

2025

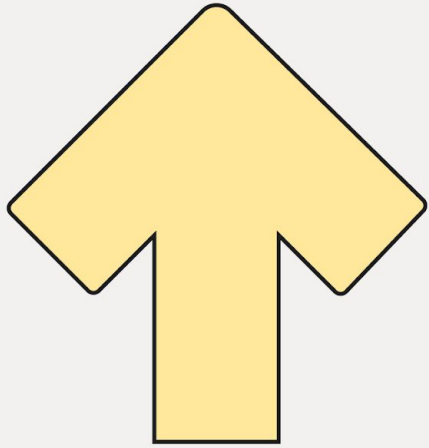
# Do Monolito aos Microserviços: fundamentos, práticas e desafios

João Oliveira, Lucas Milhorança, Tulio Olivieri



Google  
Developer  
Groups

01



# Arquitetura de Design de Software



Google  
Developer  
Groups

# O que são?

- Arquitetura de software
- Design de software



Google  
Developer  
Groups

# Arquitetura de Software

“A arquitetura de software é o processo de converter características de software como flexibilidade, escalabilidade, viabilidade, reutilização e segurança em uma solução estruturada que atenda às expectativas técnicas e de negócios.”

Exemplos de arquiteturas: Monolito, Microsserviços, Serverless, Arquitetura orientada a eventos...

Fonte: Eduardo Rabelo - Arquitetura de Software: A diferença entre Arquitetura e Design -  
<https://oieduardorabelo.medium.com/arquitetura-de-software-a-diferen%C3%A7a-entre-arquitetura-e-design-964ab38ea61a>



Google  
Developer  
Groups

# Design de Software

“A arquitetura do software é responsável pelo esqueleto e pela infraestrutura de alto nível de um software, o design do software é responsável pelo design do nível de código, como o que cada módulo está fazendo, o escopo das classes e os objetivos das funções, etc.”

Exemplos de abordagens: DDD, SOLID, Design Patterns

Fonte: Eduardo Rabelo - Arquitetura de Software: A diferença entre Arquitetura e Design -  
<https://oieduardorabelo.medium.com/arquitetura-de-software-a-diferen%C3%A7a-entre-arquitetura-e-design-964ab38ea61a>

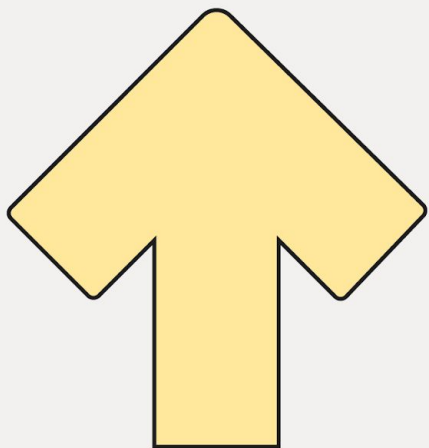


Google  
Developer  
Groups

02

# Monolitos

Um artefato.



Google  
Developer  
Groups

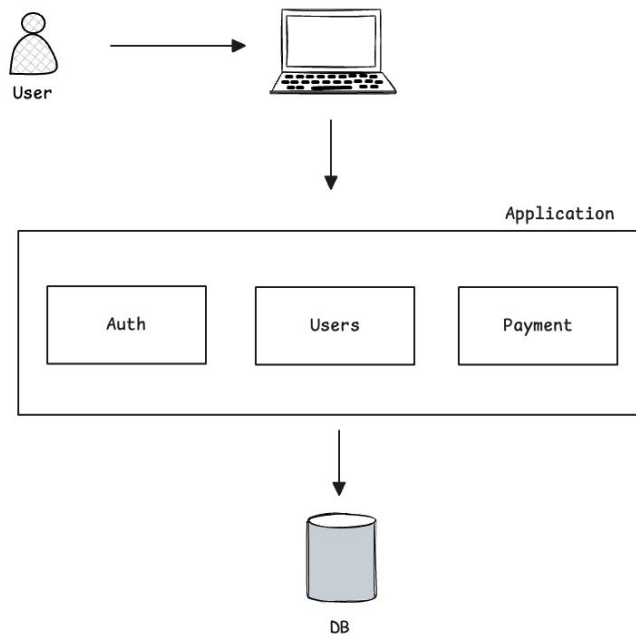
# O que é uma aplicação monolítica

- “All-in-one”
- **Um artefato ou unidade de deploy**
- Aplicações consideradas “tradicionais”



Google  
Developer  
Groups

# O que é uma aplicação monolítica



Google  
Developer  
Groups



# Desvantagens das aplicações monolíticas

- Não escalam
- Alto acoplamento
- Ultrapassado



Google  
Developer  
Groups

# Desvantagens das aplicações monolíticas

- Não escalam
- Alto acoplamento
- Ultrapassado

**NÃO É BEM ASSIM**



Google  
Developer  
Groups

# Tipos de monolitos

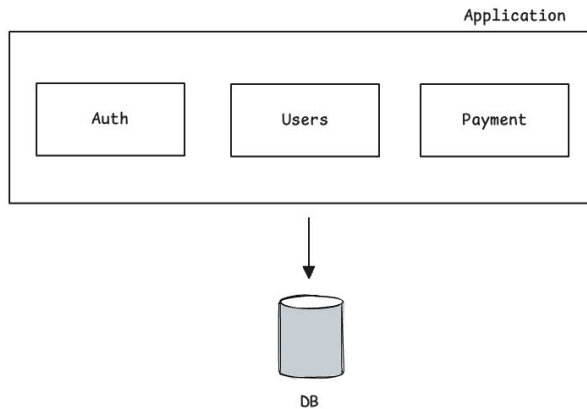
- “Single process”
- Monolitos distribuídos
- Monolito Modular



Google  
Developer  
Groups

# Single Process

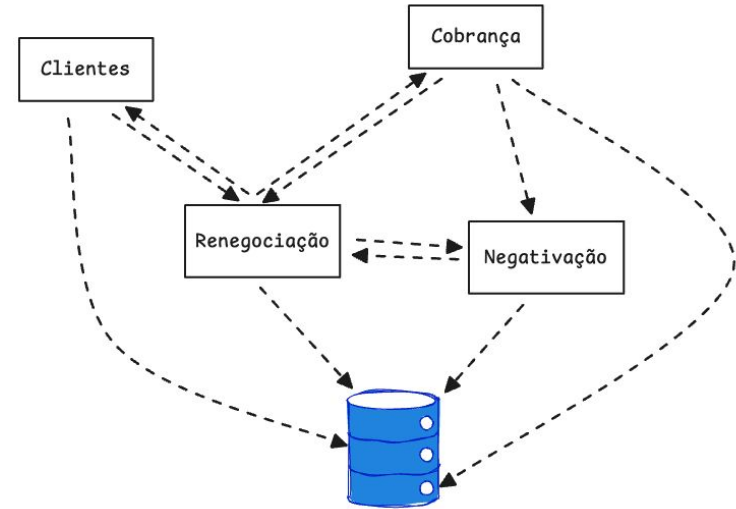
- Alto acoplamiento



Google  
Developer  
Groups

# Monolito distribuído

"Um Monolito Distribuído é um sistema que, apesar de ser implantado como múltiplos serviços independentes (distribuídos), ainda mantém um alto grau de acoplamento lógico e de dados, tipicamente através de um banco de dados compartilhado. Ele herda os problemas de complexidade do monolito e adiciona a complexidade operacional da distribuição, sem entregar os benefícios esperados de uma arquitetura de microsserviços."

