

Estudo Exploratório Sobre o Uso da Arquitetura de Microsserviços em Empresas da Cidade de Maringá - PR

Tamires G. B. Deprá¹, Douglas R. Tanno¹, Aline Maria M. M. Amaral¹,
Thelma E. C. Lopes¹

¹Departamento de Informática – Universidade Estadual de Maringá (UEM)
Maringá – PR – Brasil

{tami.g.bsp,douglas.tanno}@gmail.com, {ammamaral,thelma}@din.uem.br

Resumo. *A arquitetura de microsserviços, embora ainda muito recente, já demonstra grande impacto na indústria de software, onde empresas dos mais variados tamanhos e segmentos vêm adotando esse estilo arquitetural. Neste trabalho buscou-se identificar se as empresas de software da cidade de Maringá - PR utilizam a arquitetura de microsserviços e como a utilizam. Para este propósito, um questionário foi aplicado em uma amostra de empresas da cidade de Maringá - PR. Com os dados resultantes da aplicação do questionário, foi possível observar que um número expressivo de empresas utilizam a arquitetura de microsserviços e que as mesmas aplicam as boas práticas observadas na literatura para seu uso.*

Abstract. *The microservice architecture, although still very recent, already shows great impact in the software industry, where companies of the most varied sizes and segments have been adopting this architectural style. In this work, we aimed to identify whether software companies in the city of Maringá - PR use microservices architecture and how they use it. For this purpose, a questionnaire was applied to a sample of companies in the city of Maringá - PR. Considering the data resulting from the application of the questionnaire, it was possible to observe that a significant number of companies use the microservice architecture and that they apply the good practices observed in the literature for their use.*

1. Introdução

A arquitetura de software de um programa ou sistema computacional é a estrutura ou estruturas do sistema, que abrange os componentes do software, as propriedades externamente visíveis desses componentes e as relações entre eles [Bass et al. 2003].

Segundo [Newman 2015], microsserviços são pequenos serviços autônomos que trabalham juntos. A arquitetura de microsserviços propõe o desenvolvimento de um sistema distribuído, onde cada pequena parte do sistema é um serviço independente, facilitando a criação de novos serviços para acoplamento no sistema e manutenção dos serviços que já se encontram em produção.

Muitas questões com relação a adoção da arquitetura de microsserviços por empresas de desenvolvimento de software ainda estão em aberto, tais como: se esta tecnologia tem um impacto positivo em empresas de pequeno e médio porte; qual estratégia

para implantação está sendo realizada pelas empresas; além das perspectivas futuras com relação ao uso dessa tecnologia. Nesse sentido, o objetivo deste trabalho foi realizar um estudo exploratório sobre a arquitetura de microsserviços, entendendo melhor o seu uso em empresas, por meio da aplicação de um questionário em empresas de desenvolvimento de software da cidade de Maringá - PR. Com base nas respostas do questionário, foi realizada uma análise para determinar se as empresas utilizam essa arquitetura e de que forma elas a utilizam.

Este trabalho está organizado da seguinte forma: a Seção 2 apresenta uma descrição da arquitetura de microsserviço; na Seção 3 o método de pesquisa utilizado no trabalho é descrito; na Seção 4 são detalhados os resultados obtidos com a realização deste trabalho juntamente com a discussão dos mesmos; finalmente, na Seção 5 são apresentadas as conclusões do trabalho, assim como as motivações para a realização de trabalhos futuros.

2. Microsserviços

Microsserviços são pequenos serviços autônomos que trabalham juntos. Além disso, cada microsserviço possui seus próprios dados, sua própria regra de negócio e uma interface bem definida, de modo que cada serviço possa ter seu próprio repositório, controlador de versões, ciclo de vida e controlador de tarefas. Além disso, outra proposta desse estilo arquitetural é que cada microsserviço tenha sua própria equipe de desenvolvimento e implantação. A arquitetura de microsserviços trabalha com extensa granularidade do sistema e alta escalabilidade possibilitando o desenvolvimento em diversas linguagens e plataformas diferentes, com banco de dados distribuídos e até equipes distribuídas [Meloca 2017, Newman 2015].

