

CENTRO PAULA SOUZA
ETEC PROF. MARIA CRISTINA MEDEIROS
Técnico em informática para Internet Integrado Ensino Médio

Gabriel Caspirro Demarchi

**APLICAÇÕES WEB E SISTEMAS DISTRIBUIDOS: COMPARAÇÃO
ENTRE ARQUITETURA MONOLÍTICA E DE MICROSERVIÇOS**

Ribeirão Pires
2025

Gabriel Caspirro Demarchi

**APLICAÇÕES WEB E SISTEMAS DISTRIBUIDOS: COMPARAÇÃO
ENTRE ARQUITETURA MONOLÍTICA E DE MICROSERVIÇOS**

Trabalho sobre Aplicações Web, Sistemas Distribuídos e Comparação entre Arquitetura Monolítica e Arquitetura de Microserviços apresentado ao Curso Técnico de Fundamentos da Internet integrado ao Ensino Médio da Etec Maria Cristina Medeiros, orientado pelo Prof. Anderson Vanin para obtenção de componente em Sistemas Web II.

**Ribeirão Pires
2025**

RESUMO

Este trabalho apresenta um estudo sobre aplicações web e sistemas distribuídos, destacando as diferenças entre arquitetura monolítica e arquitetura de microsserviços. São abordadas as principais características dessas tecnologias, bem como suas vantagens e desvantagens no desenvolvimento de software. A pesquisa visa proporcionar uma visão geral sobre as escolhas arquiteturais e seus impactos na escalabilidade, manutenção e desempenho dos sistemas modernos

SUMÁRIO

1	INTRODUÇÃO	4
2	SISTEMAS WEB	5
3	SISTEMAS DISTRIBUÍDOS	6
4	ARQUITETURA MONOLÍTICA.....	7
5	ARQUITETURA DE MICROSERVIÇOS.....	8
6	CONCLUSÃO.....	9
	REFERÊNCIAS	10

1 INTRODUÇÃO

Com a crescente digitalização dos processos empresariais e sociais, as aplicações web e os sistemas distribuídos desempenham um papel essencial na infraestrutura tecnológica moderna. As aplicações web, que operam por meio de navegadores e utilizam redes para oferecer funcionalidades diversas, são amplamente adotadas em setores como comércio eletrônico, serviços financeiros, redes sociais e sistemas de gestão corporativa. Já os sistemas distribuídos surgiram como uma resposta à necessidade de maior escalabilidade, resiliência e desempenho computacional, permitindo que múltiplos dispositivos trabalhem de forma coordenada para fornecer serviços eficientes e de alta disponibilidade.

No contexto do desenvolvimento de software, duas abordagens arquiteturais se destacam: a monolítica e a baseada em microsserviços. A arquitetura monolítica, caracterizada por sua estrutura unificada, tradicionalmente facilitou a construção e manutenção de sistemas menores, mas apresenta desafios conforme a complexidade do software aumenta. Por outro lado, a arquitetura de microsserviços fragmenta o sistema em componentes independentes que se comunicam via interfaces padronizadas, promovendo escalabilidade, flexibilidade e melhor gerenciamento de falhas.

Este trabalho tem como objetivo explorar esses conceitos fundamentais, apresentando as vantagens e desvantagens das diferentes abordagens arquiteturais e destacando sua relevância para o desenvolvimento de sistemas modernos. Para isso, serão discutidos os principais aspectos das aplicações web, dos sistemas distribuídos e das arquiteturas de software, oferecendo uma visão ampla das escolhas tecnológicas disponíveis para desenvolvedores e empresas.

2 SISTEMAS WEB

As aplicações web são sistemas de software que funcionam por meio de um navegador da web, acessíveis via internet ou intranet. Essas aplicações utilizam tecnologias como HTML, CSS, JavaScript e frameworks backend para processamento de dados. Exemplos incluem redes sociais, sistemas de gestão empresarial e plataformas de e-commerce.

Os sistemas web são compostos por um conjunto de camadas que incluem o frontend, responsável pela interface do usuário, o backend, que processa as requisições e lógica de negócio, e o banco de dados, onde as informações são armazenadas. O funcionamento dessas aplicações depende da comunicação entre cliente e servidor, muitas vezes utilizando protocolos como HTTP/HTTPS e APIs RESTful ou GraphQL.

Os benefícios das aplicações web incluem acessibilidade, pois podem ser utilizadas em diferentes dispositivos sem necessidade de instalação, além de possibilitarem atualizações centralizadas e distribuição facilitada. No entanto, desafios como segurança, tempo de resposta e disponibilidade da rede devem ser considerados no desenvolvimento desses sistemas.

3 SISTEMAS DISTRIBUÍDOS

Os sistemas distribuídos são compostos por múltiplos computadores interconectados que trabalham juntos para executar tarefas de forma coordenada. Eles são amplamente utilizados em serviços de grande escala, como computação em nuvem, bancos de dados distribuídos, redes peer-to-peer e blockchain. Diferente de sistemas centralizados, os sistemas distribuídos oferecem maior escalabilidade, tolerância a falhas e eficiência no processamento de grandes volumes de dados.