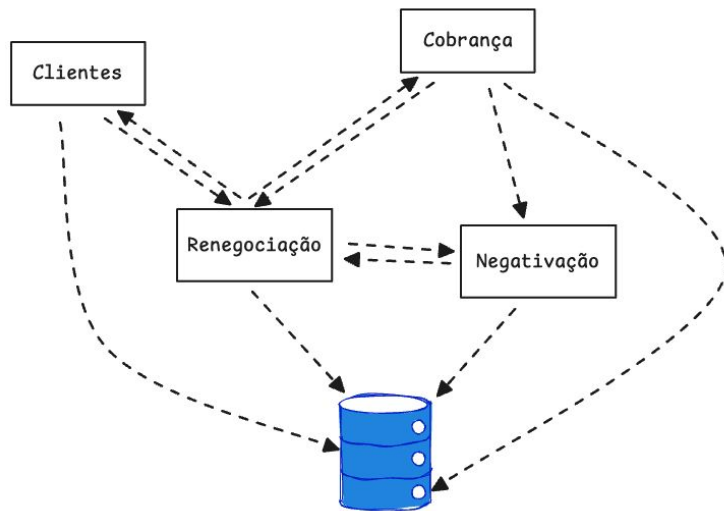


Google  
Developer  
Groups

# Monolito distribuído

"Um Monolito Distribuído é um sistema que, apesar de ser implantado como múltiplos serviços independentes (distribuídos), ainda mantém um alto grau de acoplamento lógico e de dados, tipicamente através de um banco de dados compartilhado. Ele herda os problemas de complexidade do monolito e adiciona a complexidade operacional da distribuição, sem entregar os benefícios esperados de uma arquitetura de microsserviços."

## ANTI-PADRÃO



Google  
Developer  
Groups

# Vantagens das aplicações monolíticas

- Deploy único
- Operação fim a fim em um único sistema
- Observabilidade facilitada
- Única linguagem
- Manutenção facilitada



Google  
Developer  
Groups

**“Quase todas** as histórias de microsserviços bem-sucedidas **começaram com um monólito** que ficou muito grande e **depois** foi quebrado e dividido em **microsserviços.**”

- Martin Fowler



Google Developer Groups



“**Quase todos** os casos em que ouvi falar de um sistema que foi **criado do zero** como um sistema de **microsserviços** acabaram tendo **sérios problemas.**”

- Martin Fowler

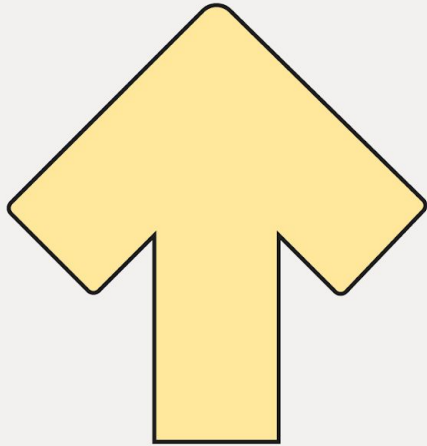


Google Developer Groups

03

# Microserviços

Flexibilidade,  
resiliência e  
entrega contínua



Google  
Developer  
Groups

# Por que Microserviços?

- Independência
- Escalabilidade e agilidade
- Times autônomos cuidando do próprio domínio
- Velocidade de entrega



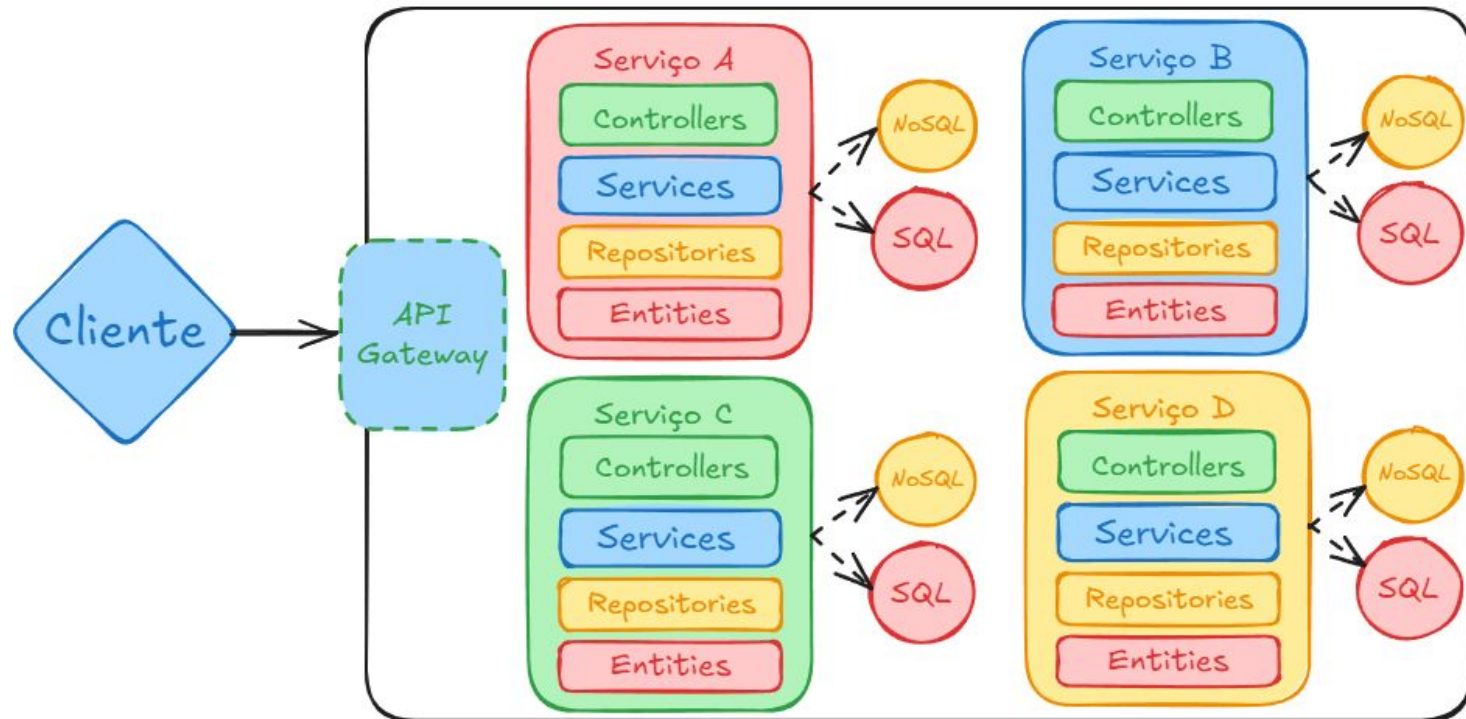
Google  
Developer  
Groups

# O que é um serviço?

- Uma unidade pequena e independente
- Responsável por um único domínio do negócio
- Tem seu próprio ciclo de vida



