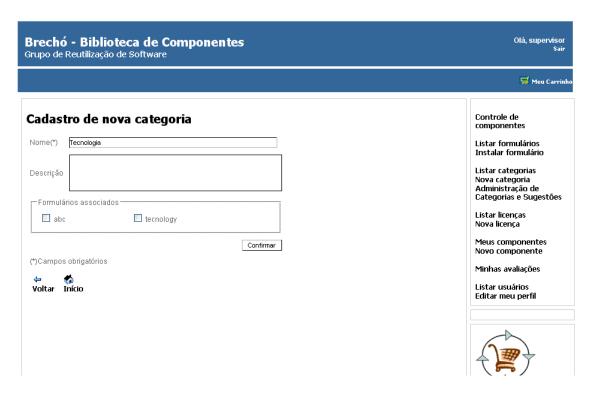


Figura 4.36: Tela onde o administrador cria uma nova categoria a partir de uma sugestão dada pelos usuários.

4.6. Considerações sobre o Capítulo

Neste capítulo, foi apresentada a implementação da abordagem Brechó-ABC. Foram vistos, para os mecanismos de avaliação, busca e categorização, os aspectos de implementação, com detalhes sobre as funcionalidades que cada módulo possui, além da descrição de algumas classes e seus atributos associados. Foi descrito também de que forma, para cada um dos módulos do Brechó-ABC, estas funcionalidades estão acessíveis ao usuário através da interface gráfica da Brechó. Além disso, foram apresentados alguns exemplos de utilização para diferentes situações que podem ocorrer

na biblioteca. As telas ajudam a mostrar como cada mecanismo da Brechó-ABC funciona quando os produtores, consumidores ou o administrador fazem suas tarefas.

No próximo capítulo, serão expostas as conclusões sobre este trabalho. Deste modo, serão apresentadas as considerações finais, as limitações, as contribuições e algumas sugestões de trabalhos futuros.

Capítulo 5 Conclusão

Neste capítulo, serão apresentadas as conclusões sobre este trabalho. Na seção 5.1, serão listadas as considerações finais sobre este trabalho. Na seção 5.2, serão mostradas as contribuições tanto gerais quanto específicas que este trabalho vem acrescentar. Na seção 5.3, as limitações de cada mecanismo serão apresentadas. Finalmente, na seção 5.4, serão sugeridos alguns exemplos de trabalhos futuros que podem ser realizados a partir deste trabalho.

5.1. Considerações Finais

Este trabalho descreveu as dificuldades encontradas em uma biblioteca de componentes quando esta é carente de ferramentas que auxiliem consumidores e produtores no processo de recuperação de componentes. Estes problemas comprometem o funcionamento de bibliotecas que têm como um de seus objetivos principais a disponibilidade de uma infra-estrutura de componentes.

A partir de análise de abordagens feitas no capítulo dois e da proposta apresentada neste trabalho, é possível estabelecer determinadas relações e algumas comparações. Na parte de avaliação, as normas da ISO-15504 serviram como base para a construção do mecanismo. O consumidor responde duas questões: (i) o nível de aceitação do componente, de acordo com o funcionamento do mesmo sob determinadas condições; e (ii) a avaliação do componente, de acordo com os requisitos especificados. Neste sentido, as tarefas relacionadas à avaliação de componentes citadas na norma são cobertas por essas duas questões.

Quanto à busca, a utilização de mais de uma técnica de busca ajuda a aumentar a precisão dos resultados. Com a flexibilização e o refinamento das buscas, os problemas que cada técnica possui podem ser reduzidos, pois os usuários podem "ampliar" a pesquisa utilizando as várias técnicas. Além disso, a busca passa a se utilizar de informações semânticas.

Na parte de categorização, pode-se confrontar cada abordagem citada no capítulo dois com o mecanismo de categorização da Brechó-ABC. A Tabela 6.1 apresenta a comparação de cada abordagem com a proposta neste trabalho, mostrando em que ponto a Brechó-ABC atua de maneira diferente.

