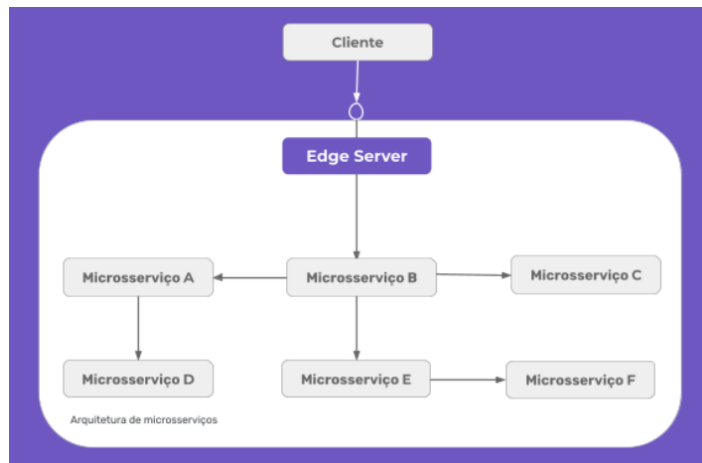




Introdução

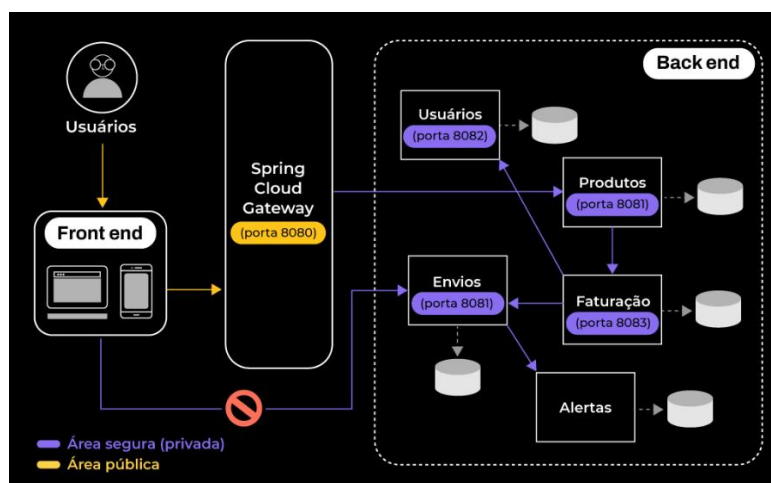
Em resumo, este padrão nos permite comunicar o mundo externo com nosso ecossistema de microsserviços, tendo acesso apenas aos microsserviços que precisam ser consumidos por um usuário externo (entendendo o usuário como uma aplicação frontal, um sistema monolítico, etc.), com o objetivo de proteger a integridade e o uso de nossas API. Aqui está um lembrete de uma imagem que vimos nas primeiras aulas que será útil para o que está por vir.



