

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

GABRIEL DA SILVA SIMÕES

**Qualificando Autores em um Processo
Aberto para Editoração de Artigos**

Dissertação apresentada como requisito parcial
para a obtenção do grau de
Mestre em Ciência da Computação

Prof. Dr. José Palazzo Moreira de Oliveira
Orientador

Prof. Dr. Leandro Krug Wives
Co-orientador

Porto Alegre, agosto de 2008

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Simões, Gabriel da Silva

Qualificando Autores em um Processo Aberto para Editoração de Artigos / Gabriel da Silva Simões. – Porto Alegre: PPGC da UFRGS, 2008.

76 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2008. Orientador: José Palazzo Moreira de Oliveira; Co-orientador: Leandro Krug Wives.

1. Colaboração. 2. Confiança. 3. MediaWiki. 4. Qualificação. 5. Reputação. 6. Wiki. I. Oliveira, José Palazzo Moreira de. II. Wives, Leandro Krug. III. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. José Carlos Ferraz Hennemann

Vice-Reitor: Prof. Pedro Cezar Dutra Fonseca

Pró-Reitora de Pós-Graduação: Prof^a. Valquíria Linck Bassani

Diretor do Instituto de Informática: Prof. Flávio Rech Wagner

Coordenadora do PPGC: Prof^a. Luciana Porcher Nedel

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

*“Não a potência, mas a duração de um sentimento
elevado forma os homens superiores.”*

— NIETZSCHE

AGRADECIMENTOS

A arte de fazer Ciência requer uma rotina diária de superação. Para isto, é fundamental buscar constantemente por apoio e colaboração. Este é o momento de lembrar daqueles que não se omitiram e contribuíram para a realização deste trabalho.

Giseli Lopes foi minha colega durante todo o período em que frequentei o II. Por mais engraçado que possa parecer, fomos descobrir depois de algum tempo de convivência que eramos conterrâneos. Dois riograndinos que acabaram dividindo o mesmo espaço. Agradeço à Giseli por todo o companheirismo durante esta jornada, especialmente pela ajuda nos cálculos e equações. Agradeço também pela paciência na hora das músicas.

Outro conterrâneo que também conheci aqui no II foi o Eduardo Borges. Não nos conhecíamos por muito pouco, porque tínhamos amizades bem próximas. O Eduardo me ajudou bastante também nos cálculos, já que assim como a Giseli, ele tem formação em Engenharia. Quando comparado a um Analista de Sistemas, um Engenheiro vai ser sempre infinitamente melhor em cálculo. Viajamos para João Pessoa, para o SBBB, e para Gramado, para o Webmedia. Foi muito legal. Agradeço muito pela ajuda e pelo companheirismo.

Amigos dos tempos de Graduação, Gustavo Piltcher, Marcos Nunes e Otávio Acosta chegaram ao II em março de 2007, iniciando seus Mestrados. Fiquei muito contente pelo ingresso deles. Continuamos aqui no II uma amizade que já vinha desde 2003. Procurei passar para eles as experiências que vivi junto ao Programa desde 2006, quando concluí os 24 créditos. Em 2007, os 3 também concluíram seus 24 créditos. Fico feliz em ver que estão todos com seus trabalhos bem adiantados, e que com certeza vencerão esta etapa com muito sucesso. Parabéns para vocês. Obrigado pela ajuda e pela companhia também aqui no II.

Além dos conterrâneos, mantive contato com vários outros colegas que, no passar destes 2 anos, contribuíram com apoio e descontração. Vou procurar citar todos, mas já peço desculpas caso esqueça de alguém. Alex, Sergio, Eduardo Pivetta, Deise, Euler, Adrovane, Maurício, Mariusa, Carolina, Lucineia, Guilherme, Rúbia, Renata, Kelly. Obrigado por tudo. Foi muito bom conviver com vocês.

Quando se trabalha com ciência, os exemplos são fundamentais. Caminhos errados são infinitos, enquanto que os certos são bem poucos. Orientação é fundamental na vida de todo o pesquisador, especialmente para os mais jovens como eu. Agradeço ao professor José Palazzo Moreira de Oliveira por me orientar durante estes mais de vinte meses de trabalho. Além de ser uma pessoa com um astral altíssimo, sempre motivado a encarar os desafios com criatividade, o professor Palazzo carrega consigo uma experiência invejável, formando dezenas de Mestres e Doutores em seus mais de trinta anos de atividade acadêmica. Obrigado pela confiança e pelo auxílio durante esta jornada. Preciso lembrar também daquele que, como eu, um dia também contou com a orientação do Professor

Palazzo e hoje faz parte do corpo docente do II. Agradeço ao Professor Leandro Krug Wives, que me co-orientou nesta jornada, pela ajuda e dedicação.

Recentemente, o II recebeu da CAPES a qualificação nível 6 no ranking dos Programas de Pós-Graduação em Computação do país. Somente três Programas em todo o Brasil conseguiram alcançar esta margem, a máxima que os Programas de Computação chegaram na última avaliação. O PPGC/II UFRGS está de parabéns por atingir esta marca. Esta conquista não veio por acaso. Nos dois anos que freqüentei estas dependências, pude sentir a vibração que este objetivo gerou junto à Direção. Parabenizo a todos por esta conquista. É muito bom poder contar com esta marca em nossos Currículos.

Jovens recém-graduados geralmente imaginam que o mundo após a formatura se transforma radicalmente. Infelizmente, às vezes, isto não é verdade. Muitas vezes, encontrar colocação no mercado torna-se uma tarefa bem mais difícil que concluir a própria Graduação. Para aqueles que optam por esticar a vida acadêmica, esta história se repete. Surge então o desafio de conseguir uma bolsa de estudos. Agradeço à CAPES por custear minha bolsa de estudos. Considero-me privilegiado em poder contar com este auxílio. Em um país como o nosso, que não olhou para a educação com bons olhos por muito tempo, ser pago para estudar é algo místico. Espero que, no futuro, esta visão mude bastante, permitindo que todos possam seguir seus estudos o quanto julgarem necessário.

Por fim, reservo este parágrafo para minha família, em especial meus pais. Por menos que possa parecer, a vida acadêmica também é repleta de angústias e desafios. Continuar estudante em um mundo que valoriza muito mais o material do que o saber exige estabilidade e o conforto daqueles poucos que seguem uma postura diferente. Obrigado pelo apoio e pelo incentivo. Não fosse por poder contar com este porto seguro, certamente eu não chegaria ao fim de mais esta jornada.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	8
LISTA DE FIGURAS	10
LISTA DE TABELAS	11
RESUMO	12
ABSTRACT	13
1 INTRODUÇÃO	14
1.1 Organização do Texto	15
2 TRABALHOS RELACIONADOS	17
2.1 Philica	17
2.2 ACP (<i>Atmospheric Chemistry and Physics</i>)	18
2.3 JIME (<i>Journal of Interactive Media in Education</i>)	19
2.4 PLoS ONE	20
2.5 arXiv	20
3 PROCESSOS DE PRODUÇÃO DE ARTIGOS CIENTÍFICOS	22
3.1 O Processo de Revisão Tradicional (<i>Peer Review</i>)	22
3.2 O Processo de Produção Utilizado	24
3.3 Classes de Usuários	26
3.3.1 Autores	26
3.3.2 Revisores	26
3.3.3 Comentaristas	28
4 MÉTODOS PARA QUALIFICAÇÃO	29
4.1 O Método EQ1	31
4.1.1 Normalizando Qualificações em EQ1	32
4.1.2 Definição do Método EQ1	34
4.2 O Método EQ2	34
4.2.1 Definição do Método EQ2	35
4.2.2 Exemplo de indicação de pontuações utilizando EQ2	36
4.3 Ranking de Qualificação	36

5	O AMBIENTE MEDIAWIKI	38
5.1	Uma Breve História Sobre Ambientes Wiki	39
5.2	Funcionalidades Originais do MediaWiki	39
5.3	Fatores de Implementação do Mediawiki	40
5.4	A Linguagem de Marcação <i>Wikitext</i>	41
5.5	Arquitetura do MediaWiki	42
5.6	A Hierarquia de Classes	42
5.6.1	A Classe Parser	43
5.7	A Arquitetura da Base de Dados	44
5.8	<i>Namespaces</i>	44
5.9	Mecanismos de Extensão	45
5.9.1	<i>SpecialPages</i>	46
5.9.2	<i>Hooks</i>	46
5.10	Controle de Versões	46
5.11	Documentação	47
6	O PROTÓTIPO IMPLEMENTADO	50
6.1	Cadastro de Usuários	51
6.2	Classes de Usuários	52
6.3	Criação e Edição de Artigos	52
6.4	Colhendo Dados e Eventos para Qualificação	54
6.4.1	Dados Quantitativos	54
6.4.2	Dados Qualitativos	54
6.5	Qualificando os Usuários com EQ1	55
6.6	Rankings de Qualificação	56
6.7	<i>Dashboards</i>	56
6.8	Outras Aplicações para o Protótipo	58
7	EXPERIMENTOS	60
7.1	Experimento 1 - Utilizando Notícias de Times de Futebol	60
7.1.1	O Experimento	61
7.1.2	Resultados	61
7.2	Experimento 2 - Utilizando Artigos da Wikipedia	64
7.2.1	O Experimento	64
7.2.2	Resultados	64
8	CONCLUSÕES E TRABALHOS FUTUROS	66
8.0.3	Publicações Resultantes	67
8.1	Trabalhos Futuros	67
	REFERÊNCIAS	69
	APÊNDICE A ARQUITETURA DA BASE DE DADOS DO MEDIAWIKI	73
	APÊNDICE B COMPARANDO QUALIFICAÇÕES GERADAS NO EXPERIMENTO 1	75

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
ACP	<i>Atmospheric Chemistry and Physics</i>
ACPD	<i>Atmospheric Chemistry and Physics Discussions</i>
ADO	<i>ActiveX Data Objects</i>
AJAX	<i>Asynchronous Javascript And XML</i>
ASP	<i>Active Server Pages</i>
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CC	Coeficiente de Competência
CCAL	<i>Creative Commons Attribution License</i>
CGI	<i>Common Gateway Interface</i>
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
EDI	<i>Electronic Data Interchange</i>
EGU	<i>European Geosciences Union</i>
EQ1	Equação para Qualificação 1
EQ2	Equação para Qualificação 2
FAPERGS	Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul
FTP	<i>File Transfer Protocol</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
II	Instituto de Informática
JIME	<i>Journal os Interactive Media in Education</i>
JDBC	<i>Java Database Connectivity</i>
JSP	<i>Java Server Page</i>
KB	<i>Kilobyte</i>
KBPS	<i>Kilobits per second</i>
KMI	<i>Knowledge Media Institute</i>

OO	Orientação a Objetos
PHP	<i>Hypertext Processor</i>
PPGC	Programa de Pós-Graduação em Computação
SGBD	Sistema Gerenciador de Bancos de Dados
SMRC	<i>Sweden Medical Research Council</i>
SOAP	<i>Simple Object Access Protocol</i>
SQL	<i>Structured Query Language</i>
TCL	<i>Tool Command Language</i>
UFRGS	Universidade Federal do Rio Grande do Sul
URL	<i>Uniform Resource Locator</i>
W3C	<i>World Wide Web Consortium</i>
WYSIWYG	<i>What You See Is What You Get</i>

LISTA DE FIGURAS

Figura 2.1:	A interface para submissão do Philica	18
Figura 2.2:	Processo de publicação utilizado pelo JIME. (ONLINE JIME, 2008) .	19
Figura 2.3:	<i>e-print</i> sobre Bibliotecas Digitais no arXiv	21
Figura 3.1:	Processo aberto produção e revisão de artigos científicos	25
Figura 3.2:	Diferentes versões após revisão realizada pelo usuário Gsimoes . . .	27
Figura 3.3:	Ciclo de versões e contribuições	28
Figura 4.1:	Atribuindo pontuação com o método EQ1	32
Figura 5.1:	Wikimarkup e sua respectiva saída	42
Figura 5.2:	Os métodos públicos da classe Parser	43
Figura 5.3:	A interface de documentação gerada pelo Doxygen	48
Figura 6.1:	Definição de módulos por funcionalidades	50
Figura 6.2:	Formulário de cadastro de novos usuários	51
Figura 6.3:	Formulário de gestão de privilégios de usuários	52
Figura 6.4:	Ligações internas em azul e vermelho	53
Figura 6.5:	Interface de apoio à edição de artigos	54
Figura 6.6:	Componente visual junto de um artigo intitulado <i>Delphi</i>	55
Figura 6.7:	Interface do <i>ranking</i> de qualificações	56
Figura 6.8:	Os gráficos A e B que aparecem no <i>dashboard</i> de usuários	57
Figura 6.9:	A ligação que aponta para os <i>dashboards</i> de um usuário	58
Figura 7.1:	Qualificações após Experimento 1 (Todos os 10 usuários)	62
Figura 7.2:	Qualificações após Experimento 1 (Excluindo o usuário A)	63
Figura 7.3:	Qualificações após Experimento 1 (Excluindo os usuários A e B) . .	63

LISTA DE TABELAS

Tabela 4.1:	Valores atribuídos para avaliações utilizando EQ1 e EQ2	31
Tabela 4.2:	Exemplo de normalização de qualificações	33
Tabela 4.3:	Exemplo de indicação de pontuações utilizando EQ1	35
Tabela 4.4:	Exemplo de indicação de pontuações utilizando EQ2	36
Tabela 4.5:	Exemplo de <i>Ranking</i> de qualificação utilizando EQ1	37
Tabela 5.1:	<i>Namespaces</i> e suas páginas relacionadas	45
Tabela 5.2:	Exemplos de hooks e seus respectivos eventos	47
Tabela 7.1:	Avaliações recebidas pelos usuários A e B	62
Tabela 7.2:	Qualificações EQ1 e EQ2 no experimento 1	64
Tabela 7.3:	Dados de cada trabalho	65

RESUMO

A Web 2.0 coloca em evidência iniciativas que valorizam atividades comunitárias, colaborativas e democráticas. Os processos de filtragem e seleção de artigos para publicação científica caminham na contra-mão destes novos conceitos. A seleção de artigos para publicação, tanto em conferências quanto em periódicos, utiliza abordagens fechadas. O objetivo deste processo fechado é garantir que somente artigos de qualidade sejam publicados. Por outro lado, devido a falta de transparência, o processo fechado está sujeito a uma infinidade de problemas que podem surgir através de equívocos ou até mesmo de tendências anti-éticas. Constam na literatura vários exemplos de problemas com o processo de seleção fechado.

Com a consolidação da Web 2.0, ambientes abertos de edição de conteúdo conquistam cada vez mais espaço. Aproveitam contribuições espontâneas de usuários para crescer rapidamente. A aplicação do processo de edição utilizado por estes ambientes em um contexto de edição e revisão de artigos científicos pode modificar radicalmente o processo utilizado atualmente. Por outro lado, por estarem sujeitos ao acesso de usuários não comprometidos, ambientes abertos não apresentam fortes argumentos para estabelecer confiança entre usuários.

Este trabalho apresenta um método dinâmico para indicação de pontos de qualificação aplicado em um protótipo desenvolvido para editoração de artigos. O protótipo segue um processo onde todas as etapas que envolvem edição, revisão e comentários dos artigos são abertas e transparentes. As qualificações indicadas pelo método são geradas a partir de dados quantitativos e qualitativos e são utilizadas para gerar um *ranking* de qualificação dos usuários. Este *ranking* tem o objetivo de contribuir para o estabelecimento de confiança entre usuários.

Palavras-chave: Colaboração, confiança, mediaWiki, qualificação, reputação, wiki.

A open process for edition and review of articles implemented over Mediawiki

ABSTRACT

Web 2.0 evidence initiatives over communitarian, collaborative and democratic activities. The filtering processes for selection of papers for scientific publication goes against these new concepts. The selection of paper for publications in conferences and periodics uses closed review. The objective of this closed process is to guarantee the quality of these published papers. On the other hand, the closed process has transparency lack. It is subject to a infinity of problems that can even appear through mistakes or of anti-ethics trends. Some examples of problems with the closed process consists in literature.

The consolidation of Web 2.0 brings more attention to open content edition environments. These environments use spontaneous user's contributions to enlarge quickly. The application of the process used by open content edition environments can strongly change the currently closed process used to selection of scientific publications. On the other hand, the open process is subject to fake or not compromised users. This problem starts a lack of trust for unknown users.

This work presents a method for qualification points generation applied in a prototype for edition and review of articles that extends MediaWiki, the Wikipedia's environment. The prototype implements a process where all steps of edition, comment and review of articles are open. To complete lacks of trust between users, the prototype uses a dynamic qualification method. The qualifications indicated by this method are based in quantitative and qualitative data and the users are listed in a ranking ordered by best ratings.

Keywords: collaboration, mediaWiki, qualification, reputation, trust, wiki.

1 INTRODUÇÃO

Ambientes abertos de edição de conteúdo, conhecidos como ambientes Wiki¹, são tão confiáveis quanto os tradicionais ambientes fechados (GILES, 2005). Ambientes abertos permitem que o montante de informação disponível cresça rapidamente, já que visam a edição coletiva. A Wikipedia², maior exemplo de aplicação Wiki disponível na Internet, surgiu em 2001 e atualmente, com relação a tópicos gerais, cobre conteúdo equivalente ao coberto por enciclopédias fechadas (HALAVAIS; LACKAFF, 2008). Quando utilizados no meio empresarial, ambientes Wiki ajudam na colaboração entre equipes de trabalho (MAJCHRZAK; WAGNER; YATES, 2006). Nestes ambientes, cada documento disponível é constantemente revisado por membros da comunidade. Os membros da comunidade são usuários que acessam os ambientes freqüentemente. As revisões permitem que a versão final de um documento torne-se um somatório de contribuições de várias pessoas com habilidades diferentes em tópicos específicos do assunto abordado pelo trabalho. Estas abordagens trazem um grande montante de conhecimento, criando um repositório anteriormente deficitário na Web (SPROULL; KIESLER, 1992).

Diferente da abordagem aberta encontrada em Wikis, o processo de edição e revisão tradicional de artigos, empregado atualmente na maior parte dos eventos e periódicos científicos, utiliza revisões “cegas” (*blind review*). Neste processo, o número de revisores é geralmente muito restrito frente à crescente quantidade de submissões. Isto pode diminuir o número de contribuições que um revisor propõe para cada trabalho, limitando a qualidade das revisões. Santini (2005) levanta o seguinte questionamento: o que pode acontecer com um bom trabalho caso um revisor tenha passado por um dia ruim? Outro problema é que nem sempre o revisor mais qualificado para revisar um trabalho é conhecido (POSCHL, 2004). Bons trabalhos podem perder espaço quando comparados a outros superficiais, um problema freqüente levantado por Parnas (2007). Este problema é decorrente da valorização da quantidade de publicações frente à qualidade das mesmas. Como pode ser observado em Ren (2007), muitos *rankings* de qualificação de autores levam em conta somente o número de publicações e citações, não estabelecendo critérios mais aprofundados na avaliação.

Um sistema Web, aberto à comunidade, que permita criar, editar, indexar e buscar artigos pode contribuir positivamente no contexto das revisões, tornando o processo mais transparente e democrático. Se este sistema permitisse também que os artigos fossem discutidos e revisados, utilizando uma abordagem aberta, seria criado um novo canal para contribuições científicas, ajudando a aumentar a qualidade das publicações. Análoga a estas idéias está a proposta dos ambientes Wiki, visando a disponibilização de conteúdo

¹ Abreviação de *Wikiwiki*, termo havaiano que significa “rápido”.

² <http://www.wikipedia.org>

na Web de forma aberta e colaborativa, permitindo que qualquer usuário possa participar da construção deste (CURRAN; DOHERTY; POWER, 2004).

Ao mesmo tempo em que contribuem para a transparência e para a democracia, ambientes abertos tendem a oferecer problemas referentes à confiança³ entre usuários. Estes problemas são causados principalmente pela distância entre os participantes, que geralmente não se conhecem. Confiança é fundamental em qualquer relacionamento onde não se podem controlar as atitudes dos envolvidos (JARVENPAA; TRACTINSKY; VITALE, 2000).

O problema da confiança na Internet vem sendo tratado utilizando sistemas de reputação. Sistemas de reputação são mecanismos computacionais que colhem e distribuem informações referentes ao comportamento regresso dos indivíduos (RESNICK et al., 2000). Aproveitam a troca de informações que acontece naturalmente entre pessoas que experimentam algo novo. Em síntese, um sistema de reputação pode ser descrito como uma implementação computacional da disseminação de informações “boca-a-boca” (HU; PAVLOU; ZHANG, 2006). Indivíduos que se relacionaram previamente disponibilizam informações referentes à experiência vivida para que, em outras ocasiões, outros indivíduos possam definir suas decisões.

Aliar as potencialidades dos ambientes wiki com a produção de conteúdo científico, construído de forma coletiva e revisado abertamente, converge para a produtividade, para a integração de pesquisadores e para o surgimento de um processo de revisão mais transparente e interativo. Neste contexto, pode-se contornar o problema da confiança entre indivíduos fisicamente distantes aplicando uma variação de sistemas de reputação. Desta forma, disponibilizam-se ferramentas que auxiliam no processo de geração de confiança, promovendo a integração.

Este trabalho apresenta um método para indicação de pontos de qualificação, definindo pontuações dinâmicas, aplicado em um protótipo construído conforme proposta de Oliveira et al (2005). O protótipo herda as funcionalidades do MediaWiki⁴, o ambiente utilizado pela Wikipedia, adaptando-o para permitir a edição, revisão e indexação aberta de artigos. Este método é baseado em sistemas de reputação para indicação de qualificação de usuários. O método utiliza dados quantitativos e qualitativos, colhidos através da captura de eventos e de avaliações feitas por usuários. Um *ranking* utiliza as qualificações indicadas pelo método para auxiliar na geração de confiança entre os usuários. Espera-se, com a aplicação prática desta proposta, auxiliar no processo de produção científica, permitindo a criação, edição e revisão de artigos com uma abordagem comunitária, interativa e aberta, nos moldes da Web 2.0 (MILLARD; ROSS, 2006).

1.1 Organização do Texto

Este texto está organizado em 8 capítulos, explorando os conteúdos conforme esta breve descrição:

- o capítulo 2 lista e comenta trabalhos relacionados que encontram-se disponíveis;
- o capítulo 3 aborda processos de produção de artigos. Neste capítulo, é descrito o processo aplicado no protótipo implementado por este trabalho. O processo tradicional também é comentado;

³Convicção em se tratar de uma fonte idônea e coerente.

⁴<http://www.mediawiki.org>

- o capítulo 4 descreve os métodos para indicação de qualificação utilizados neste trabalho. É descrito o método EQ1, uma proposta definida para aplicação no protótipo. É descrito também o método EQ2, definido para comparar resultados com EQ1;
- no capítulo 5 é descrito o MediaWiki, software utilizado como base para a implementação do protótipo;
- a descrição do protótipo é detalhada no capítulo 6;
- o capítulo 7 descreve alguns experimentos realizados e os resultados obtidos;
- conclusões e trabalhos futuros encontram-se no capítulo 8;
- os apêndices A e B apresentam informações relevantes produzidas pelo trabalho com relação à estrutura da base de dados e aos experimentos.

