



Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Engenharia de software

Proposta de uma arquitetura integrativa baseada em serviços para um ambiente virtual

Autora: Beatriz Ferreira Gonçalves
**Orientador: Professor Doutor Sérgio Antônio Andrade de
Freitas**

Brasília, DF
2016



Beatriz Ferreira Gonçalves

Proposta de uma arquitetura integrativa baseada em serviços para um ambiente virtual

Monografia submetida ao curso de graduação em Engenharia de software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de software.

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Professor Doutor Sérgio Antônio Andrade de Freitas

Brasília, DF

2016

Beatriz Ferreira Gonçalves

Proposta de uma arquitetura integrativa baseada em serviços para um ambiente virtual/ Beatriz Ferreira Gonçalves. – Brasília, DF, 2016-

42 p. : il. (algumas color.) ; 30 cm.

Orientador: Professor Doutor Sérgio Antônio Andrade de Freitas

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB
Faculdade UnB Gama - FGA , 2016.

1. Palavra-chave01. 2. Palavra-chave02. I. Professor Doutor Sérgio Antônio Andrade de Freitas. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Proposta de uma arquitetura integrativa baseada em serviços para um ambiente virtual

CDU COLOCAR CDU

AQUI - DADOS DA FICHA CATALOGRAFICA

Beatriz Ferreira Gonçalves

Proposta de uma arquitetura integrativa baseada em serviços para um ambiente virtual

Monografia submetida ao curso de graduação em Engenharia de software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de software.

Trabalho aprovado. Brasília, DF, XX de Junho de 2016:

**Professor Doutor Sérgio Antônio
Andrade de Freitas**
Orientador

Titulação e Nome do Professor
Convidado 01
Convidado 1

Titulação e Nome do Professor
Convidado 02
Convidado 2

Brasília, DF
2016

ESCREVER AQUI A DEDICATÓRIA

Agradecimentos

ESCREVER AQUI OS MEUS AGRADECIMENTOS

ESCREVER AQUI A MINHA EPÍGRAFE

Resumo

TODO: ESCREVER O RESUMO. DICAS SOBRE COMO ESCREVER O RESUMO ESTÃO LOGO ABAIXO

O resumo deve ressaltar o objetivo, o método, os resultados e as conclusões do documento. A ordem e a extensão destes itens dependem do tipo de resumo (informativo ou indicativo) e do tratamento que cada item recebe no documento original. O resumo deve ser precedido da referência do documento, com exceção do resumo inserido no próprio documento. (...) As palavras-chave devem figurar logo abaixo do resumo, antecidas da expressão Palavras-chave:, separadas entre si por ponto e finalizadas também por ponto. O texto pode conter no mínimo 150 e no máximo 500 palavras, é aconselhável que sejam utilizadas 200 palavras. E não se separa o texto do resumo em parágrafos.

Palavras-chaves: latex. abntex. editoração de texto.

Abstract

TODO: ESCREVER AQUI O ABSTRACT DO TRABALHO This is the english abstract.

Key-words: latex. abntex. text editoration.

Lista de ilustrações

Figura 1 – Interoperabilidade em uma arquitetura baseada no modelo SOA.	31
Figura 2 – Proposta da arquitetura baseada no modelo SOA com o uso de um ESB.	32

Lista de tabelas

Tabela 1 – Cronograma de atividades relacionadas ao TCC 1	34
Tabela 2 – Cronograma de atividades relacionadas ao TCC 2	34

Lista de abreviaturas e siglas

SOA	<i>Service-Oriented Architecture</i>
SOAP	<i>Simple Object Access Protocol</i>
REST	<i>Representational State Transfer</i>
ESB	<i>Enterprise Service Bus</i>

