

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
BIBLIOTECA UNIVERSITÁRIA**

Carlos Bonetti

**SILQ 2 - SISTEMA DE INTEGRAÇÃO LATTES-QUALIS:
REESTRUTURAÇÃO E MELHORIAS**

Florianópolis

2016

Carlos Bonetti

**SILQ 2 - SISTEMA DE INTEGRAÇÃO LATTES-QUALIS:
REESTRUTURAÇÃO E MELHORIAS**

Tese submetida ao Programa de Graduação
em Ciência da Computação para a obtenção
do Grau de Bacharel.
Orientador: Prof. Dr. Carina F. Dor-
neles

Florianópolis

2016

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

SobrenomeAutor, NomeAutor

Título do trabalho : Subtítulo do trabalho / NomeAutor

SobrenomeAutor ; orientador, NomeOrientador

SobrenomeOrientador ; coorientador, NomeCoorientador

SobrenomeCoorientador. - Florianópolis, SC, 2014.

100 p.

Tese (doutorado) - Universidade Federal de Santa
Catarina, Centro Tecnológico. Programa de Pós-Graduação em
Ciência da Computação.

Inclui referências

1. Ciência da Computação. 2. Exemplo de ficha
catalográfica. I. SobrenomeOrientador, NomeOrientador. II.
SobrenomeCoorientador, NomeCoorientador. III. Universidade
Federal de Santa Catarina. Programa de Pós-Graduação em
Ciência da Computação. IV. Título.

Carlos Bonetti

**SILQ 2 - SISTEMA DE INTEGRAÇÃO
LATTES-QUALIS: REESTRUTURAÇÃO E MELHORIAS**

Esta Tese foi julgada aprovada para a obtenção do Título de “Bacharel”, e aprovada em sua forma final pelo Programa de Graduação em Ciência da Computação.

Florianópolis, 15 de maio 2016.

Prof. Dr. Renato Cislaghi
Coordenador de Projetos

Banca Examinadora:

Prof. Dr. Carina F. Dorneles
Orientador

Prof. Dr. Raul Sidnei Wazlawick

Prof. Ronaldo dos Santos Mello

RESUMO

O texto do resumo deve ser digitado, em um único bloco, sem espaço de parágrafo. O resumo deve ser significativo, composto de uma sequência de frases concisas, afirmativas e não de uma enumeração de tópicos. Não deve conter citações. Deve usar o verbo na voz passiva. Abaixo do resumo, deve-se informar as palavras-chave (palavras ou expressões significativas retiradas do texto) ou, termos retirados de thesaurus da área.

Palavras-chave: Palavra-chave 1. Palavra-chave 2. Palavra-chave 3.

ABSTRACT

Resumo traduzido para outros idiomas, neste caso, inglês. Segue o formato do resumo feito na língua vernácula. As palavras-chave traduzidas, versão em língua estrangeira, são colocadas abaixo do texto precedidas pela expressão “Keywords”, separadas por ponto.

Keywords: Keyword 1. Keyword 2. Keyword 3.

LISTA DE FIGURAS

Figura 1	Exemplo de conjunto de dados do Qualis 2014.	23
Figura 2	Nova arquitetura do SILQ.	38
Figura 3	Novo esquema do modelo lógico do SILQ.	40

LISTA DE TABELAS

Tabela 1	Número de periódicos extraídos dos dados Qualis	37
----------	---	----

LISTA DE ABREVIATURAS E SIGLAS

AJAX	Asynchronous JavaScript and XML
API	Application Program Interface
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
CSS	Cascading Style Sheets
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
MCTI	Ministério de Ciência, Tecnologia e Inovação
REST	Representational state transfer
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
XML	Extensible Markup Language

SUMÁRIO

1	INTRODUÇÃO	17
1.1	OBJETIVOS	18
1.2	PROCEDIMENTOS METODOLÓGICOS	19
2	CONCEITOS E FUNDAMENTAÇÃO TEÓRICA	21
2.1	PRODUÇÃO E AVALIAÇÃO CIENTÍFICAS NO BRASIL	21
2.1.1	Qualis	22
2.1.2	A Plataforma Lattes	23
2.2	ARQUITETURA DE SISTEMAS BASEADOS NA <i>WEB</i>	24
2.2.1	<i>Web Services</i> e a arquitetura <i>REST</i>	24
2.2.2	Teste automatizado de software	26
2.3	INFORMATION RETRIEVAL E SIMILARIDADE TEXTUAL	27
2.3.1	Precision	27
2.3.2	Recall	27
3	TRABALHOS CORRELATOS	29
4	O SISTEMA DE INTEGRAÇÃO LATTES-QUALIS (SILQ)	31
4.1	HISTÓRIO E VISÃO GERAL DO SILQ-1	31
4.1.1	“Trabalhos futuros” citados pelo SILQ-1	32
4.2	SILQ 2	35
4.2.1	Extração e inserção dos novos dados Qualis	36
4.2.2	Criação do <i>Web Service</i>	37
4.2.3	Alterações no <i>front-end</i>	38
4.2.4	Alterações no modelo lógico	39
4.2.5	Garantia da qualidade	40
5	CONCLUSÕES E TRABALHOS FUTUROS	43
	REFERÊNCIAS	45

1 INTRODUÇÃO

A Plataforma Lattes, criada e mantida pela CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), é um sistema de informação responsável pela integração da base de dados de currículos, grupos de pesquisa e instituições. O Currículo Lattes se tornou o padrão nacional no registro da vida científica de estudantes e professores e é hoje adotada por institutos e universidades de todo o país. (CNPQ, 2015b)

No Currículo Lattes pode-se inserir dados gerais do pesquisador, produção bibliográfica, orientações, citações, participações em eventos científicos entre outros dados. No módulo Produção Bibliográfica, por exemplo, é possível a inserção de artigos publicados ou aceitos para publicação em periódicos indexados pelo ISSN. (CNPQ, 2015a)

A classificação da qualidade da produção intelectual é realizada pela CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) através de um conjunto de procedimentos denominado Qualis. O Qualis afere a qualidade de artigos e outras produções a partir da análise da qualidade dos veículos de divulgação, ou seja, periódicos científicos. Esta classificação é realizada pelas áreas de avaliação em um processo anual de atualização, sendo os veículos enquadrados em estratos indicativos de qualidade (A1 - o mais elevado; A2; B1; B2; B3; B4; B5; C - com peso zero. (CAPES, 2015a)

Apesar da Plataforma Lattes possuir um módulo de inclusão de publicações e permitir a definição do veículo onde este foi publicado, não há qualquer tipo de conexão entre os sistemas Lattes e Qualis, ou seja, o processo de avaliação de uma publicação (que é feita através da avaliação do veículo onde este foi publicado) era realizado de forma manual.