Abordagem x Brechó-ABC	Comentários
DesCOTS x Brechó-ABC	Apesar do sistema DesCOTS ter uma hierarquia de
	categorias, ela é rígida, pois as categorias foram
	previamente definidas. No Brechó-ABC, as
	categorias estão em constante modificação. Os
	módulos de sugestão e organização de categorias
	garantem estas mudanças de categorias. Além disso,
	está em construção a hierarquização das categorias
	da Brechó.
SemaCS x Brechó-ABC	O mecanismo de categorização da Brechó-ABC foi
	inspirado no método SemaCS. Este método só
	permite a sugestão manual de categorias, além de
	procurar a partir de alguns sites outras sugestões. O
	Brechó-ABC apóia os produtores trazendo
	sugestões de sites da Web. Eles, então, podem
	escolher as que mais satisfazem às especificações do
	componente. Além disso, o Brechó-ABC traz uma
	forma distinta de organizar as categorias, sem a
	necessidade de construção de ontologias.
Componentes educacionais	Nestas duas abordagens, além das categorias serem
(LALEUF e SPALTER, 2001) x	previamente definidas, elas são classificadas em
Brechó-ABC	grandes grupos. Apesar de serem bem definidos, os
	grupos poderiam gerar maior informação semântica e ser mais flexíveis. O Brechó-ABC, além de ter
Componentes de projeto	categorias em permanente mudança, pode ter
Componentes de projeto (SAMETINGER e KELLER,	componentes classificados em várias categorias.
2002) x Brechó-ABC	Com isso, a árvore de categorias é menos rígida e os
2002) A BIECHO-ABC	
	componentes, por gerar mais informações, têm maiores chances de serem encontrados na busca.

Tabela 5.1: Comparativo das abordagens de categorização citadas na literatura com o Brechó-ABC.

O uso de mecanismos de avaliação, busca e categorização permitem, nestas condições: (i) o controle organizado dos componentes, que estarão classificados de maneira coesa; (ii) o surgimento de técnicas automatizadas e flexíveis de busca, com a finalidade de oferecer resultados cada vez mais precisos; e (iii) a criação de novas formas de agregação de informação por parte dos consumidores dos componentes, tanto quantitativa quanto qualitativa.

O usuário tem função importante em todos os mecanismos. Na avaliação, ele qualifica o componente e deixa os seus comentários para que outros usuários se utilizem destas informações ao decidir se vale a pena adquirir o componente. Além disso, ele conta com técnicas que limitam o espaço de busca e facilitam a pesquisa de componentes.

5.2. Contribuições

Podem-se destacar, entre as principais contribuições gerais deste trabalho, as seguintes:

- ✓ A proposta de uma abordagem de avaliação, busca e categorização no contexto de uma biblioteca de componentes, denominada Brechó-ABC;
- ✓ A descrição de uma arquitetura de alto nível para mecanismos que utilizem esta abordagem;
- ✓ A implementação de um protótipo que automatiza a abordagem Brechó-ABC e sua integração à biblioteca Brechó.

Existem, ainda, contribuições específicas de cada mecanismo. Na parte de avaliação, podem ser destacadas as seguintes:

- ✓ A indicação de informações quantitativas sobre o componente, como por exemplo, o nível de satisfação atribuído pelos consumidores e o gráfico de setores que mostra a porcentagem de avaliações, de acordo com cada um dos níveis de satisfação do componente;
- ✓ A indicação de informações qualitativas sobre o componente, como por exemplo, os comentários que os usuários fazem sobre o componente adquirido.