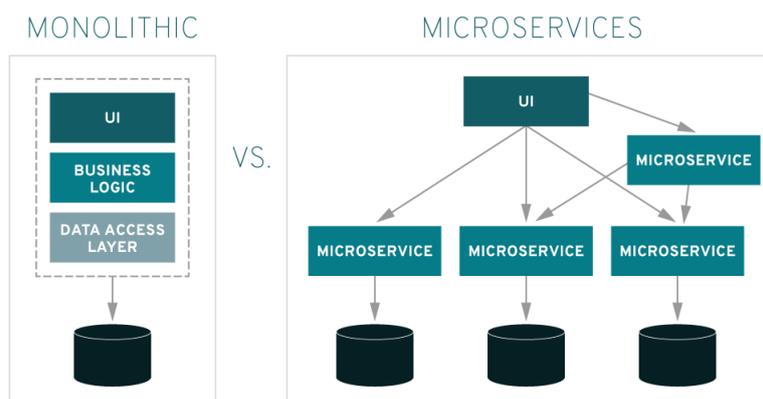


Figura 1. Critérios utilizados para decisão sobre a extração dos microsserviços.

Na Figura 1 pode-se observar um comparativo entre a estrutura de um sistema monolítico e um sistema que utiliza o estilo arquitetural de microsserviços.

A arquitetura de microsserviços é particularmente adequada para infraestruturas de nuvem, pois se beneficia muito da elasticidade ativada pela nuvem e do rápido provisionamento de recursos. Fazer uso da arquitetura de microsserviços, no entanto, não é uma tarefa fácil, pois exige o gerenciamento de uma arquitetura distribuída e seus desafios, que incluem latência e falta de confiabilidade da rede, tolerância a falhas, orquestração

de serviços complexos, consistência de dados e gerenciamento de transações e balanceamento de carga. Infraestruturas em nuvem e novas tecnologias desempenham um papel fundamental no uso de arquiteturas de microsserviços e no gerenciamento dos desafios e complexidades associados [Di Francesco et al. 2019].

É comum comparar a arquitetura de microsserviços com a arquitetura monolítica, pois a arquitetura monolítica é amplamente utilizada e difere em diversos aspectos da arquitetura de microsserviços. Uma aplicação monolítica é feita como uma única unidade lógica executável. A arquitetura de microsserviços deriva da SOA (do inglês *Service Oriented Architecture*) cuja finalidade é reduzir o acoplamento entre os diversos módulos de uma aplicação com o propósito de facilitar sua escalabilidade. Na arquitetura monolítica nenhuma funcionalidade do sistema existe e opera por si só [Fowler and Lewis 2014, Meloca 2017].

Como resultado da adoção da arquitetura de microsserviços espera-se adquirir benefícios, por exemplo, escalabilidade e responsabilidade, mas também pode-se ter alguns malefícios, por exemplo, dificuldade na refatoração e dificuldade na comunicação distribuída, a depender do aspecto sobre o qual se olha, de maneira que há um limiar a ser estudado entre as curvas da arquitetura monolítica e da arquitetura de microsserviços considerando o número de usuários pelo tempo de resposta [Meloca 2017, Newman 2015].

3. Método de Pesquisa

A pesquisa foi conduzida em 4 etapas: Levantamento Bibliográfico, Definição do Método de Avaliação, Coleta de Dados e Avaliação dos Resultados.

Para a realização do estudo exploratório, foi necessário, inicialmente, realizar o levantamento bibliográfico sobre as características da arquitetura de microsserviços, seus benefícios e detalhes de implementação, com objetivo de explorar a aplicação dessa arquitetura na prática por empresas de software da cidade de Maringá-PR e validar os dados a serem obtidos na pesquisa.

