

**CENTRO UNIVERSITÁRIO DE JOÃO PESSOA - UNIPÊ**  
**CURSO DE SISTEMAS PARA INTERNET**

# **Arquitetura de micro serviços em Node.js e MongoDB**

Thomas Cristanis Cabral Nogueira

**Arquitetura Monolítica**

**Ecosystem**

**Node.js**

**Microserviços**

**Princípios Gerais  
Microservices**

## Arquitetura em três camadas

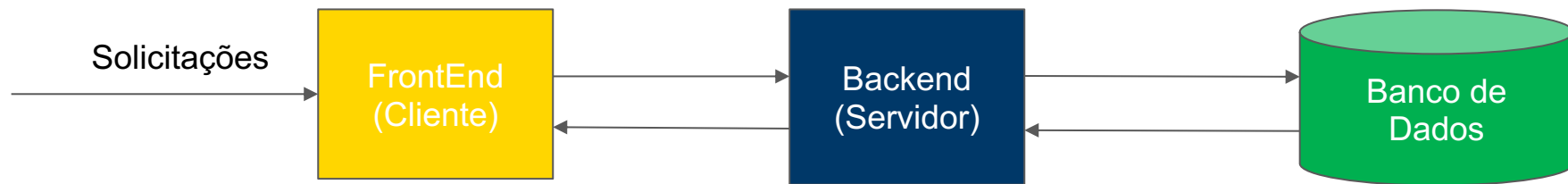


Figura 1 - Arquitetura em três camadas

- Frontend: uma interface com o cliente. Exemplos: páginas HTML, Mobile.
- Backend: faz o tratamento das requisições HTTP, executa a lógica do domínio e recebe atualização dos dados para retornar ao Frontend.
- Banco de Dados: armazena os dados da aplicação.

## Arquitetura em três camadas

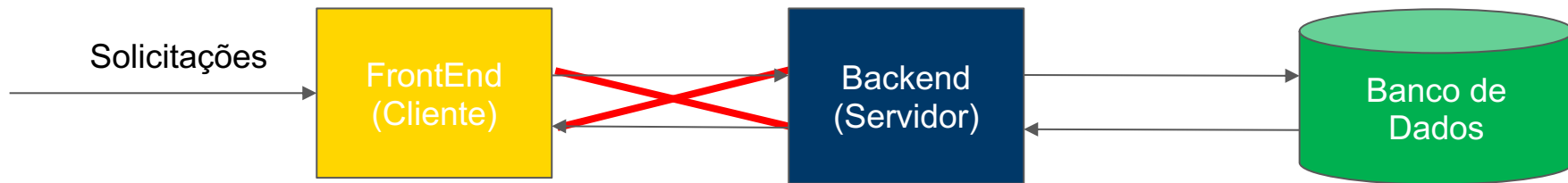
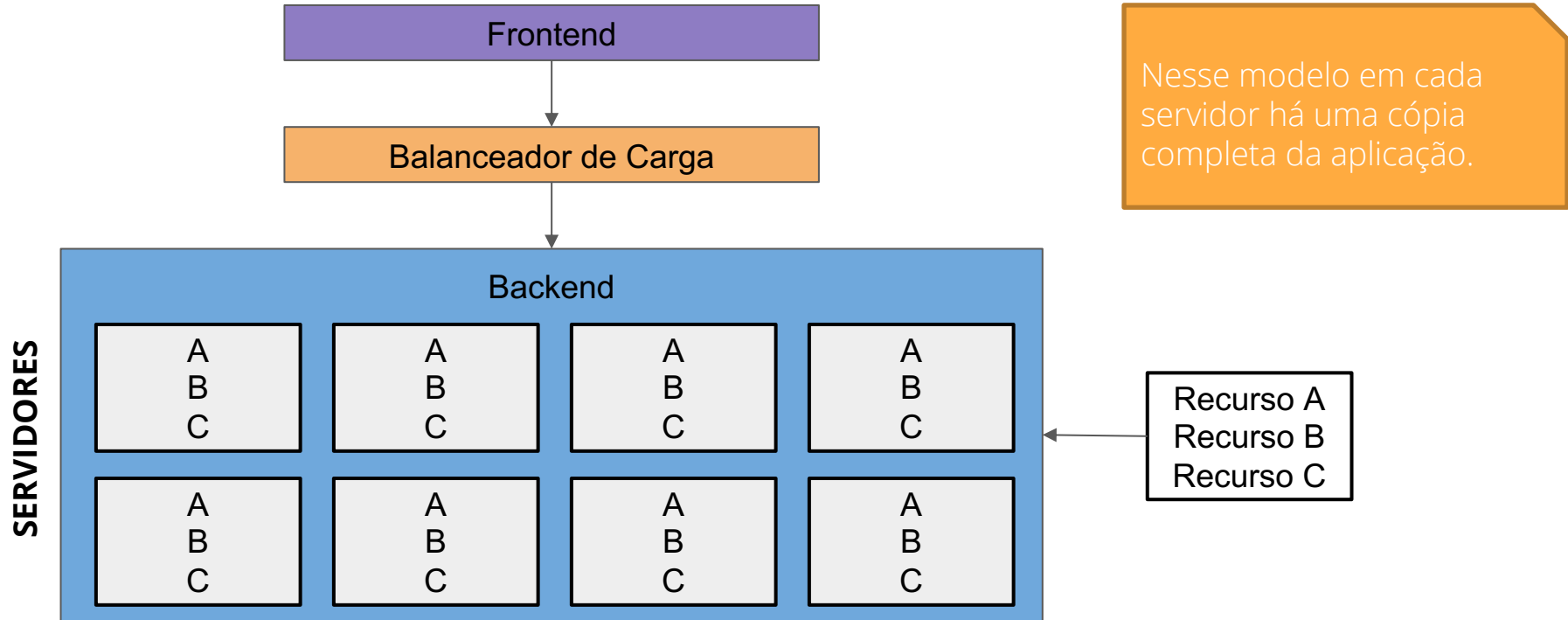