Introdução a Spring Cloud Gateway

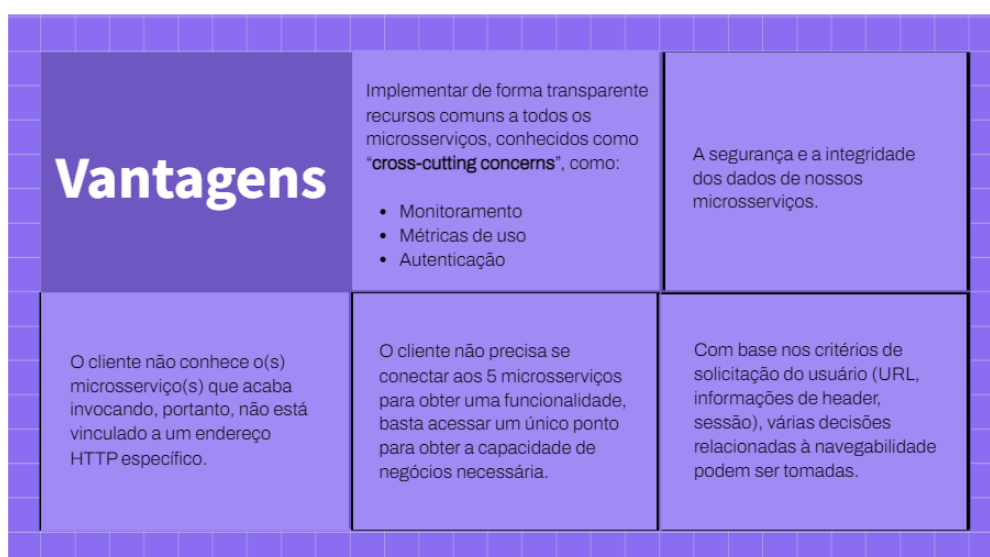
A solução Spring Cloud nos fornece uma estrutura chamada Spring Cloud Gateway, que implementa o padrão Edge Server e nos dá várias ferramentas para tirar proveito deste ponto em comum de comunicação com o ecossistema de microsserviços de sistemas externos.

Vamos lembrar da nossa DH Fintech, que vimos na primeira aula, e analisar a forma como a solução é comunicada.



