

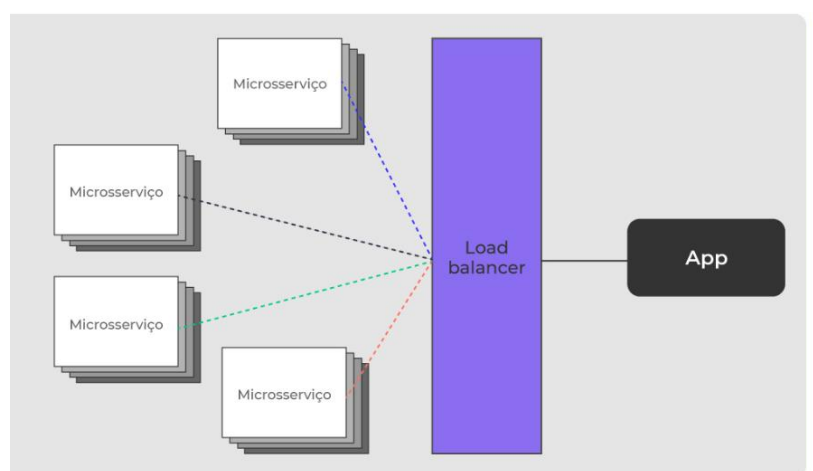


## O que devemos fazer?

Chegamos ao final da semana onde aprendemos sobre Load balancer e API Gateway. Aqui está um resumo do que temos visto.

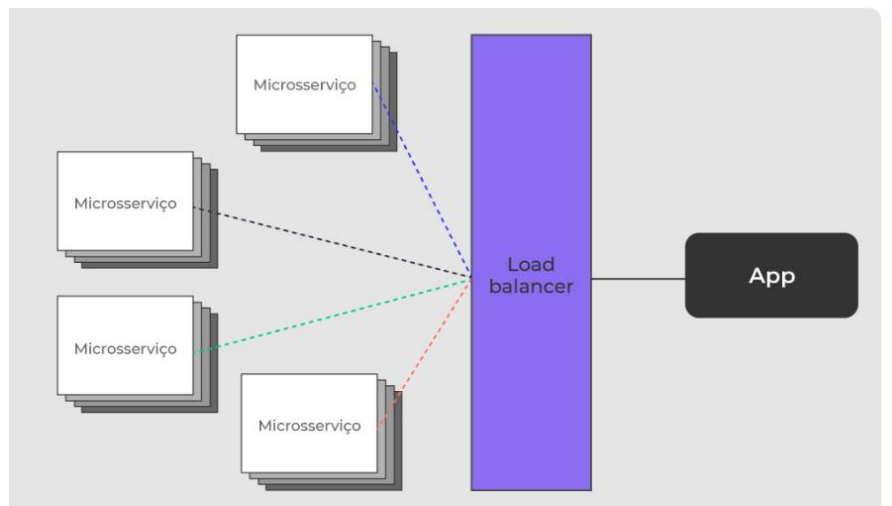
## Load balancer

A melhor alternativa para lidar com o balanceamento de carga é adicionar uma camada dele do lado do servidor. Desta forma, somente o cliente conhece a URL do servidor de balanceamento de carga, e este é feito internamente.



Com esta arquitetura, as vantagens são óbvias. O cliente só tem que executar uma única URL, onde, se adicionarmos uma nova instância ao cluster, só temos que atualizar o Load balancer e para as aplicações seria transparente, pois eles não saberiam da mudança. Além disso, podemos realizar o balanceamento de carga usando o algoritmo que determinamos ser o mais adequado e, desta forma, temos a certeza de que a carga está devidamente distribuída entre todos os servidores.

Finalmente, conseguimos aumentar a segurança, pois só expomos o equilibrador de carga à Internet (rede pública), enquanto o resto dos serviços pode estar em uma rede sem saída para a Internet. Isto reduz a possibilidade de um ataque de fora da rede.



## Exemplo de ações tomadas por uma empresa

Abaixo, mostramos as decisões de infraestrutura que uma start-up vem tomando ao longo do tempo. Isso começa a ser possível graças a uma arquitetura baseada em microserviços:

Dia 1: A empresa contrata uma instância da AWS com um objetivo claro: ir rapidamente para a produção. Implementa seu aplicativo e banco de dados. Na primeira semana eles têm 0 usuários.

Dia 90: A empresa tem 1.000 usuários. Um grande número de novas funcionalidades solicitadas por eles começa a ser implementada. Por enquanto, nenhuma mudança de infraestrutura. O crescimento é de 30% ao mês.

Dia 180: A empresa agora conseguiu crescer para 2.000 usuários. O servidor está começando a ficar sem capacidade. Mais memória/CPU/disco é adquirido (escalonamento vertical).

Dia 360: O crescimento começa a disparar, chegando a 26.000 usuários, e já não é mais possível escalar, ou seja, continuar adicionando memória e processamento no mesmo servidor. A empresa precisa adquirir outro servidor para balancear a carga (escalonamento horizontal).

## API Gateway

Abaixo, revisaremos os recursos e funcionalidades oferecidos pelo Spring Cloud Gateway.



Routing Handler	Spring Cloud Gateway envia as solicitações para o Gateway Handler Mapping, que determina o que efetuar com as solicitações que correspondem a uma rota específica.
Dynamic Routing	Spring Cloud Gateway fornece meios para encaminhar solicitações a diferentes serviços.
Routing Factories	Spring Cloud Gateway combina as rotas usando a estrutura Spring WebFlux HandlerMapping. Inclui muitas built-in Route Predicate Factories. Todos esses Predicates correspondem a diferentes atributos da solicitação HTTP, podendo combiná-los.
WebFilter Factories	Os filtros de roteamento permitem modificar a solicitação HTTP de entrada ou a resposta HTTP de saída.
Spring Cloud DiscoveryClient Support	Spring Cloud Gateway pode ser facilmente integrado com bibliotecas Service discovery e registry, como Eureka server e Consul.
Monitoring	Spring Cloud Gateway usa a API Actuator, uma biblioteca Spring Boot popular que fornece vários serviços prontos para monitorar o aplicativo.