O Sistema de Integração Lattes Qualis (SILQ) surgiu no ano de 2015, desenvolvido como Trabalho de Conclusão de Curso dos alunos Felipe Nedel Mendes de Aguiar e Maria Eloísa Costa do curso de Ciência da Computação da Universidade Federal de Santa Catarina (UFSC). Orientados pela Prof. Carina F. Dorneles, da mesma instituição, o objetivo do sistema é a classificação automática de produções científicas do currículo Lattes do pesquisador, no quesito artigos e participações de conferências, através de busca de similaridade de dados extraídos do WebQualis, a partir de uma interface web amigável. (AGUIAR; COSTA, 2015)

A primeira versão do sistema foi finalizada em 2015 e desde então

encontra-se disponível de forma pública e gratuita através do endereço <http://silq.inf.ufsc.br/>.

A primeira versão do SILQ, no entanto, possui em seu banco de dados de eventos e periódicos extraídos do Qualis apenas dados referentes ao triênio 2010-2012, já que, até metade de 2015, o Qualis realizava classificações de forma trienal. Em 2016, no entanto, após a finalização do projeto SILQ, o Qualis alterou seu modo de operação para classificações anuais, já disponibilizando dados referentes aos anos 2013 e 2014, que ainda não estava incluídos nesta primeira versão da base de dados SILQ.

Outra questão relacionada à primeira versão do SILQ é a falta de alguns recursos que poderiam facilitar o processo de avaliação e acompanhamento de currículos, como gráficos de classificações dentro de grupos de pesquisa. Outra característica importante seria a capacidade de integração com outros sistemas, facilitando o reuso do serviço disponibilizado pelo SILQ em trabalhos futuros.

A proposta deste trabalho, portanto, é a criação de uma segunda versão do SILQ, desenvolvida a partir do código existente da primeira, a fim de realizar as mudanças sugeridas acima, junto com a atualização da base de dados e arquitetura do sistema para a inclusão das novas qualificações Qualis em um ritmo anual.

Outra proposta do trabalho é levar em consideração feedbacks de usuários ao avaliar currículos. Ou seja, permitir ao usuário sugerir o melhor *matching* caso o sistema não encontre um candidato satisfatório. A hipótese de pesquisa a ser avaliada a partir desta ideia, portanto, é avaliar se o feedback explícito de usuários pode aumentar a acurácia do SILQ, um sistema de avaliação baseada em similaridade textual.

1.1 OBJETIVOS

O objetivo geral deste trabalho é a reestruturação arquitetural do Sistema de Integração Lattes-Qualis (SILQ), incluindo a criação de API de integração com outros sistemas e atualização da base de dados conforme novas classificações Qualis, junto com uma análise do uso de feedback explícito de usuários ao realizar avaliações de publicações, baseadas em similaridade textual, e seu impacto na precisão dos resultados.

Os objetivos específicos são os seguintes:

- Reestruturação da arquitetura e banco de dados do SILQ a fim de suportar classificações de eventos e periódicos disponibilizados

em um ritmo anual;

- Atualização do banco de dados do sistema com as últimas classificações disponibilizadas pelo Qualis (anos 2013 e 2014);
- Criação de uma API pública de disponibilização dos serviços do SILQ, via camada de aplicação REST para integração com outros sistemas;
- Criação de uma documentação da nova API REST criada, com o objetivo de auxiliar outros serviços a se integrarem com o SILQ;
- Alterações na interface do sistema incluindo migração de framework de interface, novos gráficos de acompanhamento de grupos de pesquisa e melhorias gerais de usabilidade;
- Verificar se a precisão da avaliação de currículos, baseada em algoritmos de similaridade textual, pode ser melhorada utilizando feedback explícito de usuários.

1.2 PROCEDIMENTOS METODOLÓGICOS

A primeira etapa do trabalho envolveu a migração tecnológica da camada de interface e desacoplamento entre *server* e *client side*, originando uma camada RESTful de integração. Para isto, foram utilizadas práticas ágeis de desenvolvimento de software incluindo desenvolvimento orientado a testes e eventuais modelagens lógicas de arquitetura de sistemas usando práticas da Engenharia de Software.

Concomitantemente, ocorreu a alteração da base de dados para suporte aos novos registros disponibilizados pelo Qualis. Certos aspectos do banco foram alterados para suportar tais registros novos. Os dados também passaram por um processo de *data-cleaning* manual antes de serem inseridos no banco.

A segunda etapa do trabalho, ainda a ser desenvolvida, envolverá a medição da acurácia das avaliações realizadas pelo SILQ utilizando métricas de *precision* e *recall*. Para isso, um subconjunto de dados será analisado, utilizando o algoritmo atual de avaliação. Após estabelecida a métrica da acurácia atual, será levado em conta feedbacks explícitos de usuários, e uma nova medição de acurácia será realizada e comparada com os resultados anteriores.

2 CONCEITOS E FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão feitas uma breve descrição e conceituação básica do processo de produção e avaliação científica no Brasil, com o objetivo de familiarizar o leitor com a problemática envolvida neste processo e o que o SILQ propôs para automatizá-lo. Também serão introduzidos os conceitos tecnológicos e computacionais que de alguma forma foram explorados pelo sistema para chegar a este objetivo.

Esta revisão de literatura, porém, não será exaustiva ao ponto de revisitar os mesmos assuntos abordados no trabalho original de (AGUIAR; COSTA, 2015) e focará somente nos pontos onde este novo trabalho divergiu. Alguns assuntos fundamentais, entretanto, foram expostos novamente com o intuito de facilitar o processo de entendimento deste artigo como um trabalho independente. É o caso, por exemplo, das seções relacionadas aos Qualis, Lattes e das tecnologias base utilizadas no sistema.

2.1 PRODUÇÃO E AVALIAÇÃO CIENTÍFICAS NO BRASIL

A produção científica é uma das atividades universitárias que cumprem uma função básica dentro das instituições e que merece notável destaque. É através dela que o conhecimento produzido na Universidade é difundido à sociedade, externalizando descobertas para outros pesquisadores e para a própria comunidade. (CRISPIM, 2014)

No Brasil, o Ministério de Ciência, Tecnologia e Inovação (MCTI) é o órgão de administração direta que possui, entre sua lista de competências, a política nacional de pesquisa científica, tecnológica e inovação; e o planejamento, coordenação, supervisão e controle das atividades da ciência e tecnologia. (MCTI, 2016?)

Uma das agências subordinadas ao MCTI é o CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), órgão que tem como principais atribuições “fomentar a pesquisa científica e tecnológica e incentivar a formação de pesquisadores brasileiros” (CNPQ, 2016).

