Resumo

TRABALHO FEITO PARA A PRIMEIRA ENTREGA DO CHALLENGE DA DISCIPLINA DE CLOUD COMPUTING E DEVOPS TOOLS, REFERENTE AO CHALLENGE DA FIAP EM PARCERIA COM A PLUSOFT.

Documentação - primeira entrega cloud computing e devops tools

Os benefícios da utilização da virtualização em nosso projeto de imersão gastronômica

**Grupo Poagilers**

Componentes do grupo –

Henrique Neves Lago – RM84549

Larissa Alves – RM86351,

Giovanna de Mello Leiva – RM85817,

Daniel Sanchez Melero – RM85109,

Kaue Augusto - RM85707,

Eric Pessoa – RM86405.

**Resumo do nosso projeto para contextualização**

Atualmente, mesmo diante a pandemia, um dos lugares mais frequentados são áreas do ramo alimentício. Esses locais sofreram um grande avanço tecnológico na última década, mas, mesmo com esse fato, ainda existem problemas que não foram solucionados de maneira efetiva. Por isso, nós do grupo Poagilers ofereceremos por meio de um aplicativo uma nova experiencia gastronômica utilizando IOT.

A ideia é que, por meio da leitura de um QR code do celular, seria possível acessar um cardápio digital personalizado de cada restaurante que se inscreveu no nosso aplicativo, mas não seria apenas uma maneira nova e intuitiva de se ler o cardápio, teria funcionalidades como chamar o garçom, pedir músicas, avaliação pessoal de pratos e acompanhamento de pedido, uma nova experiência para qualquer restaurante, reduzindo o custo da necessidade de produzir e atualizar cardápios físicos, além de modernizar e inovar na maneira de comer fora.

**Benefício da virtualização no nosso projeto**

Para o nosso aplicativo funcionar, precisamos alugar um servidor que, com todas as informações do aplicativo lá dentro, consiga receber as requisições e enviar as respostas dos usuários que estejam interagindo com o app.

Diagrama

Descrição gerada automaticamente

Figura – Arquitetura padrão / Sem virtualização.

E como a virtualização poderia ajudar nisso? Simples, usando a tecnologia do load balancer em conjunto com máquinas virtuais, Teremos a seguinte arquitetura de projeto:

Diagrama

Descrição gerada automaticamente

Figura - Arquitetura com a virtualização implementada

Diagrama

Descrição gerada automaticamente

Figura - Arquitetura com a virtualização implementada

Essa arquitetura implementa o load balancer, que é uma configuração feita para redistribuir várias requisições entre todos os servidores de algum cluster com o intuito de repartir recursos de software ou hardware e com isso melhor a disponibilidade do serviço. Com isso, vamos ter certeza que nossa aplicação não sofrerá lentidão nem ficará fora do ar por tempo algum, oferecendo um aplicativo estável e confiável para nosso usuário final, independente do número de requisições que serão cobradas do servidor, fazendo com que, de forma inteligente, possamos usufruir 100% dos recursos de infraestrutura que planejamos para nosso projeto.

**Como será utilizado o load balancer e como ele será implementado/trabalhado**

Já que trabalharemos com um servidor alugado, pretendemos criar 4 máquinas virtuais que trabalhem como servidores independentes, virtualizando nosso servidor principal e dividindo seus recursos, para que possamos implementar o load balancer.

.Com o load balancer, iremos implementar o método Round Robin, que irá receber as requisições dos usuários e distribuir igualmente entre todos os servidores virtualizados que fazem parte de nosso cluster, de forma com que nenhum fique sobrecarregado, com a vantagem também de que se algum servidor apresentar algum problema e parar de funcionar, o load balancer irá tirá-lo do cluster e distribuir igualmente entre os servidores funcionais restantes.

Não implementamos o método Master-Slave pois a operação de nosso aplicativo não é vital e não iremos perder muito dinheiro por hora no eventual caso do nosso servidor falhar, além do que não tem como encaixarmos no nosso orçamento um servidor que ficará ocioso, esperando apenas o servidor mestre cair para trabalhar. Por isso, em nossa concepção, o Round Robin é a opção mais inteligente para fazer o melhor com o recurso que temos.