**UNIMETROCAMP WYDEN**

CARLOS FILIPE PRADO MENEZES

RA: 1510028840

10º SEMESTE ENG. DA COMPUTAÇÃO

**MICRO SERVIÇOS**

SISTEMAS DISTRIBUÍDOS

**Campinas - SP**

**2019**

# 1 - INTRODUÇÃO

De forma bem resumida vamos dar uma pequena introdução no termo micro serviços que segundo dados históricos sua filosofia tinha surgido em 1978 por Mcllroy, Pinsen e Tague, pioneiros do sistemas UNIX em uma publicação “Unix Time-Sharing System: Forward” porém surgiu de forma oficial em 2011 em uma conferência de arquitetura de software onde foi apresentado um novo estilo de arquitetura de sistemas com o intuito de serem mais flexíveis, escaláveis e com uma manutenção mais simples comparada a monolíticos.

# 2 - Caso real

O caso que iremos apresentar ocorreu em um projeto da Ambev que tinha sua arquitetura atual subdividas em vários servidores, tendo cada um deles uma devida aplicação, como redis, app gateway e etc.

Apesar de possuir vários servidores tudo aquilo era apenas um projeto porém isso dificultava na manutenção e na realização de deploys e principalmente no suporte.

Analisando todo o contexto e analisando a arquitetura atual foi realizado um novo desenho de arquitetura onde iria contemplar todas as devidas aplicações porém em uma arquitetura diferente. Na época em que estávamos realizando esse projeto optamos por swarm e docker, creio que no cenário atual nós teríamos alterado apenas o swarm pelo K8. Cada aplicação que eles tinham em servidores nos transformamos em micro serviços através do swarm, ou seja, app gateway estava em um container, o redis estava em outro container e todas as imagens ficava armazenadas em um registry na Azure.

# 3 - Pontos positivo

* Os deploy ficaram muito mais rápidos e flexíveis. Rodando o comando docker deploy apontamento para o registry ele realiza o deploy podendo realizar a alteração apenas em uma determinada imagem.
* O projeto ficou extremamente escalável, onde iniciou de forma bem pequena em um ambiente de QA indo para uma estrutura enorme.
* Ficou muito fácil de dar suporte e debugar erros de aplicação.

# 4 - Pontos negativos

* Foi extremamente complexo a realização do projeto tanto para o time de infra e principalmente para o time de dev, que tinha que criar o docker file e composer.