2 TRABALHOS RELACIONADOS

Existem trabalhos relacionados com a proposta apresentada por esta pesquisa disponíveis na Web. Até o momento, nenhum trabalho utiliza um processo de produção aberto baseado em ambientes Wiki. Além disso, mesmo que estes trabalhos permitam que os leitores comentem os artigos disponíveis, somente um deles permite avaliações qualitativas feitas pelos leitores.

Algumas iniciativas com processos de revisão semelhantes ao utilizado pelo protótipo implementado neste trabalho já são conhecidas. Estas iniciativas estão disponíveis na Web e se assemelham muito aos sistemas de submissão de trabalhos encontrados nas conferências científicas. Elas integram interfaces para geração de cadastros e submissão dos manuscritos.

Foram encontradas na Web cinco iniciativas relacionadas com este trabalho. Cada uma delas surgiu por necessidades diferentes, reservando particularidades distintas. Nota-se também que a maior parte destas iniciativas surgiu abordando somente física como área de interesse, passando depois para outras áreas como Matemática, Biologia e Computação. A seguir, serão detalhadas as iniciativas encontradas.

2.1 Philica

O Philica¹ é um *journal* que publica artigos científicos de qualquer área. Foi lançado em março de 2006, criado por dois psicólogos ingleses. Ian Walker, professor de psicologia na Universidade de Bath e Nigel Holt, professor de psicologia na Bath Spa University.

Duas motivações básicas levaram a criação do Philica. Primeiramente, Walker e Holt não aceitam a forma com que o sistema de publicações tradicional funciona. Para eles, é um contra-senso que pesquisadores tenham seus salários pagos pelo governo, tenham seus projetos de pesquisa financiados pelo governo e ao fim tenham que entregar os direitos de publicação de seus resultados para editoras. Desta forma, as editoras utilizam-se do trabalho de pesquisadores e do financiamento governamental para enriquecerem. Além disso, Walter e Holt criticam o sistema de publicações tradicional pelo fato deste ser fechado e defendem que as revisões dos trabalhos precisam ser públicas.

O Philica é um meio de publicação disponível na Web. Não requer cadastros para acesso e não cobra taxas para acesso ou publicação de trabalhos. Os trabalhos submetidos são automaticamente publicados. O processo de revisão acontece abertamente. O autor tem acesso ao revisor e pode comentar as revisões. Os trabalhos publicados no Philica seguem a CCAL², sendo permitida sua distribuição e modificação, desde que o autor

¹<http://www.philica.com>

²*Creative Commons Attribution License*

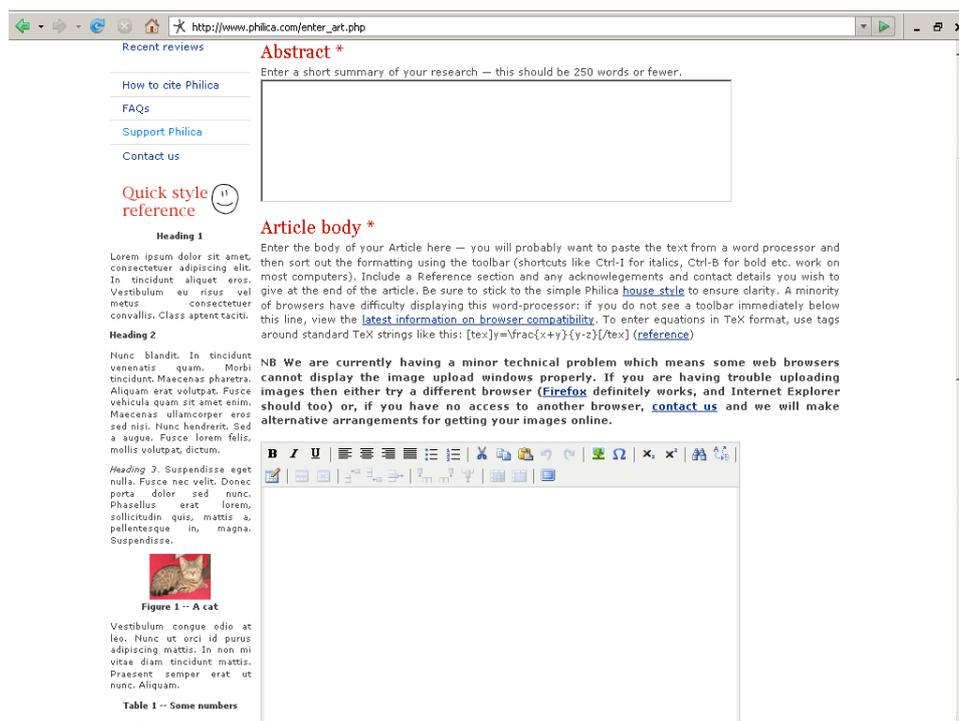


Figura 2.1: A interface para submissão do Philica

original seja devidamente citado.

No Philica são aceitos artigos completos e resumos. A submissão é feita através do próprio site. Na submissão, é preciso preencher um formulário que solicita o nome do trabalho, sua área de conhecimento e o resumo. O conteúdo do trabalho deve ser inserido através de uma interface WYSIWYG³ disponível no mesmo formulário, que pode ser observada na figura 2.1.

O Philica não cria indicativos de qualificação dos usuários. Desta forma, identificar o nível de confiança que pode ser depositado em um trabalho requer leitura e pesquisa para confirmar a coerência do mesmo. Não existe um método para definição de qualificação.

2.2 ACP (*Atmospheric Chemistry and Physics*)

O ACP⁴ é um *journal* mantido pela *Copernicus Publications* e coordenado pela EGU⁵. Seu objetivo é fornecer um meio para publicação e discussão científica a respeito da atmosfera terrestre e seus processos químicos e físicos. Assim como as outras iniciativas encontradas, o ACP também tem o objetivo de contribuir para a disseminação rápida e gratuita de conteúdo científico.

Como um *journal* tradicional, o ACP possui um corpo editorial. Todos os trabalhos submetidos ao ACP passam por processo de *peer review* (revisão em pares), onde os membros do corpo editorial ficam responsáveis pelas revisões. O que torna o ACP diferente dos meios de publicação tradicionais é a indicação dos revisores de um trabalho, que no caso vem do próprio autor. Outro fato que diferencia o ACP é a existência do ACPD⁶.

³What You See Is What You Get

⁴<http://www.atmospheric-chemistry-and-physics.net>

⁵European Geosciences Union

⁶Atmospheric Chemistry and Physics Discussions

Submissões ao JIME devem ser feitas em formato *Word*, utilizando um *template* próprio, disponível no site. O arquivo deve ser enviado por email. Caso exista alguma restrição por tamanho, é possível enviar o arquivo através do FTP ou por correio, em mídia óptica. O JIME não cobra taxas para publicação ou para acesso aos artigos disponíveis.

O JIME não oferece informações sobre qualificação de usuários. Não é possível avaliar os trabalhos publicados. É aplicado um processo de revisão fechada. O JIME não oferece interfaces para edição de conteúdo.

2.4 PLoS ONE

O PLoS ONE⁹ publica trabalhos de pesquisa básica de várias disciplinas. Sua proposta é facilitar a descoberta de conexões entre trabalhos, já que não exclui trabalhos de sua base. O PLoS ONE possui um corpo editorial que acessa cada trabalho submetido. Os trabalhos passam pelo processo de *peer review*.

No processo de *peer review* são considerados aspectos técnicos. Conceitos referentes ao assunto abordado são discutidos com avaliadores independentes. Quando aceitos, os trabalhos tornam-se disponíveis para revisão aberta. Esta revisão aberta envolve discussões e avaliações.

Ao enviar um trabalho para o PLoS ONE o autor aceita os termos da CCAL. Trabalhos assim licenciados continuam sendo de propriedade do autor. Por outro lado, seguindo a CCAL, o autor do trabalho admite que o mesmo seja utilizado para todos os fins, desde que sua autoria seja devidamente citada.

Todos os recursos que envolvem o PLoS ONE utilizam modelos de negócios. O corpo editorial, a produção do *jornal* e os serviços de hospedagem têm seus custos cobertos pelos autores. Para cobrir os custos é cobrada uma taxa para todo aquele usuário que mantém publicações. As submissões são feitas através do *upload* de arquivos no formato PDF.

2.5 arXiv

Desenvolvido por Paul Ginsparg, o arXiv¹⁰ é um repositório de artigos acessível via Web. Os artigos submetidos ao arXiv são chamados de *e-prints*. O arXiv iniciou sua atividade em 1991, com *e-prints* referentes à física. Com o tempo passou a abordar outras áreas do conhecimento.

O arXiv colabora para a distribuição aberta de conteúdo científico. Muitos autores, após terem seus trabalhos aceitos em *journals* e conferências tradicionais, enviam seus trabalhos. Outros autores, ainda antes de enviar seus trabalhos para *journals* tradicionais, submetem seus manuscritos para o arXiv para identificar seu impacto frente à comunidade. Desta forma, o arXiv colabora para uma revolução na divulgação científica, permitindo acesso completo aos *e-prints*, como pode ser observado na figura 2.3.

Os trabalhos enviados para o arXiv não passam por processo de *peer review*. Para garantir a relevância dos conteúdos foi adotado um sistema de endosso. O endosso é dado por outros autores com trabalhos submetidos, que já possuem endosso. Alguns trabalhos podem receber o endosso automaticamente. Os critérios utilizados para o endosso automático não são abertos. Os autores que endossam não são convidados a revisar trabalhos,

⁹<http://www.plosone.org>

¹⁰<http://arxiv.org>

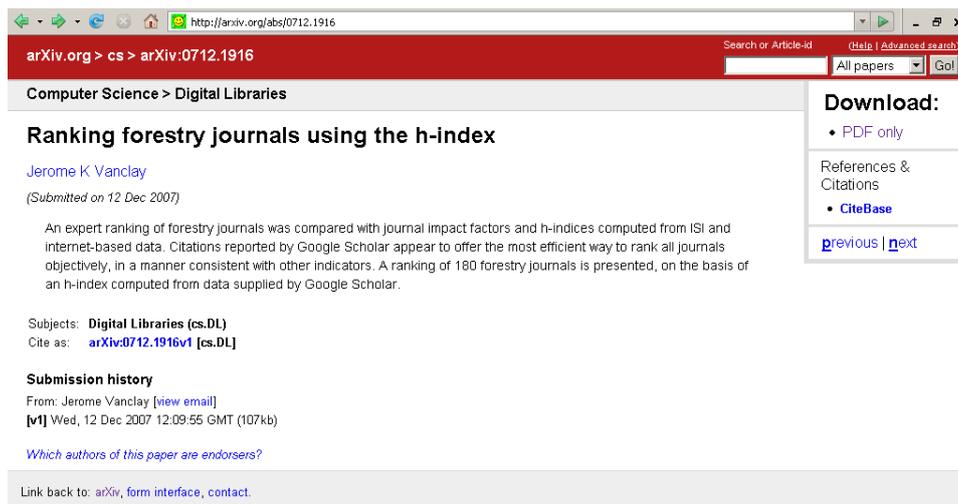


Figura 2.3: *e-print* sobre Bibliotecas Digitais no arXiv

mas sim a avaliar se eles são coerentes com a área de conhecimento para a qual foram submetidos.

A submissão de *e-prints* para o arXiv deve ser feita na forma de arquivos. Não existe uma interface para produção direta, como acontece no Philica. São aceitos arquivos em formato PDF e LaTeX.

3 PROCESSOS DE PRODUÇÃO DE ARTIGOS CIENTÍFICOS

Todo o tipo de submissão científica segue um processo. Estes processos são semelhantes, mas guardam particularidades inerentes a cada evento. Em geral, os manuscritos são enviados através de sistemas eletrônicos que disponibilizam o conteúdo aos editores ou diretamente aos revisores. Quando os editores recebem o material, fica destinado a eles a tarefa de encaminhar os trabalhos aos revisores. Nestes casos, o intermédio entre autor e revisor fica a cargo dos editores. Em outros casos, o encaminhamento é feito direto aos revisores que avaliam e devolvem as avaliações para os autores, sem haver identificação das partes.

O protótipo desenvolvido neste trabalho utiliza um processo diferenciado. As submissões são realizadas através de um ambiente on-line, seguindo os padrões Wiki. Não existe a figura do editor e os revisores são identificados. O processo de revisão é aberto e dinâmico, não existindo um número pré-estabelecido de revisores ou revisões. O critério para aceitação de revisões é estabelecido pelo próprio autor. Todos os usuários podem acessar o conteúdo dos trabalhos, deixando comentários quando julgarem necessário.

3.1 O Processo de Revisão Tradicional (*Peer Review*)

Os primeiros relatos a respeito de *peer review* são ainda do século XIX, especificamente na área médica. Ao visitarem seus pacientes, os médicos anotavam todo o quadro clínico e os procedimentos adotados. No momento em que cada paciente era curado ou nos casos em que não respondesse ao tratamento e chegasse ao óbito, as anotações eram entregues a uma junta médica. A junta avaliava, com base nas anotações, se o procedimento adotado pelo médico foi coerente. Nos casos em que a avaliação era negativa, o médico era punido com base na lei.

Atualmente, o processo de revisão tradicional, conhecido como *peer review* (revisão por pares), consiste na avaliação não identificada de trabalhos. *Peer review* é uma ferramenta destinada a ajudar editores a filtrar livros e artigos com base em qualidade e pertinência (HARRISON, 2004). Este processo oferece referências que ajudam autores a melhorar seus trabalhos.

Autores enviam seus trabalhos para editores, responsáveis pelos meios de publicação. Os editores repassam os trabalhos para os revisores, em geral são três diferentes para cada trabalho. As análises dos revisores podem incluir sugestões de pontos que devem ser ajustados no trabalho. Terminadas as análises, os revisores devolvem para os editores os trabalhos e seus pareceres. Os pareceres geralmente são os seguintes:

- fortemente Aceito;

- aceito, necessitando ajustes;
- rejeitado, mas encoraja o autor a submeter o trabalho novamente após ajustes;
- rejeitado.

Junto com os pareceres, os revisores enviam aos editores os comentários que fizeram sobre os trabalhos. As revisões e os comentários chegam também aos autores. Com estes comentários, os autores podem melhorar seus trabalhos, caso entendam que os comentários são mesmo pertinentes. Para Harrison (2004) o significado real de *peer review* é o debate aberto ao público que acontece quando um trabalho é publicado.

Quando aplicado em plenitude, *peer review* favorece avaliações justas e coerentes. O fato de não haver a identificação das partes, onde os revisores não conhecem os autores nem os autores conhecem os revisores, cria imparcialidade no processo de avaliação. Por outro lado, caso exista qualquer distorção da proposta, *peer review* pode sacrificar bons trabalhos, frustrando autores com potencial. Alguns exemplos aparecem na literatura.

Wyer (2000) comenta que na Suécia, na década de noventa, somente 25% das vagas para pós-doutorado em Biomedicina eram destinadas para mulheres. Este fraco desempenho foi atribuído aos processos de *peer review* que superestimavam os trabalhos escritos por autores do sexo masculino. O SMRC¹ é uma das mais fechadas agências para pesquisas biomédicas da Suécia.

O *journal Regulatory Toxicology and Pharmacology* foi acusado de favorecer a indústria farmacêutica. Segundo Guterman (2002), o *journal* é comparado a uma “vitrine” de negócios industriais disfarçado de periódico que passa por processo de *peer review* (GUTERMAN, 2002). De fato, o processo de *peer review* não é uma garantia de avaliações imparciais.

Frey (2003) indica que uma das maiores críticas ao *peer review* consiste no conservadorismo. Com isto, cria-se uma barreira que dificulta a divulgação de novas idéias. Este conservadorismo favorece a manutenção de antigos paradigmas, optando por manter o antigo no lugar de analisar algo genuinamente novo (CONLEY, ???). Em geral, antes de serem aceitos pela comunidade, novos paradigmas são encarados como provocações acadêmicas, sem expressão para a aplicação prática real. Além disso, o processo de *peer review* também está sujeito a fraudes e avaliações tendenciosas, além de atrasar publicações (BENOS et al., 2007).

Santini (2005) cita uma série de exemplos onde autores consagrados, visionários por suas idéias inovadoras, tiveram trabalhos recusados por revisores despreparados. Dijkstra enfrentou forte resistência ao propor o fim do uso do comando *goto*, em “*Goto Statement Considered Harmful*” (DIJKSTRA, 1968). Revisores argumentaram que a Programação Estruturada era uma ótima experiência acadêmica, mas que nunca seria utilizada na prática. Alegavam que o comando *goto* continuaria sendo usado por muito tempo. Mesmo com seus inconvenientes para depurar códigos, os programadores teriam de continuar a conviver com o *goto*. Codd, ao propor o Modelo Relacional em “*A Relational Model of Data for Large Shared Data Banks*” (CODD, 1983) também enfrentou problemas. Revisores argumentaram que representar os dados em um banco de dados na forma de conjuntos de tuplas era interessante, mas que apresentaria problemas de expressividade e eficiência. Alguns duvidaram que qualquer elemento complexo de interesse prático pudesse ser modelado usando relações.

¹Swedish Medical Research Council

Os exemplos citados fortalecem a questão levantada por Santini (2005). Quanto “estrago” pode ser causado por um revisor que teve um dia ruim? Obviamente estes “estragos” acontecem frequentemente. A maior parte deles nunca serão conhecidos.

3.2 O Processo de Produção Utilizado

O objetivo do processo de produção utilizado no protótipo implementado neste trabalho é trazer transparência e interação para as etapas da redação e revisão de um artigo. Esta abordagem foi proposta em Oliveira et al (2005), produto do projeto DIGITEX². O DIGITEX foi um projeto financiado por órgãos de fomento, onde oito pesquisadores trabalharam com temáticas relacionadas a Bibliotecas Digitais. Nesta abordagem, imediatamente após ser submetido, cada artigo fica disponível para a comunidade. A comunidade é formada por autores, revisores e comentaristas.

O processo de produção aberto abordado neste trabalho contorna alguns problemas encontrados nos métodos tradicionais. Frequentemente são citados casos onde trabalhos que propõem idéias inovadoras ou observações não esperadas enfrentam grande resistência para publicação (BARBER, 1961). Conforme Campanario (1996), este problema não pode ser atribuído somente às pessoas, mas sim a todo o sistema. Segundo o autor, é difícil identificar a diferença entre uma técnica boa e inovadora e outra equivocada e insignificante. Mais ainda, deixar de publicar um trabalho inovador é tão ruim quanto publicar um trabalho medíocre.

No processo proposto por Oliveira et al (2005), os artigos enviados pelos autores passam por revisões. As revisões são feitas por revisores identificados. Nas revisões podem surgir sugestões ou modificações que incidem no conteúdo do artigo. Paralelamente às revisões, os comentaristas discutem o artigo. As discussões geram contribuições no sentido de preencher lacunas deixadas na redação, sem incidir diretamente em seu conteúdo.

A Figura 3.1 apresenta o processo de produção utilizado neste trabalho. O processo é iniciado no momento em que um artigo é criado. Entende-se que esta é a primeira versão do artigo, identificada no diagrama por V1. Esta versão fica disponível no ambiente, aberto para leituras, comentários e revisões. Os comentaristas podem ler o artigo e comentá-lo, mas não podem editar o conteúdo diretamente. Os revisores podem ler e, caso julguem necessário, podem modificar o conteúdo do artigo diretamente. As modificações subsequentes geram novas versões do artigo identificadas por V2. Caso o autor julgue alguma modificação incoerente, ele poderá modificar o artigo, retornando-o para o estado anterior.

A linha pontilhada central mostra as revisões. Estas revisões aproveitam as contribuições vindas das discussões, representadas pelos dois balões, paralelos à linha central. O processo de discussão e revisão é iniciado no momento em que o artigo é submetido pelo autor. Não existe um número específico de revisões a serem feitas. O processo de revisão (com as contribuições e discussões) é contínuo, gerando versões atualizadas dos trabalhos.

Existem características que diferenciam este processo aberto do processo tradicional. As características mais marcantes que a abordagem aberta utilizada por este trabalho possui são:

- introduz o conceito Wiki na criação de artigos científicos;
- elimina a figura do editor;

²CNPq - Edital 11/2005, proc. 550.845/2005-4

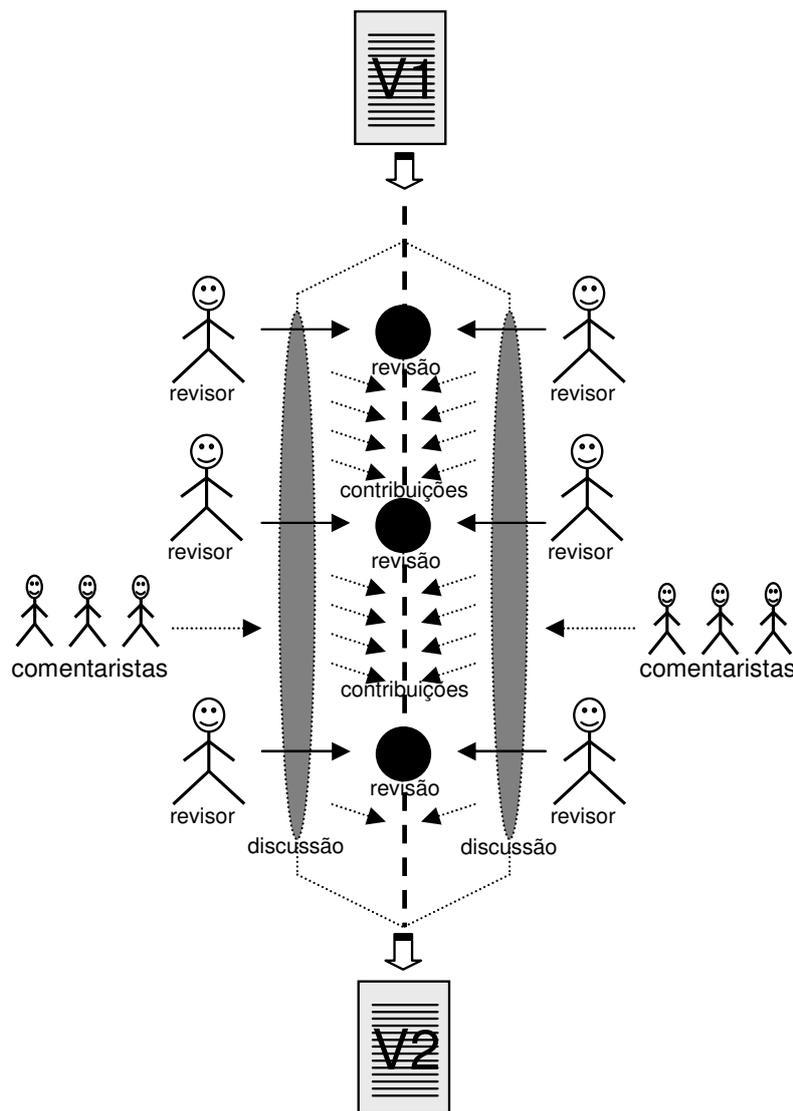


Figura 3.1: Processo aberto produção e revisão de artigos científicos

- cria a figura do comentarista;
- permite que qualquer pessoa acesse o conteúdo de determinado artigo, mesmo que este ainda não esteja concluído;
- permite revisões identificadas, realizadas diretamente pelos revisores;
- passa a responsabilidade da aceitação das revisões para o autor do artigo;
- considera a opinião da comunidade, utilizando um método dinâmico de indicação de pontos de qualificação de usuários;
- oferece um *ranking* de qualificação dos usuários.