Google  
Developer  
Groups



Google  
Developer  
Groups

# Relação de Microsserviços com o Produto

- Aplicação quebrada pelo produto, não só por questões técnicas
- Dominio como guia: DDD e Bounded Contexts



Google  
Developer  
Groups

# Autonomia das Equipes

- Cada time cuida do seu serviço
- Entregas independentes
- Stack e ritmo



Google  
Developer  
Groups

# Desvantagens

- Arquitetura complexa
- Alto custo
- Várias equipes
- Monitoramento complexo
- Indisponibilidade



Google  
Developer  
Groups



**“Não aposte** em microsserviços só porque parece legal. Separe o sistema em Jars utilizando uma arquitetura de plugins primeiro.”

- Uncle Bob



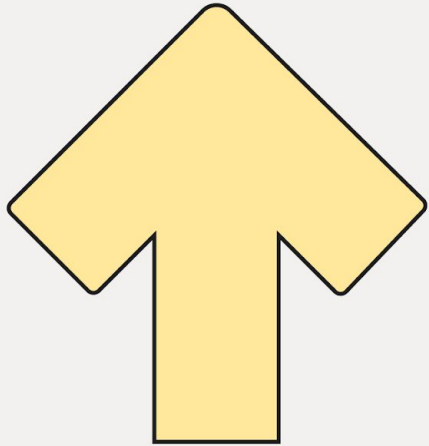
Google Developer Groups



04