A definição do método de avaliação consistiu em eleger o meio mais adequado para coletar os dados de acordo com a proposta do trabalho. Foi definido o desenvolvimento e aplicação de um questionário para realização de um estudo exploratório na cidade de Maringá-PR, contendo perguntas objetivas e discursivas sobre a arquitetura de microsserviços. As perguntas do questionário foram elaboradas nos seguintes formatos: (i) múltipla escolha, restringidas a seleção de uma opção, (ii) tipo sim/não, (iii) tipo seleção, onde é possível ao respondente selecionar mais de uma opção, (iv) e questões discursivas.

A coleta dos dados foi realizada de forma virtual, onde os participantes da pesquisa receberam um *link* para responder ao questionário. O critério de escolha das empresas para participarem da pesquisa foi o de serem indústrias de software localizadas em Maringá-PR, sem outros requisitos como número de funcionários ou ramo do desenvolvimento de software.

A última etapa da pesquisa consistiu na análise e discussão sobre os resultados obtidos com a aplicação do questionário, comparando os dados coletados com os materiais disponíveis na literatura.

O questionário completo desenvolvido nesta pesquisa pode ser consultado cli-

cando aqui.

4. Resultados e discussão

Atualmente em Maringá-PR existem aproximadamente 400 empresas de software. O questionário foi enviado à algumas dessas empresas, dentre as quais 23 participaram da pesquisa. Das 23 empresas, 39,1% utilizam a arquitetura de microsserviços, sendo esse percentual referente a 9 empresas. Considerando as respostas dadas ao questionário pelas empresas participantes da pesquisa são apresentados a seguir: os percentuais correspondentes a cada questão de múltipla escolha (opção única de resposta); o gráfico referente às questões de seleção; e as respostas das questões discursivas.

A primeira questão refere-se ao cargo ocupado pelo respondente na empresa, onde 56,5% são desenvolvedores e 17,4% são analistas. Quanto ao restante dos respondentes, há uma grande variação de cargos dentro da empresa, como: arquiteto, gerente de projetos, coordenador, testador, engenheiro de software, diretor de inovação e gerente de tecnologia.

A questão seguinte é “A empresa se organiza por área de desenvolvimento (interface, negócio, banco de dados, etc.) ou por área do negócio (exemplo: vendas, contabilidade, financeiro, etc.) com times multifuncionais?”. A maior parte das empresas, 65,2%, é dividida em áreas de negócio, 30,4% são divididas por áreas de desenvolvimento e 4,4% por ambas.

A pergunta seguinte é sobre a distribuição das equipes. É comum em empresas que utilizam microsserviços que as equipes de trabalho estejam alocadas em locais diferentes, como observado na fundamentação teórica. Nas empresas de Maringá podemos observar que a maioria, 52,2%, das empresas, possui esse esquema de trabalho, sendo que 47,8% delas não possui.

Em seguida, foi questionado se a empresa adota a arquitetura de microsserviços. Um percentual de 39,1% empresas usa microsserviços, enquanto que 60,9% delas não utiliza esse estilo arquitetural. Após essa pergunta o questionário foi dividido em duas seções diferentes, uma para as empresas que utilizam a arquitetura de microsserviços e outra seção para as empresas que utilizam outras arquiteturas para o desenvolvimento de seus sistemas.

A próxima pergunta do questionário, “Como foi a escolha pela arquitetura de microsserviços?”, já faz parte da seção para as empresas que utilizam a arquitetura de microsserviços. Na Figura 2 são apresentadas todas as opções e seus respectivos percentuais. Pode-se observar que a maioria das empresas, 55,6%, optou pela migração gradual do sistema e 11,1% alegam que o sistema está passando pela migração atualmente, também de forma gradual. O gráfico com as respostas à esta questão pode ser observado na Figura 1. Tais informações corroboram com [Balalaie et al. 2015], que afirmam que a migração do sistema para a arquitetura de microsserviços não é simples e deveria ser realizada de forma incremental e em muitos passos sem afetar o usuário final. Nesse sentido, pode-se observar que as empresas, assim como sugerido na literatura, realizam a migração para a arquitetura de microsserviços de forma gradual.