As iniciativas relacionadas a este trabalho aplicam algumas das características citadas acima. Dentre as iniciativas, as que mais se assemelham à este trabalho são: Philica, ACP e JIME. Todas as três iniciativas são *journals* abertos, disponíveis na Web. No Philica, os trabalhos podem ser comentados e revisados por qualquer usuário. No ACP, o autor

pode identificar os revisores. O JIME utiliza um processo aberto de revisão que acontece após um primeiro processo fechado. Por outro lado, nenhuma destas iniciativas utiliza o conceito Wiki ou passa a responsabilidade de aceitar as modificações indicadas pelos revisores para o autor do trabalho. Também não é possível avaliar os artigos e não é gerado nenhum *ranking* exibindo a qualificação dos usuários.

É importante salientar que cabe ao autor tomar a decisão de gerar uma nova versão do artigo original. Novas versões podem aceitar ou rejeitar as modificações e comentários realizados pelos demais membros da comunidade. Acredita-se que assim a qualidade dos trabalhos continuará evoluindo, ao mesmo tempo em que preserva sua identidade.

3.3 Classes de Usuários

Em eventos e periódicos científicos, o processo de avaliação envolve três tipos básicos de participantes, sendo estes autores, editores e revisores. Os autores escrevem trabalhos com base em suas pesquisas, os quais são submetidos para editores com a finalidade de tornar públicas suas descobertas. Os editores repassam os trabalhos para os revisores suprimindo dados de identificação, a fim de tornar o processo impessoal. Os revisores avaliam os trabalhos enviados com o objetivo de selecionar os mais adequados para publicação. O processo proposto por este trabalho mantém duas classes básicas, já encontradas nos processos tradicionais de avaliação, sendo elas autores e revisores. A figura do editor é suprimida. Surge a figura do comentarista.

A necessidade de definições de classes surgiu para estabelecer que somente autores e revisores possuem permissão para editar trabalhos. Não existe restrição por classe para fins de acessos ou avaliações. Os usuários podem estar vinculados a uma ou mais classes.

3.3.1 Autores

Para construir um documento no protótipo o usuário precisa estar vinculado à classe autor. Na versão atual do protótipo, automaticamente após a realização do cadastro, todo o usuário é vinculado à classe autor. Autores podem criar documentos sem qualquer restrição de tamanho ou conteúdo. Não existe também um número limite de documentos, ou seja, cada autor pode criar o número de documentos que julgar necessário.

Quanto às edições, todo autor pode editar inúmeras vezes qualquer documento de sua autoria. Cada diferente edição gera uma nova versão para o documento. As versões podem ser comparadas e, caso necessário, modificações podem ser desfeitas. O protótipo herda de um sistema base funcionalidades que permitem indicar o ponto exato em que cada versão difere, através de algoritmos de *diff*. Os algoritmos de *diff*, terminologia que abrevia *difference*, do inglês, realizam a comparação entre dois documentos, retornando suas diferenças (LABIO; GARCIA-MOLINA, 1996). Um autor não pode editar um documento que não seja de sua autoria.

3.3.2 Revisores

Os usuários vinculados à classe `Revisor` são os únicos que possuem permissão para modificar diretamente um artigo. Usuários vinculados a outras classes têm permissão para acessar e comentar qualquer artigo, não para modificá-los. Esta particularidade torna o processo implementado diferente do utilizado pelos Wikis tradicionais.

O papel dos usuários vinculados à classe `Revisor` é semelhante ao papel dos revisores que avaliam os trabalhos enviados para os meios de publicação tradicionais. No processo apresentado, a diferença está no fato de que os revisores são identificados. O

autor do artigo pode identificar qual revisor modificou ou sugeriu modificações em seu texto, assim como o revisor pode identificar o autor do mesmo. Nesta abordagem, o revisor não tem a função de selecionar artigos para publicação, mas sim de contribuir para o aumento da qualidade. As modificações feitas pelos revisores podem ser aceitas, sendo incorporadas ao artigo, ou desfeitas pelo autor.

Até o momento não existe um critério específico para a escolha dos revisores. Na realização dos experimentos os revisores foram selecionados aleatoriamente. Em trabalhos futuros, o critério de escolha dos revisores será dinâmico. Serão considerados revisores somente aqueles usuários que estiverem no topo do *ranking* de qualificação. Sendo assim, somente usuários com as melhores avaliações terão permissão para modificar artigos que não são de sua própria autoria.

3.3.2.1 Revisões

Nesta abordagem, entende-se por revisão todo o tipo de contribuição que um usuário vinculado à classe `Revisor` deixar em um artigo. Esta ação é simples e não requer qualquer tipo de formalismo. Quando acessa um artigo, o usuário vinculado à classe `Revisor` visualiza um *link* chamado Edição. Através deste *link* ele pode efetuar qualquer tipo de modificação, conforme os padrões criados pelos ambientes Wiki.

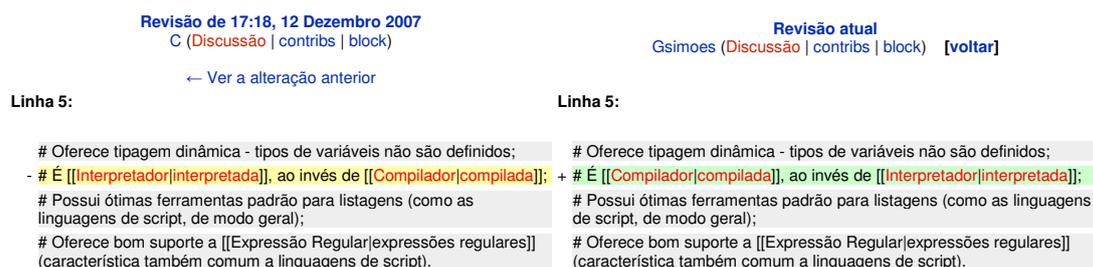


Figura 3.2: Diferentes versões após revisão realizada pelo usuário Gsimoes

O autor identifica as revisões quando compara as modificações que seus artigos sofreram, através das diferentes versões geradas. Para comparar as versões, o ambiente utiliza um algoritmo de *diff* que lista todas as diferenças encontradas no texto. O que é suprimido do texto é identificado através de um sombreamento vermelho, o que é inserido através de um sombreamento verde. A comparação pode ser vista na figura 3.2, onde um revisor identificado como Gsimoes inverte uma informação. Na figura, o texto exibido na caixa esquerda mostra um sinal negativo “-” exibido na segunda linha. Este sinal indica como a linha textual era constituída antes da edição. Na caixa da direita, a segunda linha apresenta um sinal positivo “+”, indicando a nova constituição da linha. Para desfazer uma modificação, o autor deve realizar o *diff* entre as versões e identificar suas diferenças. Após identificar as diferenças, basta editar o código fonte do texto, sobrescrevendo o conteúdo.

Uma nova versão do artigo é criada após cada contribuição, como pode ser observado na figura 3.3. Caso julgue a revisão pertinente, o autor simplesmente mantém o texto inalterado, mantendo as idéias deixadas pelo revisor. Caso o autor julgue as sugestões impróprias, o texto pode ser modificado de maneira que retorne à forma anterior à revisão. Assim, o critério de aceitação de revisões é definido pelo próprio autor.

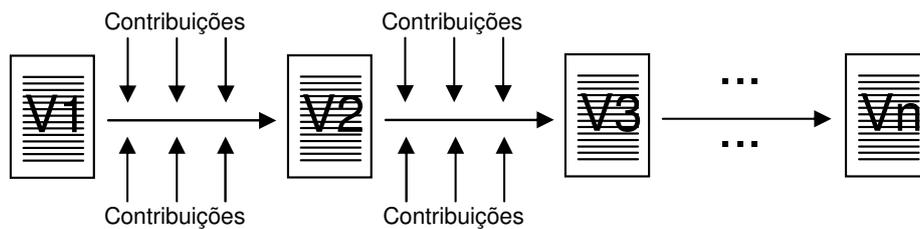


Figura 3.3: Ciclo de versões e contribuições

3.3.3 Comentaristas

Comentarista constitui a classe mais básica do ambiente. Conforme a idéia inicial do ambiente, os comentaristas desempenham um papel fundamental no processo. Todos os usuários do ambiente são automaticamente vinculados à classe *Comentarista*. Esta iniciativa deve-se a questões funcionais.

Comentaristas não podem criar ou editar artigos. Por outro lado, comentaristas podem acessar e comentar qualquer texto disponível no ambiente. Os comentários não incidem diretamente sobre o conteúdo do artigo, mas podem sugerir modificações futuras por parte do autor ou indicar pontos que devem ser melhor observados pelos revisores. Os comentaristas podem também avaliar qualquer artigo.

3.3.3.1 Comentários

Somente usuários vinculados às classes *Autor* e *Revisor* possuem o privilégio de criar ou modificar artigos. Usuários não vinculados a estas classes estão automaticamente vinculados à classe *Comentarista*. Os comentários constituem uma alternativa para que os vinculados à classe *Comentarista* possam comunicar suas opiniões.

Em um espaço de comentários pode-se discutir o conteúdo dos artigos. Caso os comentários sejam enviados por indivíduos diferentes, abre-se o espaço para discussões. Discussões iniciam um debate prévio a respeito do assunto, podendo indicar caminhos a seguir e outras formas de abordar a temática. Desta forma, quando utilizados, os comentários tornam-se uma ferramenta para a criação de possíveis novos artigos.

4 MÉTODOS PARA QUALIFICAÇÃO

Este capítulo discute **métodos** para **qualificação** de usuários. São abordados temas referentes à **avaliação quantitativa** e **qualitativa**. Neste trabalho, os termos destacados remetem diretamente para as definições de Ferreira (1986), sendo estas, respectivamente:

- método: Caminho pelo qual se atinge um objetivo;
- qualificação, habilitação: Atributos que habilitam alguém ao desempenho de uma função; qualificação;
- avaliação: Valor determinado pelos avaliadores;
- quantitativo: Relativo a, ou indicativo de quantidade;
- qualitativo: Que exprime ou determina a(s) qualidade;
- qualificar: Emitir opinião a respeito de; avaliar, apreciar.

Em ambientes abertos, as informações obrigatórias para o cadastro de usuários são superficiais. Estas informações não passam por qualquer processo de validação, característica que “democratiza” a utilização do ambiente. Por outro lado, a escassez de informações e a ausência de verificação não fornecem argumentos para auxiliar na produção de confiança entre usuários.

Iniciativas como o *h-index* (HIRSCH, 2005) procuram gerar confiança entre pesquisadores. O *h-index* constrói um *ranking* onde os pesquisadores com maior número de artigos publicados e maior número de citações para estes artigos recebem destaque. De qualquer forma, por considerar somente os artigos publicados, o *h-index* torna-se dependente do processo de avaliação tradicional (*peer review*) empregado nos periódicos e conferências.

Rech (2007) apresenta um modelo de pontuação para competência acadêmica de pesquisadores. Este modelo permite o cálculo do CC¹. O CC utiliza indicadores quantitativos, relacionados ao currículo do pesquisador, e indicativos relacionados à produção bibliográfica do pesquisador. Também são considerados os dados referentes ao impacto gerado pela produção bibliográfica do pesquisador. Os dados bibliográficos necessários para calcular o CC de um pesquisador são extraídos do currículo Lattes (CV-LATTES, 2007). Os dados relacionados à produção bibliográfica e ao impacto gerado por ela são extraídos do Google Scholar, o qual indica o número de citações de cada artigo retornado, e do sistema Qualis CAPES (CAPES, 2007) que indica o impacto da conferência ou do periódico onde o artigo foi publicado. O cálculo do CC retorna um indicativo de pontuação

¹Coeficiente de Competência

para o pesquisador. Por outro lado, ainda conforme Rech (2007), medir a competência de usuário consiste em uma tarefa complexa, visto a variedade de aspectos subjetivos envolvidos no processo. Para ele, adotar critérios mais completos, considerando as avaliações dos pares, tende a gerar indicativos mais precisos.

O problema da confiança está presente também em outros contextos, principalmente naqueles relacionados a valores monetários. Sites de comércio eletrônico, como eBay² e MercadoLivre³, facilitam negociações diretas entre usuários que normalmente estão a grandes distâncias. Devido à distância, a maior parte das negociações envolve usuários que não se conhecem. Estes sites procuram resolver o problema da confiança utilizando sistemas de reputação.

Sistemas de reputação são a pior forma de construir confiança na Web, com exceção de todas as outras alternativas já tentadas (RESNICK et al., 2000). Esta afirmativa traduz os problemas gerados pelo uso destes sistemas. Por serem dependentes de informações provenientes de usuários, sempre existirão riscos de definições inapropriadas. Por outro lado, até o momento não existe alternativa melhor para tratar o problema que a aplicação de sistemas de reputação.

Ao fim de cada transação, tanto vendedor quanto comprador avaliam-se mutuamente. Os resultados destas avaliações são exibidos para os usuários que futuramente iniciarem alguma nova negociação. Com base nas avaliações das transações anteriores (as quais podem ser positivas, neutras ou negativas), pode-se estabelecer algum nível de confiança que auxilie na decisão de continuar ou não o negócio.

Parte do objetivo deste trabalho consiste em disponibilizar um *ranking* de qualificação dos usuários. O *ranking* é uma ferramenta de apoio à confiança, já que gera uma lista de usuários ordenada por suas qualificações. Para criar este *ranking* foi definido um método que calcula pontos para a qualificação dos usuários. Este método utiliza dados quantitativos e qualitativos, traduzindo a qualificação dos usuários em valores numéricos. O método utiliza valores qualitativos objetivando complementar as lacunas encontradas em outros métodos que focam em valores quantitativos.

Um problema relacionado à qualificação automática, com base em avaliações qualitativas, consiste na definição das pontuações. Pontuações definidas estaticamente não valorizam avaliadores com maior qualificação. Definir pontuações dinâmicas, que variam de acordo com a qualificação do avaliador, tende a valorizar a opinião de usuários melhor qualificados. É muito simplista assumir que as avaliações feitas por todos os usuários tenham a mesma relevância. Uma avaliação definida como “boa”, feita por um usuário altamente qualificado, pode ser mais relevante do que uma avaliação “ótima” feita por um usuário com baixa qualificação.

Este trabalho propõe EQ1, um método para indicação de pontos de qualificação. Este método foi criado para definir a qualificação dos usuários considerando a qualificação dos avaliadores, no momento em que eles avaliam um artigo. Para contrapor EQ1 foi definido também o método EQ2. EQ2 segue os mesmos moldes de EQ1, no entanto utiliza pontuações estáticas, não considerando a qualificação dos avaliadores.

A utilização do método EQ1 permite considerar avaliações qualitativas de artigos. Nestas avaliações, o usuário que acessa o artigo pode deixar registrado o impacto que este causou. Quando realiza uma avaliação qualitativa, o usuário pode indicar uma das seguintes impressões sobre a relevância que o conteúdo trouxe para si:

- ótima;

²<http://www.ebay.com>

³<http://www.mercadolivre.com.br>

- boa;
- neutra;
- ruim;
- péssima.

As avaliações são mapeadas para valores numéricos, aplicados em EQ1 como fatores de multiplicação. Em EQ2, um método simplificado que não considera qualificação de avaliadores para definição de pontuação, os valores numéricos são aplicados diretamente como pontuações estáticas. A tabela 4.1 indica as avaliações disponíveis e seus fatores de pontuação para cada um dos dois métodos. Os fatores apresentam equilíbrio, com exceção do valor central (1) que corresponde a uma avaliação neutra. Este critério foi adotado pois acredita-se que mesmo definida como neutra, o fato do avaliador deixar sua opinião indica que o conteúdo gerou algum tipo de impacto. Para os casos em que nenhum impacto seja gerado, acredita-se que o avaliador simplesmente deixará de opinar. Neste contexto, surge o zero (0), ou seja, sem representar qualquer modificação na qualificação do autor do artigo avaliado, mantendo a qualificação anterior.

Tabela 4.1: Valores atribuídos para avaliações utilizando EQ1 e EQ2

Avaliação Informada	Valor em EQ1	Valor em EQ2
Ótima	3 como fator multiplicador	3 pontos estáticos
Boa	2 como fator multiplicador	2 pontos estáticos
Neutra	1 como fator multiplicador	1 pontos estáticos
Ruim	-2 como fator multiplicador	-2 pontos estáticos
Péssima	-3 como fator multiplicador	-3 pontos estáticos

4.1 O Método EQ1

Observando às iniciativas relacionadas para definição de qualificação de autores, nota-se que tanto *h-index* quanto CC utilizam somente dados obtidos através da aplicação de processos de revisão fechados. Nota-se também que este problema ainda é campo para novos estudos. Buscando minimizar o problema da definição automática de qualificação de autores, este trabalho define o método EQ1. O método EQ1 utiliza dados quantitativos, referentes a interações dos usuários com o protótipo, e dados qualitativos, referentes a avaliações onde os usuários indicam o nível de impacto que um artigo gerou.

O objetivo principal do método EQ1 é indicar pontos de qualificação para os usuários de um ambiente de edição e revisão aberta de artigos. Os pontos de qualificação indicados pelo método são utilizados para gerar um *ranking* de qualificação de usuários. Por tratar-se de um ambiente aberto, onde não se conhece às referências concretas com relação às competências de cada participante, o *ranking* constitui o principal recurso para geração de confiança entre os usuários. Ao utilizar dados qualitativos, referentes às avaliações dos artigos, o método EQ1 incorpora características de sistemas de reputação em uma ferramenta para definição de pontos de qualificação de autores.

O método EQ1 atribui pontos ao autor considerando a qualificação do avaliador no momento em que a avaliação é realizada. O objetivo deste método consiste em permitir

a criação de um *ranking* de qualificação que considere a qualificação dos avaliadores, de forma que avaliações feitas por usuários melhor qualificados recebam destaque. Para isso, EQ1 soma na pontuação do autor o produto da qualificação normalizada do avaliador pelo valor correspondente à avaliação, conforme a equação 4.1.

$$P_A = P'_A + (F \cdot N(P_B)) \quad (4.1)$$

Dados dois usuários, A e B, sendo A o autor do artigo avaliado e B o avaliador, P_A corresponde à pontuação final do usuário A. P'_A corresponde a pontuação do usuário A no momento em que um de seus artigos é avaliado. O fator de multiplicação, definido como -3, -2, 1, 2 ou 3, é indicado por F . N é a função que normaliza a pontuação dos usuários no intervalo entre 0 e 1. P_B corresponde a pontuação do usuário avaliador e $N(P_B)$ retorna a qualificação normalizada deste.

A figura 4.1 mostra um exemplo onde um avaliador avalia um artigo de um autor. O avaliador acessa o artigo e informa seu grau de satisfação com o conteúdo avaliado. O grau de satisfação é traduzido para um valor numérico, utilizado como argumento de entrada para a função de atribuição de pontos. A qualificação do avaliador também é utilizada como argumento de entrada. A função retorna o número de pontos resultantes da avaliação. O produto da multiplicação entre o número correspondente a avaliação informada e a qualificação normalizada do avaliador corresponde aos pontos gerados pela avaliação. A qualificação é normalizada em valores no intervalo 0 e 1, fazendo com que as pontuações geradas variem sempre entre -3 e 3. Os pontos gerados por cada avaliação são somados aos pontos de qualificação do autor do artigo avaliado.

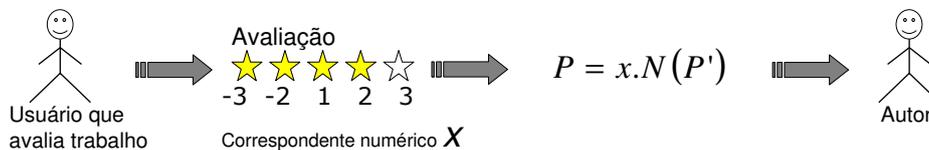


Figura 4.1: Atribuindo pontuação com o método EQ1

Com as qualificações normalizadas, a avaliação mais positiva resultará em 3 pontos, enquanto que a mais negativa resultará em -3. Toma-se por base o mapeamento entre as impressões de relevância informadas pelos usuários com relação ao conteúdo dos artigos e os valores numéricos -3, -2, 1, 2 e 3.

4.1.1 Normalizando Qualificações em EQ1

A qualificação de cada usuário é obtida através da soma dos pontos que eles acumulam durante a utilização do ambiente. Não existem limites máximos ou mínimos para número de pontos, ou seja, a pontuação pode crescer livremente. Por outro lado, já que EQ1 utiliza um fator que multiplica a qualificação dos usuários para calcular o número de pontos resultantes de uma avaliação, deve-se observar a possibilidade para o crescimento exponencial das pontuações.

Internamente, para fins de definição de qualificações e cálculo de pontuações, EQ1 utiliza uma função de normalização. A função $NormalizaQualificacao : \{\mathbb{R}\} \rightarrow \mathbb{R}_I$, sendo \mathbb{R} o conjunto dos números reais e $\mathbb{R}_I = \{x \in \mathbb{R} \mid 0 \leq x \leq 1\}$ recebe um vetor contendo o somatório dos pontos de cada usuário. Após sua execução, esta função retorna o mesmo vetor normalizado com valores entre 0 e 1. O valor 1 corresponde ao usuário melhor qualificado. O valor 0 indica a pior qualificação. Com a normalização das

qualificações é resolvido o problema do crescimento exponencial das pontuações. Também é resolvido o problema da ocorrência de erros com relação a pontuações gerados por avaliadores com qualificações negativas.

O Algoritmo 1 define a função de normalização. Na linha 1, a função $\text{MIN}(Q)$ retorna o menor valor presente no vetor Q . Na linha 2, a função $\text{MAX}(Q)$ retorna o maior valor presente no vetor Q . Se $\text{MIN}(Q)$ for igual a $\text{MAX}(Q)$, então, na linha 4, Q é retornado sem qualquer modificação. Caso contrário, a partir da linha 5, normaliza-se o vetor.

Algoritmo 1 Normaliza Qualificação

```

NORMALIZAQUALIFICACAO(Q)
1  min ← MIN(Q);
2  max ← MAX(Q);
3  if min = max
4  then return Q;
5  if min < 0
6  then div ← max + (min * -1) + 1;
7      for i ← 0 to Q.length
8          do if Q[i] > 0
9              then Q[i] ← (Q[i] - min + 1)/div;
10             else Q[i] ← (Q[i] - min)/div;
11
12     else for i ← 0 to Q.length
13         do Q[i] ← Q[i]/max;
14 return Q

```

4.1.1.1 Exemplo de Normalização de Qualificações

A Tabela 4.2 apresenta uma relação de qualificações de 10 usuários. Cada qualificação exibida é o somatório dos pontos de qualificação que cada um dos 10 usuários recebeu. São exibidos o somatório total dos pontos e estes mesmos pontos normalizados.