# Monolito Modular

Um meio caminho  
entre o monolito e  
os microsserviços



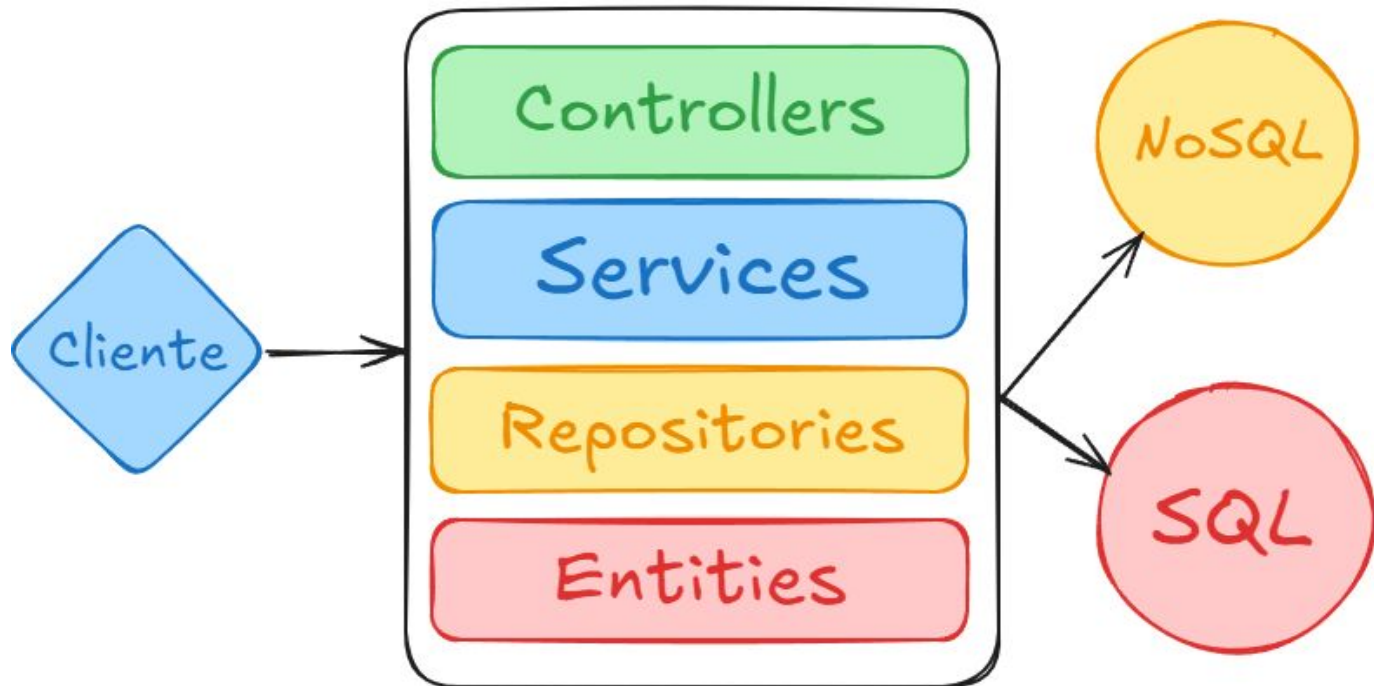
Google  
Developer  
Groups

# A junção de duas arquiteturas

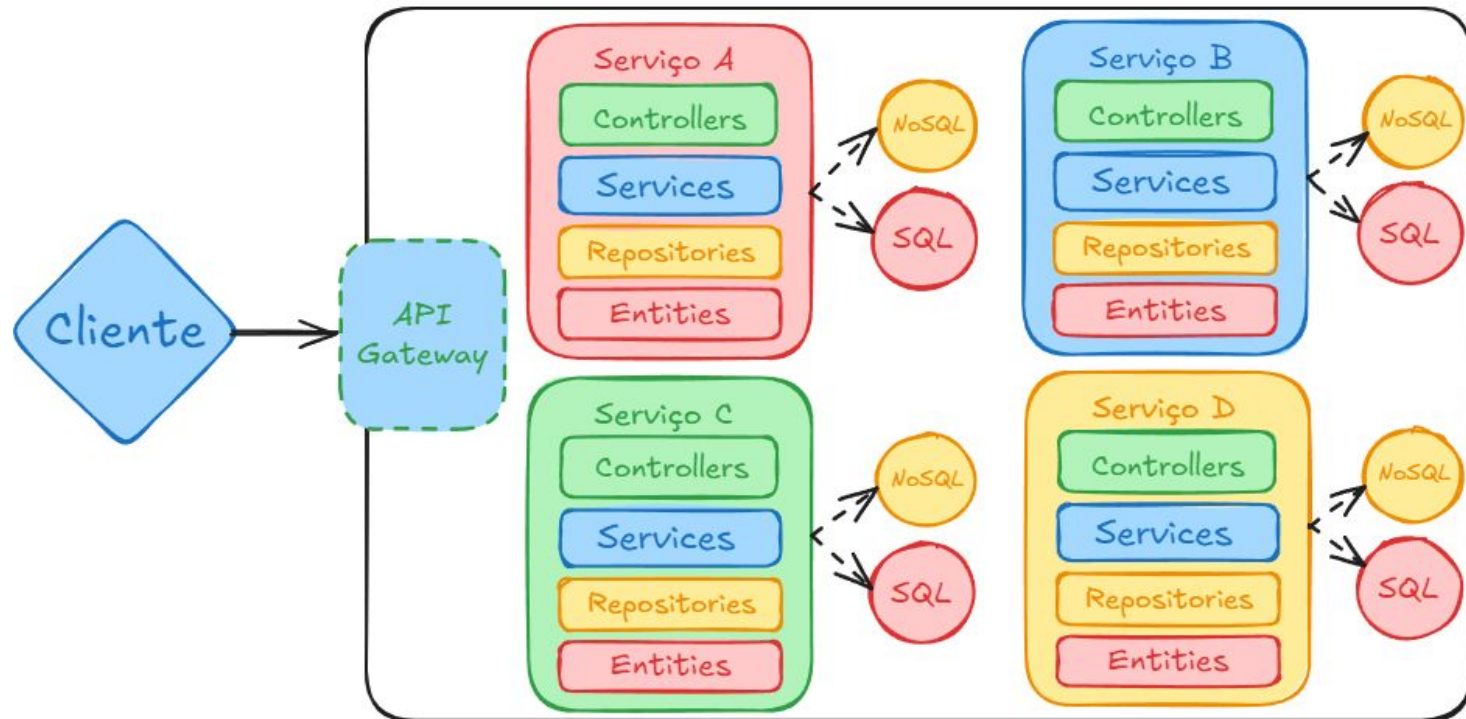
- Base de código única
- Divisão em módulos de acordo com o domínio
- Compartilhamento entre módulos e domínios
  - Chamada via interface
  - Módulo compartilhado



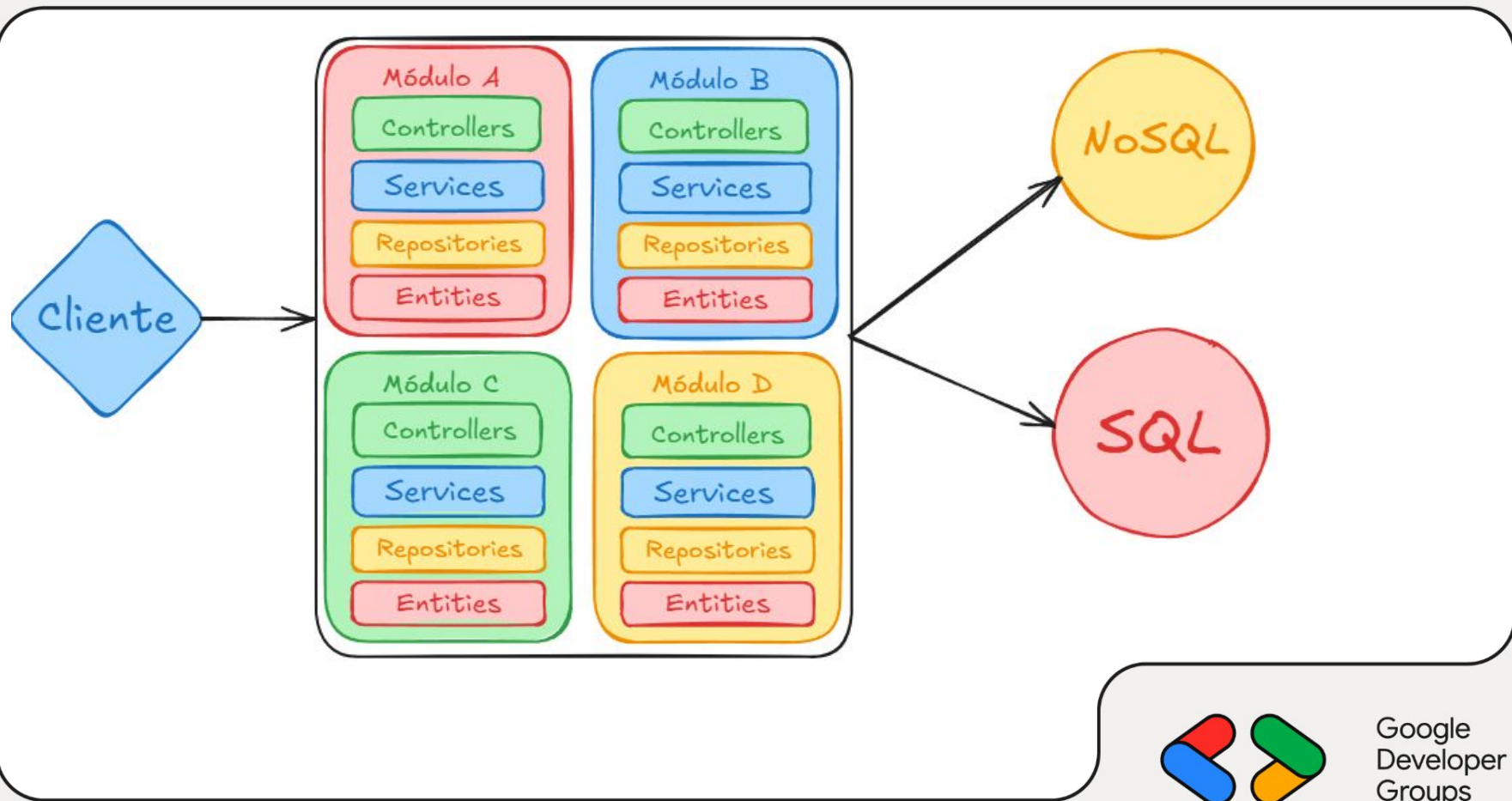
Google  
Developer  
Groups



Google  
Developer  
Groups



Google  
Developer  
Groups



## Benefícios de ambas as arquiteturas

- Mais simples de manter que microsserviços
- Segregação de interfaces entre os módulos
- Sem requisições e menos assincronicidade
- Menos fontes de logs
- Diferentes times alteram módulos diferentes



