

PROGRAMAÇÃO CONCORRENTE E DISTRIBUÍDA

Introdução à Programação Concorrente, Distribuída (e Paralela)

Aula 01 — Conceitos, sistemas distribuídos e arquitetura

Prof. Newton S. B. Miyoshi

newton.miyoshi@baraodemaua.br

Centro Universitário Barão de Mauá — Ribeirão Preto

PARTE 1

Conceitos

O vocabulário essencial antes de projetar sistemas.

PERGUNTA!



Qual a diferença entre
programação **Distribuída**,
Concorrente e **Paralela**?



UM ESPECTRO DE ACOPLAMENTO

Distribuída · Concorrente · Paralela

DISTRIBUÍDA

múltiplos computadores em
rede

◀ MENOR ACOPLAMENTO

CONCORRENTE

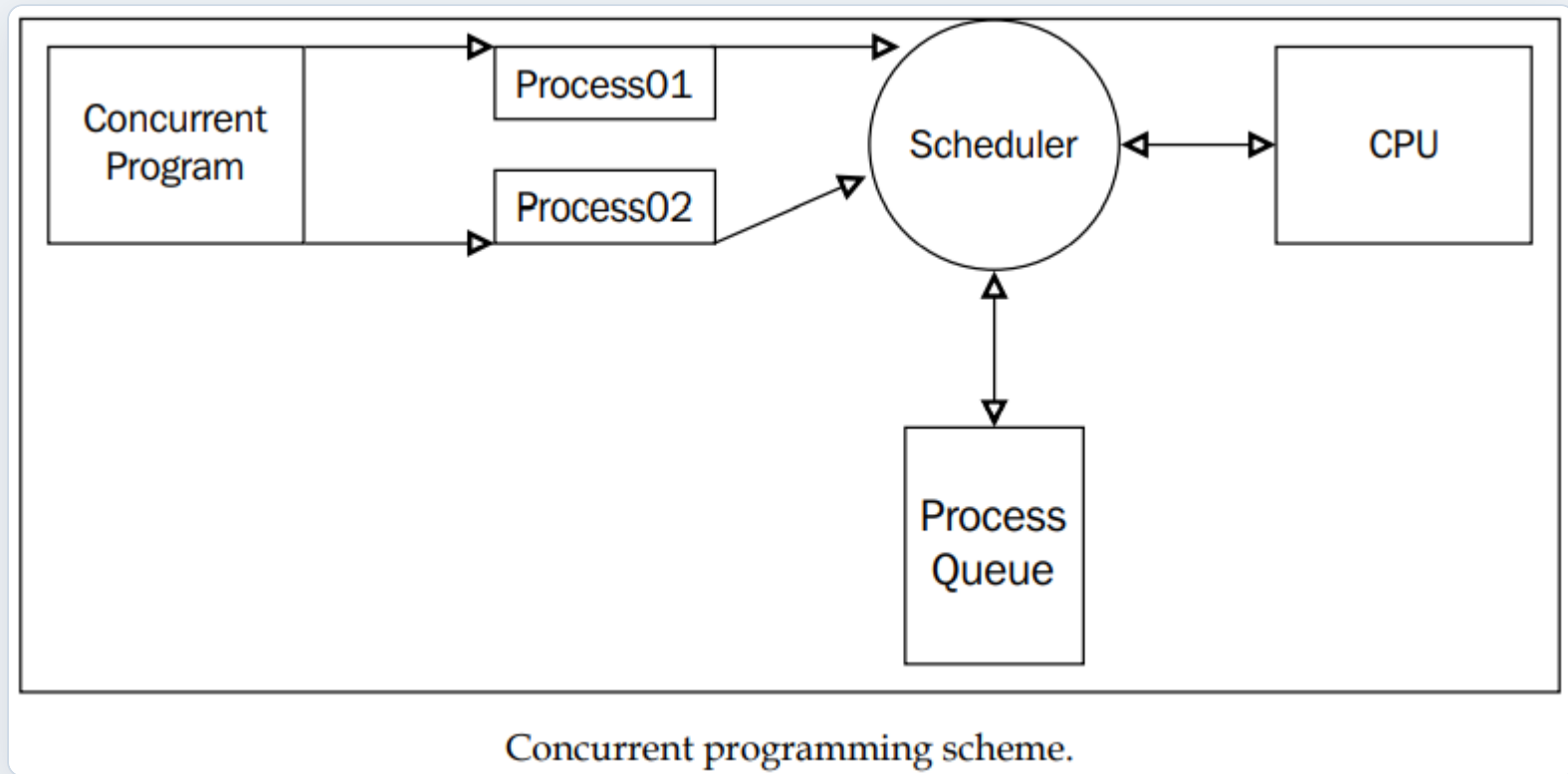
tarefas progridem
intercaladas

PARALELA

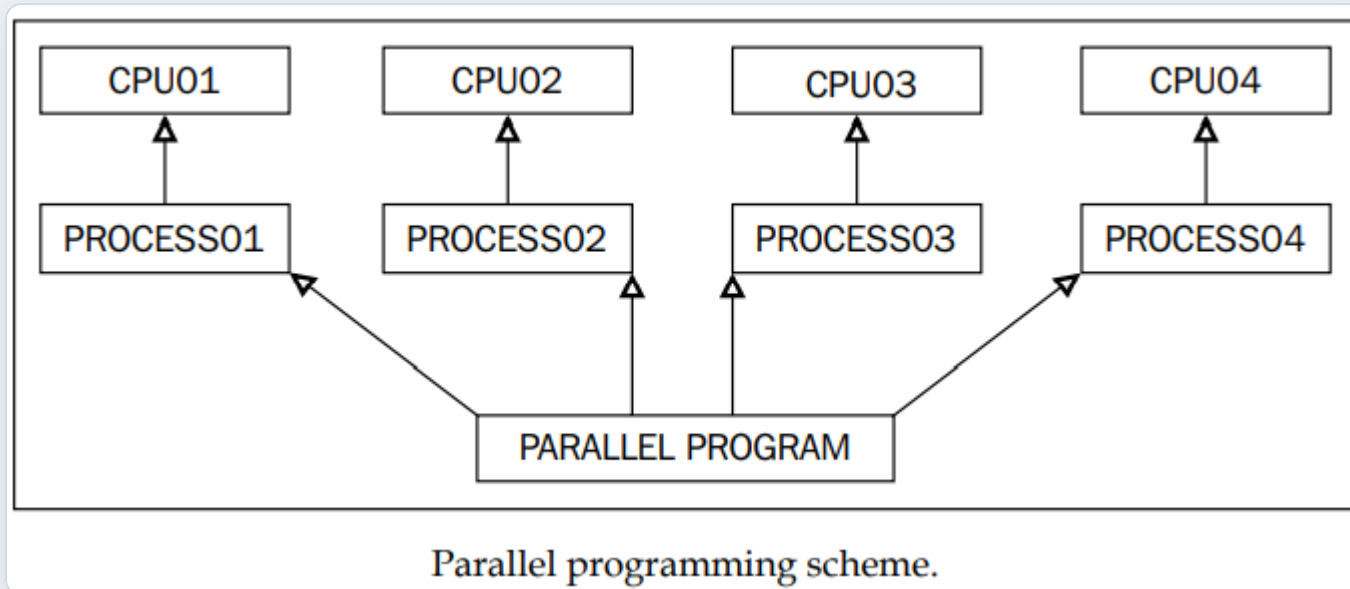
execução simultânea real

MAIOR ACOPLAMENTO ▶

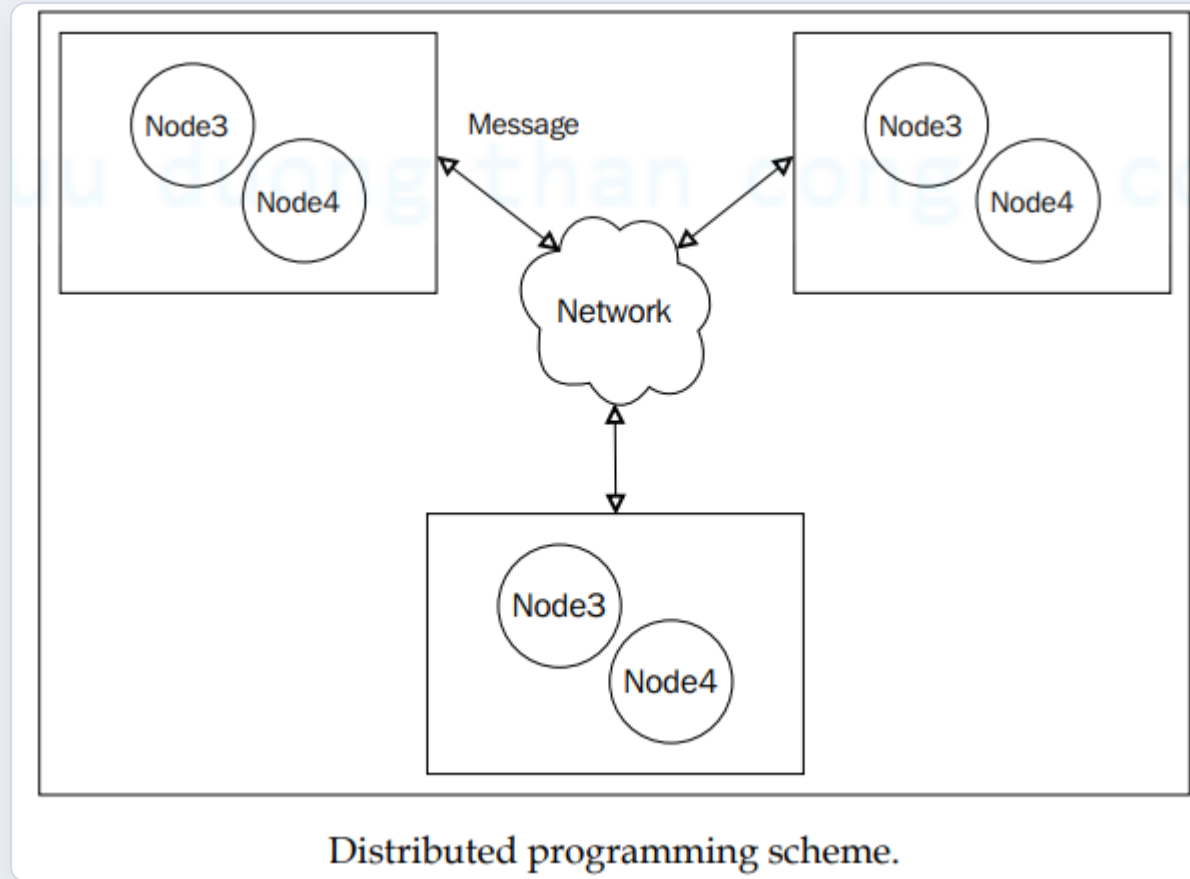
Programação Concorrente



Programação Paralela



Programação Distribuída



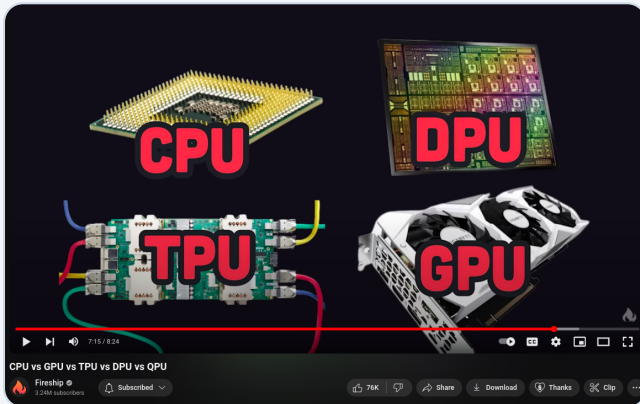
Computação Paralela

MATERIAL EM VÍDEO



YOUTUBE NVIDIA — GPU vs CPU:
pintando a Mona Lisa em 80
milissegundos youtu.be/8_ZTvG1WQxM

MATERIAL EM VÍDEO

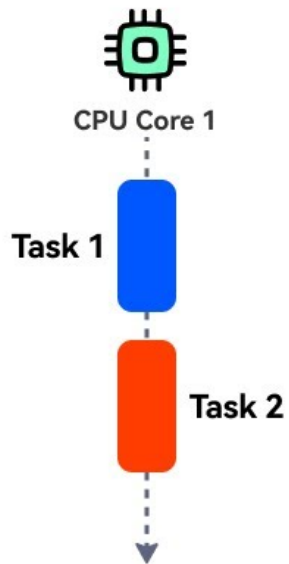


YOUTUBE CPU vs GPU vs TPU vs DPU youtu
[be.com/watch?v=r5NQecwZs1A](https://www.youtube.com/watch?v=r5NQecwZs1A)