Tabela 4.2: Exemplo de normalização de qualificações

\sum pontos	Pontos Normalizados
43,82	1
40,30	0,9518
32,25	0,8416
26,08	0,7571
8,39	0,5148
-7,44	0,2843
-12,85	0,2102
-15,10	0,1794
-20,43	0,1064
-28,20	0

4.1.2 Definição do Método EQ1

A função $GeraPontuacaoEQ1 : \{\mathbb{R}_{\mathbb{E}}, \mathbb{N}_{\mathbb{I}}, \mathbb{I}_{\mathbb{I}}\} \rightarrow \mathbb{R}_{\mathbb{S}}$, sendo \mathbb{R} o conjunto dos números reais, $\mathbb{R}_{\mathbb{E}} = \{x \in \mathbb{R} \mid 0 \leq x \leq 1\}$, $\mathbb{R}_{\mathbb{S}} = \{x \in \mathbb{R} \mid -3 \leq x \leq 3\}$, \mathbb{N} o conjunto dos números naturais, $\mathbb{N}_{\mathbb{I}} = \{x \in \mathbb{N} \mid x = 0 \vee x = 1\}$, \mathbb{I} o conjunto dos números inteiros e $\mathbb{I}_{\mathbb{I}} = \{x \in \mathbb{I} \mid -3 \leq x \leq 3\}$, implementa a Equação 4.1. A aplicação desta função permite calcular a qualificação dos usuários considerando a qualificação dos avaliadores.

A função $GeraPontuacaoEQ1$ foi implementada na forma de algoritmo. O algoritmo segue as definições da equação 4.1 e está representado no Algoritmo 2. No algoritmo, está definida uma função que recebe 3 parâmetros de entrada: Usr , referente ao usuário que realiza uma avaliação, $Tipo$, referente ao tipo de pontuação que deve ser definido e $AvaliacaoInformada$ referente a avaliação informada pelo avaliador.

Na linha 1, a função $RECUPERAQUALIFICACOES()$ retorna um vetor com o somatório de pontuações de cada usuário. A função $NORMALIZAQUALIFICACAO()$, que aparece na linha 2, foi definida no Algoritmo 1 e retorna o vetor normalizado. A função $SELECIONAUSUARIO()$, na linha 4, extrai do vetor somente a qualificação normalizada do usuário que executa a avaliação. Na linha 6, o vetor $avaliacaoNumerica$ recebe os fatores de multiplicação. Os fatores de multiplicação são mapeados para a avaliação informada pelo avaliador. O atributo de entrada $Tipo$ indica o tipo de pontuação que precisa ser gerada. Caso $Tipo$ seja igual a A , é gerada pontuação para um acesso do usuário a um artigo. Caso $Tipo$ seja igual a Q , é gerada pontuação para uma avaliação de artigo. O valor retornado na linha 13 é a pontuação referente ao acesso/avaliação.

Algoritmo 2 Gera Pontuação EQ1

```

GERAPONTUACAOEQ1( $Usr, Tipo, AvaliacaoInformada$ )
1   $qualificacoes[] \leftarrow RECUPERAQUALIFICACOES();$ 
2   $qualificacoesNormalizadas[] = NORMALIZAQUALIFICACAO($ 
3       $qualificacoes);$ 
4   $qualificacaoUsuario \leftarrow SELECIONAUSUARIO($ 
5       $qualificacoesNormalizadas, Usr);$ 
6   $avaliacaoNumerica[] \leftarrow \{-3, -2, 1, 2, 3\};$ 
7  switch
8      case  $Tipo = A :$ 
9           $pontuacao \leftarrow qualificacaoUsuario * 0.5;$ 
10     case  $Tipo = Q :$ 
11          $pontuacao \leftarrow avaliacaoNumerica[AvaliacaoInformada]*$ 
12              $qualificacaoUsuario;$ 
13  return  $pontuacao$ 

```

4.1.2.1 Exemplo de indicação de pontuações utilizando EQ1

A Tabela 4.3 apresenta uma relação de qualificações de 10 usuários. Cada qualificação exibida é o somatório dos pontos de qualificação que cada um dos 10 usuários recebeu. São exibidos o somatório total dos pontos e estes mesmos pontos normalizados.

4.2 O Método EQ2

Diferente de EQ1, EQ2 não considera a qualificação dos avaliadores para gerar a pontuação dos usuários. EQ2 foi desenvolvido unicamente para contrapor EQ1, já que utiliza

Tabela 4.3: Exemplo de indicação de pontuações utilizando EQ1

$\sum pontos'$	Pts. Normalizados	Avaliação	Pts. Indicada	$\sum pontos''$
43,82	1	3	3	46,82
40,30	0,9518	3	2,8554	43,16
32,25	0,8416	3	2,5248	34,77
26,08	0,7571	3	2,2713	28,35
8,39	0,5148	3	1,5444	9,93
-7,44	0,2843	3	0,8529	-6,59
-12,85	0,2102	3	0,6306	-12,22
-15,10	0,1794	3	0,5382	-14,56
-20,43	0,1064	3	0,3192	-20,11
-28,20	0	3	0	-28,20

uma abordagem de definição estática de pontos. EQ2 soma os fatores de multiplicação informados nas avaliações aos pontos de qualificação do usuário que tem artigos avaliados. O uso de EQ2 se justifica no momento em que se objetiva comparar diferenças na composição do ranking de qualificação, considerando ou não a qualificação dos usuários.

$$P_A = P'_A + F \quad (4.2)$$

Na equação 4.2, P_A representa a pontuação final de um usuário A. P'_A representa a pontuação do usuário A no momento em que acontece a avaliação. F indica o número que corresponde à avaliação recebida pelo documento, utilizado como fator de multiplicação em EQ1. A qualificação do avaliador não modifica qualquer resultado neste método.

O somatório dos pontos de qualificação dos usuários utilizando EQ2 tende a ser sempre maior do que utilizando EQ1. EQ1 normaliza as pontuações dos usuários em valores entre 0 e 1. O número de pontos gerados por EQ1 consiste no produto entre esta pontuação normalizada e o fator de multiplicação. Sendo assim, em EQ1, somente nos casos onde o usuário melhor qualificado for o avaliador é que o número de pontos gerados corresponderá ao maior fator de multiplicação existente. EQ2, por somar os fatores de todas as avaliações sem distinguir usuários, apresentará um crescimento de pontuação em números maior que EQ1.

4.2.1 Definição do Método EQ2

A função $GeraPontuacaoEQ2 : \{\mathbb{N}_{\mathbb{I}}, \mathbb{I}_{\mathbb{I}}\} \rightarrow \mathbb{I}_{\mathbb{I}}$, sendo \mathbb{N} o conjunto dos números naturais, $\mathbb{N}_{\mathbb{I}} = \{x \in \mathbb{N} \mid x = 0 \vee x = 1\}$, \mathbb{I} o conjunto dos números inteiros e $\mathbb{I}_{\mathbb{I}} = \{x \in \mathbb{I} \mid -3 \leq x \leq 3\}$ implementa a Equação 4.2. A aplicação desta função indica pontos estáticos de qualificação, definidos com base nos valores numéricos -3, -2, 1, 2 e 3, não considerando a qualificação dos avaliadores.

A função $GeraPontuacaoEQ2$ foi implementada na forma de algoritmo. O algoritmo segue as definições da Equação 4.2 e está representado no Algoritmo 3. Está definida no algoritmo uma função que recebe 2 parâmetros de entrada. *Tipo* define o tipo de pontuação que deve ser definida, podendo variar entre quantitativa (acesso) e qualitativa (avaliação). *AvaliacaoInformada* corresponde a avaliação informada pelo avaliador.

Na linha 1, o vetor *avaliacaoNumerica[]* recebe os valores numéricos das avaliações possíveis. Na linha 4 é definida a pontuação estática para acessos. Na linha 6 é definida a

pontuação estática para a avaliação, setada pela escolha do avaliador. A linha 7 conclui a execução da função, retornando a pontuação indicada.

Algoritmo 3 *Gera Pontuação EQ2*

```

GERAPONTUACAOEQ2(Tipo, AvaliacaoInformada)
1  avaliacaoNumerica[] ← {−3, −2, 1, 2, 3};
2  switch
3    case Tipo = A :
4      pontuacao ← 1;
5    case Tipo = Q :
6      pontuacao ← avaliacaoNumerica[AvaliacaoInformada]
7  return pontuacao

```

4.2.2 Exemplo de indicação de pontuações utilizando EQ2

A Tabela 4.4 apresenta uma relação de qualificações de 10 usuários. Cada qualificação exibida é o somatório dos pontos de qualificação que cada um dos 10 usuários recebeu. São exibidos o somatório total dos pontos e estes mesmos pontos normalizados.

Tabela 4.4: Exemplo de indicação de pontuações utilizando EQ2

$\sum pontos'$	Avaliação	$\sum pontos''$
43,82	3	46,82
40,30	3	43,30
32,25	3	35,25
26,08	3	29,08
8,39	3	11,39
-7,44	3	-4,44
-12,85	3	-9,85
-15,10	3	-12,10
-20,43	3	-17,43
-28,20	3	-25,20

4.3 Ranking de Qualificação

O ranking de qualificação é o principal recurso disponível para a criação de confiança entre os usuários do ambiente descrito neste trabalho. Nele, os usuários com maior número de pontos de qualificação são posicionados no topo. O ranking consiste de uma lista de tuplas ordenadas, composta por um identificador do usuário, pelo valor correspondente ao somatório dos pontos do usuário e pela pontuação normalizada do mesmo. A tabela 4.5 mostra o exemplo de um *ranking* gerado para um grupo de 10 usuários. Este *ranking* foi gerado utilizando EQ1.

O *ranking* desconsidera o histórico regresso do usuário, bem como suas ações externas a ferramenta. Para alcançar melhores posições, o usuário precisa acessar o ambiente e disponibilizar artigos. Estes artigos precisam ser acessados e avaliados por outros usuários,

Tabela 4.5: Exemplo de *Ranking* de qualificação utilizando EQ1

Usuário	$\sum \text{pontos}$	Pontos Normalizados
A	24.56	1
B	-13.62	0
C	16.99	0.8068
D	10.01	0.6286
E	0.38	0.3828
F	-7.90	0.1460
G	-10.29	0.0850
H	2.28	0.4313
I	2.68	0.4416
J	-9.95	0.0937

o que gera pontos de qualificação para o autor. Os pontos são incorporados à pontuação do autor, possibilitando que ele “suba” no *ranking*.

5 O AMBIENTE MEDIAWIKI

O ambiente descrito em Oliveira (2005), é aberto e colaborativo. Permite que usuários identificados produzam conteúdo científico na forma de artigos, passando por revisões abertas. A base deste ambiente é um Wiki, que segundo Kevin Curran (2004) é um servidor Web colaborativo onde vários usuários editam os mesmos documentos.

O MediaWiki, ambiente Wiki mais difundido na Internet, foi utilizado no processo. O MediaWiki foi utilizado como *framework* básico, evitando a necessidade de desenvolver todo o sistema. Quando se reutiliza um componente de software, um *framework* ou um sistema inteiro como base para o desenvolvimento de uma aplicação, além de tempo, ganha-se a segurança de estar empregando tecnologias consolidadas.

Estão disponíveis na Internet uma série de softwares para criação e gerenciamento de ambientes Wiki. Ambientes de aprendizado, como o *Moodle*¹, oferecem suporte a Wiki. O gerenciador de conteúdo *Mambo*² pode ser configurado para manter áreas Wiki. O *Mac OS X Wiki Server*³ também oferece um Wiki, neste caso, com código fonte fechado. Estes softwares não possuem mecanismos para extensão bem definidos, o que dificulta sua aplicação em atividades específicas. O MediaWiki, software utilizado pela Wikipedia, oferece mecanismos para extensão documentados.

Dentre os softwares disponíveis gratuitamente para *download* na Internet, o MediaWiki destaca-se fortemente pelos seguintes fatores:

- código fonte aberto;
- desenvolvido por uma grande comunidade;
- utiliza tecnologias abertas e difundidas: PHP e MySQL;
- oferece mecanismos para extensão;
- disponibiliza através da comunidade de desenvolvedores algum suporte e documentação.

A principal vantagem do MediaWiki frente aos outros softwares é sua utilização pela Wikipedia. A Wikipedia é uma enciclopédia *online* que reproduz as enciclopédias tradicionais impressas (CROWSTON, 1997). A Wikipedia é o maior sucesso da história da ação coletiva na Web (PRIEDHORSKY et al., 2007). Pelo volume de acessos que a Wikipedia possui, em média 5000 requisições por segundo, conclui-se que o MediaWiki é robusto

¹<http://moodle.org/>

²<http://mambo-foundation.org/>

³<http://www.apple.com/server/macosx/features/wikis.html>

e confiável. Conclui-se também que a usabilidade deste software já foi absorvida pela comunidade. O MediaWiki é desenvolvido pela *Wikimedia Foundation*⁴, uma organização que além de desenvolver o software também mantém uma série de sites wiki como a própria Wikipedia, além do Wikinews⁵, Wikibooks⁶ e Wikitionary⁷.

5.1 Uma Breve História Sobre Ambientes Wiki

Como conceito, Wiki surgiu em 1995, criado por Ward Cunningham (LEUF; CUNNINGHAM, 2001). Com grande conhecimento em *Design Patterns*⁸, Cunningham é um programador americano que começou a implementar o primeiro Wiki em 1994, conhecido como WikiWikiWeb.

A terminologia WikiWikiWeb surgiu por uma associação que Cunningham fez entre o termo *quick* e o nome da linha de ônibus que liga os terminais de um aeroporto havaiano chamada “Wiki Wiki”. Esta opção foi uma alternativa para “*Quick-web*”. Por questões de simplificação, WikiWiki foi reduzido gerando o termo Wiki.

A inspiração de Cunningham para sua criação surgiu através do *HyperCard*⁹ da Apple¹⁰. *HyperCard* são pilhas de cartões virtuais que podem ser ligados entre si. O primeiro Wiki que de fato entrou em funcionamento foi o *Portland Pattern Repository*¹¹, com o objetivo de aprofundar discussões e propor novas idéias a respeito de *Design Patterns*. Em 2006, Cunningham admitiu que pensou na possibilidade de patentear seu invento na época em que o criou.

Conceitualmente, pode-se definir como Wiki todo aquele site que dispõe de conteúdo editável por qualquer usuário. A edição deve permitir a adição de novos conteúdos e a modificação daqueles já existentes. Um site Wiki deve também, a exemplo dos *HyperCards*, permitir que sejam criadas ligações entre os conteúdos disponíveis. Não é comum a existência de controle de acesso rigoroso em sites Wiki.

5.2 Funcionalidades Originais do MediaWiki

O ambiente MediaWiki dispõe de funcionalidades básicas necessárias à proposta apresentada por Oliveira et al (2005). Com ele, é possível manter um cadastro de usuários, criar, editar e discutir artigos. Os artigos disponíveis podem conter ligações para outros artigos e para recursos externos. Dentre as funcionalidades originais do MediaWiki, as mais relevantes para a proposta são Edição e Discussão.

A Edição oferece ferramentas que auxiliam a criação e modificação de artigos. A principal ferramenta é um editor visual simplificado, que permite criar automaticamente as marcações de formatação. As marcações utilizam uma linguagem própria para apresentação visual, conhecida como *Wikitext*¹². Nos casos de usuários mais experientes com o ambiente, existe também a possibilidade de utilizar a linguagem de marcação manualmente, sem a dependência da ferramenta visual. É possível integrar uma interface WY-

⁴<http://wikimediafoundation.org>

⁵<http://www.wikinews.org>

⁶<http://wikibooks.org>

⁷<http://www.wiktionary.org>

⁸[http://en.wikipedia.org/wiki/Design_pattern_\(computer_science\)](http://en.wikipedia.org/wiki/Design_pattern_(computer_science))

⁹<http://en.wikipedia.org/wiki/HyperCard>

¹⁰<http://www.apple.com>

¹¹<http://c2.com/ppr>

¹²<http://en.wikipedia.org/wiki/Wikitext>

SIWYG¹³ para tornar a utilização mais amigável para leigos.

Neste ambiente, qualquer usuário tem acesso à criação e à edição de artigos. Desta forma, um usuário A poderá fazer modificações em um artigo escrito por um usuário B e vice-versa. Todas as edições são armazenadas criando um controle de versões. Qualquer edição incompatível pode ser desfeita. Qualquer usuário cadastrado no MediaWiki pode fazer modificações nestas versões, ficando seu usuário vinculado a cada alteração. Este controle pode ser usado como fonte de informações para avaliar o comportamento de cada usuário.

Todo o artigo submetido ao MediaWiki pode ser discutido por todos os usuários, através da interface de discussões. Uma interface própria para estas discussões está disponível para cada artigo. Ao participar das discussões, os usuários ficam livres para expor informalmente suas idéias. O conteúdo das discussões não incide diretamente no conteúdo dos artigos. Por causa disto, é comum encontrar discussões polêmicas e contraditórias em ambientes Wiki. Por outro lado, na discussão podem surgir idéias ou complementos interessantes para o conteúdo dos artigos.

5.3 Fatores de Implementação do Mediawiki

O MediaWiki é um gerenciador de conteúdo desenvolvido para a Web. Apresenta as funcionalidades através de páginas HTML exibidas pelo navegador do usuário. Nenhum processamento adicional, senão a tarefa de visualização, é realizado no computador do usuário. Todo o processamento textual, bem como os mecanismos de armazenamento, encontram-se em um equipamento servidor. Por utilizar esta estrutura, o MediaWiki foi implementado utilizando uma linguagem de *scripts* específica para a Web, chamada PHP¹⁴.

PHP é uma linguagem interpretada de alto nível e fácil utilização. Não define formalismos rígidos, não é preciso declarar variáveis ou especificar tipos de dados. Conversões de tipagem são feitas automaticamente, conforme o contexto de utilização. Suas estruturas de controle, operadores lógicos, aritméticos e relacionais seguem o padrão utilizado pela linguagem C¹⁵. É uma linguagem voltada totalmente para a Web. PHP é uma alternativa de software livre para implementar sistemas que até então deveriam utilizar ASP¹⁶ ou JSP¹⁷. O interpretador PHP é executado diretamente pelo *thread* do servidor Web, que cuida da requisição feita pelo cliente, evitando a execução de novos processos. Executar processos independentes para cada requisição exige maiores recursos de hardware, tornando PHP uma alternativa mais otimizada que o uso de CGI¹⁸.

A linguagem PHP encontra-se atualmente na versão 5, oferecendo suporte total à Orientação a Objetos. Nas versões anteriores, algum suporte parcial era fornecido. Normalmente os programas eram desenvolvidos utilizando programação estruturada, definindo estruturas de dados separadas das funcionalidades. O MediaWiki é implementado obedecendo o paradigma OO, compatível com a versão PHP 4, utilizando somente os padrões de base como herança e encapsulamento.

A maior parte dos servidores Web suportam PHP. Existem módulos PHP desenvolvi-

¹³<http://en.wikipedia.org/wiki/WYSIWYG>

¹⁴<http://www.php.net>

¹⁵[http://en.wikipedia.org/wiki/C_\(programming_language\)](http://en.wikipedia.org/wiki/C_(programming_language))

¹⁶http://en.wikipedia.org/wiki/Active_Server_Pages

¹⁷http://en.wikipedia.org/wiki/JavaServer_Pages

¹⁸<http://www.w3.org/CGI/>

dos especificamente para cada servidor disponível no mercado, geralmente distribuídos gratuitamente na Internet. Os módulos fazem com que servidores que não possuem suporte nativo interpretem PHP. O servidor Apache¹⁹, o mais utilizado na Web, oferece suporte nativo para PHP.

Com relação à persistência de dados, o MediaWiki utiliza o MySQL²⁰, o SGBD mais utilizado em associação com PHP. Na implementação original, o MediaWiki utiliza 29 tabelas. Após a implementação das extensões necessárias para criar o ambiente descrito neste trabalho foram contabilizadas 32 tabelas e uma visão. As tabelas são todas do tipo InnoDB²¹, com suporte a transações ACID²². O MySQL surgiu com uma proposta de simplicidade e desempenho, o que na prática se confirmava. Para atingir alto desempenho, o MySQL sacrificava funcionalidades essenciais como integridade referencial e controle de transações. As versões atuais oferecem novos tipos de tabelas que implementam as funcionalidades sacrificadas no passado, oferecendo menor desempenho.

O MediaWiki é um ambiente *Open Source*, seu código fonte está disponível na Internet e pode ser utilizado e modificado livremente. As tecnologias empregadas em seu desenvolvimento como PHP e MySQL também são *Open Source*. Este é mais um motivo que influenciou na escolha por estas tecnologias. Para implantar um site Wiki utilizando MediaWiki basta dispor de um servidor configurado com os serviços Apache e MySQL, baixar os arquivos de instalação do MediaWiki e executar o *script* de instalação.

5.4 A Linguagem de Marcação *Wikitext*

Linguagens de marcação são ferramentas largamente utilizadas, exemplo disto são as páginas Web, escritas em uma linguagem de marcação chamada HTML. Geralmente interpretadas, as linguagens de marcação formatam documentos que podem conter vários tipos de mídias que vão desde textos até imagens e vídeos. Existe uma série de linguagens de marcação para os mais determinados fins.

Por proposta, ambientes Wiki devem ser acessíveis a qualquer usuário, sem que estes tenham de ser *experts* em tecnologias. Por causa disto, ambientes Wiki utilizam uma linguagem de marcação simples, fácil de aprender e utilizar. *Wikitext*, também conhecida como *Wiki Markup*, permite formatar documentos eficientemente exigindo do usuário o conhecimento de uma pequena quantidade de comandos e estruturas. Sua gramática, palavras chave, estruturas e funcionalidades são dependentes do ambiente Wiki utilizado. Ainda não existe um padrão para a linguagem, mas algumas iniciativas estão surgindo em busca de uma definição. *WikiCreole* é uma proposta que busca, através da combinação não-trivial de várias linguagens, criar uma linguagem estável que permita a troca de informações entre wikis que utilizem diferentes padrões de linguagem (SAUER; SMITH; BENZ, 2007).

A figura 5.1 confronta o resultado final e o código fonte de um documento gerado com a linguagem *Wikitext*. Esta figura mostra que é possível aprender a utilizar a linguagem com base em exemplos. Já que é possível visualizar o código fonte de todos os artigos publicados em um Wiki, torna-se fácil aprender a utilizar a linguagem. Para isto, o usuário precisa analisar o código fonte de uma página que apresente alguma estrutura que ele deseje utilizar. Esta estrutura pode ser copiada e adaptada à nova página onde o usuário

¹⁹<http://httpd.apache.org/>

²⁰<http://www.mysql.com>

²¹<http://www.innodb.com/>

²²<http://en.wikipedia.org/wiki/ACID>

Uma seção de teste [editar]		
Isto é um teste		
Texto em negrito		
Texto em <i>itálico</i>		
Link: ligação externa		
Uma Tabela de Teste		
Coluna 1	Coluna 2	Coluna 3
Valor A	Valor B	Valor C
Valor A2	Valor B2	Valor C2


```

==Uma seção de teste==
Isto é um teste
----
Texto em '''negrito'''

Texto em '''itálico'''

Link: [http://www.wikimedia.org ligação externa]

{| border="1"
|+ Uma Tabela de Teste
|-
|Coluna 1 || Coluna 2 || Coluna 3
|-
|Valor A
|Valor B
|Valor C
|-
|Valor A2
|Valor B2
|Valor C2
|}

```

Figura 5.1: Wikimarkup e sua respectiva saída

deseja recriar a estrutura.