Google  
Developer  
Groups

## Com algumas desvantagens...

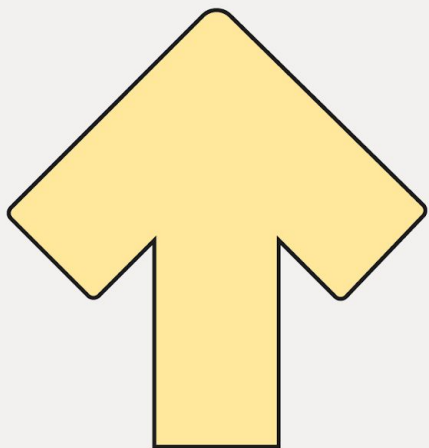
- A base de código ainda gera um único artefato
- Não é possível escalar um módulo a parte
- Uma única linguagem de programação
- Dificuldade de encontrar gargalos



Google  
Developer  
Groups



05



# Arquitetura de Microserviços

Características



Google  
Developer  
Groups

# Independência

- Cada serviço pode evoluir sozinho
- Deploy
- Banco de Dados
- Regras de Negócio
- Desacoplamento



Google  
Developer  
Groups

# DDD (Domain Driven Design)

- Cada serviço resolve um domínio específico do negócio
- Bounded Contexts (escopos definidos)
- Pensar em produto
- Estado e regras de negócio próprias



Google  
Developer  
Groups

# Stack

- Liberdade tecnológica por serviço
- Melhor ferramenta para cada domínio
- Time decide a stack
- Padronização em observabilidade, comunicação, deploy



Google  
Developer  
Groups

# Tolerância a falhas

- Projetado para falhar
- Uma falha não deve derrubar o sistema inteiro
- Isolamento de problemas entre serviços
- Recuperação automática ou mitigação de impactos

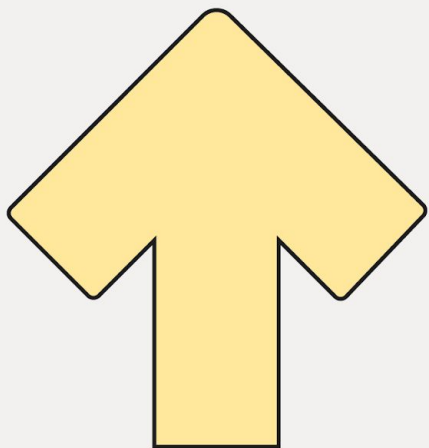


Google  
Developer  
Groups

06

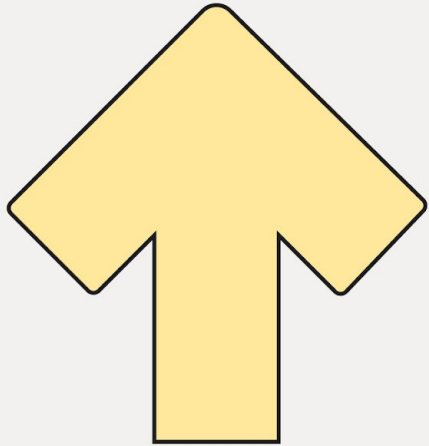
# Exemplos Práticos

Um caso de uso de  
Monolito x  
Microservicos



Google  
Developer  
Groups

07



# Resiliência

Tolerância e  
recuperação de  
falhas



Google  
Developer  
Groups

# Resiliência de microsserviços

- Em algum momento algo irá falhar...
- Essa falha não deve derrubar todo o sistema
- Recuperação após a falha
- Testes de integração
- Engenharia do Caos



Google  
Developer  
Groups



# Health Check

- Endpoint dedicado em cada microsserviço
- Valida agregados essenciais para o serviço
- Status HTTP para controle
- Resposta detalhada de cada dependência
- Monitoramento e alertas



Google  
Developer  
Groups

# Rate Limit

- Prevenção de sobrecarga intencional ou não:
  - Erros de integrações, DDoS, força bruta...
- Políticas de requisição por tempo
- Degradação do serviço como um todo
- Implementação local ou via API Gateway



Google  
Developer  
Groups

# Algoritmos de Rate Limit

- Fixed Window
- Sliding Window
- Token Bucket
- Leaky Bucket



Google  
Developer  
Groups

# Fallback

- Execução de alternativas quando um recurso falha
- Uso de outro serviço ou integração
- Uso de cache, retorno default ou filas
- Comumente utilizado junto a outro padrão



Google  
Developer  
Groups

# Circuit Breaker

- Prevenção de falhas em cascata
- Dá tempo para o serviço degradado se recuperar
- Retorno de fallbacks ou erros sem precisar checar o recurso indisponível



Google  
Developer  
Groups

## Estados de um Circuit Breaker

- Fechado: requisições funcionam normalmente, enquanto um algoritmo monitora as respostas
- Aberto: as requisições são desviadas para um fallback ou rejeitadas por um tempo
- Meio-aberto: Após um tempo de espera, algumas requisições são desviadas



Google  
Developer  
Groups

# Retry

- Reprocessamento de mensagens ou requisições que falharam
- Seletivo para requisições:
  - Retry para 429, 503, 504...
  - Mas não para 400, 401, 403, 404...
- DLQs para mensagens



Google  
Developer  
Groups

# Estratégias de Backoff

- Estratégia que define o intervalo entre tentativas:
  - Linear
  - Exponencial
  - Aleatório



Google  
Developer  
Groups



# Idempotência

- Uma mesma operação deve sempre produzir o mesmo resultado
- Operações são identificadas por Ids únicos
- Pode-se usar mais de um campo para gerar a idempotência



Google  
Developer  
Groups

# Async

- Uso de filas ou tópicos para comunicação
- Publisher faz o envio das mensagens
- Broker gerencia as filas/tópicos
- Subscriber consome as mensagens disponíveis
- Uso de webhooks ou filas/tópicos de resposta



Google  
Developer  
Groups

# Async

- Comunicação assíncrona entre os serviços
- Consistência eventual
- Queda do subscriber não prejudica o publisher
- As mensagens ficam armazenadas no broker até serem consumidas ou removidas
- Dead-Letter Queue (DLQ)



Google  
Developer  
Groups

# Observabilidade

- Monitoramento do estado de cada serviço
- Estado das máquina, banco de dados...
- Quantidade de respostas de erro
- Mensagens presas em filas ou DLQs
- Tempo de processamento de uma requisição



Google  
Developer  
Groups

# Métricas

- Valores numéricos que podem representar dados atuais ou históricos
- Normalmente utilizadas em dashboards de monitoramento
- Podem possuir alertas atrelados a um threshold de valores



Google  
Developer  
Groups

# Logs

- Mensagens de texto enviadas pelos serviços
- Essenciais para depuração em ambientes de produção
- Identificam operações e dados
- Uma mesma operação pode possuir log em N serviços