Lista de símbolos

Γ Letra grega Gama

Sumário

1	INTRODUÇÃO	25
1.1	Objetivos	25
1.1.1	Objetivos Geral	25
1.1.2	Objetivos específicos	25
1.1.3	Questão de pesquisa	25
1.2	Motivação	25
1.3	Metodologia	25
1.3.1	Classificação da pesquisa	25
1.3.2	Referencial teórico	26
1.3.3	Proposta	26
1.3.4	Engenharia de software	26
1.4	Estrutura da Monografia	26
2	REFERENCIAL TEÓRICO	27
2.1	Section 1 - Título	27
2.2	Engenharia de Software	27
2.2.1	Conceito de ESW adotados	27
3	A PROPOSTA	29
3.1	Introdução	29
3.2	Proposta de arquitetura	30
3.2.1	Requisitos	30
3.2.2	A arquitetura	30
3.2.2.1	Ferramentas ESB	33
3.2.3	Protocolo de comunicação	33
3.3	Metodologia	33
3.4	Cronograma de execução	33
4	DESENVOLVIMENTO DA PROPOSTA	35
4.1	Descrição do projeto	35
4.2	Proposta	35
4.3	O sistema	35
4.4	Definição das tecnologias	35
4.5	Cronograma de desenvolvimento	35
5	CONSIDERAÇÕES FINAIS	37

REFERÊNCIAS	39
-------------------	----

1 INTRODUÇÃO

ESCREVER O INICIO AQUI. INTRODUÇÃO DEVE SER ABRANGENTE.

1.1 Objetivos

São apresentados nessa seção os objetivos gerais, objetivos específicos e a questão de pesquisa.

1.1.1 Objetivos Geral

DESCREVER AQUI O OBJETIVO GERAL.

1.1.2 Objetivos específicos

Os objetivos deste trabalho são:

- DESCREVER OS OBJETIVOS ESPECÍFICOS.

1.1.3 Questão de pesquisa

EXPOR QUESTÃO DE PESQUISA.

1.2 Motivação

EXPOR MOTIVAÇÃO PARA A REALIZAÇÃO DO TRABALHO.

1.3 Metodologia

Essa seção apresenta a metodologia que será usada no desenvolvimento desse trabalho:

1.3.1 Classificação da pesquisa

CLASSIFICAR A PESQUISA.

1.3.2 Referencial teórico

DESCREVER COMO A PESQUISA FOI REALIZADA E ONDE O REFERENCIAL TEÓRICO ESTA CONTIDO NO DOCUMENTO.

1.3.3 Proposta

DESCREVER A PROPOSTA.

1.3.4 Engenharia de software

DESCREVER AS PRÁTICAS DE ENGENHARIA DE SOFTWARE ADOTADAS PARA O DESENVOLVIMENTO DO TRABALHO.

1.4 Estrutura da Monografia

DESCREVER COMO O DOCUMENTO ESTA ESTRUTURADO.

2 Referencial Teórico

"Este capítulo tem como objetivo apresentar o referencial teórico que embasa a pesquisa deste trabalho, incluindo conceitos abordados"

2.1 Section 1 - Titulo

2.2 Engenharia de Software

2.2.1 Conceito de ESW adotados

3 A PROPOSTA

Este capítulo apresenta a proposta do trabalho de conclusão de curso, exibindo detalhes da implementação a ser realizada.

3.1 Introdução

Avanços tecnológicos e a criação de linguagens de programação, diferentes técnicas e paradigmas e outros conceitos relacionados ao desenvolvimento de software contribui para que a necessidade de interação entre estes elementos seja emergente, de modo a viabilizar a construção de sistemas cada vez mais robustos e inteligentes. Esta interação entre elementos de software não consistem de aplicações robustas que executam todas as suas atividades de forma independente de outras aplicações: cada vez mais, diversos sistemas de software interagem com outros, podendo ser estes desenvolvidos tomando como base outros paradigmas ou escritos em outras linguagens de programação e utilizando-se diferentes técnicas.

A fim de suprir esta necessidade de interação entre sistemas diversos, foi criado um modelo arquitetural conhecido como Arquitetura Baseada em Serviços (ou *Service-Oriented Architecture* - SOA). Este modelo arquitetural utiliza o conceito de serviço como uma unidade que representa uma funcionalidade do sistema, além de trazer consigo os conceitos de interoperabilidade, flexibilidade e baixo acoplamento entre os diversos sistemas ou serviços.