5.5 Arquitetura do MediaWiki

O MediaWiki não é uma combinação de pequenos pacotes. O ambiente inteiro está disponível para *download* via Internet, em um único arquivo. Este arquivo chega ao usuário no formato *Zip*, compactado. Descompactando o arquivo, obtêm-se a estrutura do ambiente pronta para execução.

Seguindo os padrões de desenvolvimento da comunidade de Software Livre, a arquitetura do ambiente MediaWiki proporciona facilidades para o desenvolvimento distribuído, onde diversos desenvolvedores são responsáveis por várias contribuições. Para permitir esta distribuição é preciso utilizar uma estrutura de código fonte organizada, de forma que seja possível compreender o que cada trecho de código deve realizar.

O ambiente MediaWiki é desenvolvido utilizando PHP, uma tecnologia que oferece um interpretador de código que funciona como um módulo do servidor web, e uma linguagem de programação de alto nível, com sintaxe semelhante a utilizada pela linguagem C. Esta tecnologia permite o desenvolvimento baseado em alguns padrões do paradigma de programação orientada a objetos, mesmo que com algumas restrições.

Para permitir o desenvolvimento coletivo e uma melhor organização do código fonte, o ambiente MediaWiki procura utilizar sempre a orientação a objetos. Após instalado, a partir do seu diretório raiz é possível acessar o subdiretório `includes`, onde serão encontrados 166 arquivos com extensão PHP, onde ficam as implementações de todas as classes que o ambiente utiliza.

5.6 A Hierarquia de Classes

A hierarquia de classes do MediaWiki é composta por mais de 900 classes. Sua documentação não é completa, o que dificulta muito o aprendizado e as possibilidades de uso racional. A falta de documentação cria também dificuldades para o reuso das funcionalidades implementadas nas classes. Por outro lado, por se tratar de um software com

código aberto, é possível analisar o código fonte buscando compreender o funcionamento dos métodos de cada classe. A análise do código é fundamental para aqueles que precisam de uma compreensão aprofundada do funcionamento do ambiente.

O MediaWiki foi projetado para distribuir processamento entre vários servidores diferentes. Para garantir a distribuição são necessárias muitas classes que não estão envolvidas no processamento dos conteúdos disponíveis no ambiente. Não é necessário compreender o funcionamento destas classes para criar extensões que processem somente conteúdo.

O nível de importância para as tarefas de extensão começa a crescer quando surgem as classes que lidam com o código HTML e com o código *Wikitext*. Todo o HTML gerado como saída para o browser do usuário é resultado de uma sucessão de chamadas a métodos de poucas classes, dentre as 900 citadas anteriormente. O processamento de conversão entre *Wikitext* e HTML é feito pela classe `Parser`. Dentro do contexto de extensões criado por este trabalho, `Parser` é a classe mais importante no contexto de todo o ambiente. É preciso compreendê-la para criar qualquer alteração na criação dos artigos.

5.6.1 A Classe Parser

Conceitualmente, um *parser* é uma peça de software que “quebra” um documento em *tokens*, analisa a sintaxe de cada *token* para formar uma expressão válida, interpreta a expressão e executa as ações correspondentes (ZAMBON, 2007). De fato, a classe `Parser` encontrada no MediaWiki executa esta tarefa. A linguagem *Wikitext* possui uma gramática definida, permitindo que os métodos mostrados na Figura 5.2 validem todas as sentenças encontradas. As sentenças podem ser ligações, tabelas, seções, imagens, dentre outras formatações. Após validadas pelo *parser*, as sentenças de um documento são armazenadas no SGBD, no formato *Wikitext*. O caminho reverso também é previsto e pode ser executado por métodos desta classe.

No momento em que um usuário acessa um documento no ambiente ele é primeiramente recuperado pelo SGBD. O SGBD possui o documento no formato *Wikitext*. Para que possa ser exibido para o usuário, este documento precisa ser traduzido para HTML, permitindo que o browser gere o layout estabelecido pelo criador do documento. O *parser* processa o documento recuperado do SGBD, executando a tradução para HTML.

Public Member Functions

<code>Parser()</code>
<code>firstCallInit()</code>
<code>setOutType(\$ot)</code>
<code>uniqPrefix()</code>
<code>unstripforHTML(\$text)</code>
<code>magicLinkCallback(\$m)</code>
<code>makeLinkHolder(&\$nt,\$text=",\$query=",\$trail="\$prefix=")</code>
<code>armorLinks(\$text)</code>
<code>areSubpagesAllowed()</code>
<code>getCommon(\$st1,\$st2)</code>
<code>openList(\$char)</code>
<code>nextItem(\$char)</code>
<code>closeList(\$char)</code>

Figura 5.2: Os métodos públicos da classe `Parser`

É possível modificar o *parser*, gerando saídas especiais. Na implementação do protótipo abordado por este trabalho a classe `Parser` foi modificada, passando a inserir um componente visual em todos os artigos exibidos pelo ambiente. Este componente visual

é utilizado para “colher” as avaliações qualitativas informadas na avaliações dos artigos. O componente deverá aparecer em todos os documentos disponíveis no ambiente. Não é necessário inserir qualquer marcação para que o componente seja exibido. Com as modificações, a classe `Parser` escreve o código referente ao componente visual no cabeçalho de cada documento.

O *parser* do ambiente MediaWiki foi construído com base em Expressões Regulares. Com origem na álgebra formal, Expressões Regulares são linguagens generalizadas para reconhecimento de padrões, com expressivo poder (FRIEDL, 1997). Para que se possa modificar o *parser* do MediaWiki é necessário conhecer REGEX de forma aprofundada.

5.7 A Arquitetura da Base de Dados

Seguindo os padrões dos sistemas de informações contemporâneos, a retaguarda do ambiente MediaWiki é um SGBD. Este SGBD possui tabelas que armazenam todos os dados do ambiente, gerando persistência. A base de dados implementada utiliza tabelas que executam sobre a *engine* InnoDB. InnoDB é uma extensão incorporada ao MySQL que permite o uso de integridade referencial e controle de transações. Estas funcionalidades atuam diminuindo os riscos de inconsistências. Os recursos de integridade referencial não são muito explorados no MediaWiki. Por outro lado, por permitir edição colaborativa, o que implica em múltiplos acessos simultâneos aos registros, os recursos de controle de transações são críticos.

As estruturas das tabelas utilizadas no ambiente permitem expansões, já que grande parte dos dados que armazenam não aparecem na forma de atributos nas relações. As relações apresentam atributos que formam chaves de relacionamento e atributos que definem campos textuais, com estruturas não normalizadas. Mesmo não obedecendo as regras de normalização, esta abordagem é interessante por permitir que outros dados sejam armazenados sem a obrigatoriedade da criação de novas relações. O Apêndice A mostra as relações que compõe a base da dados do ambiente.

As consultas feitas sobre a base de dados utilizam um mecanismo de acesso implementado na classe `Database`. Desta forma, é possível realizar consultas e atualizações nos dados sem a necessidade de utilizar SQL. A classe `Database` implementa uma linguagem própria para estas ações, baseada em vetores e apontadores.

5.8 Namespaces

O MediaWiki apresenta ao usuário uma série de páginas. Estas páginas exibem conteúdo inserido pelos próprios usuários ou dispõem de alguma funcionalidade para administração dos usuários ou dos conteúdos. A exibição de artigos, por exemplo, é feita por um único tipo de página específica. A discussão de artigos é feita através de uma página de discussões.

Em um site comum, a implementação das páginas é realizada através de arquivos distintos. Nestes casos, uma página que exibe dados dos usuários poderia ser implementada em um arquivo chamado `usuarios.php`. A página que exibe um relatório poderia ser implementada em um arquivo chamado `relatorio.php`. Esta relação define uma nomenclatura que indica a funcionalidade de cada arquivo, correspondendo a um tipo de página. Para acessar uma página, o usuário indica o nome do arquivo correspondente à página desejada na própria URL.

O MediaWiki não implementa suas páginas nos moldes dos sites comuns. Para favore-

cer as extensões, o reuso e a distribuição de processamento, cada página é implementada através de uma classe. O núcleo do MediaWiki instancia a classe correspondente a página buscada pelo usuário, montando seu *layout* e suas estruturas de dados com base nos atributos e métodos definidos na classe. Diferente das implementações comuns, o usuário não passa na URL nomes de arquivos. Neste caso, o usuário passa um atributo que indica ao núcleo do MediaWiki qual classe deve ser instanciada.

O MediaWiki define um *namespace* associando classes que correspondem a algum tipo de página a um rótulo textual. Conhecendo este *namespace* é possível acessar diretamente a página desejada, através da passagem do rótulo via URL, na forma de atributo. Os rótulos definidos no *namespace* também ajudam programadores nas tarefas de extensão. Pode-se realizar testes para identificar que tipo de página está sendo acessada através dos rótulos. A tabela 5.1 mostra os rótulos definidos no *namespace*, seu correspondente código numérico e a funcionalidade à qual cada um está relacionado.

Tabela 5.1: *Namespaces* e suas páginas relacionadas

Identificador do <i>Namespace</i>	Código	Título
NS_MEDIA	NULL	Página de notícias
NS_SPECIAL	NULL	Página especial
NS_MAIN	0	Página principal
NS_TALK	1	Discussões
NS_USER	2	Página de usuário
NS_USER_TALK	3	Discussão sobre usuários
NS_IMAGE	6	Imagem
NS_IMAGE_TALK	7	Discussão sobre imagem
NS_TEMPLATE	10	Predefinição
NS_HELP	12	Ajuda
NS_HELP_TALK	13	Discussão sobre ajuda
NS_CATEGORY	14	Páginas de categorias
NS_CATEGORY_TALK	15	Discussão sobre categorias

5.9 Mecanismos de Extensão

Após analisar a documentação disponível e partes do código fonte do MediaWiki, pôde-se identificar os mecanismos e os procedimentos necessários para o processo de extensão. Mesmo contando com uma documentação pobre, nota-se que o ambiente foi projetado com vistas a extensões e a criação de novas funcionalidades.

Em um primeiro momento, o código fonte do ambiente parece extremamente complexo. Utiliza diversas inclusões de bibliotecas, dificultando a leitura e a compreensão. Por outro lado, por ser bem estruturado e possuir uma padronização rigorosa quanto aos nomes de classes, métodos e variáveis, parte da complexidade é abstraída. Além do *namespace*, que favorece as extensões, o MediaWiki fornece as *SpecialPages* e os *Hooks*, os quais serão melhor descritos a seguir.

5.9.1 *SpecialPages*

Para que se possa criar novas páginas contendo formulários, imagens ou textos, o MediaWiki permite a criação de *SpecialPages*. Uma *SpecialPage* é uma página que executa alguma ação que não estava prevista no ambiente original. Esta é a única forma elegante de criar novas páginas que implementem funcionalidades diferentes das originais.

É possível criar *SpecialPages* que executam qualquer ação sobre os dados disponíveis no Banco de Dados. Após criada, uma *SpecialPage* é vinculada ao ambiente através da inserção de uma linha de código em um arquivo de configuração, tornando esta nova *SpecialPage* disponível para o uso. As características de *layout* e menus são mantidas automaticamente. É necessário implementar somente a funcionalidade que se busca executar com esta nova página.

Não existem limites para a criação de *SpecialPages*, por outro lado, a profundidade de modificações que elas permitem são restritas. Não é possível, por exemplo, modificar o *parser* do MediaWiki criando uma *SpecialPage*. Para este tipo de modificação é necessário reprogramar o código da própria classe.

5.9.2 *Hooks*

Análogos aos componentes de interface gráfica, como botões e caixas de texto, que disparam eventos conforme as ações dos usuários, os *hooks* do MediaWiki são disparados conforme eventos específicos. Existe uma série de eventos deste tipo já consolidados no ambiente. Eles são disparados por ações que vão desde a autenticação de um usuário até a inserção de um artigo ou comentário. A tabela 5.2 mostra alguns exemplos de *hooks*, identificando o momento em que eles são executados.

Os *hooks* geram uma facilidade no contexto das extensões porque permitem mapear os eventos para ações. As ações são definidas em funções que podem ser incluídas em arquivos do ambiente (*includes*). É possível utilizar o *hook* de autenticação, por exemplo, para executar um processamento que identifique as classes vinculadas a um usuário detectando se este é um autor, revisor ou comentarista. Esta abordagem foi utilizada no protótipo implementado. Foi criada uma função que recupera do banco de dados as classes às quais cada usuário está atrelado.

A existência de *hooks* constitui o maior avanço com relação a extensão no MediaWiki. Caso uma estrutura semelhante não fosse disponível, seria preciso reimplementar grandes partes do ambiente original para adequá-lo às funcionalidades necessárias no protótipo. Os *hooks* tornaram viável a adoção do MediaWiki como ambiente base do protótipo construído neste trabalho.

5.10 Controle de Versões

Um ambiente Wiki deve crescer, aumentando rapidamente a quantidade e a qualidade de seu conteúdo. A exemplo da Wikipedia, o conteúdo aberto para edição estimula as contribuições. Por outro lado, a abertura também favorece o surgimento de problemas. O crescimento dos ambientes Wiki trouxe um problema conhecido como “Pixação”. Ambientes colaborativos enfrentam desafios maiores que os enfrentados pelos tradicionais quanto a problemas referentes à qualidade de seus conteúdos (GLOVER; HIRST, 1996). Alguns problemas de conteúdo podem ser motivados por subseqüentes erros humanos ou por vandalismos.

Sites fechados são constantemente invadidos. Os invasores deixam marcas no site

Tabela 5.2: Exemplos de hooks e seus respectivos eventos

Nome do <i>Hook</i>	Evento
AddNewAccount	Chamado quando uma nova conta de usuário é criada.
AbortNewAccount	Quando a criação de uma nova conta é cancelada.
ArticleDelete	Chamada quando um artigo é deletado.
ArticleSave	Quando um artigo é salvo pelo usuário.
EmailUser	Ocorrem quando um email é enviado a um usuário.
OutputPageBeforeHTML	Ocorre após a página ser montada, mas antes do código HTML ser mostrado.
UserLoginComplete	Acontece quando um usuário autentica com sucesso.
UserRights	Chamada quando um usuário é vinculado a algum grupo.

para provar seus feitos. Muitas “pixações” são motivadas simplesmente por fatores relacionados ao ego e ao exibicionismo. Existem sites específicos para divulgação de invasões. Por serem abertos, ambientes Wiki não estão sujeitos a este tipo de vandalismo, já que não oferecem desafio para o agressor. Por outro lado, existem “pixações” motivadas simplesmente pela diversidade de idéias e comportamentos.

O MediaWiki implementa uma solução simples e eficiente para solucionar o problema das pixações. Esta solução mantém a edição aberta, não comprometendo a colaboração e o acesso democrático. Para que isto aconteça, o MediaWiki armazena em seu Banco de Dados todo o conteúdo de cada artigo, cada vez que qualquer um deles for modificado. Independentemente do tipo ou tamanho da modificação, no momento em que um artigo é modificado, todo seu conteúdo é novamente armazenado gerando um novo registro no Banco de Dados.

Os armazenamentos subseqüentes criam versões diferentes do mesmo artigo. A qualquer momento um usuário pode acessar o conteúdo de qualquer versão. Pode-se também fazer comparações entre as versões. Um algoritmo de *diff* é executado, mostrando as diferenças existentes nas versões do artigo. Caso alguma “pixação” seja identificada, basta que qualquer usuário modifique o conteúdo, recuperando a qualidade do artigo.

5.11 Documentação

Seguindo a linha dos projetos *open source*, a documentação do MediaWiki é paradoxal. Existem tutoriais completos que auxiliam administradores de servidores no processo de instalação e implantação. Por outro lado, a documentação de código, fundamental para desenvolvedores que buscam estender o ambiente, ainda é restrita.

No tocante ao desenvolvimento, a documentação disponível é incompleta. Chega-se facilmente a esta constatação no momento em que se realiza qualquer pesquisa na Web em busca de informações. Basicamente, tudo o que se encontra são páginas incompletas, que não contribuem para a solução dos problemas. Alguns tópicos são levantados, citando passos iniciais para o desenvolvedor. Estes passos geralmente acabam sem chegar ao

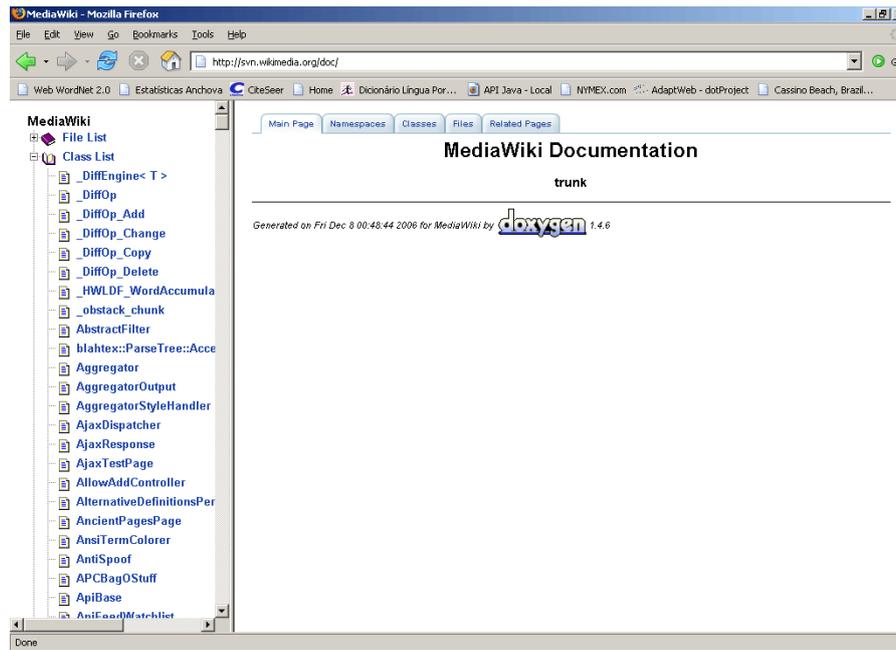


Figura 5.3: A interface de documentação gerada pelo Doxygen

objetivo final.

Existe um site dedicado a documentação do ambiente. Neste site existe um link indicado para desenvolvedores. Este link compõe a maior fonte de informações, na forma de documentação. Nele é possível acessar a documentação gerada automaticamente. A documentação descreve a hierarquia de classes e seus métodos. Informações mais completas, indicando como funcionam as estruturas de dados ou como são implementados os algoritmos, não estão disponíveis.

A documentação automática encontrada foi produzida através do software Doxygen²³. Doxygen é um gerador de documentações que utiliza marcações deixadas no código fonte. O Doxygen está implementado para linguagens como C, C++²⁴, Java²⁵ e PHP. Semelhante ao Javadoc²⁶, ele cria uma interface Web interativa, com ligações para navegação pela hierarquia de classes. Por outro lado, caso os comentários do código fonte não sejam completos, o resultado gerado pelo Doxygen também será incompleto. A Figura 5.3 mostra a interface da documentação criada pelo Doxygen para o código fonte do MediaWiki. Percebe-se, ao navegar pela documentação, que os textos encontrados são os próprios comentários presentes no código fonte. Caso não existam comentários, a documentação gerada não exibe nenhuma descrição referente à classe ou método analisado.

A falta de documentação do MediaWiki é um problema crítico para um processo de extensão. Por outro lado, já que é um software de código aberto, o MediaWiki conta com uma série de desenvolvedores espalhados pelo mundo. É relativamente fácil encontrar pessoas em fóruns e listas de discussão dispostas a colaborar. Existe uma lista de discussão aberta. Nesta lista, os principais programadores que implementaram as principais classes do ambiente, como `Parser` e `Database`, respondem dúvidas enviadas por e-mail por outros programadores menos experientes. O bom uso desta lista constitui uma

²³<http://www.doxygen.org>

²⁴<http://en.wikipedia.org/wiki/C++>

²⁵<http://java.sun.com>

²⁶<http://java.sun.com/j2se/javadoc>

ótima ferramenta de aprendizagem para compreensão do ambiente.

6 O PROTÓTIPO IMPLEMENTADO

A implementação do protótipo adaptou o MediaWiki de forma que o sistema permitisse os procedimentos definidos no processo proposto por este trabalho. As características básicas do ambiente foram mantidas. As extensões foram planejadas em módulos. Pode-se modularizar o protótipo por funcionalidades, agrupando-as por relacionamento, como mostra a figura 6.1. Não existe uma definição formal para modularizar o MediaWiki.

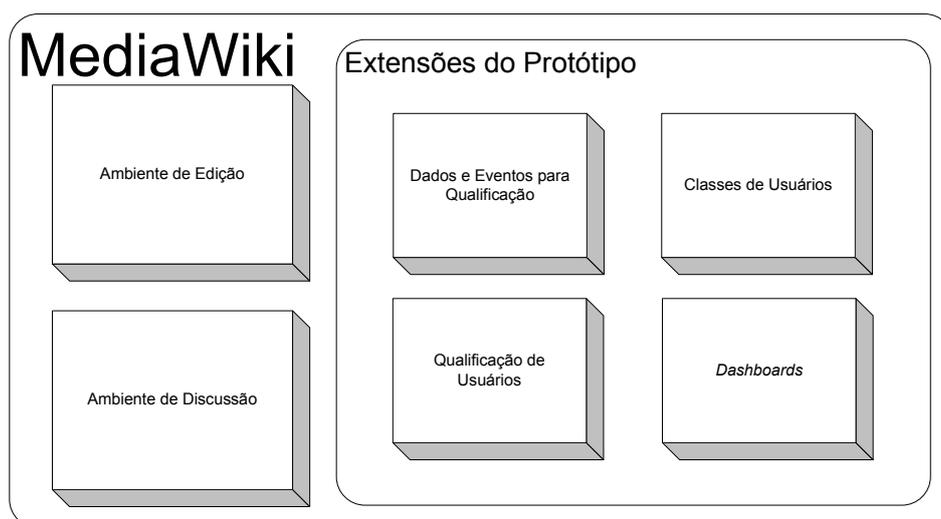


Figura 6.1: Definição de módulos por funcionalidades

É possível distribuir estas modificações para a comunidade na forma de extensões para o MediaWiki, disponibilizando-as na Web para download. Para facilitar a integração, as extensões foram implementadas em 3 arquivos PHP, os quais necessitam de mais 3 arquivos GIF (contendo imagens adicionais). Ao todo, as extensões ocupam menos de 50KB de memória.