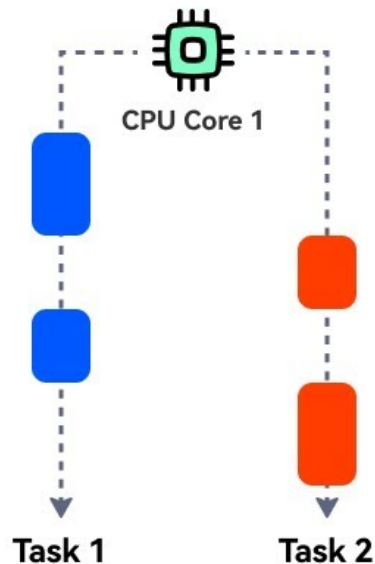
Concorrência não é Paralelismo

Concurrency is **NOT** Parallelism

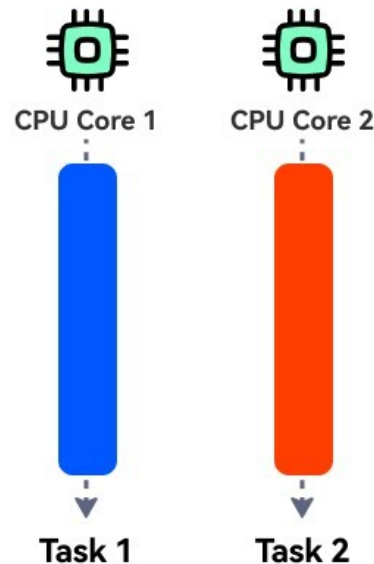
Not Concurrent,
Not Parallel



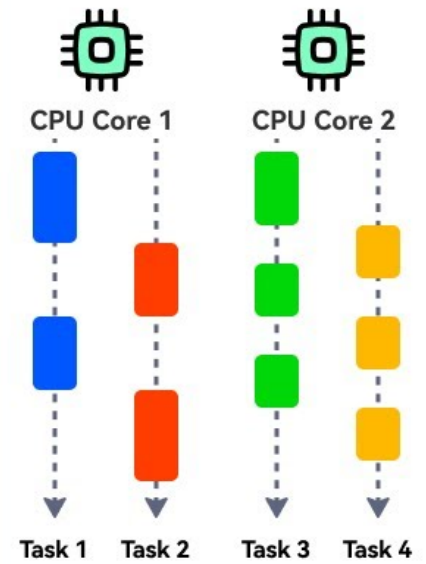
Concurrent
Not Parallel



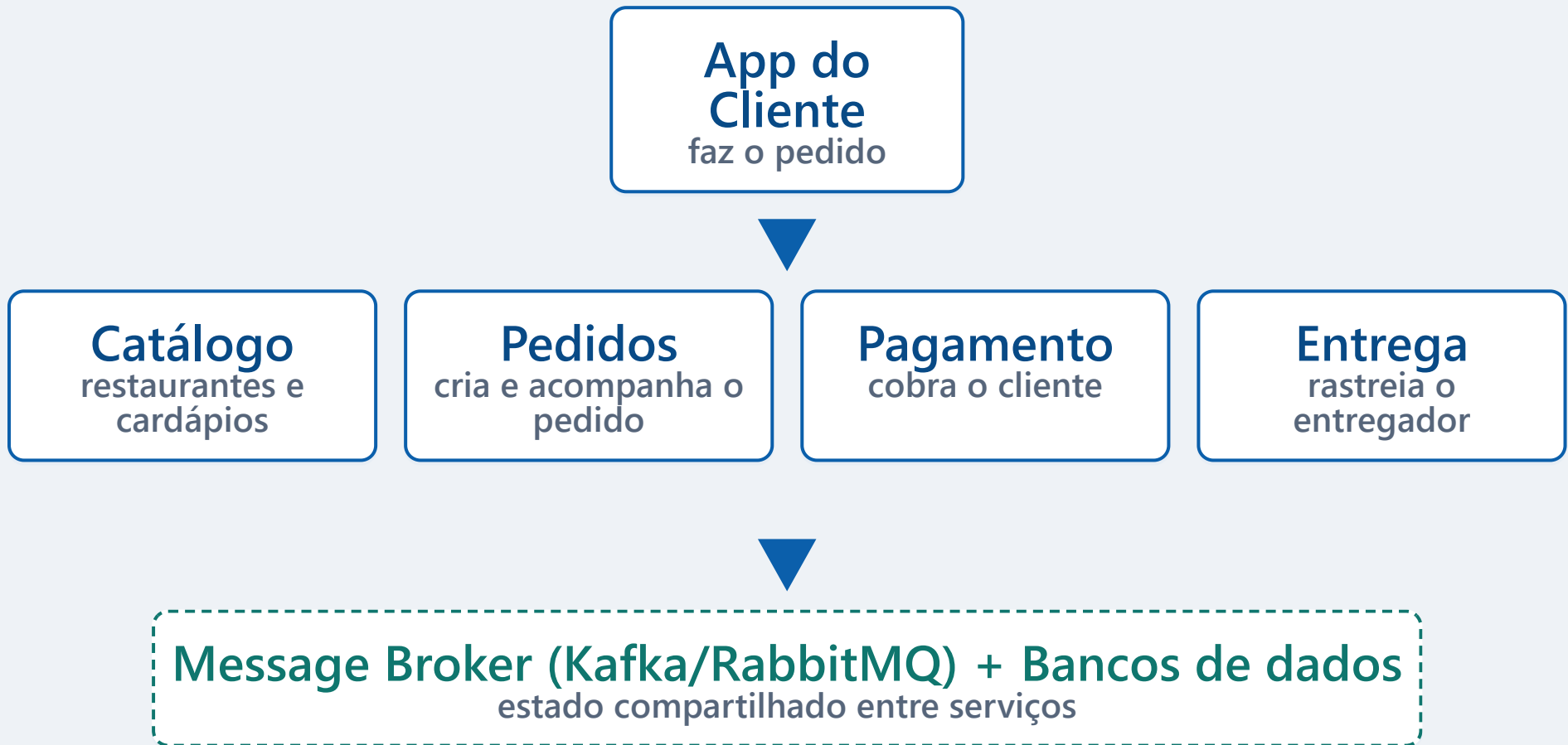
Not Concurrent
Parallel



Concurrent
Parallel



Exemplo real: app de delivery (ex.: iFood)



O que é um Sistema Distribuído?

“Uma coleção de elementos de computação independentes que aparenta a seus usuários como um sistema único e coerente.”

- Sistema construído com programação distribuída.
- Elemento de computação (computador) = nó (node).

Sistema Distribuído: duas visões



Hardware

Computadores (nós)
conectados por uma rede.



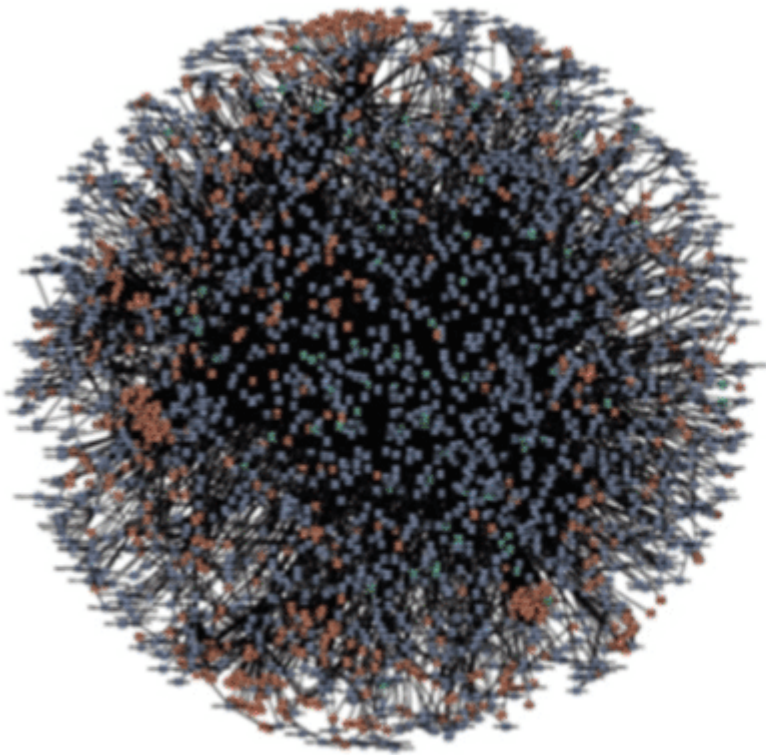
Software

Camada que dá a ilusão de um
sistema único.

Características importantes de um SD

- Sistema “aparentemente” único e coerente
 - Ilusão para usuários e desenvolvedores
 - Transparência da distribuição
 - Não se sabe qual nó realiza o processamento

A escala dos sistemas reais



amazon.com®

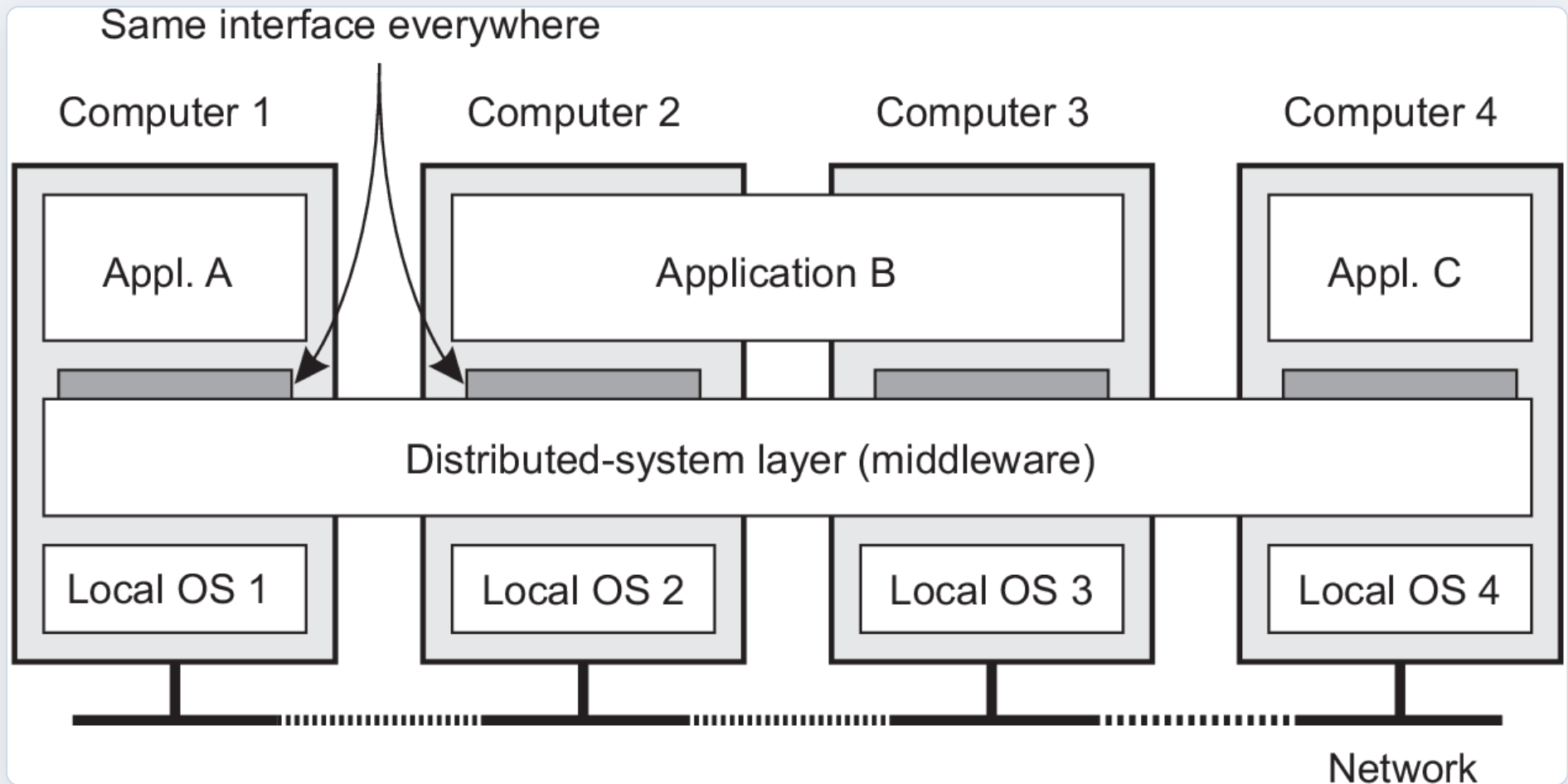


NETFLIX

O que é um Middleware?

- Camada do meio entre aplicações — **abstração** que facilita implementar SDs.
- Atua como “**broker**” / corretor.
- Define interfaces específicas.

Middleware: a mesma interface em todo lugar



Serviços providos por um Middleware

Comunicação

- Remote Procedure Call (RPC)

Transação

- Garante atomicidade
- Todas as chamadas acontecem — ou nenhuma

Composição de serviços

- Ex.: Google Maps

Confiabilidade

- Qualidade dos dados
- Entrega de mensagens

ATIVIDADE EM GRUPO

Pesquise exemplos reais

Em grupo, pesquisem e anotem pelo menos um exemplo real para cada categoria — sistemas, aplicativos, empresas ou tecnologias que vocês conheçam ou encontrem pesquisando.

Sistema Distribuído

qual sistema roda em vários computadores/nós, cooperando pela rede?

Programação Concorrente

qual sistema lida com muitas tarefas intercaladas, disputando os mesmos recursos?

Programação Paralela

qual sistema executa tarefas de verdade ao mesmo tempo, em múltiplos processadores?

Preparem-se para compartilhar os exemplos e justificar a escolha com a turma.