Na questão seguinte, “Quais critérios para decisão de extração de microsserviços?”, a escalabilidade foi apontada como critério para decisão de extração

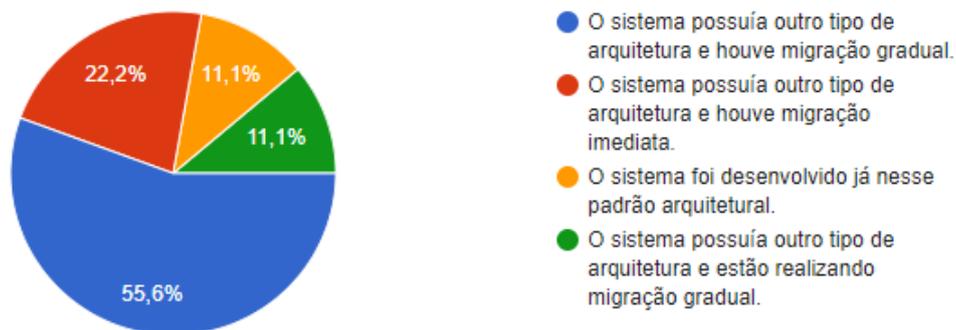


Figura 2. Estratégia para implantação da arquitetura de microsserviços.

por todas as empresas que utilizam a arquitetura de microsserviços. A segunda característica mais apontada foi acoplamento. Na Figura 3, pode-se observar as opções selecionadas pelos respondentes. Segundo [Meloca 2017], dada a adoção da arquitetura de microsserviços, como benefício, dentre outros, espera-se obter um bom nível de escalabilidade do sistema. E para [Newman 2015], uma característica da arquitetura de microsserviços é sua autonomia. Esse autor destaca que toda a comunicação entre os serviços é feita via chamadas através da rede, para forçar a separação entre eles e evitar alto acoplamento. Portanto, assim como o formato da migração, os critérios de extração também vão de encontro ao proposto pela literatura.

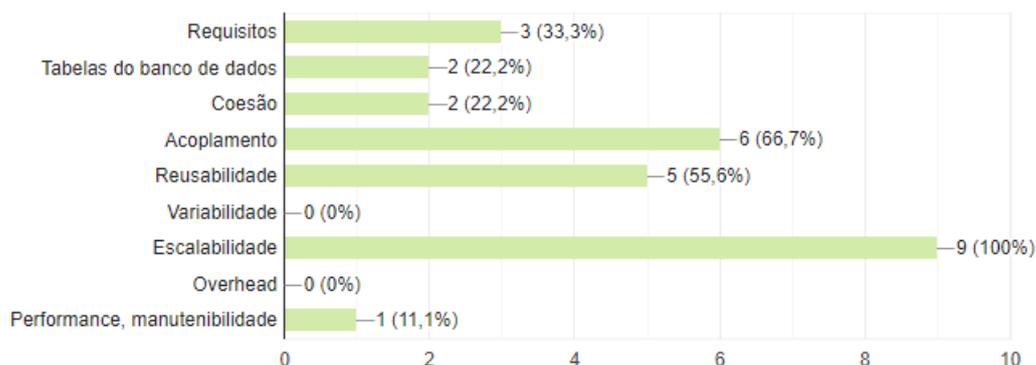


Figura 3. Critérios utilizados para decisão sobre a extração dos microsserviços.

A próxima questão do questionário aplicado refere-se aos *frameworks* utilizados no desenvolvimento de microsserviços. Na Figura 4 pode-se observar as opções selecionadas pelos respondentes, sendo ASP .Net o *framework* mais utilizado, indicado por 4 das 9 empresas que utilizam a arquitetura de microsserviços, e o Spring Boot o segundo mais utilizado, indicado por 3 empresas.