Para este trabalho de conclusão de curso, a proposta consiste em desenvolver uma arquitetura baseada no modelo SOA para um ambiente virtual, propiciando que diversos trabalhos já desenvolvidos e também aqueles em desenvolvimento não sejam mais "engavetados". Por meio do uso do modelo SOA, será possível integrar tais aplicações, ou serviços nos moldes dos termos de SOA, de modo que estas possam trocar dados e fazer uso do serviço disponibilizado por outras, independentemente das tecnologias utilizadas para o desenvolvimento das mesmas. Também faz parte da proposta, o estabelecimento de um protocolo de comunicação entre as aplicações, bem como o padrão de comunicação a ser utilizado.

Desta forma, será viável a disponibilização à sociedade de uma plataforma virtual que conterà os trabalhos realizados dentro da universidade, sejam oriundos de projetos de conclusão de curso, disciplinas ou projetos de extensão e pesquisa.

3.2 Proposta de arquitetura

3.2.1 Requisitos

A partir da necessidade identificada de disponibilizar aplicações que foram desenvolvidas, bem como aquelas que estão em desenvolvimento e que serão desenvolvidas, através de uma plataforma virtual único, algumas das principais características arquiteturais deste ambiente que influenciaram na escolha do modelo arquitetural para a construção da plataforma são:

- A comunicação entre as aplicações deve permitir a troca de dados independentemente das tecnologias utilizadas para seu desenvolvimento.
- O acoplamento entre aplicações deve ser o mínimo possível.
- Extensibilidade, permitindo que novas aplicações/componentes sejam inseridas à plataforma.
- Escalabilidade, fornecendo suporte para que diversas aplicações (ou componentes) sejam aderidas à plataforma.
- Flexibilidade, possibilitando a extensão da plataforma sem que a arquitetura original seja modificada drasticamente.

A partir das características acima, foi proposto o uso do modelo arquitetural SOA - *Service-Oriented Architecture* -, pois desta forma a plataforma virtual terá conhecimento sobre as aplicações por meio das interfaces disponibilizadas, mas não precisará ter conhecimento sobre como ou quais tecnologias foram utilizadas para o desenvolvimento das aplicações. As aplicações neste contexto também podem ser denominadas serviços ou funcionalidades da plataforma.

3.2.2 A arquitetura

A proposta de arquitetura a ser implementada faz uso da abordagem "*Hub-and-spoke*", onde a interface de comunicação entre os serviços é única e pode ser realizada com o uso de um barramento de serviços ou um Enterprise Service Bus (ESB).

O barramento de serviços é um recurso a ser utilizado na implementação da arquitetura baseada no modelo SOA para facilitar a troca entre mensagens entre as aplicações - ou serviços. Este barramento é uma ferramenta que implementa funcionalidades que roteiam as mensagens entre os usuários e provedores de um determinado serviço, transformam as mensagens e os dados para o formato aceito pelas aplicações e aceita protocolos múltiplos de comunicação através de adaptadores. Na arquitetura proposta, o protocolo

de comunicação será padronizado e unificado e apenas as outras características do barramento de serviços serão utilizadas.

Sendo interoperabilidade um dos requisitos relevantes para a escolha do modelo arquitetural, o barramento de serviços é visto como um recurso que pode ser utilizado para ajudar a promover a interoperabilidade na arquitetura definida e na validação de políticas e critérios de segurança a serem definidas posteriormente.