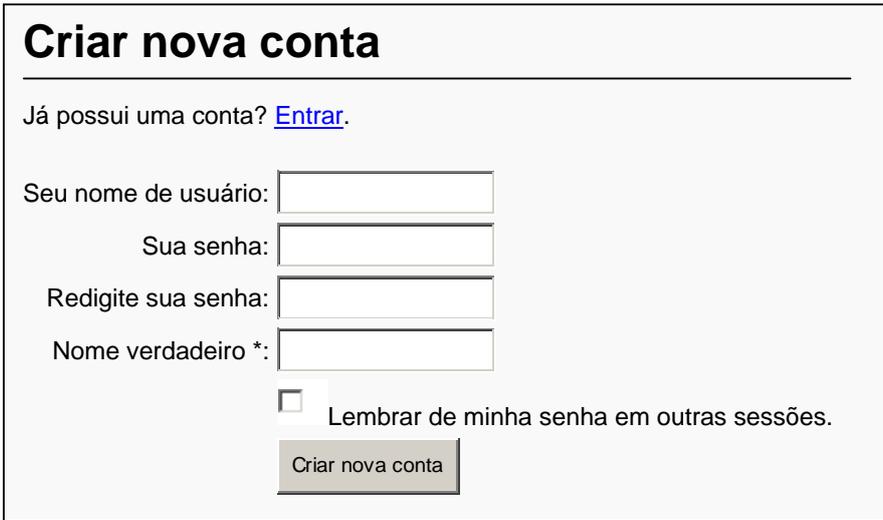
O protótipo pode ser utilizado para atuar como interface entre autores e eventos. Os autores podem criar seus artigos diretamente no ambiente. O processo de revisão pode ocorrer de forma aberta e identificada. Além dos eventos, este protótipo pode também ser aplicado em ambientes acadêmicos. Pode-se utilizar este protótipo como uma interface entre alunos de uma turma de graduação, por exemplo, e o professor. Desta forma, os artigos dos alunos podem ser criados e disponibilizados no ambiente, ficando disponíveis para toda a turma. As revisões ou correções podem ser feitas pelo professor e os demais alunos podem atuar como comentaristas. O protótipo implementa um mecanismo

de qualificação que atribui pontos para os usuários de acordo com sua interação com o ambiente e com as avaliações que seus artigos recebem. Estes pontos podem ajudar o professor nas tarefas de avaliação dos alunos, oferecendo um indicativo base para definição de conceitos.

6.1 Cadastro de Usuários

O cadastro de usuários do protótipo seguiu a implementação original do MediaWiki. Não foi necessário modificar as estruturas da base de dados original. Foi criado um *hook* que vincula os usuários à classe *Comentarista*, automaticamente, após a realização do cadastro.

Seguindo o conceito Wiki, onde as exigências de informações por parte dos usuários são mínimas, o protótipo solicita ao usuário somente login e senha para acesso. Existe um campo onde poder ser informado o nome do usuário, mas seu preenchimento é opcional. Não é permitida a duplicidade de login de acesso. O usuário pode informar se deseja que sua senha seja armazenada em uma sessão persistente. Os dados inseridos pelo usuário não passam por qualquer processo de validação adicional. O formulário pode ser observado através da figura 6.2.



Criar nova conta

Já possui uma conta? [Entrar](#).

Seu nome de usuário:

Sua senha:

Redigite sua senha:

Nome verdadeiro *:

Lembrar de minha senha em outras sessões.

Figura 6.2: Formulário de cadastro de novos usuários

Caso o protótipo tenha de ser utilizado em um evento que necessite de informações adicionais, pode-se estender o cadastro, incluindo novos campos no formulário. Estas informações adicionais podem ser armazenadas na entidade original do ambiente. Esta entidade é a tabela *User*, que utiliza uma estrutura não normalizada, onde pode-se definir atributos adicionais. O MediaWiki trabalha com entidades não normalizadas, com campos textuais onde os dados são armazenados seguindo uma estrutura própria. Esta abordagem só é aplicada em entidades com um número de acessos reduzido, evitando problemas de desempenho para o sistema. A tabela *User* só é acessada no momento em que um usuário realiza o *login* no ambiente. As informações recuperadas da tabela ficam temporariamente armazenadas em variáveis de sessão, evitando novos acessos consecutivos à base de dados.

6.2 Classes de Usuários

Seguindo a proposta apresentada neste trabalho, o processo de edição e revisão dos artigos deve ser aberto. Esta abordagem é utilizada para aumentar a transparência. Por outro lado, algumas permissões precisam ser mantidas fechadas. Não condiz com a proposta permitir que qualquer usuário modifique qualquer artigo. É exclusivo ao autor e aos revisores este tipo de privilégio.

Para definir as permissões foram implementadas novas classes de usuários. As classes implementam bloqueios que restringem o acesso a algumas funcionalidades. Ao todo foram criadas 3 novas classes: *Autor*, *Revisor* e *Comentarista*. A implementação seguiu o padrão encontrado no MediaWiki para classes de usuários. A integração das novas classes com o ambiente ficou facilitada, já que reutilizou todas as estruturas e interfaces disponíveis no ambiente. Basta que o administrador do sistema acesse a interface de gestão de privilégios dos usuários para manipular os vínculos dos mesmos com as classes, como pode ser observado na figura 6.3. A interface de administração de privilégios dos usuários pode ser acessada através de uma ligação disponível no menu de opções, localizado à esquerda da interface geral do protótipo.

Editar grupos do utilizador

Editando A

Membro de:

Grupos disponíveis:

Selecione os grupos no qual deseja que o utilizador seja removido ou adicionado. Grupos não seleccionados, não serão alterados. Pode seleccionar ou remover a selecção a um grupo com CTRL + Click esquerdo

Figura 6.3: Formulário de gestão de privilégios de usuários

Automaticamente, após a realização do cadastro, os usuários são vinculados à classe *Comentarista*, que permite que todos realizem acessos e comentários aos conteúdos. Usuários que desejarem criar novos artigos são vinculados também à classe *Autor*. Somente os usuários vinculados à classe *Revisor* podem modificar artigos que não forem de sua própria autoria. O vínculo dos usuários com a classe *Revisor* é feito manualmente, pelo administrador do protótipo.

6.3 Criação e Edição de Artigos

Análogo à Wikipedia, que também utiliza o MediaWiki, este protótipo dispõe de interfaces específicas para criação e edição de artigos. No protótipo, o MediaWiki foi estendido de forma a obedecer as definições do processo proposto neste trabalho. As interfaces para criação e edição de artigos foram mantidas. As modificações surgem somente nas permissões de usuários.

O protótipo segue o esquema de ligações encontrado nos ambientes Wiki tradicionais. Estas ligações acontecem quando um artigo possui um *link* para outro artigo, também

disponível no ambiente. Caso seja criada uma ligação para um artigo que ainda não está disponível, é aberta a oportunidade para criação de um novo artigo.

Para criar novos artigos basta editar o conteúdo de um artigo existente, inserindo uma nova ligação apontando para um artigo que ainda não foi criado. Esta ligação fica visível, destacada por uma cor diferenciada, indicando que aquele conteúdo ainda não está disponível. Ao acessar esta ligação o usuário tem a oportunidade de criar o novo artigo e disponibilizar o conteúdo. Depois de criado o artigo o apontador, que antes aparecia em uma cor diferenciada, passa a seguir o padrão de todos os outros apontadores para artigos disponíveis internamente.

A figura 6.4 mostra um artigo com uma série de ligações. As ligações em azul apontam para artigos disponíveis. As ligações em vermelho apontam para artigos ainda não disponíveis.

Ambientes de desenvolvimento

[\[editar\]](#)

É possível desenvolver aplicações em Java através de vários [ambientes de desenvolvimento integrado](#) (IDE's). Dentre as opções mais utilizadas pode-se destacar:

- [Eclipse](#) — um projeto [aberto](#) iniciado pela [IBM](#);
- [NetBeans](#) — um ambiente criado pela empresa [Sun Microsystems](#);
- [JBuilder](#) — um ambiente desenvolvido pela empresa [Borland](#);
- [JDeveloper](#) — uma IDE desenvolvida pela empresa [Oracle](#);
- [JCreator](#) — um ambiente desenvolvido pela Xinox.
- [BlueJ](#) — um ambiente desenvolvido por uma faculdade australiana (muito bom para iniciantes).
- [JGRASP](#) — bom para intermediários, feito pela equipe do projeto GRASP.
- [IntelliJ IDEA](#) — uma IDE desenvolvida pela JetBrains (considerada por muitos a melhor IDE do mercado).

Figura 6.4: Ligações internas em azul e vermelho

O conteúdo dos artigos deve ser inserido no formato *wikitext*, através de uma interface visual que ajuda o usuário no processo de formatação. A interface de apoio à edição, ilustrada na Figura 6.5, é uma ferramenta simplificada que ajuda o usuário a formatar um artigo utilizando *wikitext*. Suas funcionalidades possibilitam que um usuário inexperiente crie e formate corretamente um artigo simples. Por outro lado, formatações avançadas como tabelas, resumos e sumários não estão disponíveis na interface. Para utilizar estas formatações o usuário deverá dispor de algum conhecimento prévio de *wikitext*. A interface de apoio à edição é automaticamente exibida na tela de edição de artigos.

Para editar um artigo, o protótipo segue também o padrão estabelecido pelos ambientes Wiki. Cada artigo disponível possui um apontador para a interface de edição. Quando acessada, a interface exibe o conteúdo do artigo no formato *wikitext*. O conteúdo pode ser completamente modificado e armazenado. As modificações geram um histórico de versões, permitindo que em qualquer momento sejam desfeitas as modificações.

O protótipo não modificou os procedimentos básicos de criação e edição. Somente as permissões passaram a obedecer políticas diferentes. Estas políticas foram definidas nas novas classes implementadas. No protótipo, somente usuários vinculados à classe `Autor` podem criar novos artigos. Usuários não vinculados à classe `Autor` podem acessar qualquer trabalho mas quando acessam uma ligação para um artigo não existente, a interface para criação de novo artigo não é exibida.

No tocante às edições de artigos existentes também foram implementadas novas políticas. Somente terão acesso a edição de um artigo o próprio autor do mesmo. Demais autores não poderão modificar um artigo existente. Usuários vinculados à classe `Revisor` também podem fazer edições em artigos existentes.

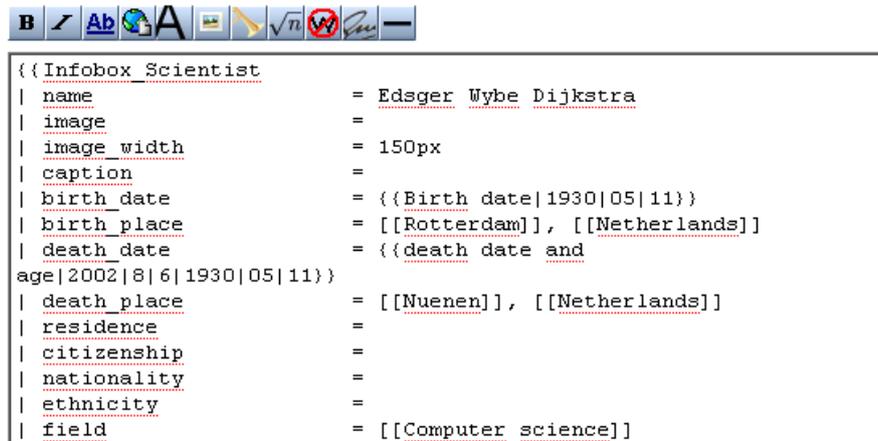


Figura 6.5: Interface de apoio à edição de artigos

6.4 Colhendo Dados e Eventos para Qualificação

O protótipo dispõe de uma área que exibe a qualificação dos usuários. A qualificação é calculada com base em dados obtidos através do sistema. Estes dados são provenientes dos eventos de interação dos usuários e das avaliações as quais os artigos criados por eles são submetidos. Os dados são convertidos para valores numéricos. Os valores são passados como argumentos de entrada para uma equação(EQ1), que retorna os pontos de qualificações de cada usuário.

6.4.1 Dados Quantitativos

Os dados provenientes de interações são quantitativos. O protótipo armazena os dados de cada acesso feito por cada usuário. Foi criada uma função para armazenar os dados no SGBD. Esta função foi vinculada ao *hook* `OutputPageBeforeHTML`, que é chamado sempre que um artigo é convertido de *Wikitext* para HTML, antes de ser exibido para o usuário.

Acessos a artigos de autoria do próprio usuário não computam pontos. Acessos consecutivos só são computados no momento em que o intervalo entre eles for superior a 24 horas. Não são computados acessos às páginas com informações dos usuários. Somente artigos são considerados.

6.4.2 Dados Qualitativos

Dados qualitativos são colhidos pelo protótipo através de avaliações de artigos feitas por usuários. Todos os artigos disponíveis podem ser avaliados por qualquer usuário. A avaliação segue o modelo encontrado em sistemas de reputação, onde o avaliador indica seu grau de satisfação com o recurso avaliado. Para colher os dados provenientes das avaliações foi inserido um componente visual que aparece junto aos artigos. Este componente exibe uma seqüência de 5 estrelas paralelas que correspondem respectivamente a péssimo, ruim, neutro, bom e ótimo. O componente pode ser observado na figura 6.6.

Para que um usuário realize uma avaliação o mesmo deve clicar na estrela que corresponde a sua satisfação com relação ao conteúdo em questão. O componente aparece mostrando as 5 estrelas sombreadas, indicando que aquele artigo ainda não foi avaliado. Ao clicar em uma estrela ela e as outras estrelas que estiverem à esquerda ficam coloridas, indicando a avaliação. Caso o usuário realize alguma avaliação errada, basta clicar nova-

Delphi



Predefinição:Info/IDE Delphi (em português, pronuncia-se "dél-fi"; em inglês, "del-fai") é um compilador e um [Ambiente de Desenvolvimento Integrado \(IDE\)](#) para o desenvolvimento de [softwares](#). Ele é produzido pela [Borland Software Corporation](#) (que, por algum tempo, foi chamada de [Inprise](#)).

Figura 6.6: Componente visual junto de um artigo intitulado *Delphi*

mente na estrela correta para corrigir o erro. O componente foi implementado utilizando AJAX¹, melhorando a experiência de utilização do usuário, já que o envio de dados não necessita de recarga do site.

6.5 Qualificando os Usuários com EQ1

Em sua forma pura, os dados colhidos pelo protótipo apontam duas informações. Estes dados indicam a frequência de interatividade de cada usuário com o ambiente. Com base nestes dados pode-se definir valores numéricos que indiquem diferenças entre usuários constantes e usuários ocasionais. Os dados indicam também informações com relação ao impacto gerado por cada trabalho.

As avaliações qualitativas apontam a opinião de cada usuário com relação a um determinado artigo. Quando avaliado por usuários diferentes, é provável que este artigo receba também avaliações diferentes. Demonstrar a qualificação do autor deste artigo atribuindo pontuações estáticas para as avaliações não valoriza a opinião de usuários mais experientes.

A qualificação dos usuários utilizando o método EQ1 considera a qualificação dos avaliadores. O objetivo deste método é fazer com que a opinião de usuários melhor avaliados prevaleça frente a outras opiniões. Entende-se que as avaliações de usuários com boa qualificação devem pesar mais sobre os resultados. As avaliações feitas por usuários com pior qualificação não são descartadas. Suas avaliações também são computadas e contribuem para a formação da qualificação. Por outro lado, seu impacto sobre a qualificação é menor.

Poucas avaliações positivas realizadas por usuário bem qualificado podem indicar uma qualificação muito positiva para o autor de um artigo. Caso este artigo receba também algumas poucas avaliações negativas, feitas por usuários com baixa qualificação, as qualificações positivas prevalecerão. Por outro lado, por serem mantidos os dados das avaliações feitas pelos usuários com baixa qualificação, caso o número de avaliações negativas for bastante superior, a tendência pode ser invertida.

O objetivo da aplicação deste método é gerar confiança. É natural acreditar que os artigos escritos pelos usuários melhor qualificados sejam mais completos e coerentes. Sendo assim, torna-se natural também atribuir a estes usuários maior expressividade com relação as suas opiniões manifestadas. Ao comparar este método com outros que não distinguem

¹<http://en.wikipedia.org/wiki/AJAX>

a qualificação do usuário avaliador (EQ2) percebe-se uma variação nas pontuações atribuídas, além de mudanças na ordem de qualificação dos usuários.

6.6 Rankings de Qualificação

O *ranking* de qualificações desenvolvido para o protótipo exibe dados de qualificação de todos os usuários cadastrados, incondicionalmente. O fato de usuários serem vinculados à classes diferentes não modifica nenhum procedimento para geração e ordenação. Da mesma forma, o *ranking* está disponível para todos os usuários.

O *ranking* disponibiliza os dados referentes a qualificação utilizando uma abordagem visual. No lugar de ser exibida em números, a pontuação é apresentada por conjuntos de 5 estrelas, com características semelhantes às encontradas no componente de avaliação que os usuários utilizam para avaliar artigos. 5 estrelas amarelas correspondem a melhor qualificação, enquanto que 5 estrelas cinzas correspondem a pior. A Figura 6.7 exibe um *ranking* de qualificação para 7 usuários fictícios.

Nome do Usuário	Qualificação Atual
usuarioA	★ ★ ★ ★ ★
usuarioB	★ ★ ★ ★ ☆
usuarioC	★ ★ ★ ★ ☆
usuarioD	★ ★ ☆ ☆ ☆
usuarioE	★ ★ ☆ ☆ ☆
usuarioF	★ ☆ ☆ ☆ ☆
usuarioG	☆ ☆ ☆ ☆ ☆

Figura 6.7: Interface do *ranking* de qualificações

A ordenação do *ranking* é decrescente, com base na pontuação de cada usuário. Sua geração é dinâmica, com base nos dados de qualificação disponíveis no momento da requisição, tornando impossível a modificação de qualquer critério para trocar as ordens ou os resultados. Nenhuma informação referente ao histórico anterior do usuário, como informação extraída de currículos ou de mecanismos externos de qualificação, influenciam na geração do *ranking*. Somente as interações e as avaliações dos artigos disponibilizados no protótipo podem alterar sua geração.

6.7 Dashboards

O protótipo implementado neste trabalho apóia a integração entre autores. Por ser aberto, entende-se também por se tratar de um ambiente democrático, que instiga a participação. Estes atributos, ao mesmo tempo que seguem os padrões da Web 2.0, abrem espaço para problemas referentes à credibilidade e confiança.

Em sistemas abertos os usuários geralmente não se conhecem. A interação entre eles é restrita ao conteúdo trocado via Web. Deste modo torna-se difícil estabelecer critérios para gerar confiança. Esta problemática aparece fortemente em sites de comércio com negociações envolvendo somente usuários. O comprador precisa confiar no vendedor

para fechar o negócio. Caso o vendedor não seja idôneo, o comprador corre o risco de não receber suas mercadorias.

Sistemas de reputação são utilizados com o objetivo de gerar confiança entre usuários da Web. Mesmo sem conhecer diretamente o vendedor um comprador pode estabelecer um nível de confiança acompanhando as avaliações de outros usuários que negociaram com este mesmo vendedor anteriormente. Os sites disponibilizam aos negociantes as avaliações que cada vendedor recebe. Geralmente os dados são exibidos na forma numérica, indicando a quantidade de avaliações positivas e negativas que cada usuário recebeu. Estes dados poderiam ser agrupados e representados em forma gráfica, gerando *dashboards*.

Dashboards são representações gráficas que permitem a visualização e compreensão rápida de uma série de informações (FEW, 2006). São amplamente empregados no âmbito empresarial, disponibilizando informações críticas para a tomada de decisão. Diretores podem identificar o desempenho de um determinado setor da empresa sem a necessidade de examinar relatórios, bastando a visualização gráfica das informações.

Este trabalho procurou extrair as vantagens da visualização gráfica proporcionada pelos *dashboards*. Para tanto, foi implementada uma extensão para o MediaWiki que monta gráficos informativos. Estes gráficos exibem informações referentes à qualificação de cada usuário. As informações são disponibilizadas utilizando dois tipos de gráficos: gráfico de barras e gráfico de bala.

O gráfico de barra utilizado no protótipo exibe somente as avaliações que os artigos de cada usuário receberam. Optou-se por utilizá-lo porque é adequado para a comparação de 2 ou mais valores. São comparados 5 valores, exibidos em 5 barras paralelas. Cada barra indica a quantidade de avaliações péssimas, ruins, neutras, boas e ótimas que os artigos receberam. Usuários com artigos bem avaliados terão as duas barras à direita da barra central maiores que as barras à esquerda da barra central. A figura 6.8 B mostra um gráfico de barra exibindo 12 avaliações para os artigos de um usuário do protótipo.

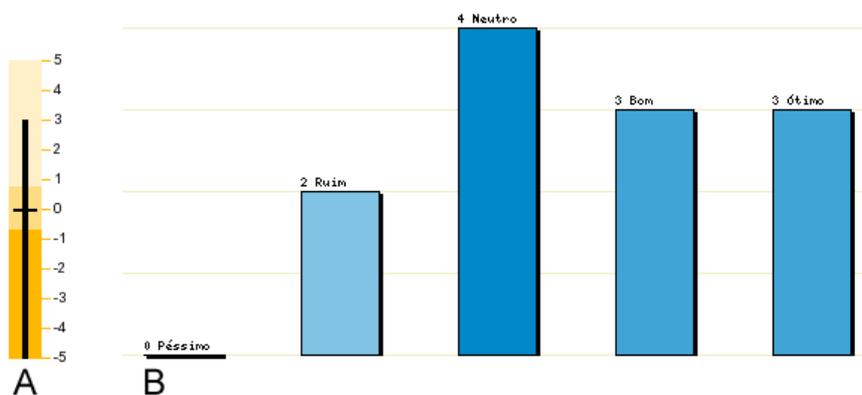


Figura 6.8: Os gráficos A e B que aparecem no *dashboard* de usuários

Gráficos de barra ajudam na comparação entre valores, mas não são apropriados para exibir informações mais completas. O protótipo precisava exibir informações quanto à quantidade de artigos publicados por cada usuário, fazendo um comparativo entre a quantidade média publicada. Estas mesmas comparações precisavam também ser aplicadas em quantidades de acessos a artigos e para valores de qualificação. A solução adotada foi o gráfico de bala.

O gráfico de bala foi criado por Stephen Few e está descrito em Few (2006). Um gráfico de bala pode representar uma série de informações relativas a comparações de

desempenho. O gráfico é composto por uma linha vertical central que indica o desempenho atingido pelo indivíduo avaliado. Uma linha horizontal central indica a média de desempenho de todos os indivíduos. O gráfico aponta ainda uma área sombreada que se estende acima e abaixo da linha central, indicando o desvio padrão dos desempenhos de todos os indivíduos. Este protótipo implementa uma extensão que cria gráficos de bala dinamicamente, com base nos dados calculados para cada usuário.

A figura 6.8 A mostra um gráfico de bala exibido pelo protótipo. Nesta figura o gráfico de bala indica o desempenho de qualificação de um determinado usuário. Nota-se que este usuário possui boa qualificação, já que a linha vertical central (que representa a qualificação do usuário analisado) ultrapassa tanto a linha horizontal central (que indica a média de todas as qualificações) quanto a área situada entre o intervalo -1 e 1. Este intervalo representa o desvio padrão das qualificações de todos os usuários.