Analisando não só a produção científica, cabe ao Ministério da Educação (MEC), através de sua fundação subordinada, a CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) a avaliação dos programas de pós-graduação das universidades Brasileiras (CAPES, 2015b). Um dos principais critérios de avaliação se dá pela quantidade e qualidade das produções científicas, através de um conjunto de

procedimentos denominado Qualis (CAPES, 2015a).

2.1.1 Qualis

A classificação da qualidade da produção intelectual é realizada pela CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) através de um conjunto de procedimentos denominado Qualis. O Qualis afere a qualidade de artigos e outras produções a partir da análise da qualidade dos veículos de divulgação, ou seja, periódicos científicos. Esta classificação é realizada pelas áreas de avaliação em um processo anual de atualização, sendo os veículos enquadrados em estratos indicativos de qualidade (A1 - o mais elevado; A2; B1; B2; B3; B4; B5; C - com peso zero. (CAPES, 2015a)

O resultado desta estratificação é uma lista com a classificação dos veículos de publicação utilizados pelos programas de pós-graduação (*journals*, revistas, periódicos etc), separados por área e ano de avaliação (um mesmo veículo tem uma avaliação diferente para cada ano de avaliação e área de conhecimento). Desta forma, a avaliação da qualidade de uma produção de um pesquisador é realizado de forma indireta, atribuindo-se a esta produção o estrato dado àquele veículo onde ela foi publicada.

Note-se que o mesmo periódico, ao ser classificado em duas ou mais áreas distintas, pode receber diferentes avaliações. Isto não constitui inconsistência, mas expressa o valor atribuído, em cada área, à pertinência do conteúdo veiculado. Por isso, não se pretende com esta classificação que é específica para o processo de avaliação de cada área, definir qualidade de periódicos de forma absoluta. (CAPES, 2015a)

Até o final do ano de 2015, como cita o trabalho de (AGUIAR; COSTA, 2015, p. 31), os dados os resultados das avaliações Qualis encontravam-se estruturados em documentos no formato PDF e XLS. Também encontravam-se disponível apenas dados referentes ao triênio de 2010-2012. No início do ano de 2016, porém, a Qualis alterou seu modo de operação e passou a realizar avaliações anuais, já disponibilizando os novos dados de 2013 e 2014, junto com os antigos, em formato CSV¹ através do WebQualis² (Hoje plataforma Sucupira).

¹Comma-separated values

²<http://qualis.capes.gov.br/webqualis/>

Figura 1 – Exemplo de conjunto de dados do Qualis 2014.

1041-4347	IEEE Transactions on Knowledge and Data Engineering (Print)	CIÊNCIA DA COMPUTAÇÃO	A1
0018-9464	IEEE Transactions on Magnetics	CIÊNCIA DA COMPUTAÇÃO	B4
0278-0062	IEEE Transactions on Medical Imaging (Print)	CIÊNCIA DA COMPUTAÇÃO	A1
1536-1233	IEEE Transactions on Mobile Computing	CIÊNCIA DA COMPUTAÇÃO	A2
1520-9210	IEEE Transactions on Multimedia	CIÊNCIA DA COMPUTAÇÃO	A2
2162-237X	IEEE Transactions on Neural Networks and Learning Systems	CIÊNCIA DA COMPUTAÇÃO	A1
0018-9499	IEEE Transactions on Nuclear Science	CIÊNCIA DA COMPUTAÇÃO	B1
1045-9219	IEEE Transactions on Parallel and Distributed Systems (Print)	CIÊNCIA DA COMPUTAÇÃO	A2
0885-8950	IEEE Transactions on Power Systems	CIÊNCIA DA COMPUTAÇÃO	B2
0098-5589	IEEE Transactions on Software Engineering	CIÊNCIA DA COMPUTAÇÃO	A1
1083-4427	IEEE Transactions on Systems, Man and Cybernetics. Part A	CIÊNCIA DA COMPUTAÇÃO	A2
1094-6977	IEEE Transactions on Systems, Man and Cybernetics. Part B	CIÊNCIA DA COMPUTAÇÃO	A2
0018-9545	IEEE Transactions on Vehicular Technology	CIÊNCIA DA COMPUTAÇÃO	A1
1063-8210	IEEE Transactions on Very Large Scale Integration (VLSI) Systems	CIÊNCIA DA COMPUTAÇÃO	A2
1077-2626	IEEE Transactions on Visualization and Computer Graphics	CIÊNCIA DA COMPUTAÇÃO	A2
1536-1276	IEEE Transactions on Wireless Communications	CIÊNCIA DA COMPUTAÇÃO	A1
1536-1284	IEEE Wireless Communications	CIÊNCIA DA COMPUTAÇÃO	A1
2162-2337	IEEE Wireless Communications Letters	CIÊNCIA DA COMPUTAÇÃO	B4
1932-4537	IEEE eTransactions on Network and Service Management	CIÊNCIA DA COMPUTAÇÃO	B3
1932-8540	IEEE-RITA	CIÊNCIA DA COMPUTAÇÃO	B5
1545-5963	IEEE/ACM Transactions on Computational Biology and Bioinformatics	CIÊNCIA DA COMPUTAÇÃO	B1
0916-8532	IEICE Transactions on Information and Systems	CIÊNCIA DA COMPUTAÇÃO	B1
1751-861X	IET Computers & Digital Techniques (Online)	CIÊNCIA DA COMPUTAÇÃO	B1
1751-8601	IET Computers & Digital Techniques (Print)	CIÊNCIA DA COMPUTAÇÃO	B1
1751-8806	IET Software (Print)	CIÊNCIA DA COMPUTAÇÃO	B1
1091-9856	INFORMS Journal on Computing	CIÊNCIA DA COMPUTAÇÃO	B1
1526-5528	INFORMS Journal on Computing (Online)	CIÊNCIA DA COMPUTAÇÃO	B1
2090-4355	ISRN Communications and Networking	CIÊNCIA DA COMPUTAÇÃO	C
2237-4523	Iberoamerican Journal of Applied Computing	CIÊNCIA DA COMPUTAÇÃO	B5
0262-8856	Image and Vision Computing	CIÊNCIA DA COMPUTAÇÃO	A1

2.1.2 A Plataforma Lattes

Desde a década de 80 notava-se a necessidade de um formulário padrão para registro dos currículos de pesquisadores brasileiros que possibilitasse a busca e seleção de especialistas bem como a geração de um mapa sobre a distribuição da pesquisa científica no Brasil. Em 1999, o CNPq lançou a Plataforma Lattes³, um sistema de informação para registro e consulta de currículos acadêmicos, e padronizou o Currículo Lattes como o formulário de currículo a ser utilizado no âmbito do Ministério da Ciência e Tecnologia e CNPq. (CNPQ, 2016, 2015b)

Entre os dados contidos em um currículo Lattes, estão os dados gerais do pesquisador, produção bibliográfica, orientações, citações, formação acadêmica, etc. No módulo Produção Bibliográfica, por exemplo, é possível a inserção de artigos publicados em periódicos indexados pelo ISSN⁴, livros e capítulos, textos em jornais ou revistas, trabalhos publicados em anais de eventos, entre outras formas de produções

³<http://lattes.cnpq.br/>

⁴International Standard Serial Number

(CNPQ, 2015a). Estes dados de publicações seriam usados pela primeira versão do SILQ (AGUIAR; COSTA, 2015) para a avaliação automática de currículos.