Para a parte de busca, destacam-se as seguintes contribuições:

- ✓ As técnicas de busca com refinamentos e flexibilizações limitam o espaço de busca, facilitando a pesquisa de componentes. Isto torna o processo de recuperação de componentes mais ágil;
- ✓ A correção de erros de digitação ajuda os usuários a perceber os erros cometidos de maneira rápida e automatizada;
- ✓ A pesquisa em dicionários na Web por sugestões de palavras com grafias similares mostra uma forma diferente de busca corretiva, sem a necessidade de se carregar um dicionário.

Para o mecanismo de categorização, as seguintes contribuições podem ser citadas:

- ✓ O fato da sugestão de categorias e a organização das categorias e sugestões serem técnicas semi-automáticas, ajudando os usuários e os administradores nestas tarefas, que são complicadas se feitas manualmente;
- ✓ A pesquisa por sugestões e relações de similaridade na Web expõe um modo diferente de recuperar informação, sem a necessidade de se carregar, por exemplo, um tesaurus ou uma ontologia;
- ✓ A relação de similaridade retirada de (CILIBRASI e VITANYI, 2007) é mais uma variável que pode ser utilizada, sem a necessidade de se carregar um tesaurus ou uma ontologia para se descobrir esta relação.

5.3. Limitações

Cada mecanismo possui algumas limitações específicas. Na parte de avaliação, existem apenas duas perguntas que o consumidor responde. Poderia-se incluir questões mais específicas sobre o componente. Estas questões poderiam ser sobre assuntos relevantes como, por exemplo, segurança e desempenho, entre outros. Desta forma, as respostas poderiam agregar mais informações quantitativas e qualitativas para outros usuários que desejam comprar um componente.

Na parte de busca, duas limitações foram identificadas. Uma diz respeito aos elementos presentes na documentação e que podem ser refinados depois da busca por categoria. Estes elementos aparecerem numa tela como campos. Exemplos de campos são botão de escolha (*checkbox*), botão de rádio (*radio button*), caixa de texto (*text box*), entre outros. A limitação é que alguns campos não podem ser utilizados para ajudar neste refinamento. Campos como área de texto (*textarea*) e outros campos para armazenamento de figuras se tornam difíceis de serem refinados, pois é pouco provável que existam duas áreas de texto exatamente iguais. Por isso, estes campos não estão sendo utilizados para refinamento de buscas.

Outra limitação na parte de busca é a possibilidade do dicionário da Web estar fora do ar ou estar muito ocupado. Isto pode comprometer o funcionamento da busca sintática e não retornar nenhuma sugestão de palavra com grafia similar. Na parte de

categorização, esta limitação também existe, pois se algum dos motores de busca estiver com problemas, a técnica de sugestão de categorias e a organização das sugestões e categorias estará também ameaçada de não funcionar corretamente.

Na parte de categorização, também existe o problema de se levar muito tempo para se retornar as sugestões da Web ou para se encontrar as relações de similaridade e, assim, a biblioteca ficar lenta para estas tarefas. Aliás, a tecnologia para a comunicação entre os robôs e os sites teve de ser feita a partir de páginas HTML, pois, durante a implementação, ocorreram mudanças em alguns dos sites de busca, como o Google e o Yahoo, que mudaram a forma como se buscava informações a partir de serviços Web. Logo, para não se arriscar e ter que alterar toda a implementação por causa desta mudança, a comunicação entre robôs e sites acabou sendo feita deste modo.

Além disso, alguns sites de busca podem trazer resultados imprecisos ou incoerentes, fazendo com que apareçam para o usuário. Por esta razão, existem quatro opções de sites que podem ser pesquisados, e sempre que aparecer um novo mecanismo de busca, este pode ser configurado e fazer parte desta lista.

5.4. Trabalhos Futuros

Como sugestão de trabalhos futuros, na parte de avaliação, pode-se citar a inclusão de questões ligadas a desempenho e segurança, entre outros assuntos. Além disso, é sugerida a utilização de um modelo de qualidade para componentes baseado nas normas vistas no capítulo dois, com as perguntas referentes a cada um dos indicadores presentes nas normas.

