



Migração de Múltiplos Sistemas Monolíticos para Microserviços

Guilherme Villaca

Orientador: Ivonei Freitas da Silva



Introdução

- Reúso de Software: Processo de design para reduzir tempo e custos de desenvolvimento. Evitar trabalhos repetidos, aproveitar conhecimento e experiência adquiridos previamente. Foco em não desenvolver do zero.
- Reengenharia: Pode ser uma forma de atingir o reúso. Usar o sistema existente como base para a continuidade e evolução. Manter lógicas programadas, decisões, requisitos e regras de negócio;

Introdução

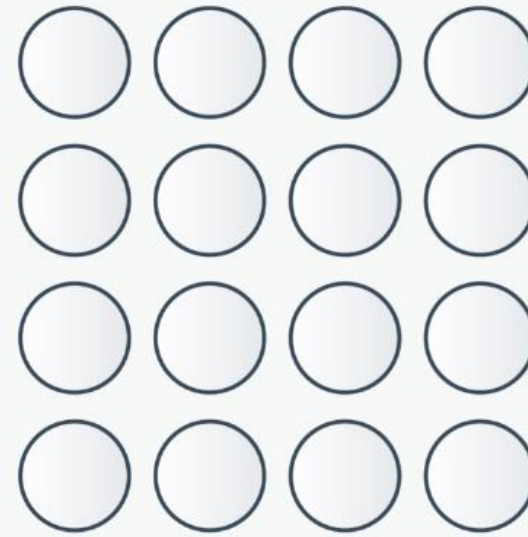
- Sistemas monolíticos: Sistemas compostos por módulos que não são independentes da aplicação.
- Microserviços: Para resolver as limitações de monolíticos, microserviços estão sendo utilizados por estimular uma melhor modularização e gestão através de serviços menores e autônomos.
- Objetivo: Verificar como seria a migração em um ambiente com múltiplos sistemas interligados.

Monolithic vs. Microservices



Monolithic

Single Unit



Microservices

Fine-grained

Sistemas monolíticos

- Problemas Comuns:
 - Manutenibilidade: Capacidade de um software ser modificado. Demanda de constantes mudanças podem deixar a manutenção mais complexa;
 - Escalabilidade: Capacidade de um software acomodar crescimento em tamanho e complexidade. Eficácia quando software usado em um contexto maior em escopo e complexidade;
 - Confiabilidade: Sistema livre de falhas. Quanto mais complexo menos confiável;
 - Qualidade: Garantia que o software atenda a todos os requisitos. Capacidade do software atender às necessidades dos usuários.

Motivações

- Delegação de responsabilidades: Microserviços podem ser desenvolvidos por times diferentes e independentes. Redução de sobrecarga e comunicação;
- Diminuição de custos: Melhora significativa em relação a manutenibilidade, portanto a longo prazo os custos são menores se comparados a sistemas monolíticos;
- Novas tecnologias: Serviços independentes torna possível desenvolver microserviços com diferentes tecnologias.

Riscos

- Dados: Um dos desafios da migração para microserviços, pois como cada microserviço deve ser independente, cada microserviço deve ser responsável pelos seus próprios dados;
- Resistência: Dificuldade em aceitar mudanças. Apego ao software desenvolvido;
- Custos iniciais: Investimento em infraestrutura e equipe;

Estratégias para migração

- Novas funcionalidades:
- Desenvolvimento baseado em código fonte existente;
- Manter as duas arquiteturas (monolítico e microserviço) em paralelo:
Feature Toggle é umas das técnicas para alternar entre os dois ambientes

Trabalhos Relacionados

- Um guia para apoiar a migração de sistemas legados para microserviços;
- Modernização de sistemas legados para disponibilização em dispositivos móveis utilizando microserviços;
- Microservices migration patterns;
- Artigo disciplina Engenharia de Requisitos: Um Processo para Migrar Sistemas Legados Monolíticos para Microserviços

Metodologia

- Estudo de caso: Estudo de um fenômeno em um ambiente real. Análise com base em um projeto de engenharia de software, metodologia de engenharia de software e seu uso dentro da organização, todo ou parte específica de um novo projeto ou manutenção de um projeto em andamento;
- Ambiente real: Núcleo de Tecnologia da Informação (NTI) UNIOESTE.

Resultados esperados

- Nova arquitetura baseado em microsserviços;
- Migração de um sistema para microsserviços;
- Analisar os efeitos da migração em relação a dificuldades encontradas;
- Melhoraria do processo para posterior migração de outros sistemas;
- Contribuição com organizações que vivenciam o mesmo cenário.

Cronograma

- Levantar histórico dos sistemas existentes no NTI;
- Identificar dependências;
- Definição da nova arquitetura;
- Decomposição da lógica, definição protocolos de comunicação, frameworks e ferramentas, definição do plano B;
- Decomposição base de dados;
- Implementação gradual, sistema monolítico funcionando em paralelo;
- Análise de possíveis problemas e melhoria do processo.



Obrigado!