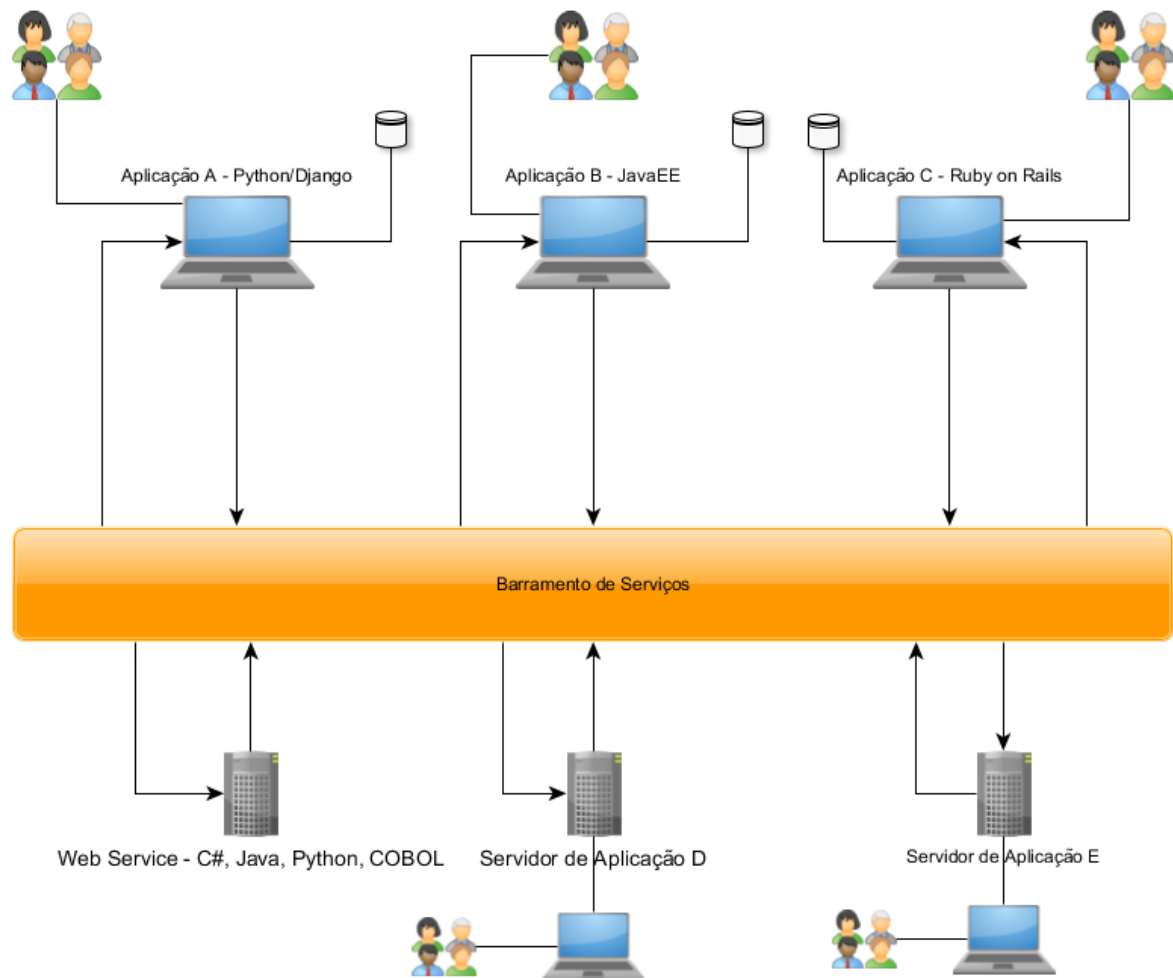


Figura 1: Interoperabilidade em uma arquitetura baseada no modelo SOA.

A figura acima visa demonstrar a interoperabilidade em uma arquitetura baseada no modelo SOA: é possível inferir que as diversas aplicações fazem a requisição dos serviços disponíveis por meio do uso do barramento de serviços, que também pode ser interpretado como um barramento de aplicações. As aplicações podem ser desenvolvidas utilizando-se tecnologias e paradigmas distintos e a troca de dados entre elas se dará via mensagens de requisição e resposta entre as aplicações usuário, que requisitam operações dos serviços, e os serviços, responsáveis por processar as requisições e fornecer uma resposta.

A figura também exibe um fato interessante na arquitetura proposta: não é ne-

cessário que um serviço seja apenas um serviço hospedado em um servidor. As aplicações poderão operar tanto em modo *standalones*, sendo executadas de forma independente dos outros serviços ou aplicações, quanto como um serviço para a plataforma virtual unificada ou para outras aplicações que tenham conhecimento da existência e do protocolo de uso deste serviço.

O ESB é uma ferramenta que fornece as funcionalidades do barramento de serviços. Seu irá garantir que, sempre que este estiver ativo, as requisições que serão realizadas sempre terão uma resposta, mesmo sendo algo que indique a inatividade do serviço requerido ou a não autorização para acesso à este. Ao adicionar um novo serviço à arquitetura utilizando o ESB, deverão ser especificados tanto os procedimentos quando uma nova requisição for feita quanto aqueles passos que conduzem ao envio de uma resposta de acesso ao serviço especificado, sendo esta resposta advinda do próprio serviço ou uma resposta padrão, também especificada, em caso de falha ou sucesso ao executar a requisição.