Na parte de busca, sugere-se a criação de buscas textuais que considerem sinônimos e palavras relacionadas. Além disso, que se utilizem formas sintáticas baseadas em assinaturas de serviços para oferecer mais precisão nos resultados da pesquisa. Outros técnicas de busca, como por exemplo, a de redes neurais (BARROS, 1995), podem ser utilizadas para se tornarem mais uma opção para o usuário.

Na parte de categorização, trabalhos futuros podem envolver a criação de métodos que façam a comunicação entre os motores de busca por meio de serviços Web e a criação de um mecanismo que sugira, automaticamente no ato da publicação, uma categoria (dentre as que existem) para o componente.

Referências Bibliográficas

ALVARO, A., MEIRA, S. R. L., ALMEIDA, E. S., 2005, "Quality Attributes for a Component Quality Model". In: 10th International Workshop on Component-Oriented Programming (WCOP) in Conjunction with the 19th European Conference on Object Oriented Programming (ECOOP), Glasgow, Escócia, julho.

APACHE, 2007, Apache Struts. In: < http://struts.apache.org>, acessado em agosto de 2007.

AURÉLIO, B. H. F., 2005, "Mini Dicionário Aurélio da Língua Portuguesa", 6^a ed., Curitiba, Brasil: Positivo.

BAAEZA-YATES, R.A., RIBEIRO-NETO, B.A., 1999, "Modern Information Retrieval". ACM Press Series/Addison Wesley, Nova Iorque, maio.

BARROS, M.O., "Recuperação de Componentes em Bibliotecas de Software: Uma Abordagem Conexionista". Tese de M.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 1995.

BASS, L., BUHMAN, C., COMELLA-DORDA, S., LONG, F., ROBERT, J., SEACORD, R., WALLNAU, K., 2000, "Market Assessment of Component-Based Software Engineering, Technical Report, Software Engineering Institute", CMU/SEI-2001-TN-007.

BOEHM, B.W., BROWN, J.R., LIPOW, M., 1976, "Quantitative Evaluation of Software Quality". In: *Proceedings of the 2nd International Conference on Software Engineering*, San Francisco, California, Estados Unidos, pp. 592-605, outubro.

BRAGA, R.M.M., WERNER, C.M.L, MATOSO, M., 2006, "OdysseySearch: A Multi-Agent System for Component Information Search and Retrieval", Journal of Systems and Software, (vol. 79, n° 2), pp. 204-215, ISSN 0164-1212.

CILIBRASI, R., VITANYI, P., 2007, "Automatic Meaning Discovery Using Google". In: http://xxx.lanl.gov/abs/cs.CL/0412098>, accessado em agosto de 2007.

COMPONENT SOURCE, "Component Source®", 2007. In: http://www.componentsource.com/index.html, acessado em agosto de 2007.

DROMEY, R.G., 1995, "A model for software product quality". In: *IEEE Transactions on Software Engineering 21*, pp. 146-162, fevereiro.

FISCHER, G., HENNINGER, S., REDMILES, D., 1991, "Cognitive Tools for Locating and Comprehending Software Objects for Reuse". In: *Proceedings of 13th International Conference on Software Engineering (ICSE'91)*, IEEE Computer Society, Austin, TX, pp. 318–328, maio.

FRAKES, W.B., 2007, "Software Reuse". In: http://frakes.cs.vt.edu/SEportalReuse.htm, acessado em agosto de 2007.

FRAKES, W.B., KYO KANG, K., 2005, "Software Reuse Research: Status and Future", IEEE Transactions on Software Engineering, v.31, n.7.