A Plataforma Lattes, além de apresentar seus currículos através de uma interface web usando HTML, também disponibiliza os dados em formato XML⁵, uma linguagem de marcação semi-estruturada legível por humanos e máquinas (AGUIAR; COSTA, 2015, p. 34). Foi este o formato utilizado pelo SILQ para a extração dos dados Lattes de um pesquisador, conforme relatado em (AGUIAR; COSTA, 2015).

2.2 ARQUITETURA DE SISTEMAS BASEADOS NA WEB

Nesta seção serão introduzidos conceitos e fundamentos relacionados à construção, interoperabilidade e arquitetura geral de sistemas baseados na Web. A maioria destes conceitos já estão consolidados na indústria de software e são aqui explanados com o objetivo de familiarizar o leitor com os assuntos discutidos no decorrer do desenvolvimento deste trabalho, bem como para apresentar as justificativas tecnológicas que motivaram as alterações arquiteturais realizadas.

2.2.1 *Web Services* e a arquitetura *REST*

Segundo a W3C, Web Services são serviços que provêm meios de interoperabilidade padronizados entre diferentes aplicações de software rodando em ambientes e/ou plataformas heterogêneas. Um Web Service, portanto, oferece tal serviço de interoperabilidade através da *World Wide Web*, expondo uma interface descrita em um formato legível por máquinas. Diferentes entidades consomem tal serviço através de trocas de mensagem utilizando o formato especificado (BOOTH et al., 2004).

Representational state transfer (REST) é um estilo arquitetural utilizado para a implementação de um Web Service, baseado no protocolo HTTP⁶ (FIELDING; TAYLOR, 2002). Serviços web que implementam uma API⁷ no estilo REST também são chamadas de *RESTful APIs* e comumente utilizam JSON⁸ como formato de intercomunicação, podendo utilizar também XML, Atom, texto, entre outros (RICHARD-

⁵Extensible Markup Language

⁶Hypertext Transfer Protocol

⁷Application Program Interface

⁸JavaScript Object Notation

SON, 2013).

Um exemplo de comunicação entre um servidor web rodando um serviço de consulta de uma biblioteca e um cliente (outra aplicação) que desejasse saber a coleção de livros de seu acervo seria o seguinte:

1. O cliente envia um requisição HTTP do tipo GET para a URI `http://api.biblioteca.exemplo.com/acervo`;
2. O servidor da biblioteca responde ao pedido com a coleção de livros de seu acervo em formato JSON. Um exemplo de resposta poderia ser:

Listing 2.1 – Exemplo de uma resposta JSON dada por um serviço RESTful

```
1 {  
2   acervo: [  
3     {  
4       titulo: "Uma breve história do tempo",  
5       autor: "Stephen W. Hawking"  
6     },  
7     {  
8       titulo: "Metodologia de Pesquisa para  
9         Ciência da Computação",  
10      autor: "Raul Sidnei Wazlawick"  
11    },  
12    ...  
13  ]  
14 }
```

3. A resposta em JSON é recebida pelo cliente, interpretada e o conteúdo processado (consumido) da maneira que a aplicação necessitar.

Desta forma, clientes que necessitem da informação de acervo da biblioteca do exemplo acima não precisam se preocupar com os detalhes tecnológicos necessários para realizar essa consulta (se os dados estão em um banco de dados local ou distribuído, relacional ou não, qual linguagem é utilizada para consulta, etc), eles apenas fazem uma requisição HTTP de um certo tipo e em uma URI pré-estipulada (normalmente definidas na documentação do *web service*) e são servidos com a informação desejada.

Uma interpretação moderna de tal arquitetura é denominada *microserviço*, que compartilha das características de um *web service* clássico, mas cuja maior diferença seria que os *microservices* tem como objetivo prover um serviço específico, desempenhando uma única função minimalista, porém completa (FOWLER; LEWIS, 2014). Deste ponto de vista, um *web service* clássico monolítico⁹ poderia, na maior parte das vezes, ser decomposto em vários microserviços, cada um especializado em uma função e podendo ter características tecnológicas completamente distintas.

Outra definição comumente utilizada em aplicações que implementam a arquitetura REST (ou mesmo outras arquiteturas web) é a divisão da aplicação em dois níveis lógicos, o *server-side* e o *client-side*. Dentro do modelo cliente-servidor, o *server-side* de uma aplicação é toda infraestrutura de um servidor que recebe e processa requisições, geralmente retornando uma resposta à este pedido. O *client-side*, por sua vez, é toda infraestrutura que efetua a requisição ao servidor, recebe sua resposta e a manipula para apresentar o resultado ao usuário. No contexto das aplicações web modernas, *server/client-side* são geralmente associados aos termos *back-end* e *front-end* da aplicação, respectivamente, apesar de não possuírem significados completamente idênticos (JIA; ZHOU, 2005).

Tais conceitos estimularam a criação de uma API REST para o SILQ, com o objetivo de tornar disponível, de forma programática, os dados Qualis inseridos em sua base de dados e o serviço de avaliação de currículos. Desta forma, outras aplicações podem consumir os serviços oferecidos pelo sistema sem se preocupar com os detalhes tecnológicos utilizados pelo SILQ para implementar tais funcionalidades, aumentando a facilidade de integração com outros sistemas.

2.2.2 Teste automatizado de software

O processo de desenvolvimento de software é uma atividade em que comumente ocorrem, apesar dos esforços e do emprego de técnicas e ferramentas, diversos erros (*bugs*) no produto final. A Garantia da Qualidade de Software é uma disciplina que agrega as atividades que ocorrem de forma paralela ao desenvolvimento com o objetivo de minimizar a ocorrência de erros e riscos no produto (BARBOSA; MALDON-

⁹Dentro do conceito da arquitetura de microserviços, monolítica é uma aplicação construída como uma unidade única e indivisível, em oposição ao conceito composto e distribuído dos microserviços (FOWLER; LEWIS, 2014)

ADO; VINCENZI, 2000).

Três das técnicas utilizadas para a garantia da qualidade de teste são a Verificação, Validação e Teste (BARBOSA; MALDONADO; VINCENZI, 2000). A verificação avalia se o produto está sendo desenvolvido corretamente, enquanto a validação avalia se o produto correto está sendo desenvolvido (BOEHM, 1981). Já o teste, no contexto de software, geralmente implica na execução do próprio produto, considerando-se valores de entrada pertinentes para teste, para verificação de certas funcionalidades do sistema (BARBOSA; MALDONADO; VINCENZI, 2000).