Com base nos requisitos essenciais levantados e no estudo realizado sobre o modelo arquitetural SOA, o modelo proposto pode ser visto na imagem abaixo:

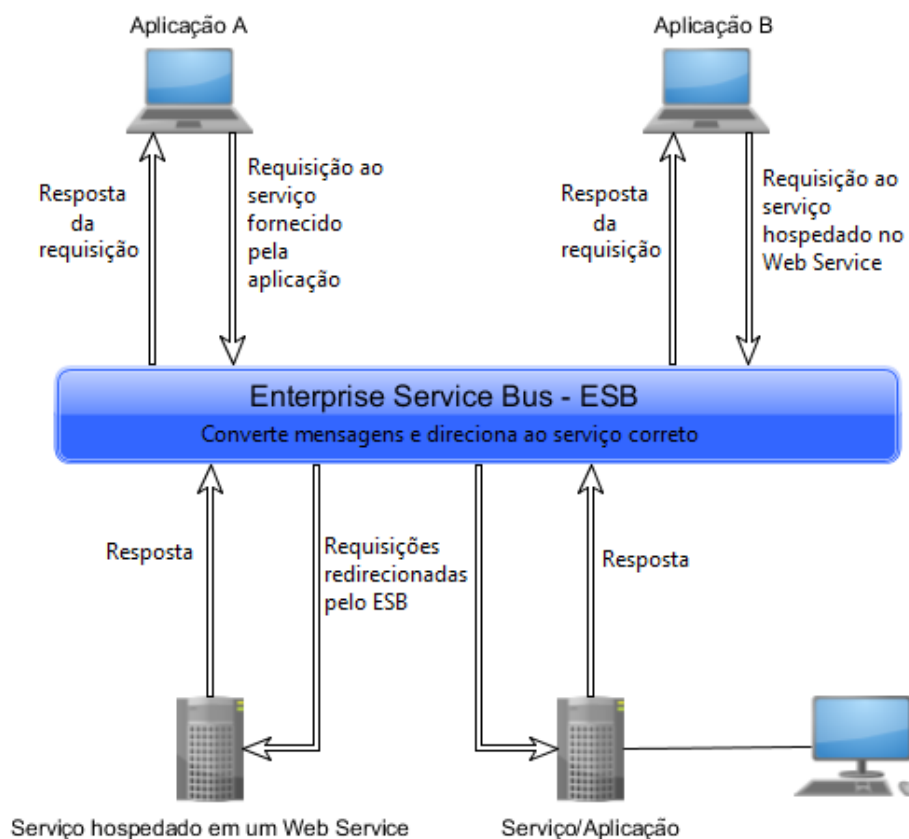


Figura 2: Proposta da arquitetura baseada no modelo SOA com o uso de um ESB.

A comunicação entre aplicações e serviços deverão seguir o padrão de protocolo também definido neste documento, afim de manter uma regra de execução na troca de

informações. O protocolo também facilitará a adição de um novo serviço à arquitetura no que diz respeito aos procedimentos de transformação dos dados e adaptação entre tecnologias e protocolos de comunicação adotados.

3.2.2.1 Ferramentas ESB

Existem diversas ferramentas deste tipo disponíveis e em uso por grandes organizações, tais como JBoss ESB, Mule ESB, Zato e WSO2 ESB. Algumas destas ferramentas já foram levantadas, e, sendo o ESB um elemento importante para a implementação aqui proposta, uma análise destas ferramentas com junto a uma avaliação baseada em critérios ainda não estabelecidos a fim de eleger uma ferramenta deste tipo para apoiar o desenvolvimento do trabalho.