Google  
Developer  
Groups

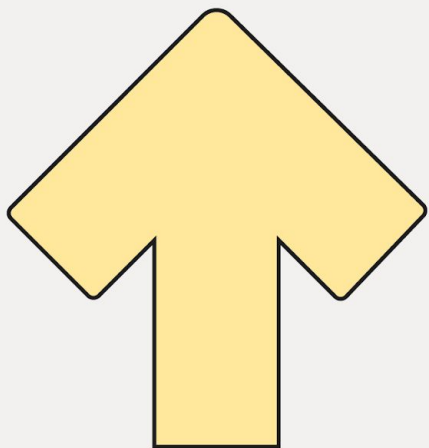
# Traces

- Representam todo o caminho que uma requisição percorreu dentro do sistema
- Identificam gargalos em um serviço ou agregado específico
- Um Id identifica a requisição de ponta a ponta



Google  
Developer  
Groups

08



# Padrões

Os direcionamentos  
para o sucesso e a  
mitigação de falhas



Google  
Developer  
Groups



## Padrões discutidos nesta seção

- Secrets Manager
- API Gateway
- BFF (Backends for frontends)
- ACL (Anti-corruption layer)
- API Composition

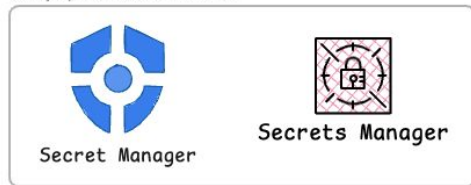


Google  
Developer  
Groups

# Secrets Manager (Vault)

## Gestão segura e centralizada de credenciais

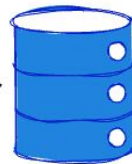
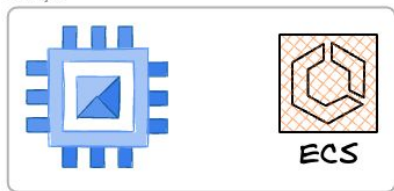
Serviço que armazena credenciais



Criptografia



Serviços



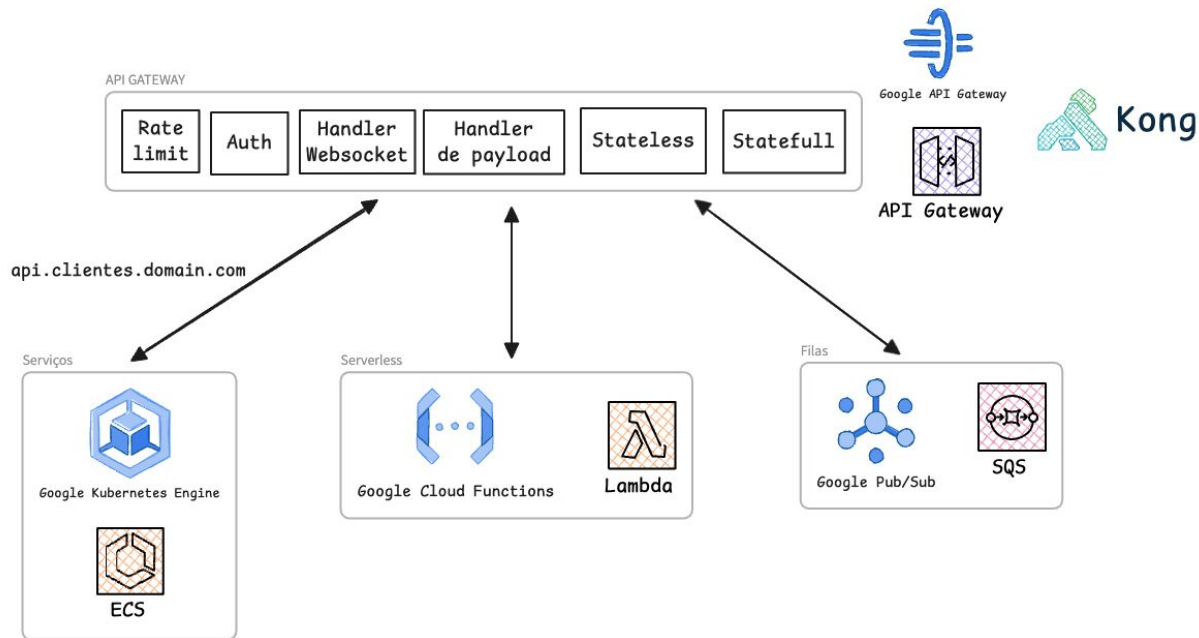
DB



Google  
Developer  
Groups

# API Gateway

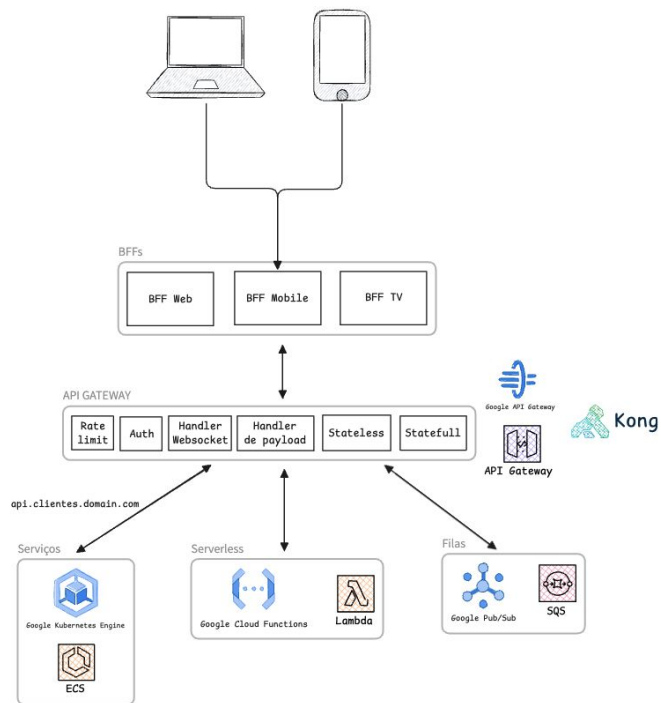
Segurança, roteamento e exposição controlados