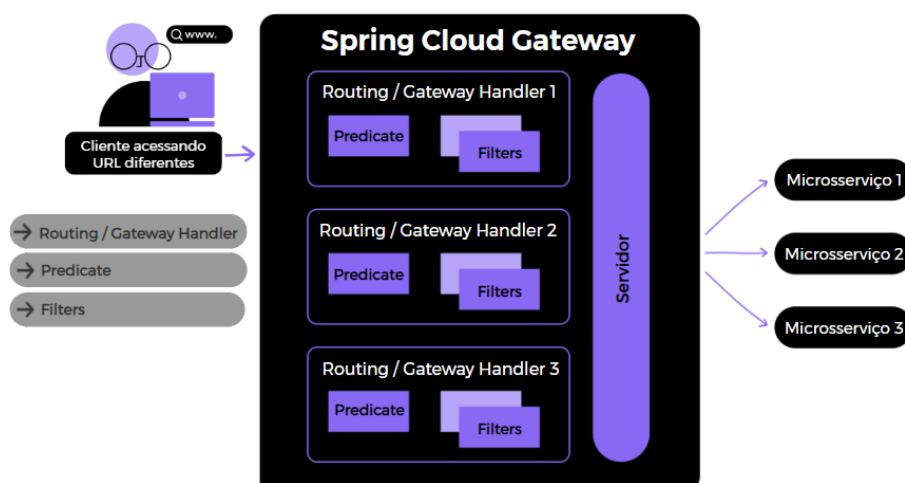


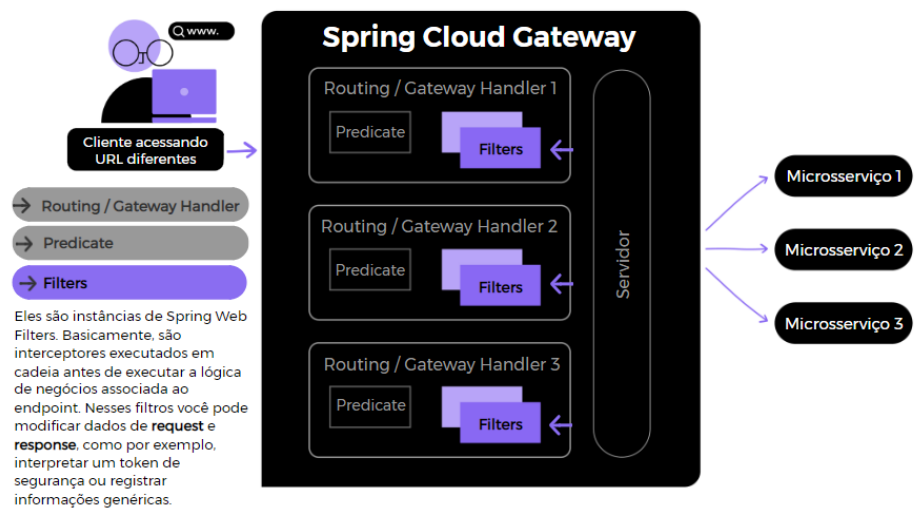
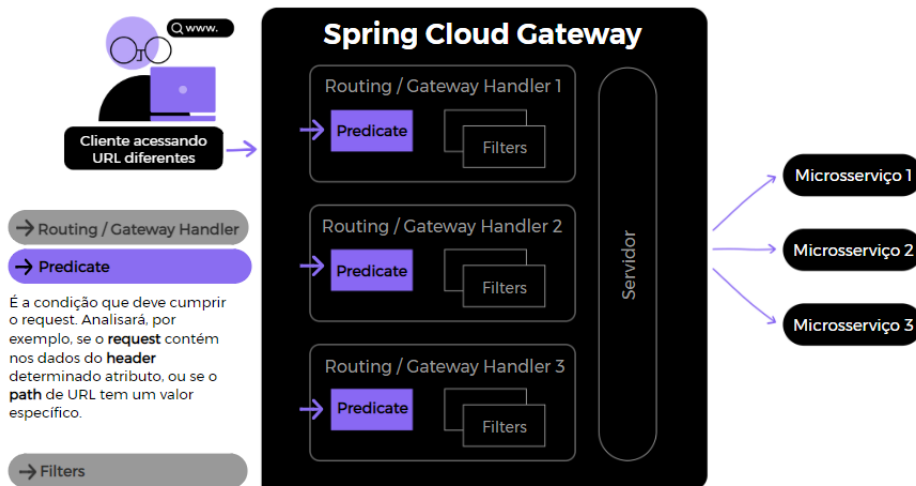
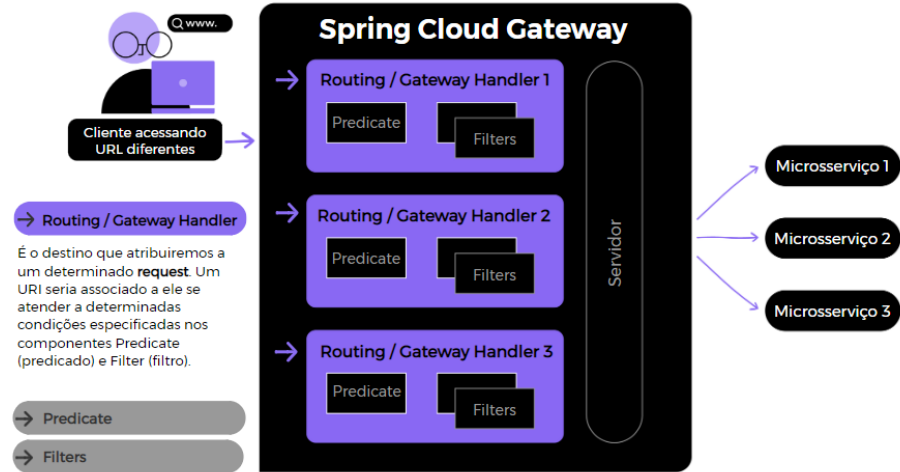
Como podemos ver na imagem, temos dois canais de comunicação: um público e um privado. O canal azul ou seguro é composto de comunicações internas entre microserviços em portas diferentes de 8080, que não podem ser acessadas de fora da rede. Por esta razão, um front end não poderia invocar estes serviços diretamente. Enquanto o canal verde ou público é usado para dar acesso a partir de nosso Cloud Gateway para o mundo exterior. A seguir, veremos algumas das vantagens desta arquitetura.



Componentes de Spring Cloud Gateway

Agora vamos ver quais são os componentes que formam o Spring Cloud Gateway.