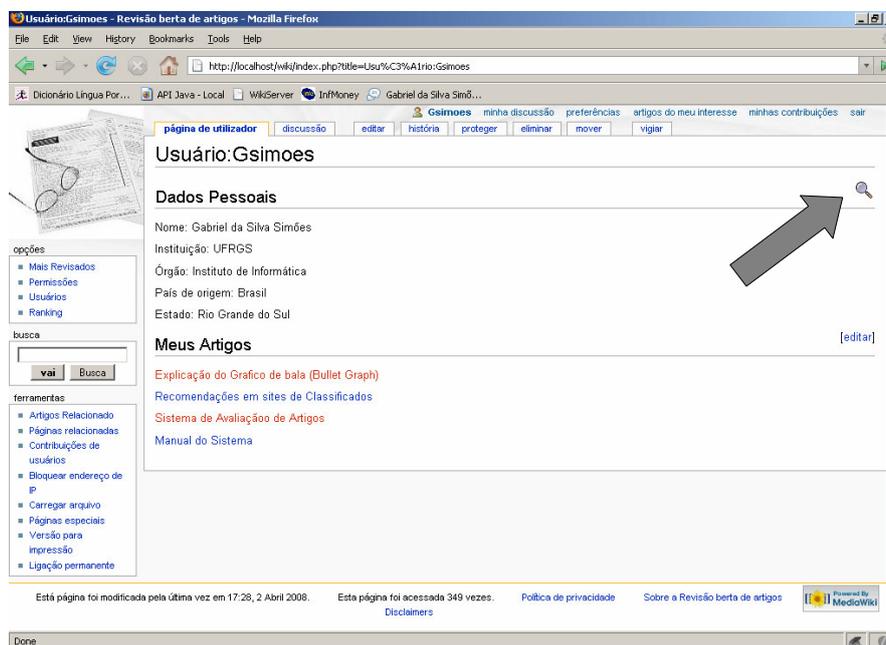


Figura 6.9: A ligação que aponta para os *dashboards* de um usuário

Os *dashboards* com as informações de qualificações e avaliações estão disponíveis para todos os usuários. O objetivo de divulgá-los está exclusivamente ligado à geração de confiança. Não existe a intenção de comparar desempenhos gerando qualquer tipo de competição entre usuários. Para ter acesso à tela que apresenta os *dashboards* é necessário acessar a página de um usuário e clicar na figura de uma lupa, situada na parte superior direita de tela. A seta presente na Figura 6.9 aponta para a lupa.

6.8 Outras Aplicações para o Protótipo

O protótipo desenvolvido para este trabalho foi experimentado em um contexto de avaliação de artigos. O processo de produção e revisão que foi apresentado tem por objetivo facilitar a colaboração no desenvolvimento dos trabalhos, além de oferecer um sistema de revisões abertas. Estas propostas podem ser aplicadas em outros contextos.

Conforme a literatura, existem várias iniciativas que inserem o conceito Wiki em ambientes de aprendizagem. Na forma original, quando utilizados em contextos de aprendizagem, os ambientes wiki geram alguns problemas. Os principais problemas consistem

na precariedade de seus acessos concorrentes, o que pode fazer com que um determinado estudante tenha suas modificações sobrescritas por outro colega. A permissão para que qualquer usuário modifique qualquer artigo também é um problema para contextos de aprendizagem. Wang (2004) e Brereton (2003) levantam estas discussões e propõem um ambiente Wiki estendido, que possa tratar cada uma destas problemáticas.

O protótipo implementado por este trabalho define políticas que facilitam sua adequação como um ambiente de aprendizagem. Configurando corretamente às classes de usuários é possível definir políticas que permitam acessos e edições somente para usuários específicos. Somado a isto, o MediaWiki implementa bloqueios que garantem a consistência dos dados para os casos de acessos concorrentes. É possível utilizar o protótipo para acompanhar os alunos de uma turma durante um curso. Com base nas avaliações que os próprios alunos fazem entre seus trabalhos, o ambiente fornecerá um ranking de qualificação que pode ser considerado para a atribuição dos conceitos finais de cada aluno.

7 EXPERIMENTOS

Para validar o método EQ1 e o protótipo foram realizados dois experimentos. Diferentes pessoas foram convidadas para participar de cada experimento. Os convidados responderam por usuários fictícios previamente cadastrados no protótipo. Notícias diversas, extraídas da Web, foram inseridas na forma de artigos vinculadas a cada usuário fictício. Foram utilizados artigos referentes a Esportes e artigos referentes a Linguagens de Programação separadamente em cada experimento. Os artigos referentes a Esportes foram extraídos de veículos de comunicação que publicam seu conteúdo na Internet (Globo Esporte¹ e UOL Esporte²). Os artigos referentes a Linguagens de Programação foram extraídos da Wikipédia em Português. Os usuários foram distribuídos dentre as categorias disponíveis no protótipo (autores, revisores e comentaristas). Os conteúdos foram previamente inseridos para agilizar os experimentos.

O experimento 1 contou com a participação de 10 usuários. O experimento 2 contou com a participação de 5 usuários. Em ambos os experimentos os usuários estavam previamente criados, identificados por letras do alfabeto. Nenhum vínculo entre o usuário e a pessoa que respondeu por ele foi estabelecido. Os artigos foram também previamente vinculados aos usuários.

A estratégia de selecionar e inserir artigos previamente surgiu devido a várias dificuldades encontradas na realização de experimentos anteriores. Por se tratar de um software que requer esforço do usuário (o usuário deve inserir e formatar seu conteúdo), tornou-se difícil contar com a participação espontânea de voluntários dispostos a colaborar. Com os artigos previamente inseridos, os convidados para o experimento precisaram somente acessar e avaliar o conteúdo. Esta estratégia teve de ser utilizada para viabilizar os experimentos. Nenhum convidado concordou em desenvolver conteúdo para os experimentos.

7.1 Experimento 1 - Utilizando Notícias de Times de Futebol

Neste experimento, foram inseridas 3 notícias referentes a times de futebol para cada um dos 10 usuários, gerando um total de 30 artigos. O tema Futebol foi escolhido com o objetivo de aproveitar o impacto emocional que gera sobre os usuários participantes. A distribuição dos artigos concentrou times locais em usuários específicos.

Um usuário recebeu notícias de 3 times locais. Foi selecionado um único time para cada um dos 9 usuários restantes, do qual foram inseridas também 3 notícias. Sendo assim, 3 usuários (A, B e C) publicaram notícias de times específicos da região (T1, T2 e T3), 1 usuário (D) publicou notícias referentes aos 3 times da região e os outros 6 usuários

¹<http://globoesporte.globo.com/>

²<http://esporte.uol.com.br/>

(E, F, G, H, I e J) publicaram notícias de diversos times de outras regiões.

O objetivo do experimento 1 consiste em validar o método EQ1. Buscou-se identificar variações nas qualificações com base no consenso dos usuários melhor posicionados no ranking de qualificações. Dentre os possíveis resultados, destaca-se a hipótese onde os usuários que publicaram artigos relacionados com os times regionais obtenham as melhores qualificações. Quando utilizando EQ1, um dos times sofre rejeição, resultado da preferência da maior parte dos participantes do experimento por um time em específico. Foram comparados os dados obtidos ao realizar os experimentos utilizando EQ1, método que considera a qualificação dos avaliadores, e EQ2, método que atribui pontuações estáticas.

7.1.1 O Experimento

Com o ambiente previamente populado com usuários fictícios e com notícias referentes aos times de futebol, foram convidadas 10 pessoas para responder pelos usuários, avaliando cada artigo publicado. Cada uma destas 10 pessoas acessou o ambiente, lendo e avaliando as notícias vinculadas aos outros 9 usuários cadastrados. Estas pessoas não possuíam qualquer responsabilidade sobre os conteúdos, já que não participaram da escolha ou da distribuição dos artigos.

Este experimento determina a qualificação dos usuários fictícios através da relevância das notícias apresentadas. Não foi objetivo deste experimento avaliar as capacidades de produção das pessoas que responderam pelos usuários fictícios. Foram utilizados dois métodos para geração dos pontos de qualificação: EQ1 e EQ2.

Os participantes que responderam pelos usuários fictícios foram convidados aleatoriamente, sendo todos estudantes do PPGC/II. Não foi mantido qualquer vínculo entre o participante do experimento e o usuário cadastrado no sistema. Por questões regionais, a maior parte dos usuários participantes eram simpatizantes dos times T1, T2 e T3, sendo observada uma forte preferência por T1 em detrimento de T2. Neste contexto regional, o time T3 também apresenta forte representação com relação à preferência.

Os 10 participantes iniciaram as avaliações em conjunto. Foi solicitado que todos avaliassem todas as notícias, exceto às 3 vinculadas a seu próprio usuário. O primeiro usuário a terminar necessitou de 8 minutos para realizar os acessos e as avaliações dos 27 artigos. O participante que necessitou de maior tempo terminou suas 27 avaliações em 20 minutos. Todos os usuários acessaram e avaliaram todos os artigos, conforme solicitado. Os critérios de avaliação utilizados foram pessoais, não sendo escopo deste trabalho discutí-los.

7.1.2 Resultados

Neste primeiro experimento foi possível avaliar o funcionamento do protótipo. Os 10 usuários não relataram qualquer dificuldade relevante na utilização da interface do mesmo. Isto se comprova no momento em que foram necessários 20 minutos para que todos terminassem suas avaliações. Por outro lado, a diferença de 12 minutos que separa o primeiro do último usuário a concluir o experimento indicam que alguns são mais “meticulosos” em suas avaliações. Avaliar 27 artigos em 12 minutos significa que foi designado, em média, menos de 30 segundos para a apreciação de cada artigo. Desta forma, entende-se que a opinião do usuário define-se basicamente no momento em que ele conhece o título do conteúdo avaliado.

Devido ao fato do experimento ser baseado em notícias de Futebol, contexto no qual alguns times da região estão inclusos, esperava-se que os resultados fossem influenciados

Tabela 7.1: Avaliações recebidas pelos usuários A e B

Usuário	Péssimo	Ruim	Neutro	Bom	Ótimo
A	1	2	6	9	9
B	10	2	8	5	2

para suas direções. Dentre os 10 times, 2 deles, T1 e T2, geram forte impacto sobre os participantes do experimento. Durante o experimento pôde-se observar que grande parte dos participantes demonstravam uma preferência em favor de T1. Além disso T1 e T2 são fortes rivais, gerando rejeição entre ambos. Os dados listados na Tabela 7.1, que exhibe o número de avaliações péssimas, ruins, neutras, boas e ótimas que os artigos vinculados aos usuários A e B receberam, confirmam a preferência em favor de T1 e a rejeição que gera sobre T2. Os dados apontam para um aparente equilíbrio, onde A ocupa a posição positiva e B a negativa.

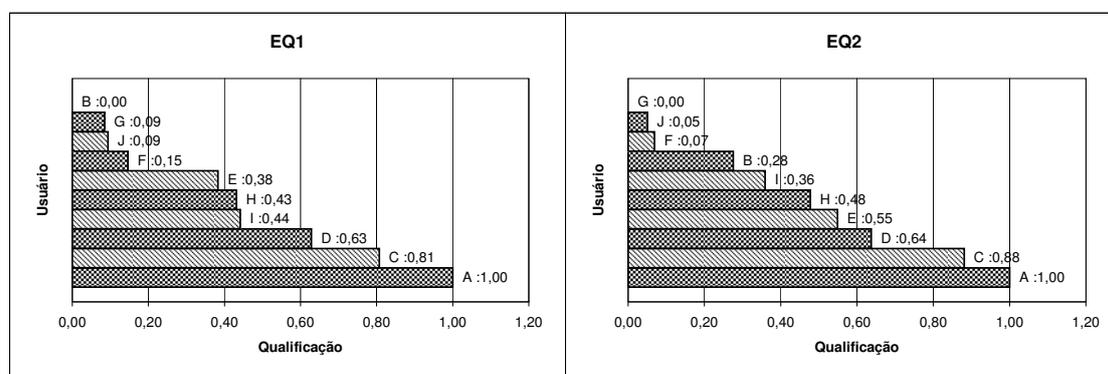


Figura 7.1: Qualificações após Experimento 1 (Todos os 10 usuários)

Aplicando EQ1 em um primeiro cenário, onde todos os usuários foram considerados, o usuário A obteve a maior qualificação seguido do usuário C. Como esperado, devido a forte rejeição entre T1 e T2, B obteve a menor qualificação entre os 10 usuários. Este resultado confirma o fato de que a maior parte dos envolvidos no experimento tendem para o time T1. Quando utilizado EQ2 a rejeição cai expressivamente, elevando a qualificação do usuário B.

Comparando os gráficos da figura 7.1, que mostram as qualificações normalizadas dos usuários, percebe-se que acontece uma modificação no *ranking* de qualificação. Utilizando EQ1, o usuário B aparece na 10^a posição. Utilizando EQ2, o usuário B ocupa a 7^a posição. As 3 primeiras posições no *ranking*, utilizando os dois métodos, permanecem inalteradas. Este resultado comprova a eficácia de EQ1. EQ1 indica pontuações considerando a qualificação dos avaliadores, portanto, privilegia o consenso e as avaliações realizadas por usuários melhor qualificados.

A tendência natural era de uma boa qualificação para o usuário C, já que o time T3 possui baixa rejeição tanto por parte dos simpatizantes dos times T1 quanto T2, além de contar com simpatia por parte da maior parte dos envolvidos no experimento. A simpatia e a baixa rejeição por T3 são confirmadas pelo fato de C manter a segunda melhor qualificação utilizando tanto EQ1 quanto EQ2.

Em um segundo cenário o *ranking* de qualificação foi gerado suprimindo o usuário A, que publicou notícias de T1. Desta forma, exclui-se a maior fonte de rejeição que T2 en-

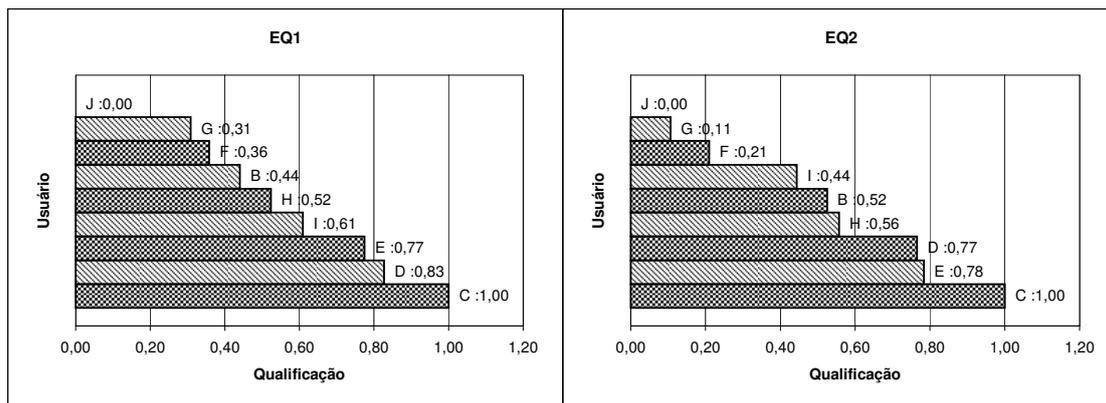


Figura 7.2: Qualificações após Experimento 1 (Excluindo o usuário A)

controu no primeiro cenário. A hipótese mais provável seria uma melhora na qualificação do usuário B.

Analisando os dados exibidos pelos gráficos da figura 7.2, que comparam as qualificações normalizadas dos usuários, percebe-se que a hipótese esperada é confirmada. Utilizando EQ1, a qualificação normalizada do usuário B sobe de 0 para 0,44. Esta melhora faz com que B suba 3 posições no *ranking*. Utilizando EQ2, a qualificação de B também melhora. No primeiro cenário, B aparecia com 0,28, enquanto que no segundo, B obtém 0,52 pontos normalizados de qualificação. Esta melhora faz com que B suba uma posição no *ranking*.

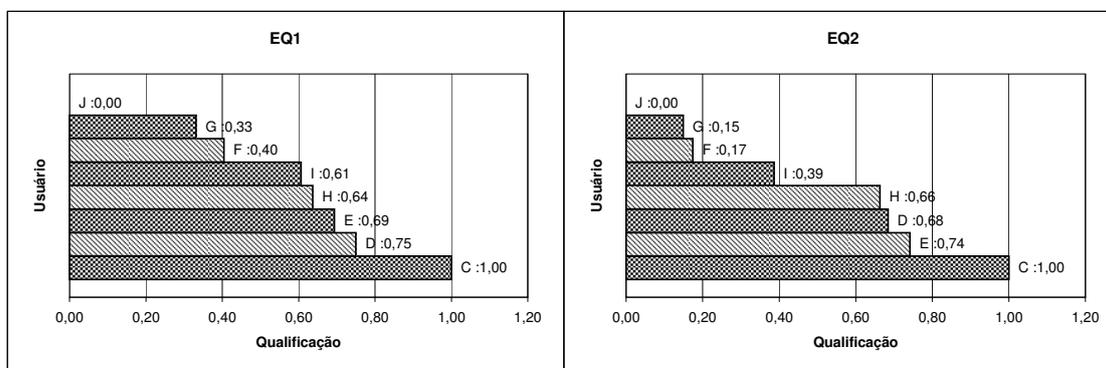


Figura 7.3: Qualificações após Experimento 1 (Excluindo os usuários A e B)

O terceiro cenário exibido pelos gráficos da Figura 7.3, que comparam as qualificações normalizadas dos usuários utilizando EQ1 e EQ2, gera o *ranking* suprimindo os usuários A e B. Naturalmente o usuário C, que aparece com a 2ª posição no cenário 1 e com a 1ª no cenário 2, mantém-se ocupando a 1ª posição. O restante dos usuários não sofrem modificações determinantes em suas qualificações.

Todos os 3 cenários comparam qualificações normalizadas em valores no intervalo 0 - 1. O valor 1 indica a melhor qualificação enquanto 0 indica a pior. Para fins de comparação direta de pontuações, pode-se observar a Tabela 7.2 que lista as qualificações não normalizadas. As qualificações listadas exibem valores correspondentes a EQ1 e EQ2. São listadas as qualificações obtidas por todos os usuários.

O Apêndice B apresenta uma série de tabelas comparativas. Estas tabelas mostram usuários e suas pontuações, indicados em valores absolutos, normalizados e em porcenta-

Tabela 7.2: Qualificações EQ1 e EQ2 no experimento 1

Usuário	EQ1	EQ2
A	24.56	47.22
B	-13.62	-6.33
C	16.99	38.08
D	10.01	19.81
E	0.38	13.17
F	-7.90	-21.84
G	-10.29	-27.01
H	2.28	7.80
I	2.68	-0.09
J	-9.95	-23.23

gem para todos os 3 cenários. As tabelas são ordenadas pela maior pontuação e fazem um comparativo dos *rankings* de qualificação para o Experimento 1 utilizando EQ1 e EQ2.

7.2 Experimento 2 - Utilizando Artigos da Wikipedia

O segundo experimento realizado utilizou artigos da Wikipedia em língua portuguesa. Diferente do anterior, este experimento não abordou assuntos de forte impacto pessoal. Este experimento teve como objetivo identificar motivos que levam os usuários a avaliar bem ou mal um trabalho.

O objetivo deste experimento está relacionado a apresentação do conteúdo. Buscou-se identificar com o experimento 2 as tendências demonstradas pelos usuários ao avaliarem artigos completos e artigos resumidos. Os métodos EQ1 e EQ2 também foram utilizados, mesmo não sendo o foco deste experimentos compará-los.

7.2.1 O Experimento

Neste experimento, foram criados 5 usuários fictícios. Foram associados 3 artigos extraídos da Wikipedia em língua portuguesa a cada usuário. Os artigos foram previamente inseridos para facilitar a realização do experimento. Todos os artigos abordavam linguagens de programação diversas, sendo que cada trabalho falava especificamente de uma única linguagem.

Análogo ao experimento anterior, 5 membros do Grupo de Pesquisa foram convidados a participar, respondendo pelos usuários fictícios. A seleção dos usuários foi feita aleatoriamente. Cada usuário acessou e avaliou os 3 trabalhos de outros 4 usuários, totalizando 12 avaliações. O experimento precisou de pouco mais de 10 minutos para ser realizado.

7.2.2 Resultados

Por ser menor que o Experimento 1 (já que contou com a metade do número de usuários) este experimento precisou de menos tempo para ser realizado. O experimento inicial necessitou de 20 minutos para ser concluído, ou seja, duas vezes o tempo necessário para a realização deste segundo. Este dado revela a relação existente entre a quantidade de avaliações e o tempo necessário para completá-las.

A Tabela 7.3 lista a relação de artigos disponibilizados no protótipo durante a reali-

zação do Experimento 2. A Tabela ordena os registros com base no número de palavras, com orientação decrescente. Além do título, pode-se observar o número de avaliações péssimas, ruins, neutras, boas e ótimas recebidas por cada artigo.

Tabela 7.3: Dados de cada trabalho

Título	Péssimo	Ruim	Neutro	Bom	Ótimo	Número de Palavras
PROLOG	0	0	1	0	3	4341
C	0	1	0	0	3	4003
Java	0	0	0	1	3	2668
PHP	0	0	0	1	3	2440
COBOL	0	1	0	3	0	1850
Delphi	0	1	0	3	0	1553
JavaScript	0	0	1	1	2	953
Pascal	0	1	0	0	3	394
Visual Basic	0	2	2	0	0	394
ADA	0	1	3	0	0	339
JDBC	0	1	2	1	0	302
TCL	0	0	2	2	0	292
ASP	1	2	1	0	0	261
XML Schema	2	1	1	0	0	123
ADO	3	1	0	0	0	37

Com base nos dados apresentados, percebe-se uma relação entre as avaliações informadas e o número de palavras que compõe cada artigo. Observa-se que os artigos compostos por mais de 400 palavras obtiveram as melhores avaliações. Com exceção do artigo intitulado “Pascal”, somente artigos com mais de 2000 palavras obtiveram 3 avaliações ótimas. Artigos com menos de 400 palavras obtiveram piores avaliações. Somente artigos com menos de 300 palavras obtiveram alguma avaliação péssima.

Os dados mostram que ao avaliarem os artigos, a maior parte dos usuários não analisou o conteúdo. A simples percepção visual de tamanho e formatação induziram as avaliações. Um experimento onde os usuários considerassem o conteúdo precisaria contar com o comprometimento dos participantes.

8 CONCLUSÕES E TRABALHOS FUTUROS

O processo tradicional utilizado na seleção de trabalhos para publicação, conhecido como *peer review*, apresenta uma série de atributos. O anonimato torna a seleção impessoal, preservando os trabalhos. O fato de existirem em média 3 revisores ajuda a manter a imparcialidade do processo. Geralmente os revisores são pessoas qualificadas, com experiência em publicações nas áreas alvo do evento em questão. Principalmente nos casos onde artigos são rejeitados, o *feedback* gerado pelos comentários dos revisores contribui para que autores possam melhorar seus trabalhos, possibilitando aceitações futuras. Por outro lado, este sistema fechado está sujeito a distorções que além de desconstituir todos os seus atributos podem também acabar com a divulgação de idéias importantes e inovadoras.

