

# USO DE ALGORITMOS DE BUSCA PARA RECOMENDAR A MIGRAÇÃO DE ARQUITETURA DE SOFTWARE MONOLÍTICA PARA ARQUITETURA ORIENTADAS A MICROSSERVIÇOS

Janaína Aparecida de Freitas<sup>1</sup>, Thelma Elita Colanzi Lopes<sup>2</sup>

<sup>1</sup>Acadêmica do Mestrado em Ciência da Computação, Universidade Estadual de Maringá. japfreitas@gmail.com

<sup>2</sup>Orientadora, Doutora, DIN, Universidade Estadual de Maringá. thelma@din.uem.br

## RESUMO

Algoritmos baseados em busca podem desempenhar um importante papel quando aplicados a atividades de recomendação para migração de arquiteturas de software e podem ajudar os arquitetos de software a encontrarem boas soluções, ou próximas do ideal, de forma automatizada. Considerando este contexto, este projeto de pesquisa tem como o objetivo propor o uso de algoritmos de busca para recomendar a migração de arquitetura de software monolítica para arquitetura orientada a microsserviços. A pesquisa justifica-se, pois aborda temáticas atuais e também visa mostrar que é possível recomendar a migração através de algoritmos de busca a de arquitetura monolítica para baseada em microsserviços através de pesquisa, de estudos e de fomento a novas tecnologias, para tornar viável uma solução automática para este propósito.

**PALAVRAS-CHAVE:** Engenharia de Software; Problemas de Otimização, Arquiteturas de Software.

## 1 INTRODUÇÃO

A Engenharia de Software, como uma disciplina de engenharia, é um campo onde existem muitos problemas e os algoritmos de busca têm sido aplicados em diversas áreas para a resolução desses problemas complexos, tais como avaliação da qualidade, teste de software, projetos de software entre outros. Tais problemas complexos podem ser descritos pela busca da solução em um espaço onde tenha possíveis soluções e que permitem buscar configurações em que as características desejadas sejam otimizadas (FREITAS et al., 2009).

O campo de pesquisa conhecido como Engenharia de Software Baseada em Busca (em inglês *Search Based Software Engineering* (SBSE)) vem sendo estudado com o objetivo de encontrar soluções apropriadas para os diferentes problemas de otimização de diversas áreas de engenharia de software (HARMAN e JONES, 2001). As técnicas exploradas pela SBSE permitem encontrar soluções suficientemente boas entre as possíveis soluções, de acordo com métricas de adequação adotadas.

Segundo Bass et al (1998) arquitetura de software de um programa é a estrutura do sistema, que compreendem elementos de software, as propriedades visíveis externamente desses elementos e as relações entre eles. A arquitetura é constituída por decisões, algumas grandes e outras pequenas, e a maioria precisa ser tomada no início do projeto e podem ter impacto profundo em outras fases do projeto que estão por vir. Projetar arquiteturas de software é uma tarefa crítica e altamente exigente (BASS et al., 1998).

O estudo da arquitetura de software evoluiu pela observação dos princípios do projeto que os engenheiros seguem e as ações que eles tomam quando se trabalha em sistemas reais. É uma tentativa de abstrair os pontos comuns inerentes ao projeto do sistema, como as atividades, conceitos, métodos, abordagens e resultados esperados (BASS et al., 1998).

Alguns padrões representam soluções conhecidas para problemas de desempenho, outros se prestam bem a sistemas de alta segurança, outros ainda foram usados com

sucesso em sistemas de alta disponibilidade. A escolha de um padrão arquitetural é frequentemente a primeira grande escolha do arquiteto de projeto (BASS et al., 1998).

Padrões arquiteturais são criados com o objetivo de melhorar a qualidade das aplicações desenvolvidas, neles se destacam dois modelos: monolítico e micro serviço. Atualmente a arquitetura monolítica é a mais tradicional e a mais utilizada para o desenvolvimento de aplicações devido à sua popularidade no mercado. Em uma arquitetura monolítica os serviços que compõem a aplicação são organizados de forma lógica no mesmo código fonte e unidade de instalação, sempre com um alto acoplamento entre os componentes (SOUZA, 2016).

As arquiteturas monolíticas têm uma limitação relevante, que é o fato de todos os módulos pertencerem a um único e mesmo programa. E isso faz, com que, ao passar do tempo, conforme novas funcionalidades vão sendo adicionadas, se torne muito grande e complexos, impactando nas atividades de construção e implantação. Para resolver essas limitações, surgiu a arquitetura de microsserviços. (SOUZA, 2016).