Possíveis respostas

Distribuído

- WhatsApp, e-mail (SMTP)
- Netflix, Amazon, iFood
- DNS, sistema bancário

Concorrente

- Sistema operacional (multitarefa)
- Servidor web atendendo várias requisições
- Navegador com várias abas/downloads

Paralelo

- Renderização gráfica em GPU
- Edição/compressão de vídeo (multi-core)
- Treinamento de modelos de IA

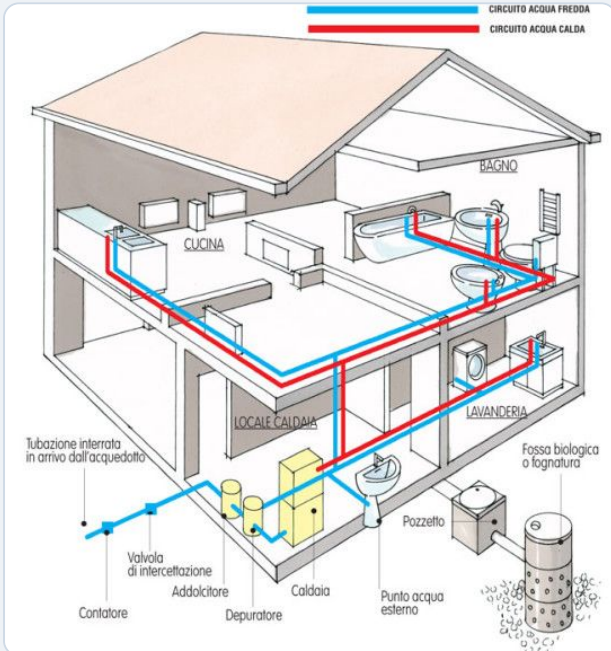
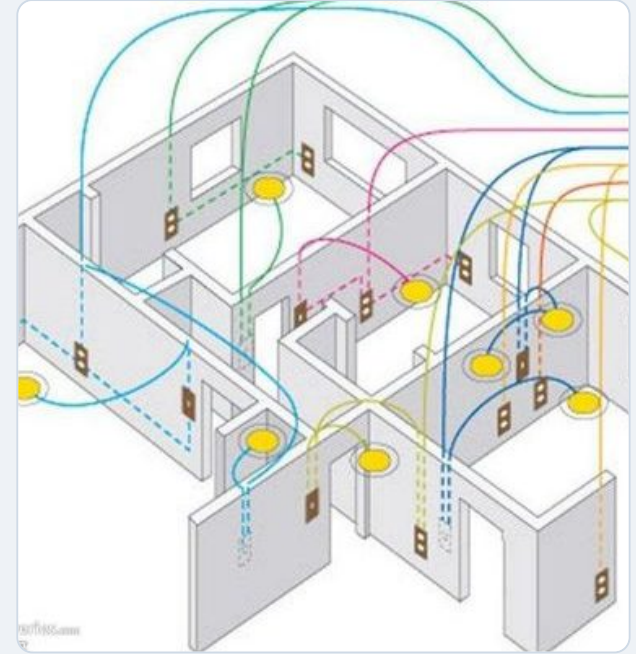
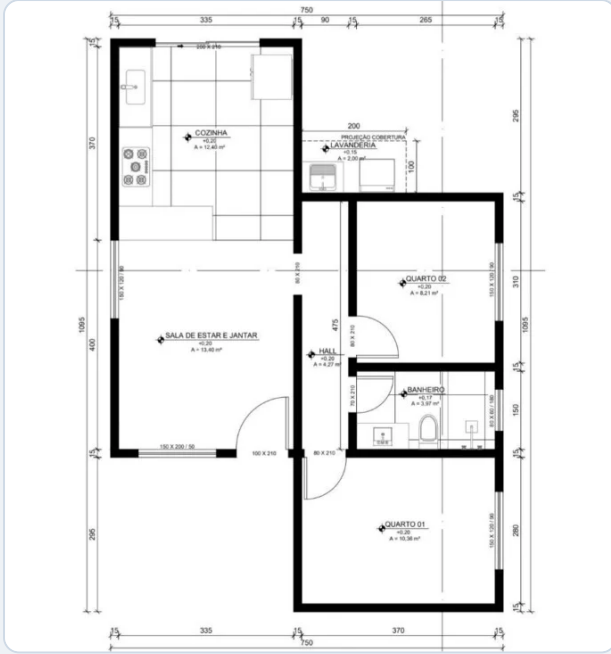
Não há resposta única — o importante é justificar por que o exemplo se encaixa na categoria. Muitos sistemas reais combinam as três!

PARTE 2

Arquitetura de Sistemas

System Design — middleware e componentes.

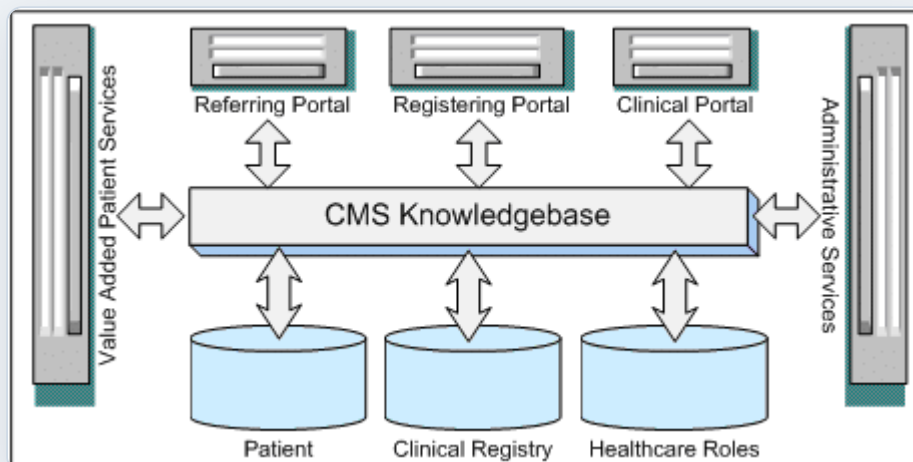
Projetar software é como projetar construções