Google  
Developer  
Groups

# BFF (Backends for frontends)

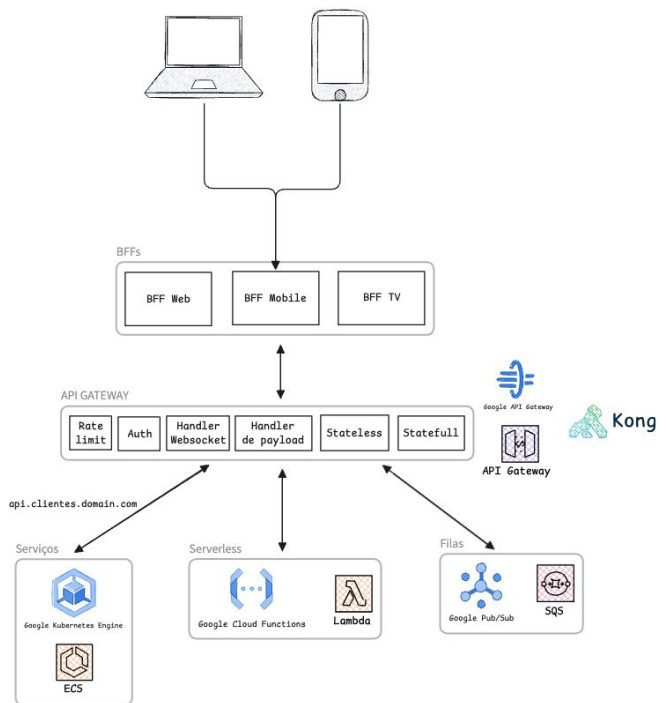
Gateway para clients



Google  
Developer  
Groups

# BFF (Backends for frontends)

Gateway para clientes

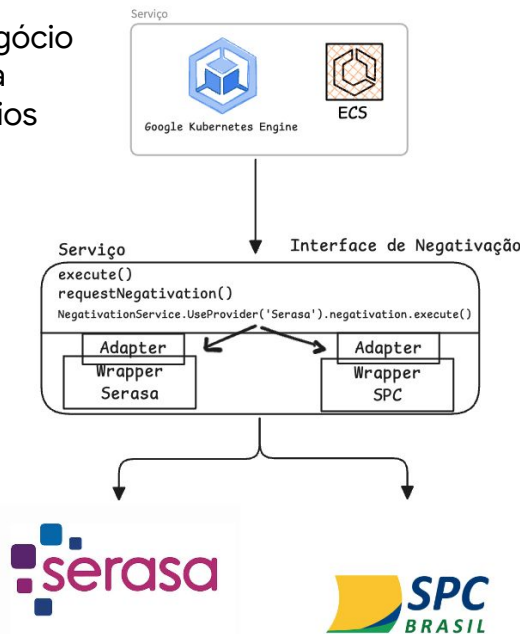


Google  
Developer  
Groups

# ACL (Anti-corruption layer)

## Compatibilidade de modelos de domínio

- Pode possuir regras de negócio
- Pode simplesmente fazer a “tradução” entre os domínios



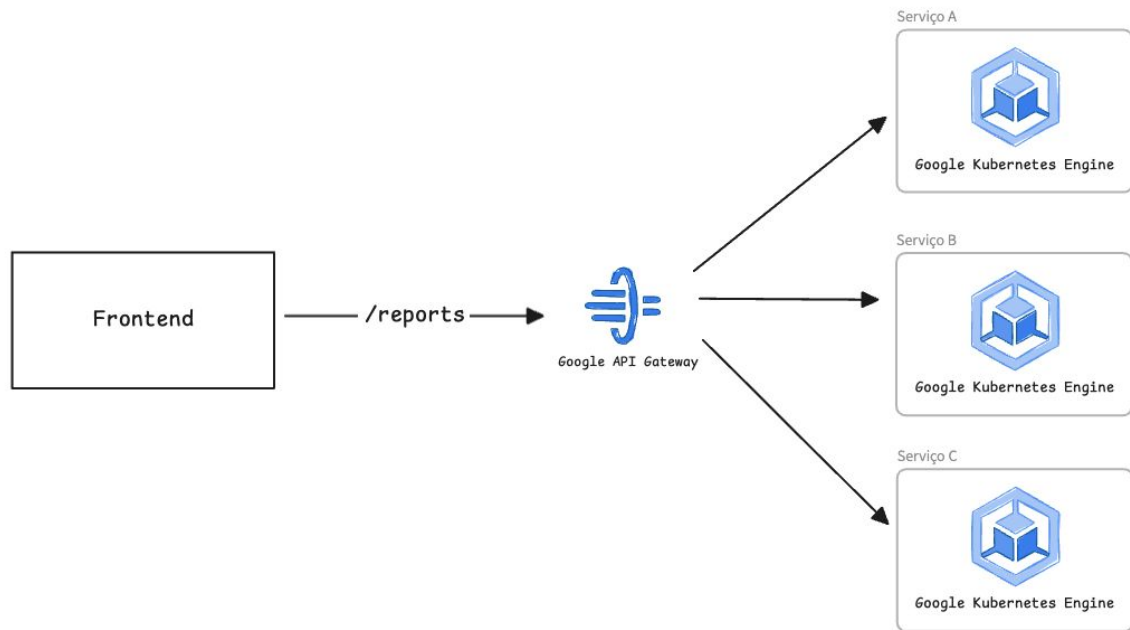
Visa garantir que o modelo de domínio externo jamais corrompa o modelo de domínio do Core da aplicação.



Google  
Developer  
Groups

# API Composition

Tratamento de consulta centralizado



Google  
Developer  
Groups

# API Composition

Tratamento de consulta centralizado

## Trade-offs

- Latência
- Disponibilidade
- Criar mais um serviço
- Depender de requests síncronas