Na Figura 5 são apresentados os resultados referente a questão “Qual o meio de comunicação utilizado entre os microsserviços?”. Pode-se observar que predominantemente é utilizada a comunicação por APIs REST, abrangendo 66,7% e a

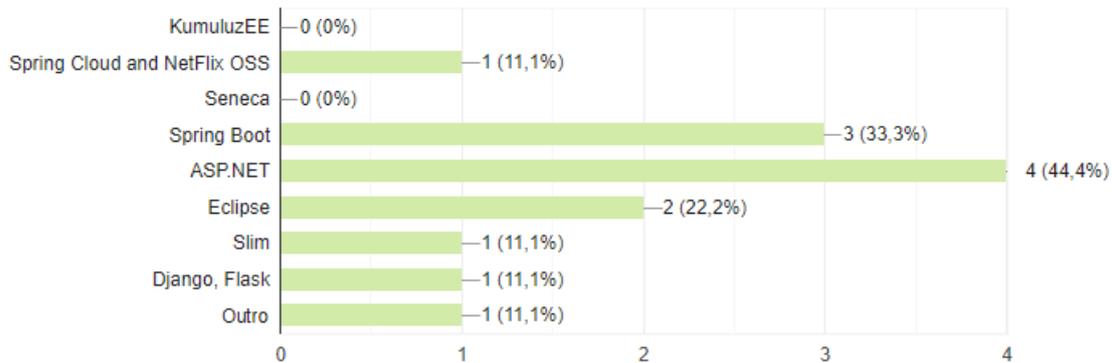


Figura 4. Frameworks utilizados para desenvolvimento de microserviços.

opção SOAP não é utilizada por nenhuma das empresas. Para [Balalaie et al. 2015] e [Fowler and Lewis 2014] microserviços visam transformar sistemas de software em pacotes de pequenos serviços, cada um entregável em uma plataforma diferente e rodando seus próprios processos através de uma comunicação com mecanismos leves como, por exemplo, RESTFull APIs ou API HTTP. Sendo assim, as empresas que utilizam APIs REST ou HTTP, que na pesquisa realizada corresponderam a 77,8%, também trabalham de acordo com o proposto pela literatura.

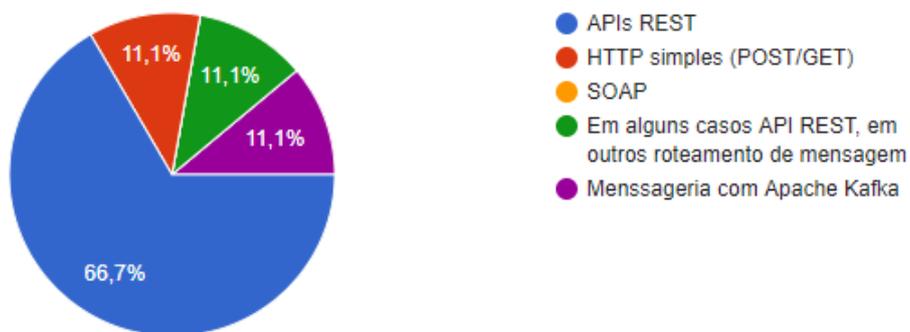


Figura 5. Tipo de comunicação utilizada entre os microserviços.

A Figura 6 apresenta os percentuais obtidos com a aplicação de uma questão sobre a estrutura do banco de dados utilizada pelas empresas que utilizam microserviços. Dentre as respostas, a estrutura mais utilizada é a de um banco de dados dedicado a cada microserviço, 55,6 %. Para [Meloca 2017], cada microserviço possui seus próprios dados, sua própria regra de negócio e uma interface bem definida, de modo que cada serviço deve ter seu próprio repositório, controlador de versões, ciclo de vida e controlador de tarefas. Levando isso em conta, pode-se entender que, ao optar por um banco de dados para

cada serviço, espera-se ter melhor desempenho e menor acoplamento entre os serviços, benefícios almeçados ao utilizar a arquitetura de microsserviços.

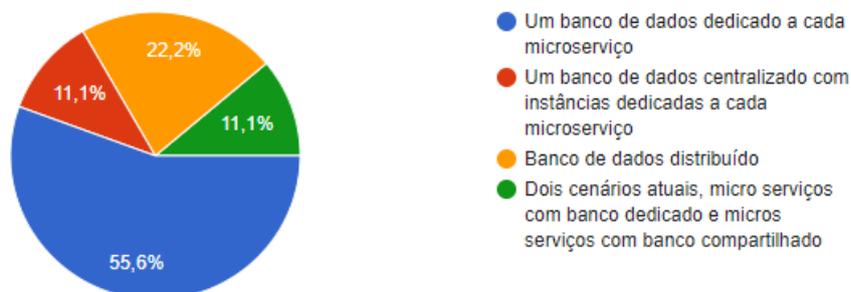


Figura 6. Estrutura de banco de dados para suporte aos microsserviços.