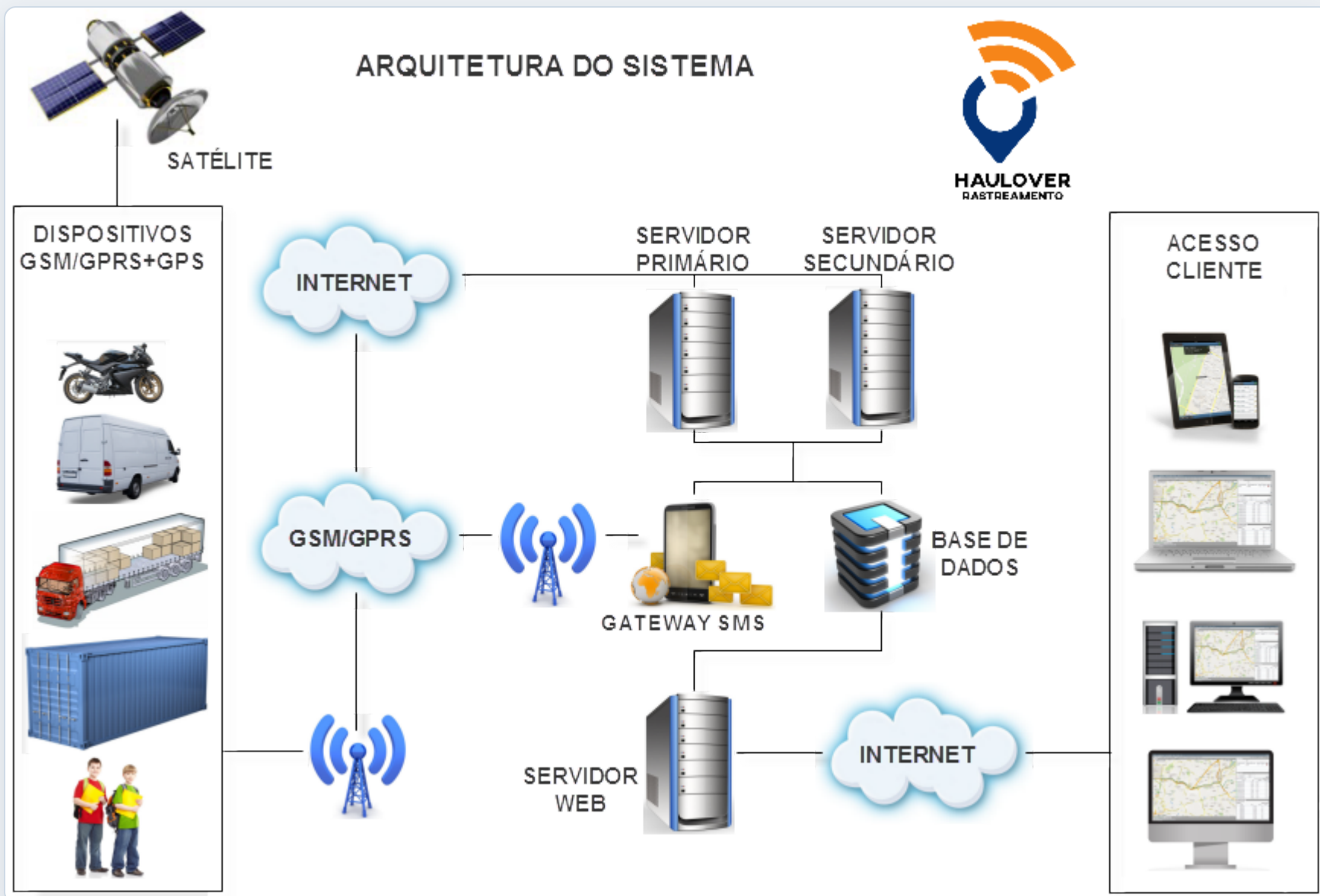


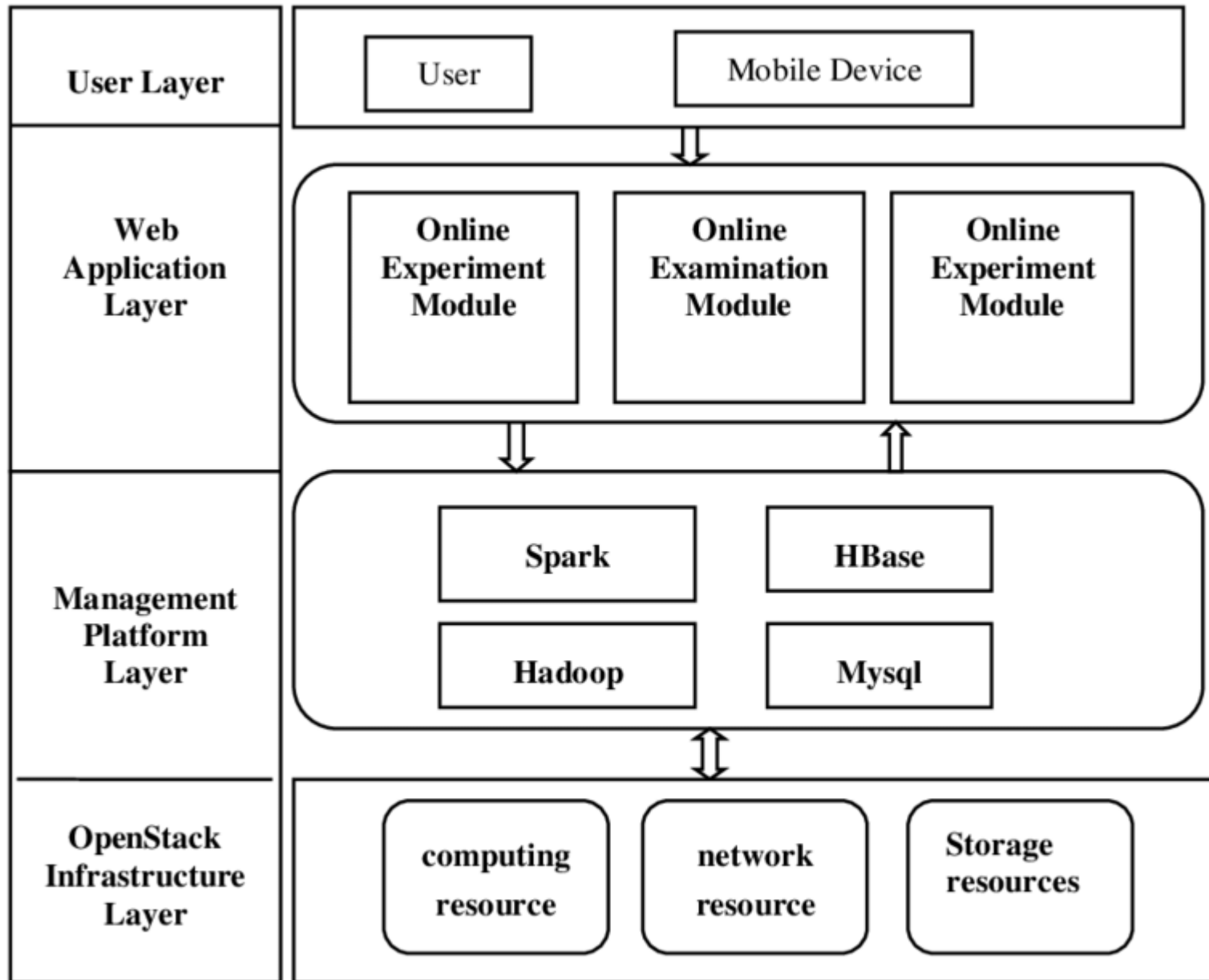
PARTE 3

Exemplos de arquiteturas reais

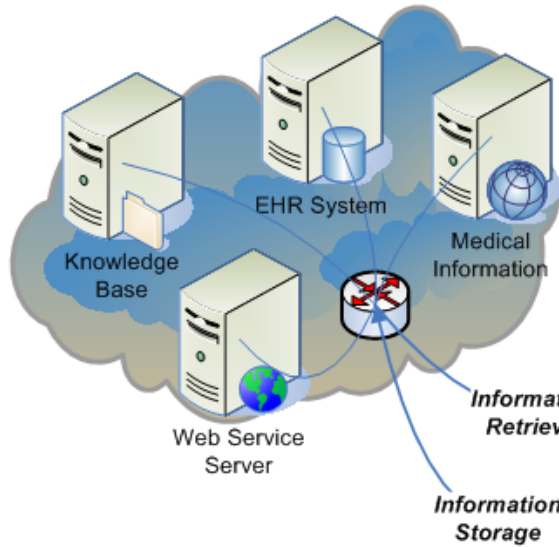
Sistemas distribuídos em diferentes domínios.