FANCHAO, M., DECHEN, Z., XIAOFEI, X., 2006, "A Specification-Based Approach for Retrieval of Reusable Business Component for Software Reuse". International

- Journal of Computer Science Volume 1 Number 4, ISSN 1306-4428, pp. 283-290, dezembro.
- GOOGLE, 2007, Google Diretório. In: http://www.google.com.br/dirhp?hl=pt-BR>, acessado em agosto de 2007.
- GOULÃO, M., BRITO e ABREU, F., 2004, "Software Components Evaluation: an Overview". In: *Proceedings of 5^a Conferência da Associação Portuguesa de Sistemas de Informação*, Lisboa, Portugal, novembro.
- GRADY, R.B., CASWELL, D.L., 1987, "Software Metrics: Establishing a Company-Wide Program". Englewood Cliffs, New Jersey, Estados Unidos: Prentice-Hall.
- GRAU, G., CARVALLO, J.P., FRANCH, X., QUER, C., 2004, "DesCOTS: A Software System for Selecting COTS Components". In: *Proceedings of the 30th EUROMICRO Conference*, Rennes, France, setembro, pp 118-126, IEEE Computer Society.
- GUO, J., LUQI, 2000, "A Survey of Software Reuse Repositories". In: 7th IEEE International Conference and Workshop on the Engineering of Computer Based Systems, Endinburgo, Escócia, abril, pp. 92-100.
- ISO/IEC 9126, 1991, "Information Technology Software Product Evaluation Software Quality Characteristics and Metrics", Genebra, Suíça: International Organization for Standardization.
- ISO/IEC TR 15504, 1999, "Information Technology Software Process Assessment Part 5: An Assessment Model and Indicator Guidance", Genebra, Suíça: International Organization for Standardization.
- KARTOO, 2007, KartOO Metamotor de Pesquisa. In: http://www.kartoo.com, acessado em agosto de 2007.
- KIM, Y., STOHR, E.A., 1998, "Software reuse: survey and research directions". In: *Journal of Management Information Systems*, v.14 n.4, pp.113-147, março.
- LALEUF, J.R., SPALTER, A.M., 2001, "A Component Repository for Learning Objects: A Progress Report". In: *Proceedings of the first ACM/IEEE-CS joint conference on Digital Libraries*, Roanoke, Virginia, Estados Unidos, junho, pp. 33-40, ACM Press.
- LUCRÉDIO, D., PRADO, A.F., ALMEIDA, 2004, E.S., "A Survey on Software Components Search and Retrieval". In: *Proceedings of the 30th EUROMICRO Conference*, Rennes, França, setembro, pp. 152-159
- McCALL, J., 1994, "Quality Factors". In: *Encyclopedia of Software Engineering*, vols. I & II, pp. 958-ff: John Wiley & Sons.
- MITTERMEIR, R.T., POZEWAUNIG, H., MILI, A., MILI, R., 1998, "Uncertainty Aspects in Component Retrieval". In: *Proceedings of the 7th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Paris, França, julho, pp. 564-571.
- MYSQL AB, 2007, Sistema Gerenciador de Banco de Dados MySQL. In: http://www.mysql.com>, acessado em agosto de 2007.
- NETSCAPE, 2007, Dmoz Open Directory Project. In: <<u>http://www.dmoz.org</u>>, acessado em agosto de 2007.

NYMAN, M., NÅLS, A., 2004, "Software Component Quality". In: *Seminar in Software Quality (G717)*, Faculty of Chemical Engineering, Åbo Akademi University, Finlândia, dezembro 1.

OBJECT REFINERY, 2007, JFreeChart. In: http://www.jfree.org/jfreechart, acessado em agosto de 2007.

PRESSMAN, R. S., 2002, "Engenharia de Software", 5^a ed., Rio de Janeiro, Brasil: McGraw-Hill.

PRIBERAM, 2007, Dicionário da Língua Portuguesa On-line. In: http://www.priberam.pt/dlpo/dlpo.aspx, acessado em agosto de 2007.

RAVICHANDRAN, T., ROTHENBERGER, M.A., 2003, "Software reuse strategies and component markets", *Communications of the ACM*, v.46 n.8, p.109-114, agosto.