Segundo (PRESSMAN, 2001) e (BOURQUE; FAIRLEY, 2014, cap. 5), diferentes tipos de testes podem ser agrupados em níveis de acordo com o grau de especificidade do módulo ou do requerimento sendo testado. Testes unitários têm como objetivo verificar o funcionamento de unidades indivisíveis e isoladas de um software, garantindo que cada uma delas funcione conforme especificado. Testes de integração, por sua vez, tem por objetivo encontrar falhas na integração entre as unidades de um sistema. Testes de sistema, ou testes *end-to-end* (fim a fim), executam a validação do sistema final como um todo, simulando as ações que um usuário real realizaria ao utilizar a aplicação.

2.3 INFORMATION RETRIEVAL E SIMILARIDADE TEXTUAL

Introdução sobre busca por similaridade e information retrieval. Como o SILQ realiza suas buscas.

2.3.1 Precision

Conceituação.

2.3.2 Recall

Conceituação.

3 TRABALHOS CORRELATOS

<http://qualis.renesp.com.br/index.php>

4 O SISTEMA DE INTEGRAÇÃO LATTES-QUALIS (SILQ)

O presente trabalho é uma continuação dos esforços iniciados por (AGUIAR; COSTA, 2015). Neste capítulo, portanto, será apresentado o histórico do trabalho original, os motivos que levaram ao desenvolvimento de um novo trabalho para sua continuação, comparações do que foi de fato alterado no sistema e por que estas alterações foram necessárias.

Para mais fácil compreensão e contextualização, durante o texto serão utilizados os termos “trabalho original”, “artigo original”, “SILQ-1” ou ainda “primeira versão do sistema” ou sinônimos para referir-se ao trabalho de (AGUIAR; COSTA, 2015). Em contrapartida, os termos, “SILQ-2”, “nova versão do sistema” e derivados referem-se a este trabalho.

4.1 HISTÓRIO E VISÃO GERAL DO SILQ-1

Este trabalho é uma continuação de (AGUIAR; COSTA, 2015), um Trabalho de Conclusão de Curso de alunos do curso de Ciência da Computação da UFSC, orientados pela Professora Carina F. Dorneles. O objetivo deste trabalho de 2015 era a criação de um sistema que deveria ser capaz de “qualificar produções científicas, no quesito artigos e participações em conferências, por busca por similaridade de dados com os dados extraídos do WebQualis” (AGUIAR; COSTA, 2015, p. 26-27). Este objetivo foi alcançado com a criação da primeira versão do Sistema de Integração Lattes-Qualis (SILQ), lançado no segundo semestre de 2015 e disponível no sítio <http://silq.inf.ufsc.br/>.

Apesar de estável e com sua função principal sendo desempenhada de forma satisfatória, o SILQ-1 deixou algumas lacunas e melhorias a serem desenvolvidas por trabalhos futuros. Segundo os próprios autores, “[...] o SILQ foi concebido para ser uma ferramenta de domínio público e vários projetos devem nascer a partir dele. A continuidade do projeto só tem a acrescentar ao mundo acadêmico [...]” (AGUIAR; COSTA, 2015, p. 79), o que motivou a criação deste trabalho para a continuação do projeto original. O ponto de partida deste trabalho, portanto, foi a seção “Trabalhos futuros” do artigo original.

4.1.1 “Trabalhos futuros” citados pelo SILQ-1

Na seção de “Conclusões e Tabalhos Futuros” do artigo original, (AGUIAR; COSTA, 2015) citam 8 itens de melhoria que poderiam ser revisitados por trabalhos futuros. Cada um destes itens está exposto abaixo junto uma explicação de quais foram as atitudes tomadas por este trabalho para resolvê-las.

1. Controle de persistência e transações

definir um novo container de execução para automatizar o uso de uma API de persistência. No momento, as transações são controladas de forma manual; (AGUIAR; COSTA, 2015, p. 79)

Esta melhoria trata-se de um detalhe de implementação do SILQ-1 que, devido à sua arquitetura, forçava a criação e gerenciamento manual de transações ao realizar acessos ao banco de dados. A automatização desta tarefa foi rapidamente alcançada no SILQ-2 com a utilização do *Spring Framework*, conforme relatado na seção 4.2.2.

2. Automatização da atualização dos dados Qualis

as publicações periódicas dos documentos de avaliação do WebQualis levam à necessidade de atualizar o banco de dados periodicamente. A construção de uma funcionalidade automatizada é pertinente; (AGUIAR; COSTA, 2015, p. 79)

Os dados Qualis utilizados pelo sistema para realizar a avaliação de currículos são atualmente inseridos de forma manual. Na segunda etapa deste trabalho será incluído um módulo no sistema onde administradores poderão fazer *upload* de novos dados Qualis, conforme eles forem sendo divulgados e assim não necessitando inserí-los de forma manual no banco de dados.

Os dados Qualis 2014 e 2015, divulgados somente no ano de 2016, entretanto, já foram inseridos manualmente na base de dados do SILQ-2, passando por um processo de limpeza de dados descrito na seção 4.2.1.

3. *Fine-tuning* da função de similaridade

desenvolvimento de uma nova função de similaridade ou novos procedimentos de cálculo de similaridades. Foram testadas funções já bem conhecidas pela comunidade científica, com testes concretos para levantar qual delas é a mais indicada no escopo da ferramenta. Porém, é possível que se alcance resultados melhores utilizando uma combinação de funções, ou até mesmo criando uma nova especificamente para o cálculo de similaridade de nomes de periódicos e conferências; (AGUIAR; COSTA, 2015, p. 79)

Os testes realizados pelos autores do trabalho original somente levaram em conta o tempo de processamento entre diferentes tipos de algoritmos de similaridade textual, escolhendo a função *trigram* em detrimento das outras por sua integração nativa com o PostgreSQL e por seu tempo de resposta superior aos outros métodos (AGUIAR; COSTA, 2015, p. 57-58). Não foi realizada, porém, uma comparação da acurácia dos resultados de diferentes métodos de similaridade.

Um dos objetivos do presente trabalho é a estipulação das métricas de precisão e revocação (*precision* e *recall*) para o algoritmo de avaliação atual. Esta atividade ainda será realizada (na segunda etapa deste trabalho), porém com enfoque no uso de feedback de usuários para aumento de acurácia. Após a obtenção destas métricas, entretanto, será possível comparar outros algoritmos de similaridade textual, além do *trigram* utilizado atualmente, com a finalidade de escolher o melhor para a situação. Todavia, esta comparação de diferentes algoritmos não é um dos focos deste trabalho.