3.2.3 Protocolo de comunicação

DESCREVER O PROTOCOLO, JUSTIFICAR O USO E ESTABELECEER OS PADRÕES A SEREM UTILIZADOS — USAR O REST E DESCRVER COMO ELE VAI SER UTILIZADO, QUAL O PADRÃO ESTABELECIDO DO PROTOCOLO, Formato de mensagens, DESENHAR O DIAGRAMA DE SEQUENCIA E TALS—

3.3 Metodologia

3.4 Cronograma de execução

Para o desenvolvimento da proposta e completude da memsa com sucesso, foram criados dois cronogramas distintos para as fases de estudo e elaboração da proposta e para a fase de prova de conceito e implementação da proposta desenvolvida. Os cronogramas contém a descrição das atividades realizadas, bem como os prazos relacionados a cada atividades e podem ser visualizados nas tabelas abaixo.

Tabela 1: Cronograma de atividades relacionadas ao TCC 1

Atividade	Jan	Fev	Mar	Abr	Mai	Jun
Identificação da necessidade	X					
Leituras sobre modelos arquiteturais	X					
Pesquisa sobre o modelo arquitetural mais adequado	X	X				
Pesquisa sobre trabalhos relacionados	X	X				
Levantamento superficial de ferramentas ESB		X	X			
Organização das referências			X			
Escrita formal da proposta				X		
Estabelecimento de metodologia				X		
Escrita do TCC 1					X	X

Tabela 2: Cronograma de atividades relacionadas ao TCC 2

Atividade	Jul	Ago	Set	Out	Nov	Dez
Análise de ferramentas a serem utilizadas	X					
Implantação da ferramenta escolhida		X				
Adaptação de uma aplicação já desenvolvida			X			
Implementação do uso da ferramenta com a aplicação adaptada				X		
Escrita do TCC 2					X	X

4 Desenvolvimento da Proposta

4.1 Descrição do projeto

4.2 Proposta

4.3 O sistema

4.4 Definição das tecnologias

4.5 Cronograma de desenvolvimento

5 Considerações finais

Referências

- Adaptive Ltd et al. *Service oriented architecture Modeling Language (SoaML) - Specification for the UML Profile and Metamodel for Services (UPMS)*. OMG - Object Management Group, 2009. Disponível em: <<http://www.uio.no/studier/emner/matnat/ifi/INF5120/v10/undervisningsmateriale/>>. Nenhuma citação no texto.
- BASS, L.; CLEMENTS, P. C.; KAZMAN, R. *Software Architecture in Practice*. 2. ed. [S.l.]: Addison-Wesley Professional, 2003. ISBN 0-321-15495-9. Nenhuma citação no texto.
- BELQASMI, F. et al. SOAP-Based vs. RESTful Web Services: A Case Study for Multimedia Conferencing. *IEEE Internet Computing*, v. 16, n. 4, p. 54–63, jul. 2012. ISSN 1089-7801. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6197173>>. Nenhuma citação no texto.
- BIANCO, P.; KOTERMANSKI, R.; MERSON, P. *Evaluating a Service-Oriented Architecture*. [S.l.], 2007. Disponível em: <<http://www.sei.cmu.edu/library/abstracts/reports/07tr015.cfm>>. Nenhuma citação no texto.
- BIANCO, P. et al. *Architecting Service-Oriented Systems*. Pittsburgh, PA, 2011. Disponível em: <<http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=9829>>. Nenhuma citação no texto.
- CHEN, H. M.; KAZMAN, R.; PERRY, O. From Software Architecture Analysis to Service Engineering: An Empirical Study of Methodology Development for Enterprise SOA Implementation. *IEEE Transactions on Services Computing*, v. 3, n. 2, p. 145–160, jun. 2010. ISSN 1939-1374. Disponível em: <<http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=5467023>>. Nenhuma citação no texto.
- DAI, W. et al. Bridging Service-Oriented Architecture and IEC 61499 for Flexibility and Interoperability. *IEEE Transactions on Industrial Informatics*, v. 11, n. 3, p. 771–781, jun. 2015. ISSN 1551-3203, 1941-0050. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7086296>>. Nenhuma citação no texto.
- DIKMANS, L.; LUTTIKHUIZEN, R. v. *SOA Made Simple*. BIRMINGHAM - MUMBAI: Packt Publishing, 2012. ISBN 978-1-84968-416-3. Nenhuma citação no texto.
- ERL, T. *Service-Oriented Architecture: A Field Guide to Integrating XML and Web Services*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2004. ISBN 0-13-142898-5. Nenhuma citação no texto.
- ERL, T. *Service-Oriented Architecture: Concepts, Technology, and Design*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2005. ISBN 0-13-185858-0. Nenhuma citação no texto.
- ERL, T. Computação orientada a serviços e SOA. In: GAMA, F. C. N. d.; BARBOSA, R. d. C. (Ed.). *SOA: Princípios de Design de Serviços*. São Paulo: Pearson Prentice Hall, 2009. p. 306. ISBN 978-85-7605-189-3. Nenhuma citação no texto.