RAWASHDEH, A., MATALKAH, B., 2006, "A New Software Quality Model for Evaluating COTS Components". In: *Journal of Computer Science* 2(4), pp. 373-381.

REDHAT, 2007, *Framework* Hibernate. In: < http://www.hibernate.org>, acessado em agosto de 2007.

SAMETINGER, J., KELLER, R., 2002, "Compositional Design Reuse". In: *CACIC, VIII Argentinean Conference on Computer Science*, Universidad de Buenos Aires, Argentina, outubro 15-18.

SEBASTIANI, F., 2002, "Machine Learning in Automated Text Categorization". In: *ACM Computing Surveys*, Vol. 34, N° 1, março, pp. 1-47.

SHNEIDERMAN, B., BYRD, D., CROFT, W.B., 1997, "Clarifying Search: A User-Interface Framework for Text Searches", D-Lib Magazine, janeiro, ISSN 1082-9873. In: http://www.dlib.org/dlib/january97/retrieval/01shneiderman.html>, acessado em agosto de 2007.

SJACHYN, M., BEUS-DUKIC, L., 2006, "Semantic Component Selection – SemaCS". In: *Fifth International Conference on Commercial-off-the-Shelf (COTS)-Based Software Systems*, IEEE, Los Alamitos, USA, fevereiro, pp. 83-89.

SOURCE FORGE, "Source Forge, Inc.", 2007. In: http://sourceforge.net/index.php, acessado em agosto de 2007.

SUGUMARAM, V., SOTREY, V.C., 2003, "A Semantic-Based Approach to Component Retrieval". The DATABASE for Advances in Information Systems – Summer 2003, Vol. 34, N° 3.

SUN, 2007, "Java Technology". In: < http://java.sun.com>, acessado em agosto de 2007.

VIVÍSIMO, 2007, Clusty Search. In: http://clusty.com, acessado em agosto de 2007.

WANG, L., MEHANDJISKA-STAVREVA, D., 2004, "An Initial Framework for Collaboration-Based Component Selection". In: *Proceedings of Software Engineering Research and Practice*, Las Vegas, Nevada, Estados Unidos, junho, pp. 799-806.

WERNER, C.M.L., BRAGA, R.M.M., 2000, "Desenvolvimento Baseado em Componentes". In: *Anais do Simpósio Brasileiro de Engenharia de Software*, Minicursos, João Pessoa, PB, Brasil, outubro, pp. 297-329.

WERNER, C. M. L., MURTA, L. G. P., LOPES, M., DANTAS, A., LOPES, L. G., FERNANDES, P., PRUDENCIO, J. G., MARINHO, A., RAPOSO, R., 2007, "Brechó:

Catálogo de Componentes e Serviços de Software". In: XXI Simpósio Brasileiro de Engenharia de Software, Sessão de Ferramentas, João Pessoa, outubro. XXI Simpósio Brasileiro de Engenharia de Software, Sessão de Ferramentas (aceito para publicação).

WIKIPEDIA, "Wikimedia Foundation, Inc.", 2007. In: http://pt.wikipedia.org/wiki/Categorização>, acessado em agosto de 2007.

YAHOO, 2007, Yahoo Busca Diretório. In: http://br.yahoo.com/info/diretorio.html>, acessado em agosto de 2007.

YANG, Y., PEDERSEN, J.O., 1997, "A Comparative Study on Feature Selection in Text Categorization". In: *Proceedings of the 14th International Conference on Machine Learning*, ICML'97, Nashville, Tennessee, Estados Unidos, julho 8-12, pp. 412-420.

YE,Y., 2001, "Supporting Component-Based Software Development with Active Repository Systems". Tese de PhD, Universidade do Colorado.

ZHANG, Z., 2000, "Enhancing component reuse using search techniques". In: *Proceedings of the 23rd Conference on Information System Research in Scandinavia (IRIS23)*, Lingatan, Sweden, agosto.