Entre os principais desafios dos sistemas distribuídos estão a comunicação entre os nós da rede, a sincronização de dados e a consistência das informações. Protocolos de comunicação, como HTTP, gRPC e MQTT, desempenham um papel essencial na troca de informações entre os componentes do sistema. Além disso, técnicas como replicação de dados e algoritmos de consenso, como Paxos e Raft, são empregadas para garantir a integridade e disponibilidade dos dados.

Os sistemas distribuídos podem ser classificados em diferentes modelos arquiteturais, incluindo cliente-servidor, onde os clientes fazem requisições a servidores centralizados, e peer-to-peer, onde todos os nós podem atuar como clientes e servidores simultaneamente. Com o avanço da computação em nuvem, arquiteturas serverless também têm se popularizado, permitindo que aplicações escalem dinamicamente sem a necessidade de gerenciamento explícito da infraestrutura.

A segurança é um aspecto crítico nos sistemas distribuídos, uma vez que a descentralização aumenta os riscos de ataques como interceptação de dados, ataques DDoS e falhas de autenticação. Estratégias como criptografia de ponta a ponta, autenticação multifator e monitoramento contínuo são fundamentais para mitigar esses riscos e garantir a confiabilidade do sistema.

4 ARQUITETURA MONOLÍTICA

A arquitetura monolítica é um modelo de desenvolvimento onde todos os componentes de um software estão integrados em um único código-base. Essa abordagem é tradicionalmente utilizada em aplicações menores ou em estágios iniciais do desenvolvimento de um sistema, pois oferece simplicidade na implementação e no gerenciamento.

Entre as vantagens da arquitetura monolítica, destacam-se a facilidade de desenvolvimento inicial, menor complexidade na comunicação entre componentes e implementação mais direta de funcionalidades. No entanto, conforme o sistema cresce, problemas como dificuldades de escalabilidade, maior tempo de deploy e dificuldades na manutenção tornam-se evidentes. Alterações em partes específicas do sistema podem impactar toda a aplicação, tornando as atualizações mais arriscadas.

Empresas que necessitam de rápido crescimento e escalabilidade podem encontrar na arquitetura monolítica um obstáculo, especialmente em ambientes onde novas funcionalidades precisam ser lançadas constantemente. A migração para microsserviços, nesse caso, pode ser uma solução viável.

5 ARQUITETURA DE MICROSERVIÇOS

Na arquitetura de microsserviços, a aplicação é dividida em pequenos serviços independentes que se comunicam por meio de APIs, geralmente utilizando padrões como REST ou gRPC. Cada microsserviço é responsável por uma funcionalidade específica, podendo ser desenvolvido, testado e implantado separadamente dos demais.

Essa abordagem proporciona maior flexibilidade e escalabilidade, permitindo que diferentes partes do sistema evoluam de forma independente. Além disso, facilita a adoção de diferentes tecnologias para diferentes serviços, otimizando o desempenho e possibilitando uma manutenção mais eficiente.

No entanto, a arquitetura de microsserviços também apresenta desafios, como a complexidade na orquestração dos serviços, a necessidade de estratégias eficientes de comunicação e segurança, além da exigência de um monitoramento constante para garantir a disponibilidade e integridade dos serviços.

6 CONCLUSÃO

A escolha entre arquitetura monolítica e microsserviços depende das necessidades específicas de cada projeto. Com o avanço das tecnologias em nuvem e da demanda por escalabilidade, os microsserviços têm se tornado uma alternativa viável para grandes sistemas distribuídos. No entanto, para aplicações menores e menos complexas, a arquitetura monolítica pode ser uma opção mais prática e eficiente.

Empresas e desenvolvedores devem avaliar cuidadosamente os benefícios e desafios de cada abordagem antes de tomar uma decisão. A escolha da arquitetura impacta diretamente a manutenção, o desempenho e a evolução do sistema ao longo do tempo. Dessa forma, compreender as particularidades dessas abordagens é essencial para um desenvolvimento de software bem-sucedido e sustentável.

REFERÊNCIAS

FOWLER, Martin. *Microservices: a definition of this new architectural term*. 2014. Disponível em: <https://martinfowler.com/articles/microservices.html>. Acesso em: 24 fev. 2025.

MESTRES DA WEB. Sistema Web: O que é e como funciona. Disponível em: <https://www.mestresdawe.com.br/tecnologias/sistema-web-o-que-e-e-como-funciona>. Acesso em: 24 fev. 2025.

ATLASSIAN. Arquitetura distribuída. Disponível em: <https://www.atlassian.com/br/microservices/microservices-architecture/distributed-architecture>. Acesso em: 24 fev. 2025.