Uma questão sobre às ferramentas de integração utilizadas pelas empresas também foi aplicada, sendo as opções de resposta: *TeamCity*; *Jenkins*; *Azure*; *nenhuma* e *outro* com a possibilidade do respondente citar a ferramenta utilizada. Como pode ser visto na Figura 7, a ferramenta mais utilizada é a *Jenkins*. Essa questão é relevante pois uma característica dos microsserviços é a facilidade do *deploy*, o que geralmente ocorre por meio de integração contínua, e as ferramentas mencionadas tem a proposta de facilitar essa integração contínua. Segundo [Balalaie et al. 2015] a arquitetura de microsserviços está intimamente ligada ao conceito de entrega contínua. Na prática, isso só é possível ao adotar ferramentas apropriadas que promovam essa integração do projeto com o *deploy*.

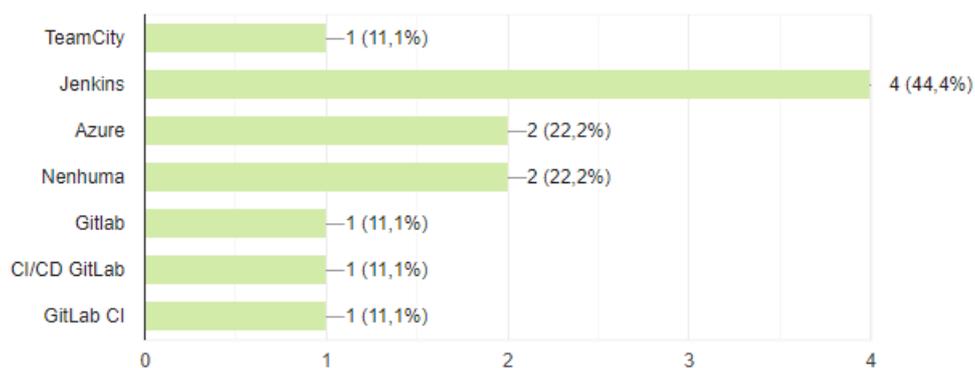


Figura 7. Ferramentas para realização de integração contínua.

A Figura 8 se refere as respostas da pergunta “Qual ferramenta de controle de log utiliza?”. As opções dadas foram: *Flume*; *ELK*; *Fluentd*; *nenhuma* e *outra* com opção de indicação da ferramenta utilizada. A maior parte das empresas, como pode ser observado no gráfico, utiliza alguma ferramenta de controle de log, porém não há uma que mais se destaca. Um dos respondentes marcou a opção *outra* e descreveu a forma do trabalho como segue: “Registros de eventos no banco de dados — log dos frameworks de roteamento”. O controle de log é uma importante ferramenta no momento de rastrear um problema e permite diagnosticar anormalidades em relação ao propósito

do sistema e questões de segurança e acessibilidade. O registro de logs é gerado e incrementado ao longo do tempo, e possui informações que permitem diagnosticar anormalidades em relação ao propósito do sistema e questões de segurança e acessibilidade. Para [Brown 2016], os seguintes conceitos são importantes na implementação da arquitetura de microsserviços: ID de Correlação, Agregador de Log e Disjuntor. Mais especificamente sobre o agregador de log, ele complementa os IDs de Correlação permitindo que os logs de diversos microsserviços diferentes sejam agregados em um único repositório que possa ser pesquisado. Juntos, eles permitem a depuração eficiente e compreensível de microsserviços, independentemente do número de serviços ou da profundidade de cada pilha de chamada.