Figura 2 - Arquitetura em três camadas

- Esse é um exemplo de cenário de aplicação monolítica
- Uma única unidade lógica
- As mudanças acarretam em publicar uma nova versão de toda a aplicação.

**Esse é um exemplo de cenário de aplicação monolítica, onde há uma única unidade lógica. Qualquer mudança consiste em publicar uma nova versão de toda a aplicação.**

# Arquitetura Monolítica



Nesse modelo em cada servidor há uma cópia completa da aplicação.

Figura 3 - Detalhe no backend da Arquitetura Monolítica

- Primeiramente, essa é a forma natural de construir sistemas;
- Uma aplicação monolítica é feita como uma única unidade;
- Toda a lógica para manipular uma requisição é executada em um único processo;
- Possibilita ser escalada horizontalmente;
- Uma pequena alteração faz com que toda aplicação seja implantada novamente;
- Para escalar horizontalmente é necessário ter instâncias inteiras da aplicação no novo servidor;
- Esse modelo não é errado, depende da necessidade da aplicação.

**Aplicações monolíticas podem ser bem sucedidas**

# Microserviços



## Microserviços Google Trends 2015 - Atual

Interesse ao longo do tempo ?



Figura 4 -Busca pelo termo "Microservice" - Google Trends

- Desafios de escalabilidade
- Baixa eficiência
- Velocidade de desenvolvimento baixa
- Dificuldade em adoção de novas tecnologias



Em suma, o estilo arquitetural de microsserviço é uma abordagem para desenvolver uma única aplicação como **um conjunto de pequenos serviços**, cada um **executando em seu próprio processo** e comunicando-se com mecanismos leves, geralmente uma API HTTP. Esses serviços são **desenvolvidos com base nos recursos de negócios** e **implantados de maneira independente** por processos de implantação (*deploy*) automatizados. Existindo um mínimo de gerenciamento centralizado desses serviços, que pode ser escrito em diferentes linguagens de programação e usar diferentes tecnologias de armazenamento de dados.

James Lewis and Martin Fowler

**Um conjunto de pequenos serviços**



**Contextos bem definidos**

**Executando em seu próprio processo**



**Isolados**

**Desenvolvidos com base nos recursos de negócios**



**Focados no negócio**

**Implantados de maneira independente**



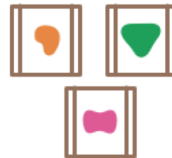
**Independentes**

# Monolítico x Microserviço

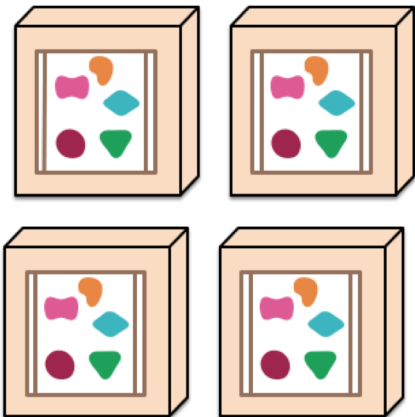
Uma aplicação monolítica coloca toda sua funcionalidade em um único processo...