O conceito de microsserviços vem se tornando forte e ganhando destaque como um estilo de desenvolver software cada vez mais atraente (FOWLER E LEWIS, 2014). Conforme Fowler e Lewis (2014), o termo "Arquitetura de microsserviços" surgiu nos últimos anos para descrever uma maneira particular de projetar aplicativos de software como suítes de serviços implementáveis independentemente. Newman (2015) define microsserviços como "pequenos serviços independentes que trabalham juntos, mas cada um como uma entidade separada".

Segundo Fowler e Lewis (2014) quase todas as histórias bem-sucedidas do uso de arquiteturas baseadas em micro serviço começaram com uma arquitetura monolítica que ficou muito grande e acabou sendo quebrado. E em alguns casos, o sistema que foi desenvolvido como um sistema de micro serviços a partir do zero, acabou com sérios problemas (FOWLER E LEWIS, 2014).

A migração de arquiteturas monolíticas para arquitetura baseadas em microsserviços apresenta muitos benefícios, como gestão diferente da disponibilidade, otimização da escalabilidade, capacidade de utilizar tecnologias diferentes e evitar o aprisionamento tecnológico, implantação (*deploy*), redução do tempo de lançamento no mercado e compreensão melhor da base de código (CARVALHO, 2017).

As dificuldades em torno da decomposição de um sistema monolítico em pequenos serviços, e monitorar e gerenciar esses serviços estão entre os fatores que tornam a migração para microsserviços uma tarefa não trivial e complexa e muitas vezes imprecisa. (CARVALHO, 2017). E a realização de tal atividade pode envolver diferentes fatores que muitas vezes são complexos e custosos.

## 2 MATERIAIS E MÉTODOS

Serão realizados mapeamentos sistemáticos buscando identificar, avaliar e analisar publicações de estudos similares que já tenham sido realizados pela comunidade científica. Para este mapeamento se definiu como fontes de pesquisa o Portal da Capes, o Portal Scielo, o Google Acadêmico, Elsevier, IEEE, procurando artigos de revistas patrocinadas por instituições reconhecidas na área, com o objetivo de colocar o pesquisador em contato direto com todo o material já escrito sobre a temática.

Neste projeto, pretende-se utilizar abordagens para o estudo do mapeamento sistemático em ES, como a proposta de Petersen *et al.* (2015), procurando estabelecer os objetivos e escopo da pesquisa, estratégias de busca para realizar a seleção e triagem dos artigos relevantes e por fim, extração de dados e mapeamentos dos resultados alcançados.

Para os estudos experimentais, serão utilizadas abordagens da Engenharia de Software Experimental onde um modelo será sugerido, com variáveis relacionadas com o objeto de estudo, contexto de realização, hipóteses, tabulação, análise e resultado dos

dados (TRAVASSOS *et al.*, 2002). A partir dos resultados obtidos com a pesquisa, serão elaborados artigos para divulgação através da submissão a eventos e periódicos científicos relacionados ao tema.

### 3 RESULTADOS E DISCUSSÕES

Tripoli e Carvalho (2016), apresentam um levantamento de características, benefícios e desvantagens da arquitetura microsserviços em relação à arquitetura monolítica que impactam na decisão do uso desta arquitetura. Alguns estudos comparativos comparativo entre as arquiteturas de software de microsserviços e às arquiteturas monolíticas (SOUZA, 2016; LUCIO 2017; RIBEIRO, 2017). Carvalho (2017) propõe uma abordagem de migração de aplicações monolíticas para o padrão de microsserviços, onde é contemplado todas as atividades que envolve o ciclo de desenvolvimento de software focada em dar suporte à migração de aplicações que já estão em produção. A otimização de Arquitetura de Software utilizando sistema de colônia de formigas, é usada para propor diretrizes e recomendações a fim de identificar elementos arquiteturais, recuperar e otimizar arquiteturas (MEDEIROS, 2016).

No entanto, foram encontrados apenas estudos com foco em conceitos de microsserviços, comparações entre as arquiteturas, estudo de caso para o desenvolvimento de aplicações de arquiteturas baseada em microsserviços e uma propõem a migração, mas não faz uso de algoritmos de busca para recomendar esta migração entre arquitetura de software.