4. Considerar outras informações do pesquisador

levantamento das outras informações contidas no currículo, como o desempenho do pesquisador antes de ser incluído em um grupo de pós-graduação ou a quantidade e a qualidade de outras atividade acadêmicas; (AGUIAR; COSTA, 2015, p. 79)

Este item envolve o processamento de outras informações contidas no currículo Lattes de um pesquisador e, em um primeiro

momento, foge do escopo original do SILQ que é a integração entre o Lattes e o Qualis, utilizando as publicações e informações de onde foram publicadas/apresentadas como item de avaliação.

Por este motivo, este item não foi abordado neste trabalho. Todavia, outros trabalhos de pesquisa relacionados estão paralelamente em desenvolvimento.

5. Relação entre orientações e produções associadas

levantamento da relação entre as atividades de orientação, como bolsas de iniciação científica, e as produções acadêmicas associadas; (AGUIAR; COSTA, 2015, p. 80)

Não foi abordado neste trabalho, pelos mesmos motivos do item anterior.

6. Busca por informações de eventos

expansão do módulo da busca por informações de eventos. A complexidade do tema impediu o desenvolvimento completo desse módulo na ferramenta SILQ. Procedimentos automáticos de busca utilizando-se web crawlers e o tratamento correto dessas informações fornecem muitos temas para desenvolvimento de módulos novos; (AGUIAR; COSTA, 2015, p. 80)

A inserção de dados do Qualis Eventos¹ no sistema se dá de forma manual, à medida em que novas informações são disponibilizadas pela CAPES. O objetivo deste item seria a criação de um *web crawler* para captura automática de dados de eventos científicos na web. Devido à complexidade deste tema e por se tratar de um assunto independente, não relacionado com o foco deste trabalho, este item não foi abordado.

7. Feedback do usuário

permitir que o usuário auxilie a ferramenta na qualificação, ou seja, no caso dele observar algum matching errado, ele próprio pode sugerir o matching correto; (AGUIAR; COSTA, 2015, p. 80)

¹dados e conceitos relacionados a eventos científicos disponibilizados pela CAPES para algumas áreas específicas

Este é o ponto de análise principal deste trabalho e que deu origem à hipótese de pesquisa. É possível aumentar a precisão da ferramenta de qualificação, de uma maneira geral, utilizando sugestões (ou feedback) de usuários? Para responder à esta pergunta, entretanto, antes é necessário estipular métricas para calcular o grau de precisão atual da ferramenta e então comparar o algoritmo antigo com um novo que leve em consideração estas sugestões. Este será o foco da próxima etapa deste trabalho.

8. Período de participação em grupo de pesquisa

indicar o período em que o professor está vinculado à um programa de pós-graduação, permitindo que as avaliações dos currículos também possam ser realizadas considerando-se apenas os períodos nos quais o pesquisador estava vinculado à esse programa. (AGUIAR; COSTA, 2015, p. 80)

Este item trata-se de um detalhe de gerenciamento de grupos e de sua avaliação, que considera publicações de um pesquisador mesmo quando este foi membro do grupo em um período que não cobre totalmente o período de avaliação estipulado. Para correção deste comportamento, basta dar ao administrador do grupo a opção de informar o período de ingresso ou de participação de um pesquisador no grupo de pesquisa e atualizar o algoritmo de avaliação para considerar somente publicações neste período. Trata-se, também, de uma modificação que será introduzida nas próximas iterações de desenvolvimento deste trabalho.

4.2 SILQ 2

É possível aumentar a acurácia de um sistema (de recomendação/avaliação) baseado em algoritmo de similaridade textual utilizando feedback do usuário?

Qual o Threshold (nível de similaridade) ideal para o SILQ?

Qual a função de similaridade ideal para o SILQ?

4.2.1 Extração e inserção dos novos dados Qualis

Os dados Qualis de eventos e periódicos foram extraídos pelo trabalho original a partir de documentos em PDF e planilhas. Como relatado em (AGUIAR; COSTA, 2015, p. 50-52), os dados foram extraídos dos PDFs utilizando bibliotecas para manipulação deste formato e passaram por um processo de limpeza. A partir deste processo, aproximadamente 107 mil tuplas foram criadas representando os periódicos do triênio 2010-2012.

No primeiro semestre do ano de 2016, porém, a CAPES disponibilizou novos dados Qualis referentes ao ano de 2013 e 2014, alterando sua periodicidade de avaliação de trimestral para anual. Estes novos dados foram divulgados em formato CSV, junto com os antigos dados do triênio 2010-2012, mas desta vez separados por ano. Estes dados foram obtidos pela plataforma Sucupira (antigo WebQualis): <http://qualis.capes.gov.br/webqualis/>.

Como o conjunto de dados do SILQ-1 contava apenas com dados trimestrais, as tuplas originais não possuíam informação de ano (só podia-se deduzir que os dados faziam parte do triênio, mas não sabia-se qual ano). Desta forma, todos as planilhas divulgadas passaram por um novo processo de extração, incluindo os dados já inseridos no SILQ-1, mas agora considerando a nova informação de ano.

As cinco novas planilhas passaram por um processo de *data cleaning* manual:

1. Substituição de entidades HTML especiais por seu caractere correspondente. Ex.: “&” por “&”;
2. Normalização do campo ISSN para o formato “9999-9999”: alguns registros não possuíam o dígito separador ou omitiam os zeros à esquerda;
3. Correção de ISSNs errôneos: alguns registros possuíam o campo ISSN com dígitos faltando. Ex.: 0034-167 (Revista Brasileira de Enfermagem), cujo número correto seria 0034-7167. Estes casos foram tratados um a um e os números corretos identificados através de pesquisas na web.

As planilhas CSV foram então dadas como *input* à base de dados resultando na criação de 339204 tuplas, referentes a cada um dos anos do período divulgado 2012-2015. A Tabela 1 mostra o número de periódicos extraídos para cada ano de avaliação do Qualis.

Tabela 1 – Número de periódicos extraídos dos dados Qualis

Ano	Nº de periódicos extraídos
2010	75786
2011	66171
2012	108272
2013	44437
2014	44538
Total	339204

Após esta etapa, verificou-se que todos os registros da base de dados antiga do SILQ-1 estavam contidos nesta nova versão. Desta forma, os dados do SILQ-1 foram mantidos, porém agora incluindo o ano de avaliação, além do nome do periódico, ISSN, conceito atribuído pela CAPES e área de avaliação.

A inclusão dos novos registros aumentou o número de publicações passíveis de serem avaliadas pelo sistema. Os dados Qualis de 2015 e 2016 ainda não foram divulgados no momento de escrita deste artigo, mas podem ser incluídos facilmente na base de dados assim que forem disponibilizados pela CAPES.