- ERL, T. Orientação a serviços. In: GAMA, F. C. N. d.; BARBOSA, R. d. C. (Ed.). *SOA: Princípios de Design de Serviços*. São Paulo: Pearson Prentice Hall, 2009. p. 306. ISBN 978-85-7605-189-3. Nenhuma citação no texto.
- ERL, T. *SOA: Princípios de Design de Serviços*. 1. ed. São Paulo: Pearson Prentice Hall, 2009. ISBN 978-85-7605-189-3. Nenhuma citação no texto.
- ESFAHANI, F. S. et al. Adaptable Decentralized Service Oriented Architecture. *Journal of Systems and Software*, v. 84, n. 10, p. 1591–1617, out. 2011. ISSN 01641212. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0164121211000744>>. Nenhuma citação no texto.
- EVDEMON, J. *Princípios do design de serviço: padrões e antipadrões de serviço*. 2005. Disponível em: <<https://msdn.microsoft.com/pt-br/library/ms954638.aspx>>. Nenhuma citação no texto.
- FERREIRA, A. Apresentação Prezi, *SOA vs ESB*. 2013. Disponível em: <https://prezi.com/na8_yc87is6s/soa-vs-esb/>. Nenhuma citação no texto.
- JANSEN, A.; BOSCH, J. Software Architecture as a Set of Architectural Design Decisions. In: . IEEE, 2005. p. 109–120. ISBN 978-0-7695-2548-8. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1620096>>. Nenhuma citação no texto.
- JÚNIOR, D.; VIEIRA, M. *Uma arquitetura para aprendizagem colaborativa utilizando a integração web e TV digital integrativa*. Tese (text) — Universidade Federal de Alagoas, set. 2012. Disponível em: <<http://www.repositorio.ufal.br/handle/riufal/849>>. Nenhuma citação no texto.
- JOSUTTIS, N. *Soa in Practice: The Art of Distributed System Design*. [S.l.]: O'Reilly Media, Inc., 2007. ISBN 0-596-52955-4. Nenhuma citação no texto.
- KAZMAN, R. The essential components of software architecture design and analysis. In: . IEEE, 2005. p. 1 pp. ISBN 978-0-7695-2465-8. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1607129>>. Nenhuma citação no texto.
- LEWIS, G. *Getting Started with Service-Oriented Architecture (SOA) Terminology*. Software Engineering Institute Carnegie Mellon University, 2010. Disponível em: <http://www.sei.cmu.edu/library/assets/whitepapers/SOA_Terminology.pdf>. Nenhuma citação no texto.
- LINTHICUM, D. et al. *Service Oriented Architecture (SOA) in the Real World*. [S.l.]: Microsoft Corporation, 2007. Nenhuma citação no texto.
- LÓPEZ-SANZ, M. et al. Modelling of Service-Oriented Architectures with UML. *Electronic Notes in Theoretical Computer Science*, v. 194, n. 4, p. 23–37, abr. 2008. ISSN 15710661. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S1571066108002028>>. Nenhuma citação no texto.
- MALLOY, B. et al. Improving the predictable assembly of service-oriented architectures. *IEEE Software*, v. 23, n. 2, p. 12–15, mar. 2006. ISSN 0740-7459. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1605172>>. Nenhuma citação no texto.

MUELLER, J. *Understanding SOAP and REST Basics And Differences*. 2013. Disponível em: <<http://blog.smartbear.com/apis/understanding-soap-and-rest-basics/>>.

Nenhuma citação no texto.