### 4 CONSIDERAÇÕES FINAIS

Como o projeto está na fase inicial, até o presente momento foi possível realizar pesquisa sobre temas correlatos ao projeto proposto. Esperamos poder traçar uma linha de pensamento e organização para ao final, termos resultados que consolidem e mostrem que é possível o uso de algoritmos de busca para recomendar a migração de arquitetura de software monolítica para arquitetura orientada a microsserviços.

### REFERÊNCIAS

BALALAIÉ, A., HEYDARNOORI, A., and JAMSHIDI, P. **Migrating to cloud-native architectures using microservices: An experience report.** In **Advances in Service-Oriented and Cloud Computing -Workshops of ESOC 2015, Taormina, Italy, September 15-17, 2015, Revised Selected Papers**, pages 201–215.

BASS, Len; CLEMENTS, Paul; KAZMAN, Rick. **Software architecture in practice / — 2nd ed.**, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1998.

CARVALHO, Luis Heustákio Lima. **Uma Abordagem de Migração para Arquitetura Microservices a partir de Aplicações Monolíticas em Produção**, Dissertação (Mestrado Acadêmico) - Universidade de Fortaleza. Programa de Mestrado Em Informática Aplicada, Fortaleza, 2017.

FOWLER, M.; LEWIS, J. **Microservices.** 2014. Disponível em [:http://martinfowler.com/articles/microservices.html](http://martinfowler.com/articles/microservices.html). Acessado em outubro de 2018.

FREITAS, Fabrício Gomes de, MAIA Camila Loiola Brito, COUTINHO Daniel Pinto, CAMPOS Gustavo Augusto Lima de, SOUZA Jerffeson Teixeira de. **Aplicação de**

**Metaheurísticas em Problemas da Engenharia de Software: Revisão de Literatura.** Revista de Sistemas de Informação da FSMA, Ceará, 2009.

HARMAN, M.; JONES, B.. Search based software engineering, **Journal of Information and Software Technology** 43 (14) (2001) 833-839, 2001.

HARMAN, M.; MANSOURI, S. A.; ZHANG, Y. **Search-based software engineering: Trends, techniques and applications.** ACM Comput. Surv., 45(1):11:1–11:61, Dec. 2012.

LUCIO João Paulo Duarte. **Análise Comparativa entre Arquitetura Monolítica e de Microsserviços.** Departamento de Informática e Estatística Universidade Federal de Santa Catarina (UFSC), 2017.

MEDEIROS, Maria Affonso. **Otimização de Arquitetura de Software Utilizando Sistema de Colônia de Formigas.** Universidade Tecnológica Federal do Paraná (UTFP), Campo Mourão, 2016.

NEWMAN, S. **Building Microservices Designing Fine-Grained Systems.** [S.l.]: O'Reilly Media, Inc., 2015. v. 2015.

PETERSEN K, VAKKALANKA S, KUZNIARZ L. **Guidelines for conducting systematic mapping studies in software engineering: An update.** *Inf. Softw. Technol.*. 2015; 64:p. 1–18.

RIBEIRO, Bruno Rafael Costa. **Estudo comparativo entre arquiteturas monolíticas e de micro serviços.** Tese de Mestrado, Sistemas e Tecnologias de Informação Para as Organizações. Instituto Politécnico de Viseu. IPV - ESTGV. 2017.

RICHARDSON, C.; SMITH, F. **Microservices From Design to Deployment.** [S.l.]: NGINX, Inc., 2016. v. 2016.

SOUZA, Emmanuel Neri de. **Um comparativo de arquiteturas de software de micro serviços em relação às arquiteturas monolíticas em ambientes corporativos.** Curitiba, 2016.

COLANZI, T E., VERGILIO, S. R., ASSUNÇÃO, W. K. G., e POZO, A. **Search based software engineering: Review and analysis of the field in Brazil.** Journal of Systems and Software, 86(4):970 { 984, 2013. Special Issue: Software Engineering in Brazil: Retrospective and Prospective Views.

TRAVASSOS, G. H.; GUROV, D.; AMARAL, E. A. G. D. **Introdução à Engenharia de Software Experimental.** COPPE/UFRJ, Rio de Janeiro, 2002. (Relatório Técnico RT -ES-590/02 do Programa de Engenharia de Sistemas e Computação).

TRIPOLI, Crislaine da Silva; CARVALHO Rodrigo Pimenta. **Microsserviços: características, benefícios e desvantagens em relação à arquitetura monolítica que impactam na decisão do uso desta arquitetura.** II Seminário de Desenvolvimento em SOA com Cloud Computing e Conectividade. Instituto Nacional de Telecomunicações – INATEL; 2016 – ISSN 2447-2352.