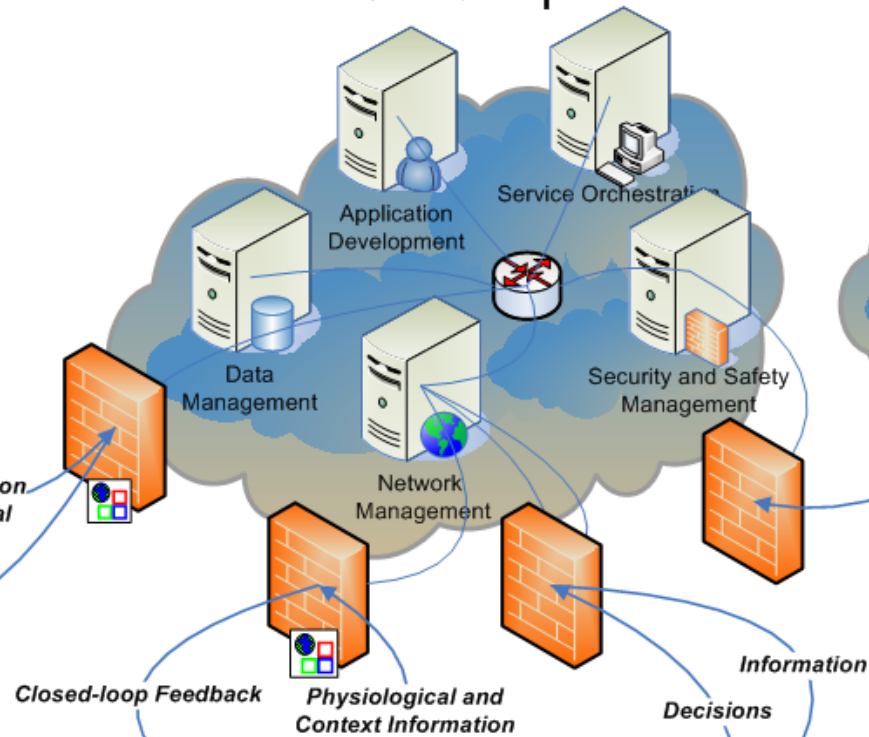




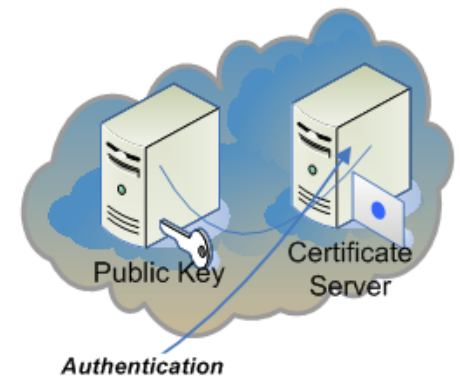
Health Information Systems



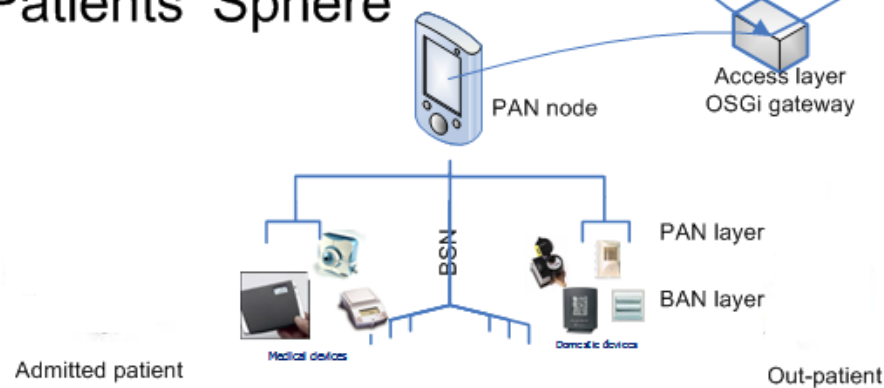
The REACTION platform



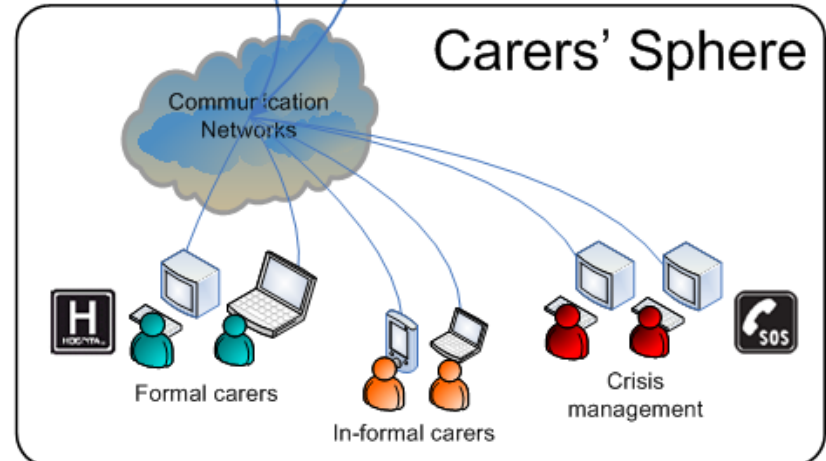
Authentication

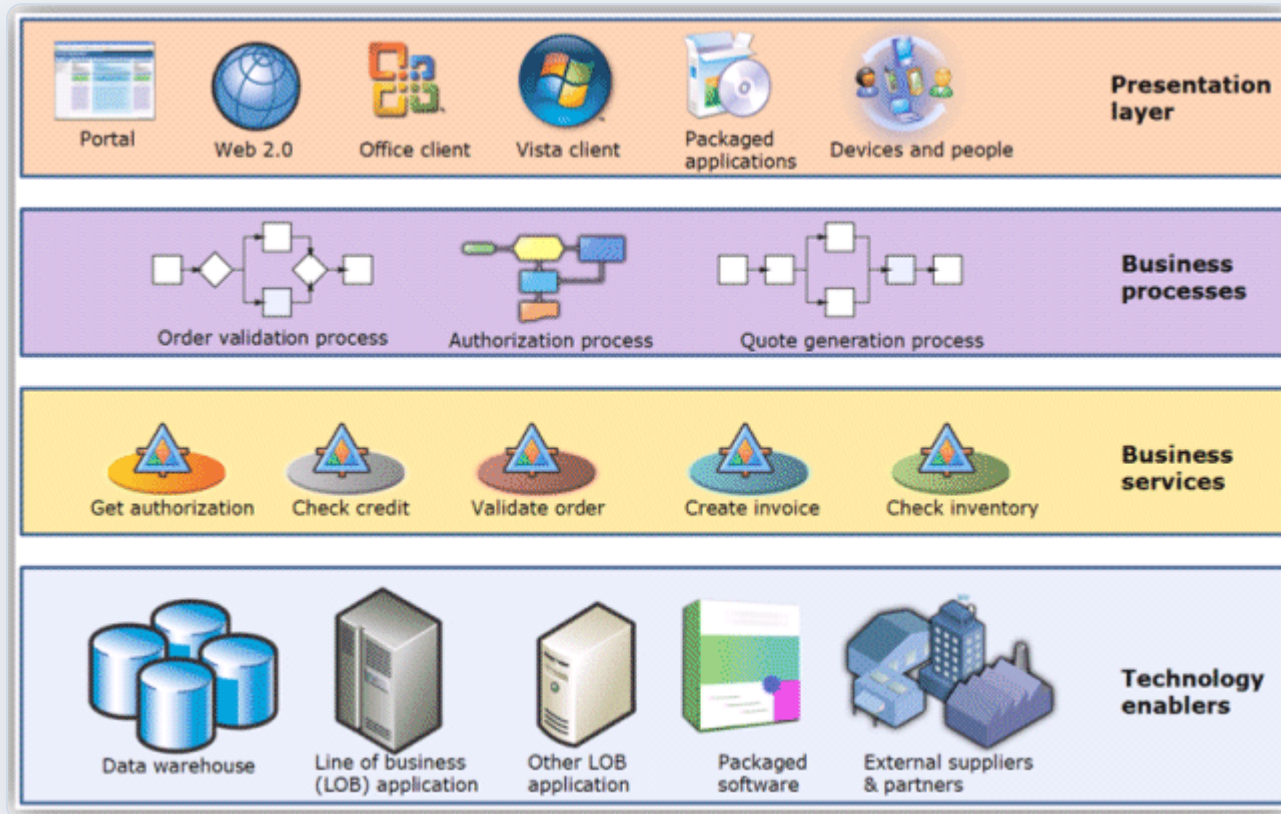


Patients' Sphere

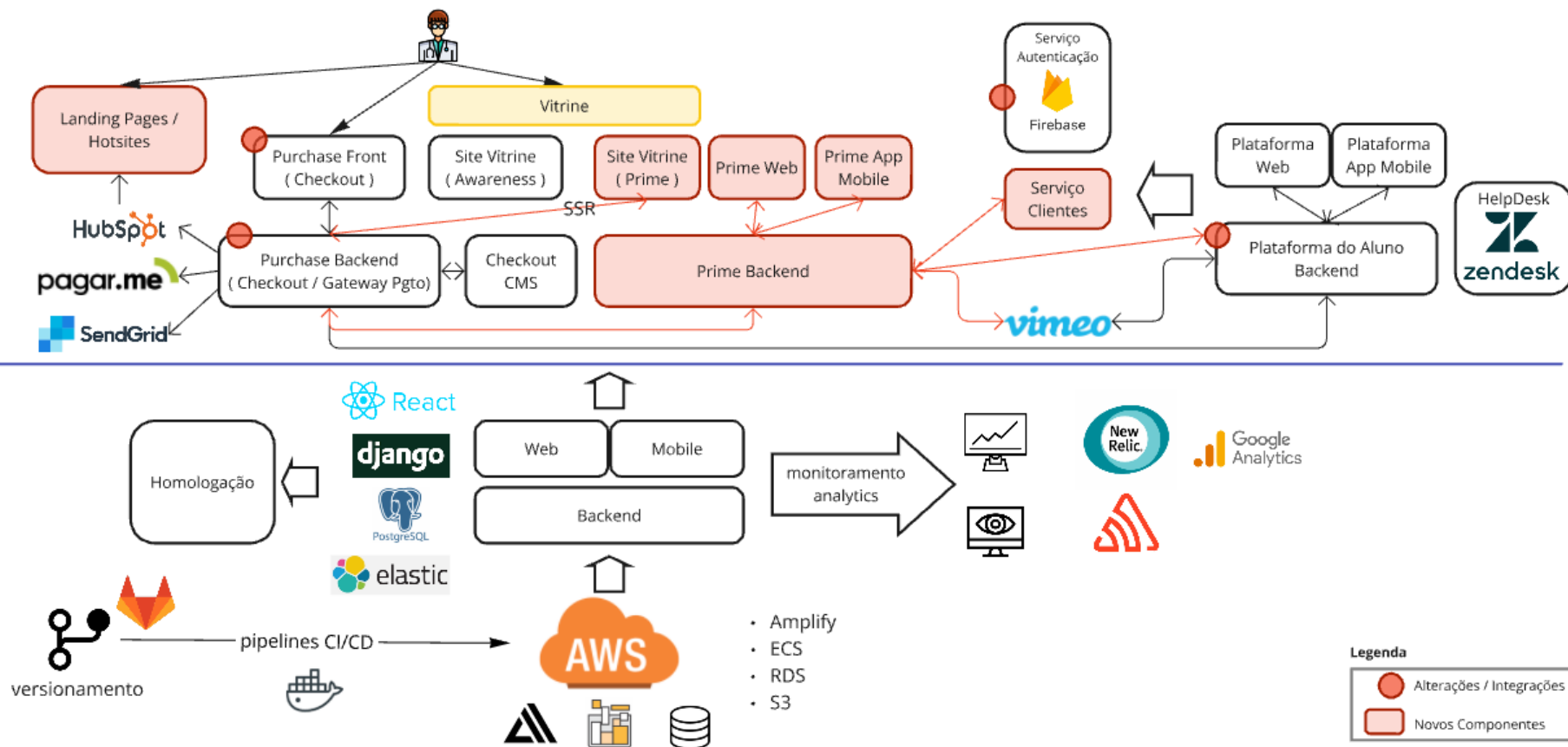


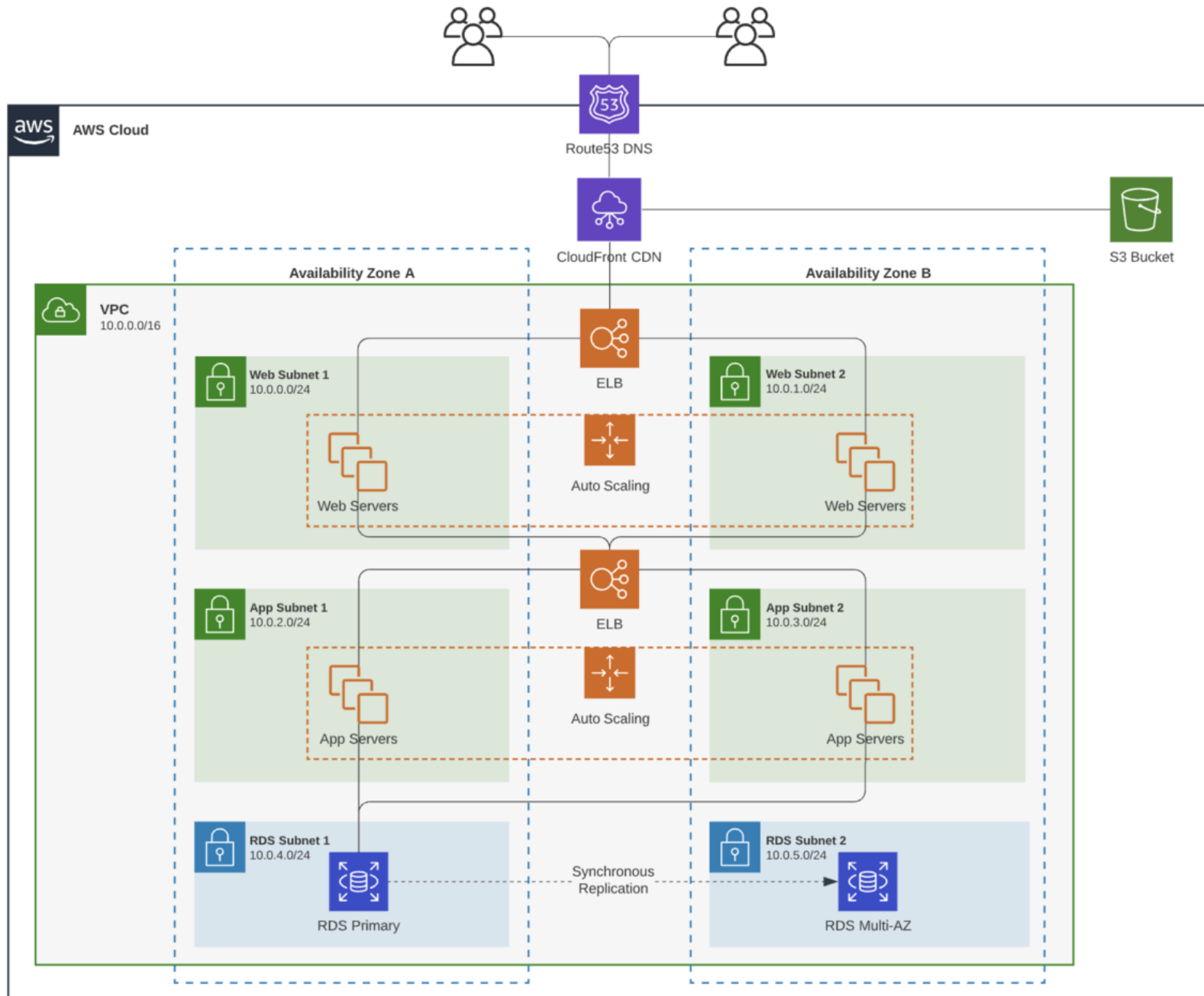
Carers' Sphere

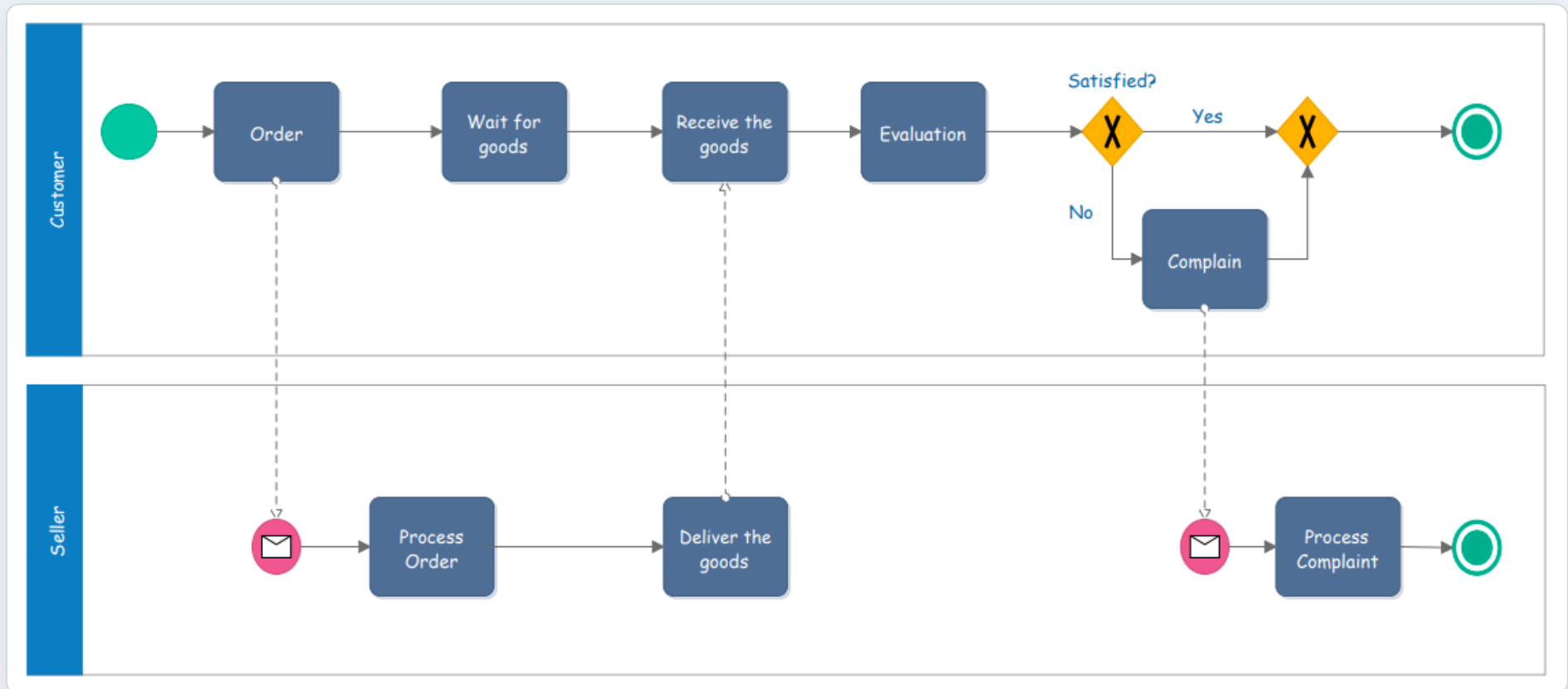


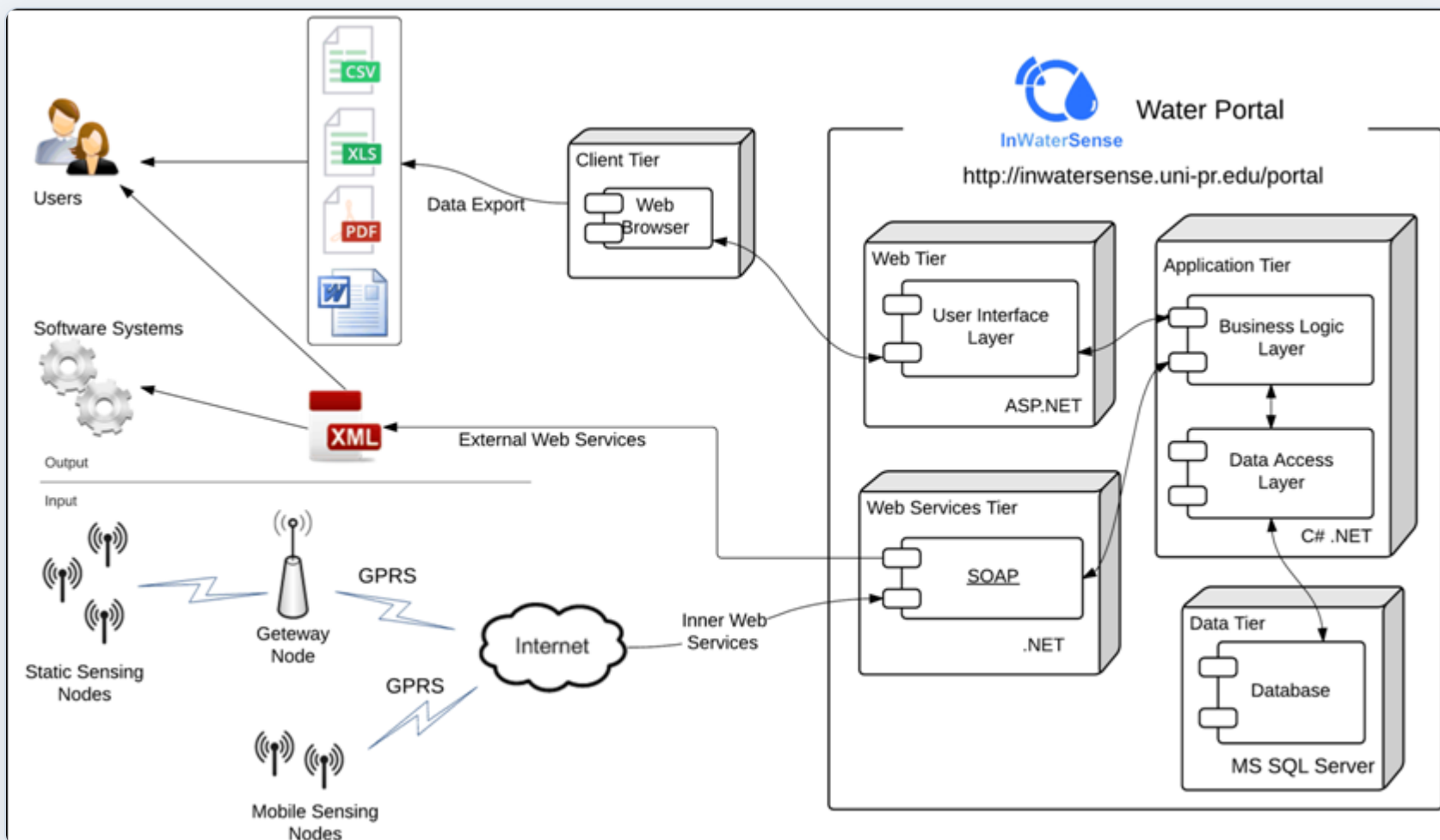


QUADRO COLABORATIVO









ATIVIDADE EM GRUPO

Cenário A — “Café com pão”

Nossa startup “Café com pão” cresceu! Precisamos projetar o sistema para dar suporte à rede de padarias e a toda a logística de entrega. Sua equipe é responsável por projetar esse ambiente.

REQUISITOS