NICKULL, D. et al. *Service Oriented Architecture (SOA) and Specialized Messaging Patterns*. San Jose, CA, USA, 2007. Nenhuma citação no texto.

O que é SOA e por que usá-la? 2010. Disponível em: <<http://www.celtainformatica.com.br/noticias/o-que-e-soa-e-por-que-usa-la>>. Nenhuma citação no texto.

OLIVEIRA, M.; NAVARRO, R. Interoperabilidade em SOA: Desafios e Padrões. *SOA na prática*. Disponível em: <<http://www.univale.com.br/unisite/mundo-j/artigos/37Interoperabilidade.pdf>>. Nenhuma citação no texto.

ONOE, A. Y. *Proposta de governança SOA utilizando capacidades dinâmicas: uma aplicação em centro de comunicação digital universitário*. Tese (text) — Universidade de São Paulo, nov. 2010. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/3/3141/tde-19012011-095120/>>. Nenhuma citação no texto.

PARK, S.; CHOI, J.; YOO, H. Integrated Model of Service-Oriented Architecture and Web-Oriented Architecture for Financial Software. *Journal of Information Science and Engineering*, v. 28, n. 5, p. 925–939, 2012. Disponível em: <http://www.iis.sinica.edu.tw/page/jise/2012/201209_07.pdf>. Nenhuma citação no texto.

PEREDO, R.; PEREDO, I. Software Architecture and Design with Semantic Web Technologies in Virtual Learning Environments. *Issues in Information Systems - A Journal of IACIS*, v. 15, n. 2, p. 190–196, 2014. Disponível em: <http://iacis.org/iis/2014/111_iis_2014_190-196.pdf>. Nenhuma citação no texto.

PEREIRA, M. Z. *PSOA: um framework de práticas e padrões SOA para projetos DDS*. Tese (masterThesis) — Pontifícia Universidade Católica do Rio Grande do Sul, Rio Grande do Sul, 2011. Disponível em: <<http://hdl.handle.net/10923/1658>>. Nenhuma citação no texto.

ROSCHELLE, J.; KAPUT, J. Educational Software Architecture And Systemic Impact: The Promise Of Component Software. *Journal of Educational Computing Research*, v. 14, n. 3, p. 217–228, 1996. Disponível em: <<https://www.sri.com/sites/default/files/publications/imports/EdSoftwareArch.pdf>>. Nenhuma citação no texto.

ROZLOG, M. *REST e SOAP: Usar um dos dois ou ambos?* 2013. Disponível em: <<http://www.infoq.com/br/articles/rest-soap-when-to-use-each>>. Nenhuma citação no texto.

SANGWAN, R. et al. Integrating a software architecture-centric method into object-oriented analysis and design. *Journal of Systems and Software*, v. 81, n. 5, p. 727–746, maio 2008. ISSN 01641212. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0164121207001872>>. Nenhuma citação no texto.

SILVA, E. *Introdução aos Conceitos do WSO2 ESB*. 2014. Disponível em: <<http://pt.slideshare.net/edgarsilva/introduo-aos-conceitos-do-wso2-esb>>. Nenhuma citação no texto.

SIRIWARDENA, P. *Enterprise Integration with WSO2 ESB*. 1. ed. BIRMINGHAM - MUMBAI: Packt Publishing, 2013. ISBN 978-1-78328-019-3. Nenhuma citação no texto.

Software Engineering Institute - Carnegie Mellon University. *Software Architecture*. Disponível em: <<http://www.sei.cmu.edu/architecture/>>. Nenhuma citação no texto.

STEENDEREN, M. v.; DYK, P. v. Standard Object Access Protocol (SOAP). *InterWord Communications - South African Journal of Information Management*, v. 2, n. 2/3, set. 2000. ISSN 1560-683X. Disponível em: <<http://www.sajim.co.za/index.php/SAJIM/article/viewFile/106/103>>. Nenhuma citação no texto.

VANTAGENS e Desvantagens de SOA. 2013. Disponível em: <<http://www.devmedia.com.br/vantagens-e-desvantagens-de-soa/27437>>. Nenhuma citação no texto.

WSO2 Enterprise Service Bus. Disponível em: <<http://wso2.com/products/enterprise-service-bus/>>. Nenhuma citação no texto.