4.2.2 Criação do *Web Service*

A primeira versão do SILQ foi desenvolvida com base no *Play² Framework*, um *web application framework* escrito em Java e Scala que simplifica o processo de construção de uma aplicação web. Este mesmo framework provê APIs e bibliotecas para o desenvolvimento tanto do *back-end* quando do *front-end* de uma aplicação web.

Para esta nova versão do SILQ, porém, um dos objetivos seria a criação de uma API de integração programática, tornando o SILQ não só um sistema para uso de usuários finais, mas também para integração com outras ferramentas, como um *web service*. Para tanto, uma alteração significativa na arquitetura da aplicação foi feita, separando o sistema em um *server-side* rodando Spring³ e servindo seu conteúdo através de uma API REST; e um *client-side* remodelado utilizando AngularJS⁴ e consumindo o serviço através de requisições HTTP.

²<https://www.playframework.com/>

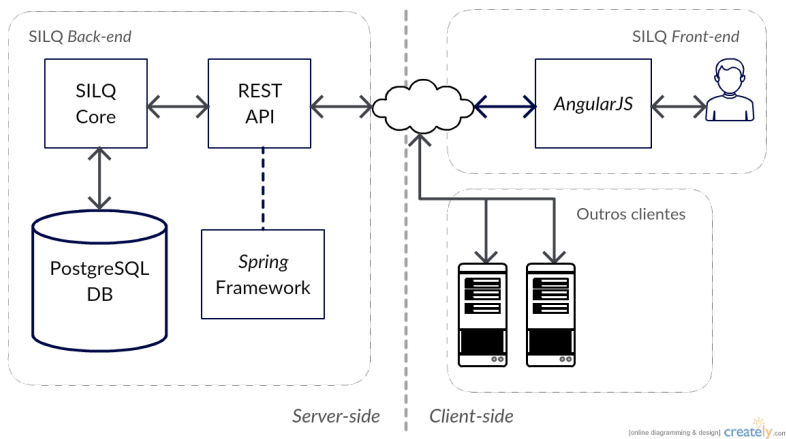
³<http://projects.spring.io/spring-framework/>

⁴<https://angularjs.org/>

Em um primeiro momento, o “núcleo” do SILQ, que inclui os algoritmos de avaliação e processamento dos currículos, não foi alterado, apenas a forma com que este serviço é disponibilizado e consumido. Desta forma, a linguagem utilizada no sistema, o Java, foi mantido. A escolha da alteração do framework back-end, de *Play!* para *Spring*, se deu ao fato da mudança no paradigma arquitetural da aplicação, com a introdução da camada REST. O Spring se trata de um framework Java *open-source* largamente utilizado pela comunidade que oferece suporte a vários aspectos de uma aplicação web, incluindo a construção de interfaces RESTful.

A figura 2 mostra um esquema da nova arquitetura do SILQ. Na figura, a caixa “SILQ Core” representa a camada de serviço da aplicação, responsável pelo acesso ao banco de dados, avaliação de currículos e gerenciamento dos dados Qualis, cuja implementação foi realizada no trabalho passado (apesar de também sofrer algumas mudanças tecnológicas no trabalho atual).

Figura 2 – Nova arquitetura do SILQ.



4.2.3 Alterações no *front-end*

Na primeira versão do SILQ, as páginas HTML do sistema eram dinamicamente renderizadas no server-side pelo *Play!* Framework e servidas conforme as requisições dos clientes. Desta forma, além de executar toda a lógica de aplicação, o servidor também fica responsável

pela lógica de apresentação do conteúdo.

Em arquiteturas REST, a lógica de apresentação de dados (ou visão), fica completamente separada do servidor, executando no *web browser* do cliente. Ou seja, o servidor só recebe requisições HTTP, as processa e retorna uma resposta em formato JSON. Esta resposta é recebida pelo cliente que então apresenta o conteúdo através da interface gráfica HTML renderizada pelo browser. Desta forma, a carga no servidor é diminuída, já que ele não está mais incumbido da tarefa de renderização das páginas HTML que serão servidas.

Utilizando esta ideia, no SILQ-2, a camada de visão foi totalmente reescrita em *Javascript*, utilizando o framework *AngularJS*. Trata-se de um framework *open-source* desenvolvido pela Google que auxilia na criação de interfaces web dinâmicas e integração com *web-services* via REST.

Desta forma, quando um cliente realiza a primeira requisição à URL do SILQ, o servidor serve uma página HTML (geralmente denominada de *index*) junto com os *assets* necessários para a construção de conteúdo dinâmico e estilização da página (imagens, arquivos CSS e Javascript). As requisições subsequentes, entretanto, utilizam somente requisições assíncronas ao servidor, utilizando Javascript: uma requisição conhecida por *Ajax* (Asynchronous JavaScript and XML). Estas requisições utilizam a camada REST do servidor que retornará respostas em JSON e cujo conteúdo será processado no cliente e montará as páginas de forma dinâmica. Desta forma, parte da lógica da aplicação como um todo é transferida para o cliente, reduzindo a carga no servidor.

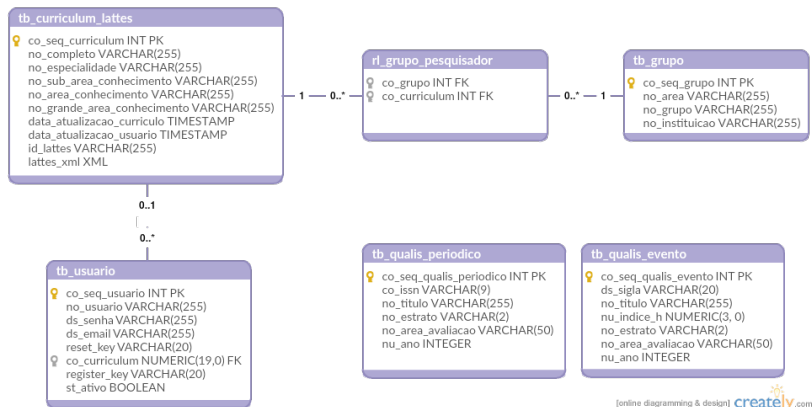
4.2.4 Alterações no modelo lógico

O banco de dados relacional implementado em *PostgreSQL* também sofreu algumas modificações em relação ao trabalho anterior. A figura 3 mostra o esquema lógico do banco de dados da versão atual do SILQ.

Algumas mudanças pequenas de nomenclatura foram realizadas. Uma mudança significativa, porém, foi a unificação das tabelas *tb_dado_geral* e *tb_profissional* para a nova tabela *tb_curriculum_lattes*. A função destas duas tabelas antigas era guardar os currículos enviados de usuários e de pesquisadores de grupos, respectivamente. Nesta nova versão do SILQ, porém, em ambos os casos os currículos Lattes enviados pelo sistema são salvos na mesma tabela de currículos (*tb_curriculum_lattes*), inclusive tendo seus registros reutilizados em caso de, por exemplo, dois

usuários enviarem o mesmo currículo. Neste caso, o sistema armazena somente uma vez o currículo e cria duas referências diferentes a ele, poupando-o do trabalho de duplicar os dados do currículo Lattes, cuja representação em XML contém em média algumas dezenas de *kilobytes*.