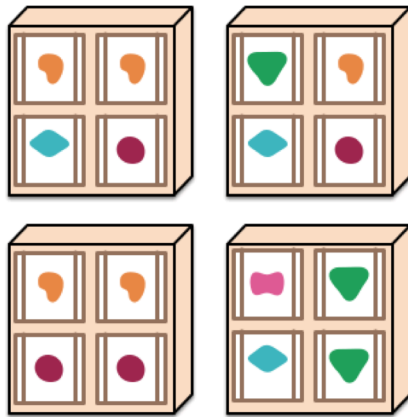
Uma arquitetura em microsserviços põe cada elemento de uma funcionalidade em um serviço separado ...



... e escala replicando a aplicação monolítica em vários servidores



... e escala distribuindo estes serviços entre os servidores, replicando quando necessário.



***"Não é justo que uma determinada tecnologia ou abordagem seja considerada totalmente correta ou errada, mas que foi apropriada ou não em certas circunstâncias."***  
(Kitchenham, 2007)

Figura 5 - Comparação entre Aplicações Monolíticas e Microserviço

amazon.com<sup>®</sup>

**NETFLIX**

**GILT**



ebay



NORDSTROM

theguardian

## NETFLIX

550 bilhões de eventos por dia

8.5 milhões de eventos por segundos

21 GB de transferência por segundo

1 Petabyte de por dia

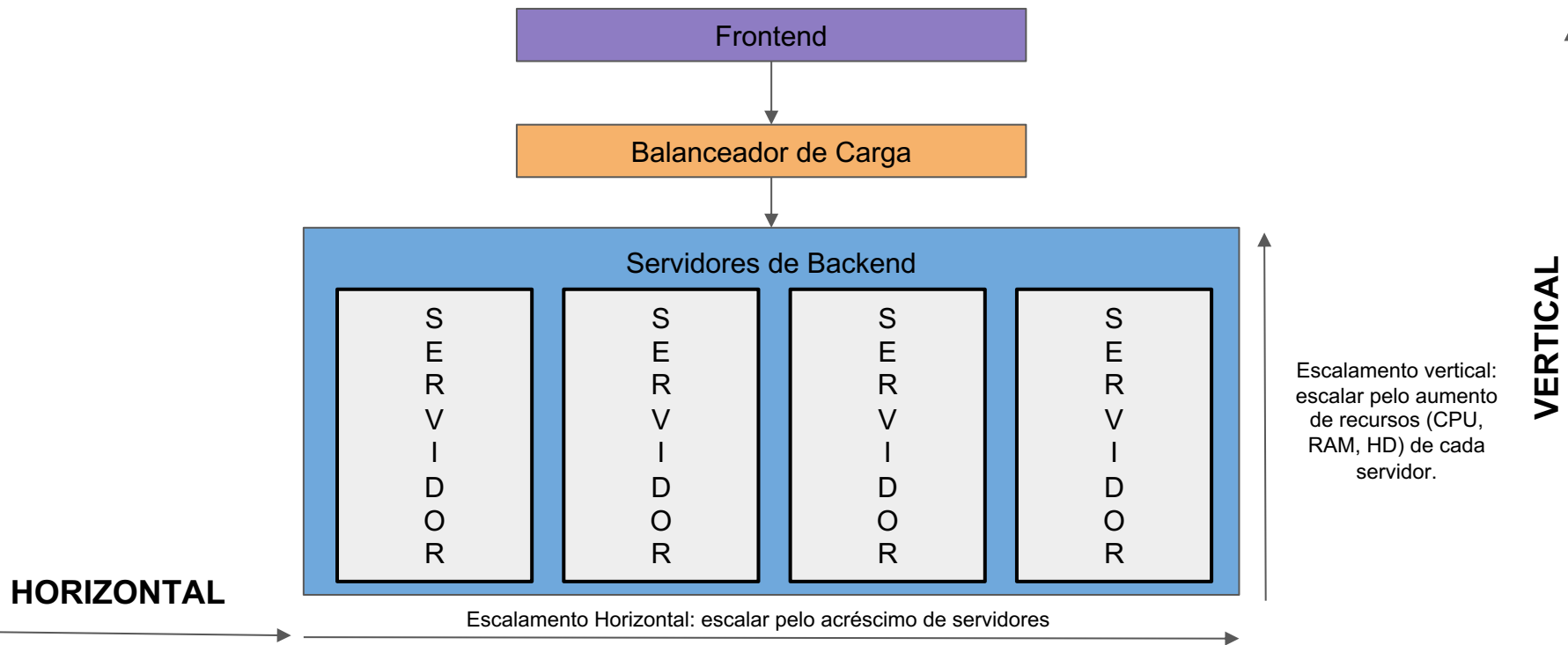


Figura 6 - Escalabilidade em Microserviços



- Escalabilidade Horizontal e Vertical
- Eficiência.
- Times mais produtivos
- Tecnologias Heterogêneas
- Resiliência
- Deployment Facilitado