Um processo aberto, que traga transparência para a publicação, pode ser uma iniciativa para contornar os problemas apresentados pela abordagem tradicional. Obviamente este novo processo precisa passar por um período de experimentações e adaptações. Por outro lado, o amadurecimento do processo não é suficiente para que este integre o cotidiano dos eventos científicos. Além disso, é preciso haver uma evolução cultural onde as pessoas passem a depositar confiança em artigos revisados pela comunidade como um todo, não por uma seleção de membros dela. Acima de tudo, para que esta abordagem possa realmente funcionar, é necessário pensar em ciência como uma atividade cooperativa, aberta a idéias e contribuições.

O processo apresentado e implementado por este trabalho é uma alternativa que pode substituir o modelo tradicional. Ele fornece instrumentos para edição e revisão de artigos e para a geração de confiança entre os envolvidos. O método EQ1, que indica pontos de qualificação de usuários com base na qualificação dos avaliadores, fornece informações orientadas pelo consenso dos usuários melhor qualificados. O *ranking* de qualificação indica a qualificação de cada usuário, considerando somente os dados gerados automaticamente pela aplicação de EQ1. Analisando estas informações é possível identificar autores e artigos relevantes, indicados pelo consenso da comunidade.

Sobre o método para indicação de pontos de qualificação EQ1, pode-se afirmar que seus resultados são relevantes. Os dados retornados pelo Experimento 1 indicam que as qualificações geradas por EQ1 tendem ao consenso dos usuários com melhores qualificações. Foram identificadas mudanças significativas na ordem do *ranking* de qualificações quando os resultados obtidos por EQ1 foram comparados aos resultados obtidos por EQ2, que não considera a qualificação dos avaliadores.

A escolha do MediaWiki como software base para a implementação do protótipo foi apropriada. Por dispor de mecanismos para extensão, foi possível adaptá-lo para funcionar conforme o processo apresentado neste trabalho. A adoção do MediaWiki, além de incorporar as funcionalidades básicas de um ambiente Wiki, evitou a criação de um ciclo

completo de desenvolvimento de software. O desenvolvimento de um novo software consumiria tempo excessivo. Estender o MediaWiki criou uma oportunidade para estudá-lo e compreendê-lo, gerando material relevante para os próximos projetos que possam vir a utilizá-lo como base. O MediaWiki também proporcionou uma experiência de utilização por parte dos usuários bastante positiva. Por ser o software empregado na Wikipedia, o MediaWiki já é conhecido pela maior parte dos usuários da Internet, o que facilita sua utilização.

8.0.3 Publicações Resultantes

A realização desta pesquisa deixa como contribuições o método EQ1, o *ranking* de qualificação de usuários e o protótipo que implementa o processo aberto para editoração de artigos. Até o momento, a partir destas contribuições, foram geradas duas publicações.

A primeira publicação trata-se de um pôster publicado e apresentado na I Sessão de Pôsteres do XXII Simpósio Brasileiro de Banco de Dados, intitulado “Arquitetura de um Ambiente Aberto de Edição e Revisão de Documentos Científicos” (SIMÕES; OLIVEIRA; WIVES, 2007a). Este trabalho abordou como tema principal o processo de produção aberto aplicado no protótipo implementado por esta dissertação.

A segunda publicação trata-se de um *paper* publicado e apresentado no *III Workshop on Digital Libraries* que aconteceu em conjunto com o XIII Webmedia, intitulado “Um processo aberto para edição e revisão de artigos implementado sobre o MediaWiki” (SIMÕES; OLIVEIRA; WIVES, 2007b). Este trabalho focou no método EQ1 aplicado ao contexto de qualificação de autores em um ambiente aberto para editoração de artigos.

8.1 Trabalhos Futuros

A seleção dos revisores, usuários que podem modificar diretamente qualquer trabalho, mesmo não sendo de sua autoria, ainda é estática. Os revisores são escolhidos aleatoriamente, ou com base em suas áreas de competência, e são vinculados à classe `Revisor` manualmente. Pode-se desenvolver critérios para seleção automática dos revisores. Poderiam ser definidos como revisores todos aqueles usuários que estiverem no topo do ranking de qualificação, bastando vincular cada um deles à classe `Revisor`. Aqueles usuários que deixarem de estar no topo do ranking de qualificação perderiam o vínculo com à classe `Revisor`.

O protótipo passou por uma série de testes, verificando a precisão dos resultados gerados pelos algoritmos e questões de usabilidade. Por outro lado, foram realizados somente dois experimentos envolvendo um número expressivo de usuários. Estes experimentos puderam validar as hipóteses esperadas, mas ainda assim não são suficientes para propor ajustes ou modificações. Experimentos mais completos precisam ser elaborados para extrair melhores conclusões sobre esta proposta. Uma idéia seria disponibilizar o protótipo na forma de ambiente de aprendizado, trabalhando com uma turma de alunos de um determinado curso ou disciplina. Deste modo os alunos poderiam utilizar o ambiente para publicar seus trabalhos e, com base nisto, serem avaliados pelos próprios integrantes da turma. Outra possibilidade de experimento que deverá ser executada é a aplicação do protótipo como sistema de submissão para trabalhos científicos em eventos. Esta proposta pode ser executada trabalhando em parceria com o II ou outra instituição que administre um evento do gênero.

Uma nova Dissertação de Mestrado, que tomará por base o método de indicação de pontos de qualificação e o protótipo descrito por este trabalho, está sendo iniciada. Estão

planejadas, para este novo trabalho, aplicações do protótipo na Semana Acadêmica e em disciplinas do PPGC/II. Estas aplicações terão potencial para gerar novos dados experimentais, permitindo uma avaliação mais detalhada do método e indicações mais precisas das características a serem trabalhadas. Está prevista, também, a integração desta nova Dissertação de Mestrado aos trabalhos de grupos vinculados a instituições internacionais, parceiros em projetos de pesquisa

REFERÊNCIAS

- BARBER, B. Resistance by Scientists to Scientific Discovery: this source of resistance has yet to be given the scrutiny accorded religious and ideological sources. **Science**, [S.l.], v.134, n.3479, p.596–602, 1961.
- BENOS, D. J. et al. The ups and downs of peer review. **Advan. Physiol. Edu.**, [S.l.], v.31, n.2, p.145–152, 2007.
- BRERETON, M.; DONOVAN, J.; VILLER, S. Talking about watching: using the video card game and wiki-web technology to engage it students in developing observational skills. In: AUSTRALASIAN CONFERENCE ON COMPUTING EDUCATION, ACE, 5., 2003, Darlinghurst, Australia. **Proceedings...** [S.l.]:Australian Computer Society, 2003. p.197–205.
- CAMPANARIO, J. M. Have referees rejected some of the most-cited articles of all times? **J. Am. Soc. Inf. Sci.**, New York, NY, USA, v.47, n.4, p.302–310, 1996.
- CAPES. **Sistema Qualis-CAPES**. Disponível em: <<http://qualis.capes.gov.br/webqualis/>>. Acesso em: set. 2007.
- CODD, E. F. A relational model of data for large shared data banks. **Communications of the ACM**, New York, NY, USA, v.26, n.1, p.64–69, 1983.
- CONLEY, J. P. Rejected: leading economists ponder the publication process. **International Review of Economics and Finance**, Greenwich, v.6, n.4, p.441–442.
- CROWSTON, K. Reproduced and emergent genres of communication on the World-Wide Web. In: HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, HICC, 30., 1997. **Proceedings...** [S.l.: s.n.], 1997. v.6, p.30–39.
- CURRAN, K.; DOHERTY, K.; POWER, R. WikiWikiWeb as a Tool for Collaboration. **Information Technology Journal**, [S.l.], v.3, n.2, p.206–210, 2004.
- CV-LATTES. Plataforma Lattes - Conselho Nacional de Desenvolvimento Científico e Tecnológico. Disponível em: <<http://lattes.cnpq.br/>>. Acesso em: mar. 2008.
- DIJKSTRA, E. W. Go To statement considered harmful. **Communications of the ACM**, New York, NY, USA, v.11, n.3, p.147–148, 1968.
- FERREIRA, A. B. H. **Novo dicionário da língua portuguesa**. 2.ed. [S.l.]: Nova Fronteira, 1986.

FEW, S. **Information Dashboard Design** : the effective visual communication of data. [S.l.]: O'Reilly Media, 2006.

FREY, B. S. Publishing as Prostitution? Choosing Between One's Own Ideas and Academic Success. **Public Choice**, [S.l.], v.116, n.1/2, p.205–223, 2003.

FRIEDL, J. E. F. **Mastering regular expressions**. Sebastopol, CA, USA: O'Reilly & Associates, 1997.

GILES, J. Internet encyclopaedias go head to head. **Nature**, [S.l.], v.438, p.900–901, Dec. 2005.

GLOVER, A.; HIRST, G. Detecting stylistic inconsistencies in collaborative writing. In: **WRITERS at work: professional writing in the computerized environment**. [S.l.]: Springer-Verlag, 1996. 147-168p.

GUTERMAN, L. Scientists Accuse Toxicology Journal of Industry Ties. **The Chronicle of Higher Education**, [S.l.], v.6, 2002.

HALAVAIS, A.; LACKAFF, D. An Analysis of Topical Coverage of Wikipedia. **Journal of Computer-Mediated Communication**, [S.l.], v.13, p.429–440, Jan. 2008.

HARRISON, C. Peer review, politics and pluralism. **Environmental Science & Policy**, Amsterdam, v.7, n.5, p.357–368, 2004.

HIRSCH, J. E. An index to quantify an individual's scientific research output. **Proceedings of the National Academy of Sciences**, [S.l.], v.102, n.46, p.16569–16572, 2005.

HU, N.; PAVLOU, P. A.; ZHANG, J. Can online reviews reveal a product's true quality?: empirical findings and analytical modeling of online word-of-mouth communication. In: **ACM CONFERENCE ON ELECTRONIC COMMERCE, EC, 7.**, 2006. **Proceedings...** New York: ACM, 2006. p.324–330.

JARVENPAA, S. L.; TRACTINSKY, N.; VITALE, M. Consumer trust in an Internet store. **Inf. Technol. and Management**, Hingham, MA, USA, v.1, n.1-2, p.45–71, 2000.

LABIO, W.; GARCIA-MOLINA, H. Efficient Snapshot Differential Algorithms for Data Warehousing. In: **INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES, VLDB, 22.**, 1996. **Proceedings...** [S.l.: s.n.], 1996. p.63–74.

LEUF, B.; CUNNINGHAM, W. **The Wiki Way**: collaboration and sharing on the internet. [S.l.]: Addison-Wesley Professional, 2001.

MAJCHRZAK, A.; WAGNER, C.; YATES, D. Corporate wiki users: results of a survey. In: **INTERNATIONAL SYMPOSIUM ON WIKIS, WIKISYM, 2006**. **Proceedings...** New York: ACM Press, 2006. p.99–104.

MILLARD, D. E.; ROSS, M. Web 2.0: hypertext by any other name? In: **CONFERENCE ON HYPERTEXT AND HYPERMEDIA, HYPERTEXT, 2006**. **Proceedings...** New York: ACM Press, 2006. p.27–30.

OLIVEIRA, J. P.; GALANTE, R.; MUSA, D.; EDELWEISS, N. Uma Proposta para Editoração, Indexação e Busca de Documentos Científicos em um Processo de Avaliação Aberta. In: WORKSHOP EM BIBLIOTECAS DIGITAIS, WDL, 2005. **Anais...** [S.l.: s.n.], 2005. v.1, p.30–39.

ONLINE JIME. Disponível em: <<http://www-jime.open.ac.uk>>. Acesso em: 30 jan. 2008.

PARNAS, D. L. Stop the numbers game. **Communications of the ACM**, New York, NY, USA, v.50, n.11, p.19–21, 2007.

POSCHL, U. Interactive journal concept for improved scientific publishing and quality assurance. **Learned Publishing**, [S.l.], v.17, p.105–113, 2004.

PRIEDHORSKY, R. et al. Creating, destroying, and restoring value in wikipedia. In: INTERNATIONAL ACM CONFERENCE ON SUPPORTING GROUP WORK, 2007. **Proceedings...** New York: ACM, 2007. p.259–268.

RECH, R. O. **Um Modelo de Pontuação na Busca de Competências Acadêmicas de Pesquisadores**. 2007. Dissertação (Mestrado em Ciência da Computação) — Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre.

REN, J.; TAYLOR, R. N. Automatic and versatile publications ranking for research institutions and scholars. **Communications of the ACM**, New York, NY, USA, v.50, n.6, p.81–85, 2007.

RESNICK, P. et al. Reputation systems. **Communications of the ACM**, New York, NY, USA, v.43, n.12, p.45–48, 2000.

SANTINI, S. We Are Sorry to Inform You ... **Computer**, Los Alamitos, CA, USA, v.38, n.12, p.126–127, 2005.

SAUER, C.; SMITH, C.; BENZ, T. WikiCreole:: a common wiki markup. In: INTERNATIONAL SYMPOSIUM ON WIKIS, WIKISYM, 2007. **Proceedings...** New York: ACM, 2007. p.131–142.

SIMÕES, G.; OLIVEIRA, J. P. M. de; WIVES, L. K. Arquitetura de um Ambiente Aberto de Edição e Revisão de Documentos Científicos. In: SIMPÓSIO BRASILEIRO DE BANCO DE DADOS, 22., 2007. **Sessão de Pôsteres: anais**. João Pessoa: SBC, 2007. p.35–38.

SIMÕES, G.; OLIVEIRA, J. P. M. de; WIVES, L. K. Um processo aberto para edição e revisão de artigos implementado sobre o MediaWiki. In: WORKSHOP EM BIBLIOTECAS DIGITAIS, WDL, 3., SBSMW, 13., 2007, Gramado. **Short Papers and Workshops**. Porto Alegre: SBC, 2007.

SPROULL, L. S.; KIESLER, S. **Connections: new ways of working in the networked organization**. Cambridge, MA, USA: MIT Press, 1992.

WANG, C.; TURNER, D. Extending the wiki paradigm for use in the classroom. In: INTERNATIONAL CONFERENCE ON INFORMATION TECHNOLOGY CODING AND COMPUTING, ITCC, 2004. **Proceedings...** [S.l.]: IEEE, 2004. p.255–259.

WYER, M. **Women, Science and Technology**: a reader in feminist science studies. USA: Routledge, 2000.

ZAMBON, G. **Beginning Jsp, Jsf and Tomcat Web Development**: from novice to professional. Berkeley: APress, 2007.

APÊNDICE A ARQUITETURA DA BASE DE DADOS DO MEDIAWIKI

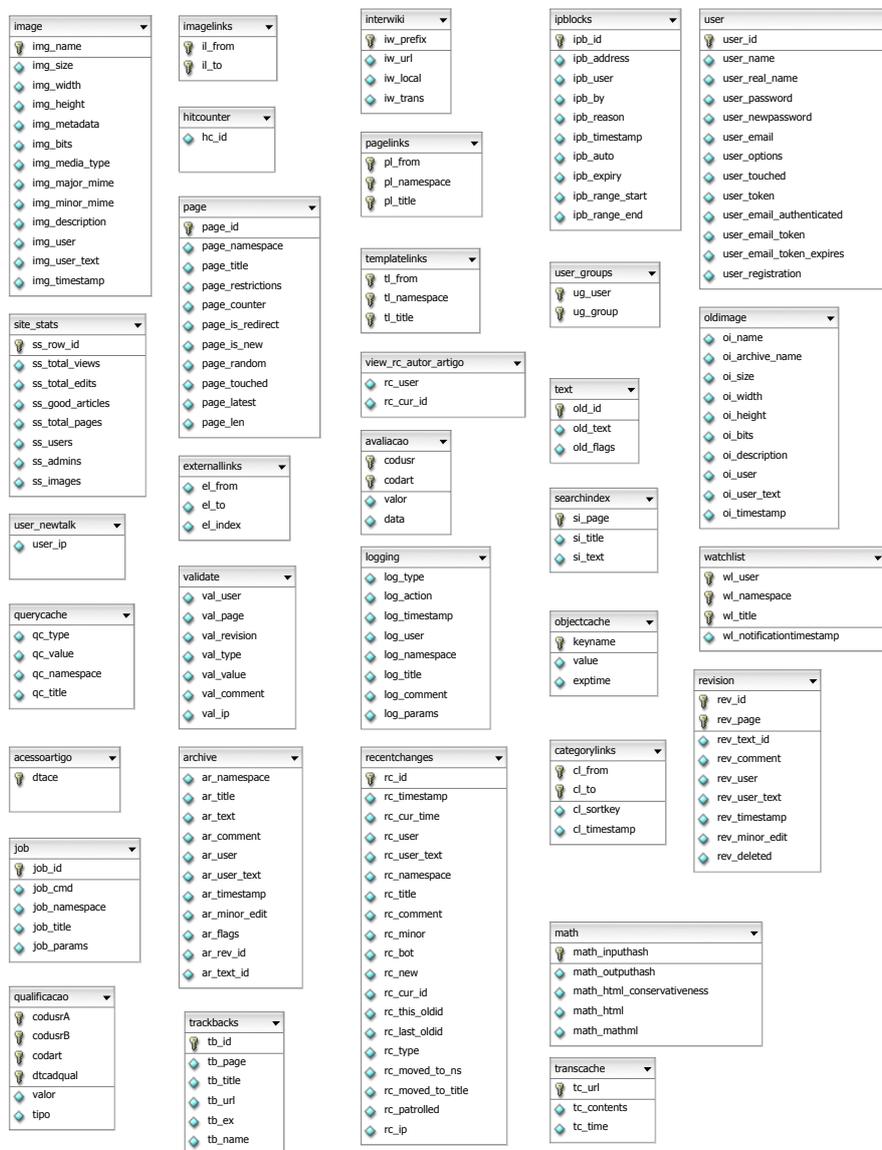


Figura A.1: Base da Dados do MediaWiki

A Figura A.1 ilustra a modelagem final da base de dados do protótipo. Dentro das entidades exibidas, somente 4 foram adicionadas ao modelo original do MediaWiki. Todas

as outras entidades foram mantidas inalteradas. As entidades adicionadas são:

- `acessoArtigo`
Entidade criada para armazenar dados referentes aos acessos realizados pelos usuários aos artigos disponíveis. Após cada acesso `acessoArtigo` recebe o identificador do usuário que realiza o acesso, o identificador do artigo acessado e a data em que acontece o acesso. Estes dados precisam ser mantidos para que o *ranking* de qualificações seja recalculado, comparando os resultados gerados por algoritmos diferentes.
- `qualificacao`
Armazena as pontuações geradas por cada evento que define pontos de qualificação. Os eventos podem ser acessos ou avaliações. A entidade armazena o identificador do usuário que recebe os pontos, o identificador do usuário que gerou os pontos, o identificador do artigo onde aconteceu o evento e a data de ocorrência do evento. O atributo `Valor` armazena os pontos gerados e o atributo `Tipo` armazena o tipo de evento que gerou os pontos.
- `avaliacao`
Nesta entidade ficam armazenados os dados referentes às avaliações feitas pelos usuários. A entidade registra o identificador do usuário responsável pela avaliação e o identificador do artigo que recebe a avaliação. Armazena também o valor numérico correspondente à relevância do artigo e a data em que ocorreu a avaliação.
- `view_rc_autor_artigo`
Esta visão efetua uma seleção na entidade `recentchanges`, retornando o identificador de um artigo e o identificador do primeiro usuário a editá-lo. Com base nos dados retornados por esta visão, é possível identificar o autor de um artigo. Conforme a proposta deste trabalho, um artigo só pode ser editado pelo primeiro usuário que modificou seu conteúdo (autor) ou por revisores.

APÊNDICE B COMPARANDO QUALIFICAÇÕES GERADAS NO EXPERIMENTO 1

Tabela B.1: Comparação entre as qualificações de todos os usuários utilizando EQ1 e EQ2

EQ1 Todos os 10 usuários				EQ2 Todos os 10 usuários			
Usuário	Pontos	Normal	%	Usuário	Pontos	Normal	%
A	24,56	1	100,00%	A	47,22	1	100,00%
C	16,99	0,8068	80,68%	C	38,08	0,8811	88,11%
D	10,01	0,6286	62,86%	D	19,81	0,6375	63,75%
I	2,68	0,4416	44,16%	E	13,17	0,549	54,90%
H	2,28	0,4313	43,13%	H	7,80	0,4774	47,74%
E	0,38	0,3828	38,28%	I	-0,09	0,3589	35,89%
F	-7,90	0,146	14,60%	B	-6,33	0,2757	27,57%
J	-9,95	0,0937	9,37%	F	-21,84	0,0689	6,89%
G	-10,29	0,085	8,50%	J	-23,23	0,0504	5,04%
B	-13,62	0	0,00%	G	-27,01	0	0,00%

MAX EQ1	MAX EQ2	Dif.	%
24,56	47,22	22,66	47,99%

Tabela B.2: Comparação entre as qualificações excluindo usuário A utilizando EQ1 e EQ2

EQ1 excluindo usuário A				EQ2 excluindo usuário A			
Usuário	Pontos	Normal	%	Usuário	Pontos	Normal	%
C	11,65	1	100,00%	C	30,53	1	100,00%
D	7,00	0,8269	82,69%	E	17,74	0,7839	78,39%
E	5,59	0,7744	77,44%	D	16,63	0,7652	76,52%
I	1,16	0,6095	60,95%	H	4,29	0,5567	55,67%
H	-0,15	0,5235	52,35%	B	2,40	0,5248	52,48%
B	-2,39	0,4401	44,01%	I	-1,40	0,4437	44,37%
F	-4,60	0,3578	35,78%	F	-15,25	0,2097	20,97%
G	-5,92	0,3086	30,86%	G	-21,36	0,1064	10,64%
J	-14,21	0	0,00%	J	-27,66	0	0,00%

MAX EQ1	MAX EQ2	Dif.	%
11,65	30,53	18,88	61,84%

Tabela B.3: Comparação entre as qualificações excluindo usuários A e B utilizando EQ1 e EQ2

EQ1 excluindo usuários A e B				EQ2 excluindo usuários A e B			
Usuário	Pontos	Normal	%	Usuário	Pontos	Normal	%
C	12,64	1	100,00%	C	28,97	1	100,00%
D	5,65	0,75	75,00%	E	15,93	0,7415	74,15%
E	4,08	0,6938	69,38%	D	13,06	0,6846	68,46%
H	2,48	0,6366	63,66%	H	11,99	0,6634	66,34%
I	1,63	0,6062	60,62%	I	-0,99	0,3863	38,63%
F	-3,01	0,4045	40,45%	F	-11,69	0,1742	17,42%
G	-5,05	0,3315	33,15%	G	-12,95	0,1493	14,93%
J	-14,32	0	0,00%	J	-12,95	0	0,00%

MAX EQ1	MAX EQ2	Dif.	%
12,64	28,97	16,33	56,36%