Clientes se cadastram, assinam cestas e fazem pedidos avulsos por app móvel e web. Entregadores recebem pedidos por app móvel. Padarias credenciadas usam uma aplicação web para receber pedidos, preparar cestas e entregar. Clientes acompanham o pedido em tempo real e pagam pelo próprio app.

Desenhe uma ideia da arquitetura inicial desta solução!

Cenário B — “BikeCare”

A “BikeCare” está revolucionando o mercado de manutenção de bicicletas. Projete o sistema que dá suporte à rede de oficinas credenciadas e à logística de coleta e entrega de bikes.

- Clientes solicitam manutenção/personalização por app ou web; escolhem levar à oficina ou agendar coleta em casa.
- Oficinas recebem pedidos e atualizam o status em tempo real (diagnóstico, peças, em andamento, concluído).
- Entregadores recebem rotas de coleta/entrega e registram fotos antes/depois.
- Pagamento pelo app, com notificações push e acompanhamento em tempo real.
- Gerenciar estoque de peças integrado a fornecedores para reposição automática.

Ferramentas para desenhar arquiteturas

draw.io

Excalidraw

Lucidchart

Google Slides

Miro

FigJam

OUTROS MATERIAIS



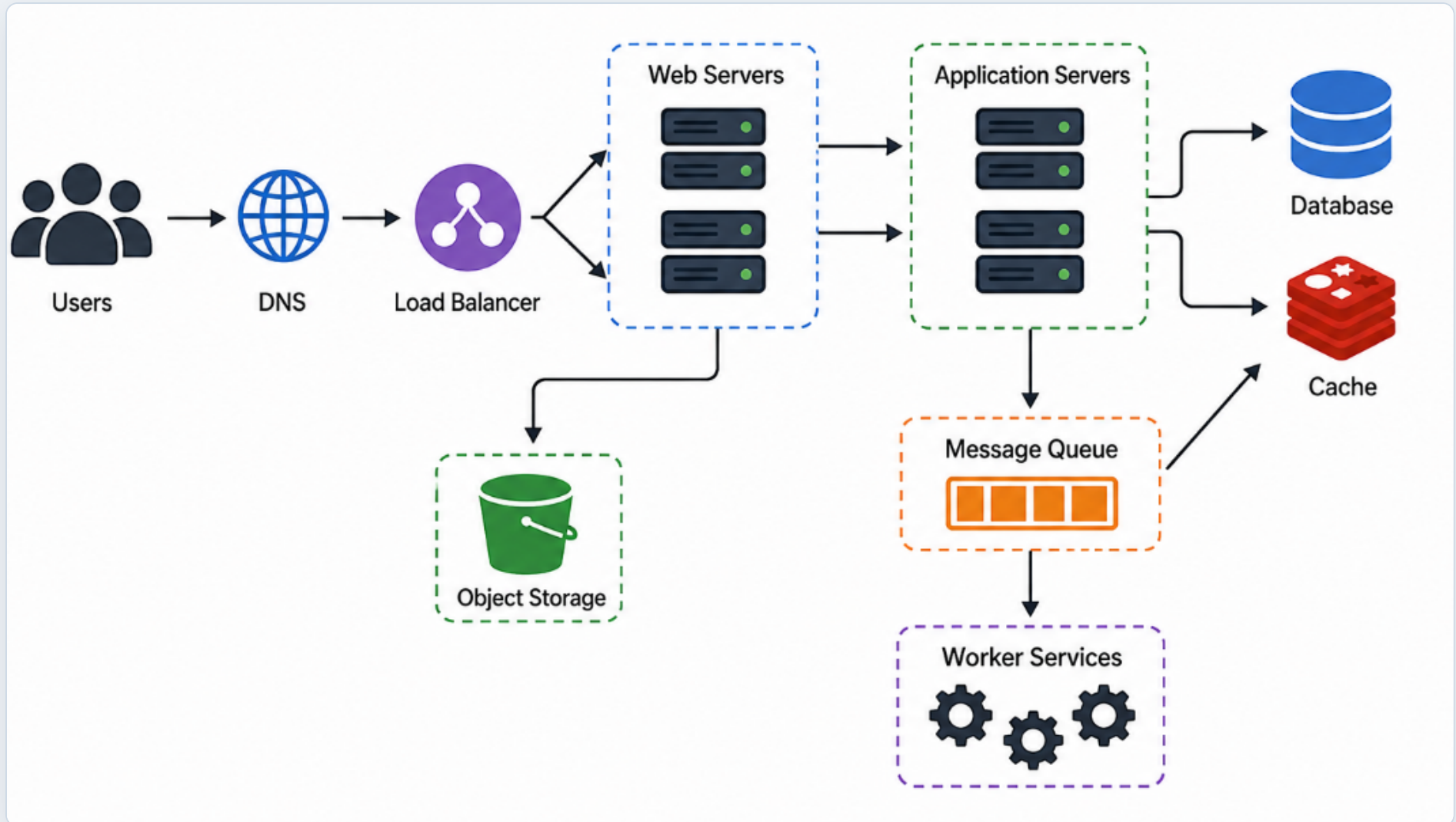
VÍDEO · HIPSTERS PONTO TUBE **Arquitetura de sistemas com Fabio Akita** [youtube.com](https://www.youtube.com) — #HipstersPontoTube

OUTROS MATERIAIS



VÍDEO Modelagem de Software é Difícil? “Ver” vs “Enxergar” [youtube.com](https://www.youtube.com)

Exemplo de arquitetura de um sistema real



Referências bibliográficas

- TANENBAUM, A. S.; VAN STEEN, M. Distributed Systems. 3. ed. 2018. ISBN 978-90-815406-2-9. distributed-systems.net
- GRAMA, A.; GUPTA, A.; KARYPIS, G.; KUMAR, V. Introduction to Parallel Computing. 2. ed. karypis/parbook
- PALACH, J. Parallel Programming with Python. Packt Publishing, 2014.