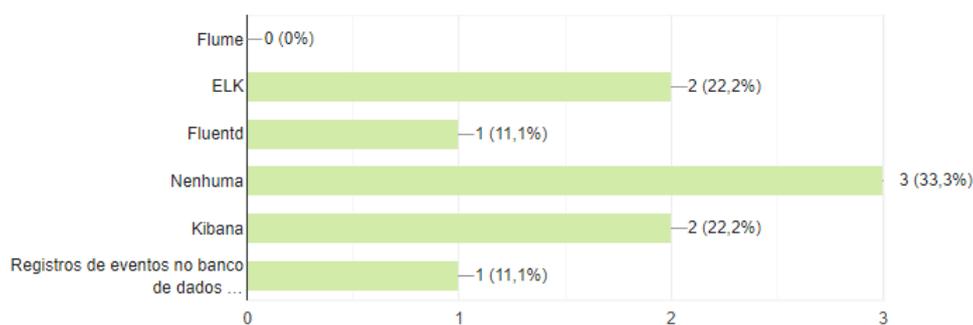


Figura 8. Ferramentas para controle de log.

As duas últimas questões da seção de perguntas para empresas que utilizam a arquitetura microsserviços foram discursivas, onde a empresa poderia responder de forma livre. Dentre as 9 empresas que utilizam a arquitetura de microsserviços apenas 5 responderam essas questões. Esses respondentes foram nomeados P1, P2, P3, P4 e P5.

A primeira questão discursiva realizada relaciona-se às preocupações da empresa com a granularidade dos microsserviços. Com base nas respostas podemos observar que a principal preocupação das empresas é o domínio dos serviços e a manutenção do código. É possível observar esse padrão na resposta fornecida por P1 “*Manutenção de código/estrutura do banco, balanceamento de carga*” ou ainda pela resposta fornecida por P2 “*Responsabilidade única, a orientação é que cada microsserviço seja responsável por um contexto específico*”. Para [Newman 2015], microsserviços são pequenos serviços autônomos que trabalham juntos. Sendo assim, um princípio base da arquitetura de microsserviços é justamente a granularidade dos serviços.

A segunda questão discursiva refere-se sobre o gerenciamento do *overhead*, “Como a empresa tem gerenciado o *overhead* (balanceamento de carga)?”. Pode-se observar que a maior parte das empresas adota como estratégia utilizar ferramentas para o controle do *overhead*. Na resposta de P3, por exemplo, “*Frameworks escolhidos para roteamento/balanceamento*”, ou ainda na resposta de P4 “*Utilizamos Kubernetes rodando no EKS da Amazon*” e na resposta de P5 “*Nginx, AWS EC2 ELB*”. Como destaca [Balalaie et al. 2015], para ser escalonável, um aplicativo deve ser capaz de distribuir a carga de um serviço individual entre suas várias instâncias. Esse é o dever de um balanceador de carga e, nesse caso, deve obter instâncias disponíveis do componente de descoberta de serviço. Como o escalonamento é um dos benefícios da arquitetura de

microserviços a empresa deve se preocupar com o *overhead* do sistema.

A parte final do questionário aplicado neste trabalho, algumas questões foram dedicadas para entender por que as empresas não utilizam a arquitetura de microserviços.

A primeira questão para as empresas que não utilizam foi se a empresa possui conhecimento sobre o tema, sendo que a grande maioria, 71,4%, apesar de não utilizar, alegou ter conhecimento sobre o microserviços, enquanto que 28,6% não possuem conhecimento.

A segunda questão foi “Qual arquitetura utiliza atualmente?”. Foi constatado que todas utilizam arquitetura monolítica. Mesmo SOA sendo uma arquitetura importante, não foram encontradas, dentro da amostra estudada, empresas em Maringá - PR que utilizem essa arquitetura.

A última questão objetivou indentificar a possibilidade/intenção de uma migração dos sistemas da empresa para a arquitetura de microserviços. Nesse sentido 42,9% dos respondentes acreditam que *talvez seja possível* a migração, 28,6% indicam que *é possível* a migração e outros 28,6% acreditam que *não é possível* a migração.