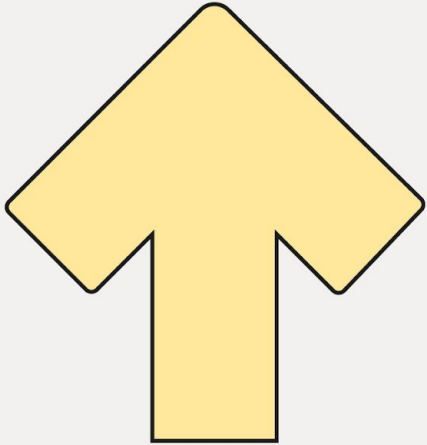
Google  
Developer  
Groups



09

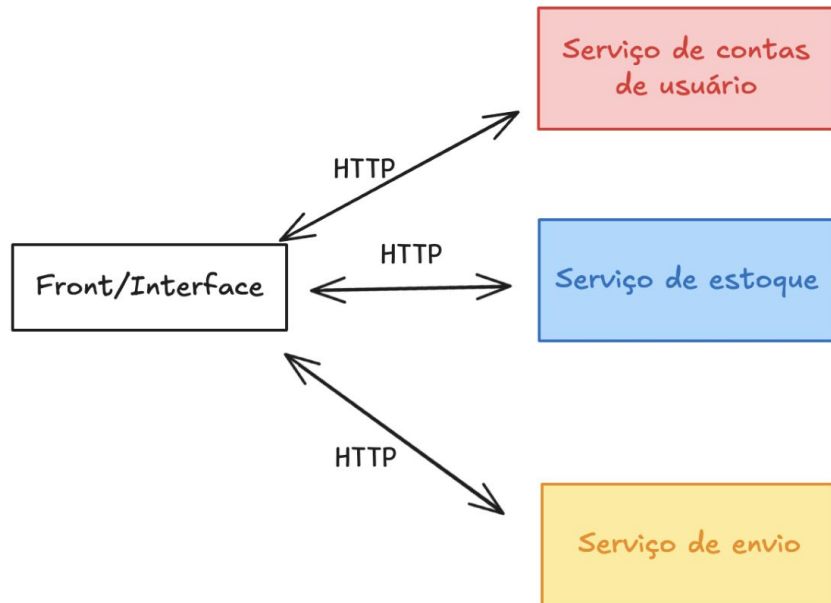
# Comunicação

Síncrona e  
Assíncrona



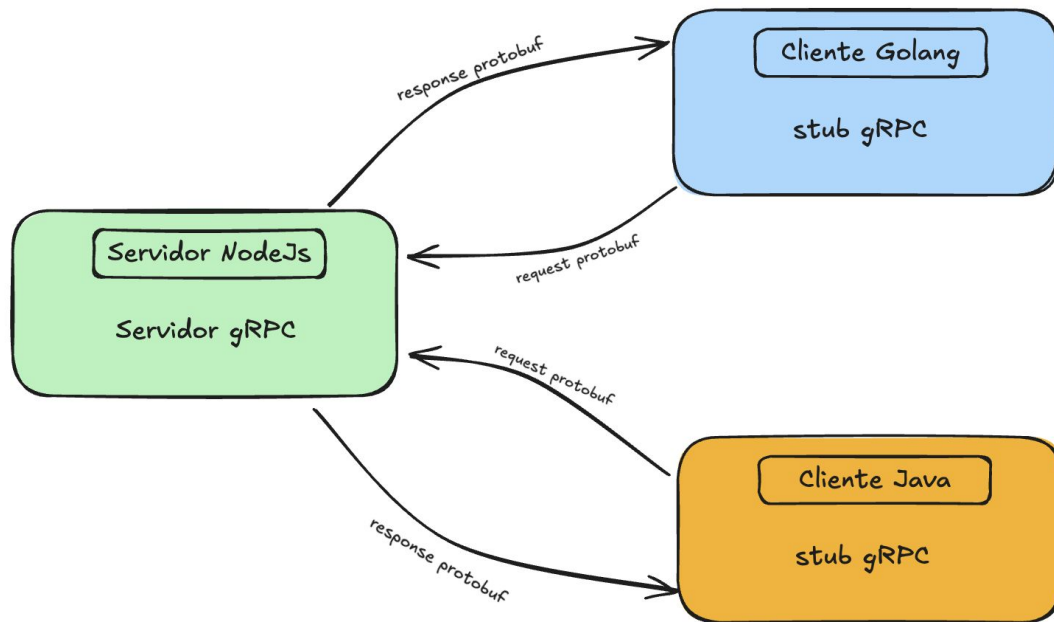
Google  
Developer  
Groups

# HTTP/REST (Síncrona)



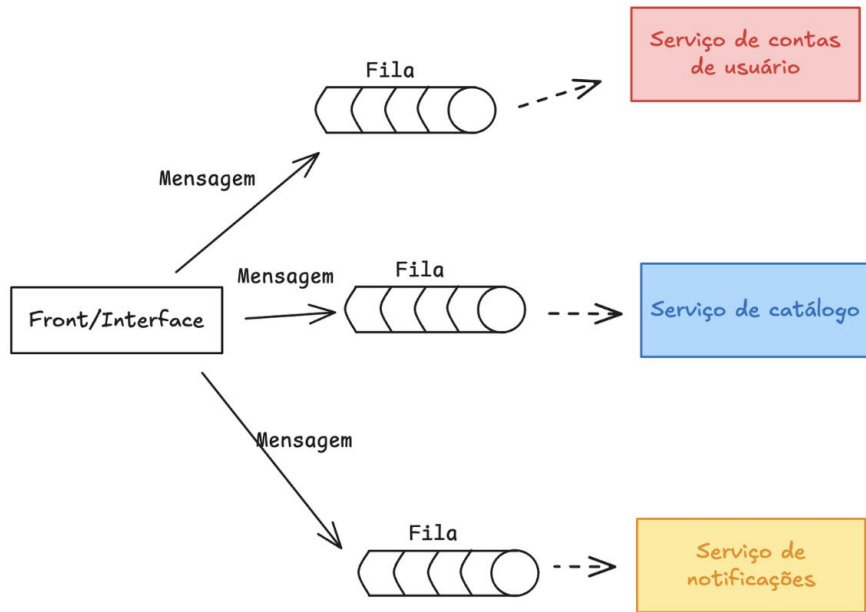
Google  
Developer  
Groups

# gRPC (Síncrona)



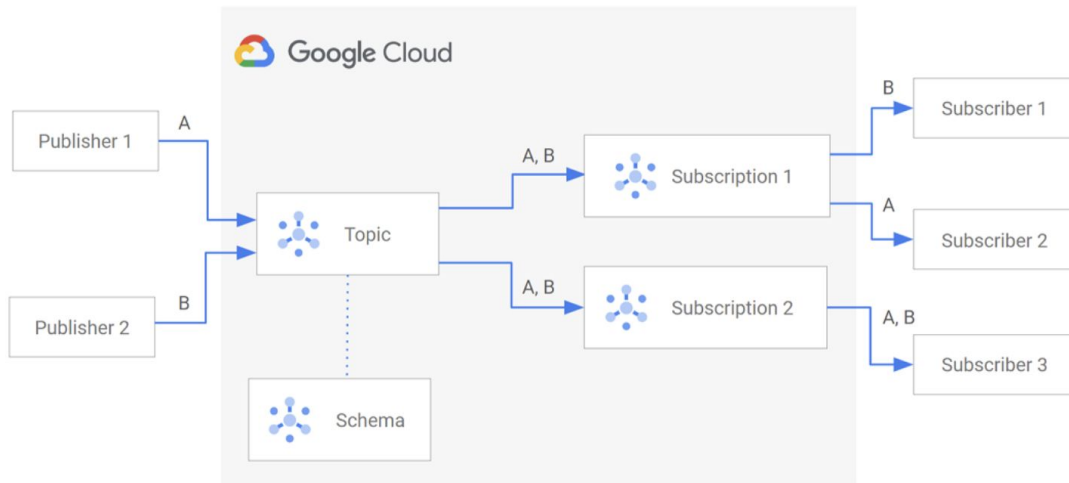
Google  
Developer  
Groups

# Mensageria (Assíncrona)



Google  
Developer  
Groups

# PUB/SUB (Assíncrona)



Google  
Developer  
Groups