# Ecossistema de Microserviços



Figura 6 - Camadas do ecossistema de Microserviços

## Hardware

*Servidores*  
*Banco de Dados*  
*Sistemas Operacionais*

## Comunicação

*EndPoints*  
*Troca de Mensagens*  
*Balaceamento de Carga*  
*RPC*

## Plataforma de Aplicação

*Ambiente de Dev*  
*Ferramenta de Testes*  
*Monitoramento*  
*Pipeline de Deployment*

## Microserviços

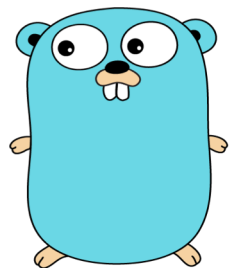
*Microservicos*  
*Configurações*

**Susan J. Fowler** criou uma espécie de ***checklist*** detalhada dos padrões que, na sua visão todo microserviço do ***Uber Technologies*** deveria atender.

1. **Estável**
2. **Confiável**
3. **Escalável**
4. **Tolerante a Falhas**
5. **Alto Desempenho**
6. **Monitorado**
7. **Documentado**
8. **Preparado para Catástrofe**

Um serviço que atendesse a critérios preparados a partir desses itens seria considerado **pronto para produção**.





**Golang**



# Node.js

- O **Node.js** é runtime JavaScript criado para o desenvolvimento no lado do servidor;
- Com o Node.js, o JavaScript foi inserido nos scripts do lado do servidor;
- Tem desempenho incrivelmente mais rápido e à capacidade de lidar com requisições simultâneas;
- Casos: Netflix, LinkedIn, Trello, Uber, PayPal, eBay, NASA, entre outros.





- Cria um projeto Node.js (package.json)
  - `npm init`
- Instala os módulos do Koa (Middleware Web API)
  - `npm install koa koa-better-router koa-body`
- Imagem Docker do MongoDB
  - `docker run -d -p 27017:27017 -v dbdata:/data/db mongo:4.0`
- Instala o mongoose (conexão com o MongoDB)
  - `npm install mongoose`

## Uma Plataforma de Viagens

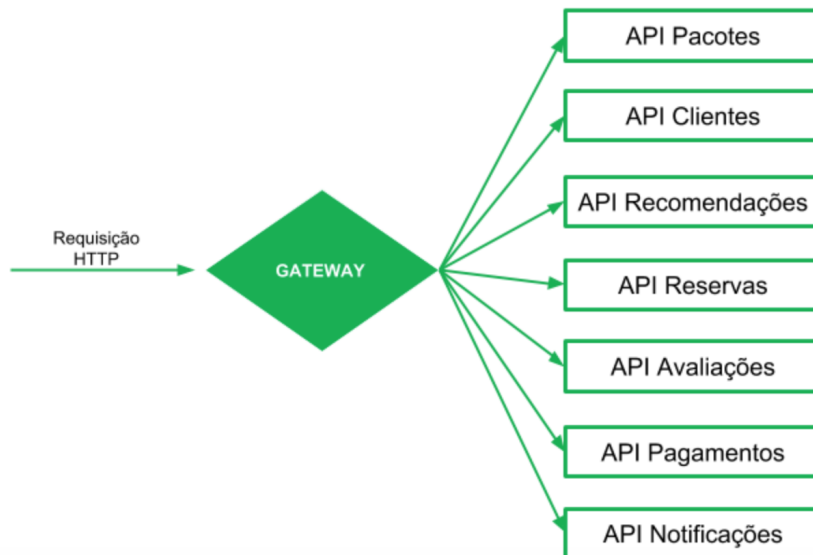


Figura 7 – Ilustração dos microserviços da prática

- Coulouris, George. Sistemas Distribuídos. Conceitos e Projeto. 4. ed. Editora Bookman,. 2007
- Newman, Sam. Building Microservices: DESIGNING FINE-GRAINED SYSTEMS. 1. ed. United States of America: O'Reilly Media, Inc,. 2015
- Fowler, Suzan J. Microserviços prontos para produção. 1. ed. Editora Novatec,. 2017
- Fowler, Martin. Microservices a definition of this new architectural term. 2014. Disponível em:
- <<https://martinfowler.com/articles/microservices.html>>. Acesso em: 25/11/2018
- Richardson, Chris. Microservices Patterns With examples in Java. 1. ed. Manning,. 2018
- Fowler, Martin. Microservices Resource Guide. 2018. Disponível em:
- <<https://martinfowler.com/microservices/>>. Acesso: 27/11/2018