Das empresas que não utilizam a arquitetura de microserviços, 71,4% alegam ter conhecimento sobre a tecnologia de microserviços. Além disso 42,9% acreditam que talvez seja possível realizar a migração para esse estilo arquitetural e 28,9% acreditam que é possível realizar tal migração. Essas informações indicam um cenário positivo para uma futura adoção desse estilo arquitetural pelas empresas participantes da pesquisa.

Com o objetivo de consolidar algumas das informações obtidas pelas empresas que utilizam microserviços a Tabela 1 apresenta o cruzamento dos dados sobre a escolha da arquitetura e os critério de extração dos microserviços adotados pelas empresas. No total, 5 empresas estão realizando a migração gradual, 2 empresas realizaram a migração imediata, 1 empresas está realizando a migração atualmente e 1 empresa construiu o software já utilizando a arquitetura de microserviços. Após observar esses dados pode-se notar que o segundo fator mais relevante para extração dos microserviços por empresas que realizaram a migração gradual é a reusabilidade. Já para as empresas que realizaram a migração imediata são os requisitos. O critério de acoplamento também é uma escolha das empresas que estão passando por migração ou o sistema já foi construído utilizando a arquitetura de microserviços.

5. Conclusão

Este trabalho teve como objetivo realizar um estudo exploratório sobre o uso da arquitetura de microserviços por empresas de software de Maringá - PR. Com base nos resultados apresentados e na análise dos mesmos, pode-se concluir que há um número expressivo de empresas de software dentre as que responderam o questionário que utilizam a arquitetura de microserviços, sendo 9 das 23 que participaram da pesquisa. Também é possível observar, conforme a análise dos dados, que as empresas que utilizam a arquitetura de microserviços, em geral, trabalham conforme as boas práticas já apresentadas e descritas na literatura. Das empresas que não utilizam a arquitetura, pode-se observar que há o conhecimento sobre o tema e muitas empresas acreditam que uma migração seria viável.

É possível expandir a pesquisa ao âmbito nacional, possibilitando a comparação entre a forma de trabalho regional, observada na cidade de Maringá - PR, e a forma de

Estratégia de migração	Crítérios para extração dos serviços	Nº de empresas
Migração gradual	Escalabilidade	5
	Reusabilidade	4
	Coesão	1
	Acoplamento	3
	Requisitos	1
Migração imediata	Escalabilidade	2
	Acoplamento	1
	Requisitos	2
	Coesão	1
	Tabelas do banco de dados	2
Em processo de migração	Escalabilidade	1
	Acoplamento	1
	Performance, manutenibilidade	1
Foi construído assim	Escalabilidade	1
	Reusabilidade	1
	Acoplamento	1

Tabela 1. Estratégia de migração e critérios para extração dos microsserviços.

trabalho encontrada em outras regiões do país. Também pode ser expandida esta pesquisa para outros aspectos não abordados ou não explorados de forma profunda nesta pesquisa, tais como aspectos da migração dos sistemas, os benefícios obtidos com o uso da arquitetura de microsserviços, entre outros tópicos.

Como forma de divulgação dos resultados da pesquisa em âmbito privado, os autores pretendem promover lives e palestras no futuro para discussão do tema.

Referências

- Balalaie, A., Heydarnoori, A., and Jamshidi, P. (2015). Migrating to cloud-native architectures using microservices: an experience report. In *European Conference on Service-Oriented and Cloud Computing*, pages 201–215. Springer.
- Bass, L., Clements, P., and Kazman, R. (2003). *Software architecture in practice*. Addison-Wesley Professional.
- Brown, K. (2016). Além de palavras da moda: um breve histórico sobre padrões de microsserviços.
- Di Francesco, P., Lago, P., and Malavolta, I. (2019). Architecting with microservices: A systematic mapping study. *Journal of Systems and Software*, 150:77–97.
- Fowler, M. and Lewis, J. (2014). *Microservices*.
- Meloca, R. M. (2017). Um comparativo entre frameworks para microsserviços. B.S. thesis, Universidade Tecnológica Federal do Paraná.
- Newman, S. (2015). *Building microservices: designing fine-grained systems*. "O'Reilly Media, Inc."