Figura 3 – Novo esquema do modelo lógico do SILQ.



4.2.5 Garantia da qualidade

Para a garantia da qualidade desta nova versão do SILQ, foram introduzidas duas camadas de testes automatizados: a primeira, envolvendo *testes unitários* e *testes de integração* escritos em Java e com o objetivo de garantir a corretude do server-side da aplicação; e a segunda, basicamente um *teste de sistema* escrito em Javascript e simulando casos de uso reais, garantindo a corretude do client-side e de sua integração com o servidor.

No SILQ, os testes unitários e de integração foram escritos em Java utilizando o framework de testes *jUnit*⁵. Eles testam cada função da camada de serviço da aplicação, por exemplo simulando o upload de um currículo qualquer e verificando se os dados extraídos e retornados são de fato aqueles contidos no currículo. Foram criados testes, inclusive, para as funcionalidades já existentes desde o SILQ-1, mas que ainda não eram cobertos com casos de teste automatizados, para assim aumentar a confiabilidade do sistema e garantir que mudanças futuras não ocasionem *bugs* nestes módulos antigos.

⁵<http://junit.org/>

Este nível de teste, porém, não valida a interface de usuário e sua integração com a camada de serviço. Ou seja, todo o client-side da aplicação ainda está “descoberto” de casos de testes. Para tanto, foram incluídos no SILQ casos de teste de sistema escritos em Javascript e utilizando o framework de testes *Protractor*⁶, criado especificamente para testes *end-to-end* de aplicações feitas com AngularJS. O Protractor simula o uso de um usuário real realizando cliques em botões, preenchendo formulários e navegando através de links da aplicação. Desta forma, são garantidos algum nível de corretude da interface do sistema e da integração entre o client e server side.

⁶<http://www.protractortest.org/>

5 CONCLUSÕES E TRABALHOS FUTUROS

O SILQ é um esforço coletivo para a automatização e consequente melhoria na qualidade de gestão de grupos de pesquisa. A constante atualização tecnológica e alimentação da base de dados da ferramenta é um processo que deve ser mantido para mantê-la viável aos seus usuários.

O objetivo inicial de refatoração arquitetural com inclusão de melhorias de interface, novas funcionalidades e criação da camada de integração REST foi alcançada. Tanto usuários como desenvolvedores de aplicações interessados no serviço se beneficiam desta alteração. As mudanças já se encontram disponíveis na página oficial do SILQ: <http://silq.inf.ufsc.br/>, em constante evolução. A documentação contendo as diretrizes para integração, porém, ainda não foi realizado, mas está programado para a entrega final deste trabalho.

A questão da precisão do algoritmo de avaliação e se sua acurácia pode ser aumentada com feedback de usuários, porém, ainda permanece. Esta hipótese será trabalhada nas próximas etapas deste projeto, conforme descritas na metodologia do trabalho, e será respondida até a entrega final.

REFERÊNCIAS

- AGUIAR, F. N. M. de; COSTA, M. E. **SILQ - Sistema de Integração Lattes Qualis**. Florianópolis: Universidade Federal de Santa Catarina, Biblioteca Universitária, 2015.
- BARBOSA, E. F.; MALDONADO, J. C.; VINCENZI, A. M. R. Introdução ao teste de software. **XIV Simpósio Brasileiro de Engenharia de Software**, 2000.
- BOEHM, B. W. **Software Engineering Economics**. Englewood Cliffs, NJ: Prentice Hall, 1981.
- BOOTH, D. et al. W3C Working Group Note, **Web Services Architecture**. fev. 2004. Disponível em: <<https://www.w3.org/TR/2004/NOTE-ws-arch-20040211>>.
- BOURQUE, P.; FAIRLEY, R. E. (Ed.). **SWEBOK: Guide to the Software Engineering Body of Knowledge**. Version 3.0. Los Alamitos, CA: IEEE Computer Society, 2014. ISBN 978-0-7695-5166-1. Disponível em: <<http://www.swebok.org/>>.
- CAPES. **Classificação da produção intelectual**. 2015. Disponível em: <<http://www.capes.gov.br/avaliacao/instrumentos-de-apoio/classificacao-da-producao-intelectual>>. Acesso em: 17/11/2015.
- CAPES. **História e missão**. 2015. Disponível em: <<http://www.capes.gov.br/historia-e-missao>>. Acesso em: 22/05/2016.
- CNPQ. **Módulo Produção Bibliográfica**. 2015. Disponível em: <http://ajuda.cnpq.br/index.php/Módulo_Produção_Bibliográfica>. Acesso em: 17/11/2015.
- CNPQ. **Sobre a Plataforma Lattes**. 2015. Disponível em: <<http://www.cnpq.br/web/portal-lattes/sobre-a-plataforma>>. Acesso em: 17/11/2015.
- CNPQ. **Conselho Nacional de Desenvolvimento Científico e Tecnológico**: Institucional. 2016. Disponível em: <http://cnpq.br/apresentacao_institucional/>. Acesso em: 22/05/2016.

CNPQ. **Histórico:** História do surgimento da plataforma lattes. 2016. Disponível em: <http://memoria.cnpq.br/web/portal-lattes/historico>. Acesso em: 31/05/2016.

CRISPIM, B. do A. (Editorial) A importância da publicação de artigos científicos: Uma abordagem para área de ciência animal. **Journal of the Selva Andina Animal Science**, v. 1, n. 1, 2014.

FIELDING, R. T.; TAYLOR, R. N. Principled design of the modern web architecture. **ACM Transactions on Internet Technology**, v. 2, p. 115–150, 2002.

FOWLER, M.; LEWIS, J. **Microservices**. 2014. Disponível em: <http://martinfowler.com/articles/microservices.html>. Acesso em: 07/06/2016.

JIA, W.; ZHOU, W. **Distributed Network Systems: From concepts to implementations**. Boston: Springer Science and Business Media Inc, 2005.

MCTI. **Ministério da Ciência, Tecnologia e Inovação:** Institucional. 2016? Disponível em: <http://www.mcti.gov.br/institucional>. Acesso em: 22/06/2016.

PRESSMAN, R. S. **Software Engineering: A Practitioner's Approach**. 5th. ed. [S.l.]: McGraw Hill, 2001. ISBN 0-07-365578-3.

RICHARDSON, L. **RESTful Web APIs**. Sebastopol, Calif: O'Reilly, 2013. ISBN 9781449358068.