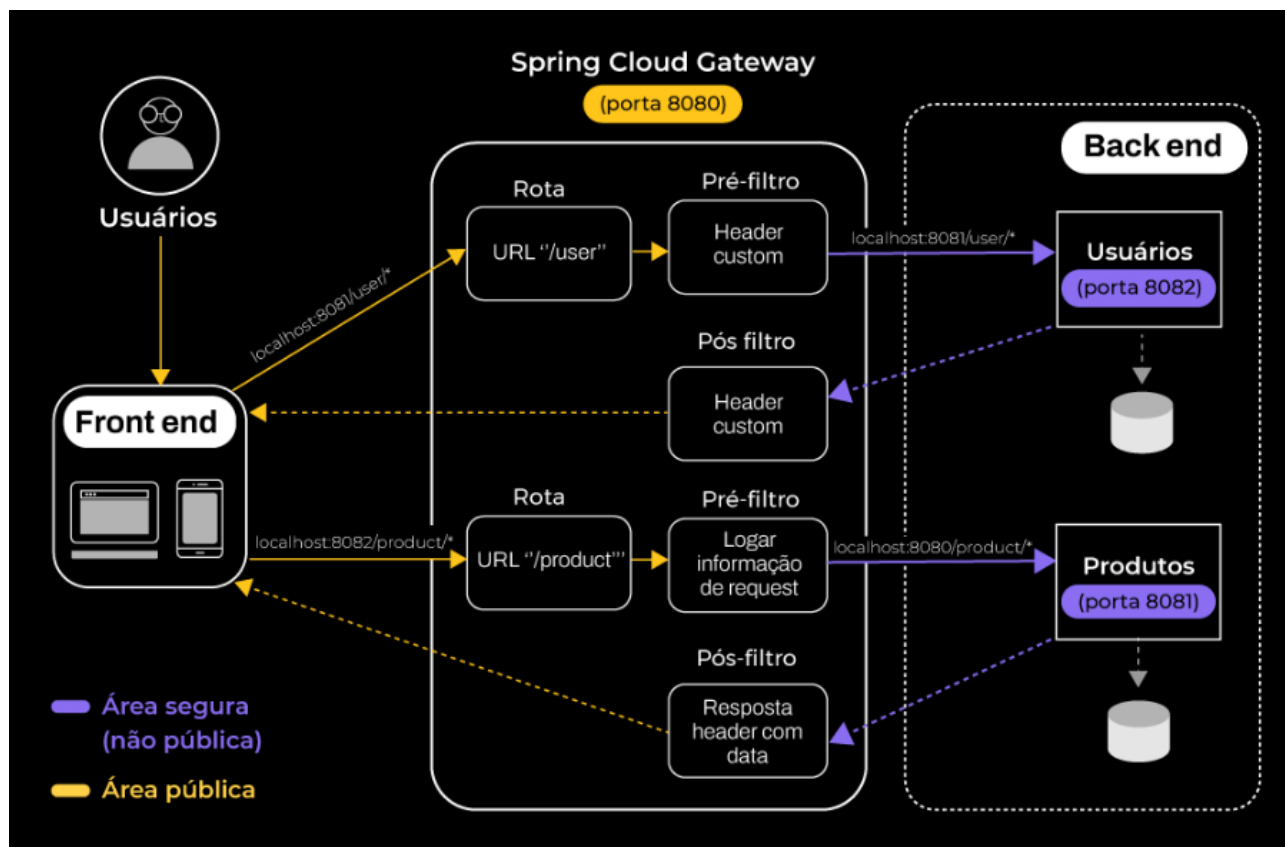






Implementação de Spring Cloud Gateway

A implementação será baseada em dois microsserviços da nossa DH Fintech: o microsserviço de usuário e o de produto. Tomamos o seguinte diagrama como ponto de partida para implementar os componentes que temos visto teoricamente.



Finalmente, observaremos a configuração de um projeto de gateway.