

Thread Corporation

Rua Redentor 818, Villa Becker
Toledo, Paraná , 85902-532
(45) 992149181

ATIVIDADE PRÁTICA - ATP - CLOUD COMPUTING

Aluno: Juan Manoel Marinho Nascimento

Visão geral da solução

O presente documento visa entender as demandas técnicas e principais requisitos para implantação de infraestrutura para o cliente. Total de usuarios: 2000, total de usuarios simultâneos: 200



Requisitos para Servidor da Aplicação

1. Cada usuário cadastrado utiliza 1 GB de espaço de armazenamento. Vamos trabalhar com discos internos no servidor configurados em RAID 1.
2. Cada usuário simultâneo utiliza 1 GB de memória no servidor de aplicação.
3. Cada grupo de cem usuários simultâneos consome 20 vCPUs (considere que cada CORE de processador equivale a 4 vCPUs).
4. Cada grupo de cem usuários simultâneos consome 10 GbE de largura de banda de rede.

Requisitos para Servidor de Banco de dados

1. Cada usuário cadastrado utiliza 5 GB de espaço de armazenamento. Vamos trabalhar com discos internos no servidor configurados em RAID 5.
2. Cada usuário simultâneo utiliza 2 GB de memória no servidor de banco de dados.
3. Cada grupo de cem usuários simultâneos consome 40 vCPUs (considere que cada CORE de processador equivale a 4 vCPUs).

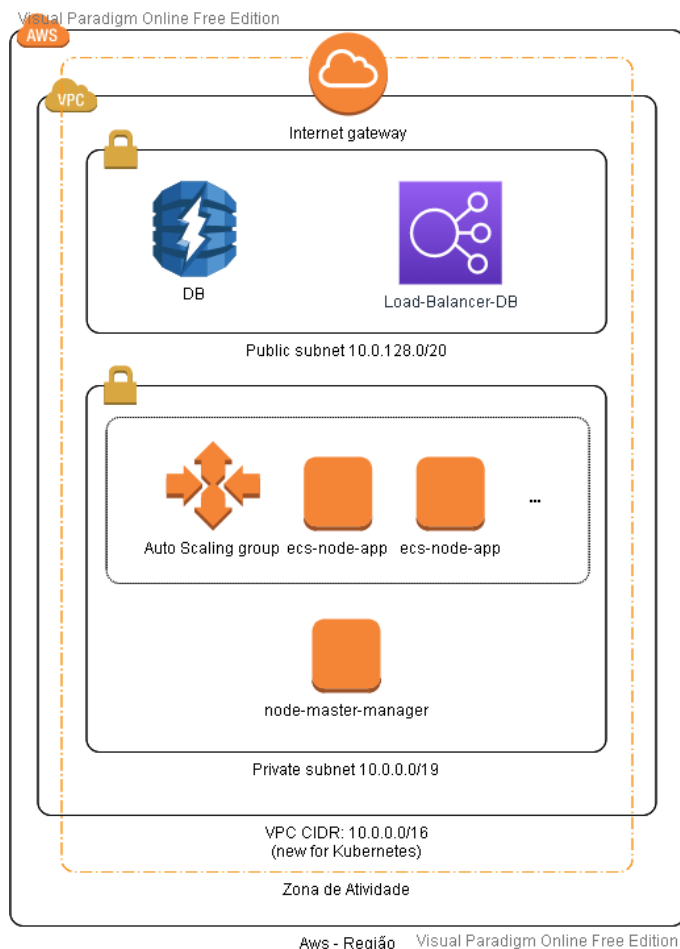
Especificações de Hardware

Usuários Unitário	db_user	RAM gb	vCPUs	Storage gb	
1	1	2	2,5	5	
Total user contínuos					
200	200	400	500	1000	
Usuários Unitário	app_user	ram	vCPUs	MB Rede	Storage
1	1	1	5	0,01	1
Total user contínuos					
200	200	200	1000	2	200

ECS INSTANCE	RAM	SD-STORAGE	vCPUS	AUTO-SCALE	DB INSTANCE
1	4		4	TRUE	LIGHTSAIL POSTGRESQL
total					
200	800	2TB	800	TRUE	4

Detalhes Técnicos - AWS - CLOUD

Foi projetado o uso da plataforma aws ecs para subir imagens docker para dentro da plataforma, permitindo autoscale e sendo gerenciado por um cluster interno, a plataforma atua como serviço, permitindo ser gerenciado seus custos pelo uso, modelo pay as you go.








Para essa aplicação foi definido usarmos arquitetura em cloud, pensando em garantir estabilidade e escalabilidade do sistema, o fluxo ao lado mostra como ambos os sistemas devem ser isolados e como devem se conversar, com um vpc, é possível gerenciar nossa rede privada e permitir acesso externo com controle mais efetivo. Pensando na aplicação, foi definido usar um balanceador de carga para gerenciar o uso de recurso em autoescalar as aplicações, de n para n, pensando no maior cenário é em momentos de pico essa parte fica abstraída da empresa visto que a aws é responsável por gerenciar infra.

Para gerenciamento disso temos um kubernetes que vai ser nosso manager e interface de controle, tudo isso é abstraído para camada de rede e gerenciamento de containers.

O banco definido para esse projeto foi o DynamoDB, visto que sozinho já tem uma interface de auto-scale e alta performance, ele consegue gerenciar de forma elástica de acordo com nossa aplicação.

Detalhes Técnicos - AWS - PRINT

 Amazon Lightsail  Início Documentação  Conta ▾  AWS Faturame




datalake-caio


1 GB RAM, 1 vCPU, 40 GB SSD
Banco de dados MySQL (8.0.23)
Virgínia, Zona A (us-east-1a)

Encerrar Reinicializar

Status: **Disponível**





Endpoint: **ls-d3897fabf406022697bd50f2f95b05b3ee52c9f3.cx8982sx1pw5.us-east-1.rds.amazonaws.com**
Porta: **3306**





Conectar-se Logs e histórico Métricas  Redes Snapshots e restauração Tags Excluir

Detalhes da conexão
Nome de usuário e senha 

dbmasteruser

[Alterar senha](#)

 Snipping Tool
 New  Mode ▾
Select the snip mode using the button.
 Snipping Tool is n
In a future update, Snipp
new home. Try improv

 Amazon Lightsail  Início Documentação  Conta ▾  AWS Faturament

Boa tarde!

Instâncias [Contêineres](#) [Bancos de dados](#) [Redes](#) [Armazenamento](#) [Snapshots](#)

Classificar por **Data ▾**

[Criar instância](#)

 <h4>gitlab-runner</h4> <p>512 MB RAM, 1 vCPU, 20 GB SSD</p> <p>Em execução</p> <p>100.27.32.118 2600:1f18:1084:f400:2b73:a25:e6ee:8615 Virgínia, Zona A</p>	 <h4>bi-metabase</h4> <p>1 GB RAM, 1 vCPU, 40 GB SSD</p> <p>Em execução</p> <p>52.20.130.186 2600:1f18:1084:f400:9797:e59f:301:a838 Virgínia, Zona A</p>
---	---

Tipos de Modelos de Disponibilidade: IaaS , SaaS e PaaS

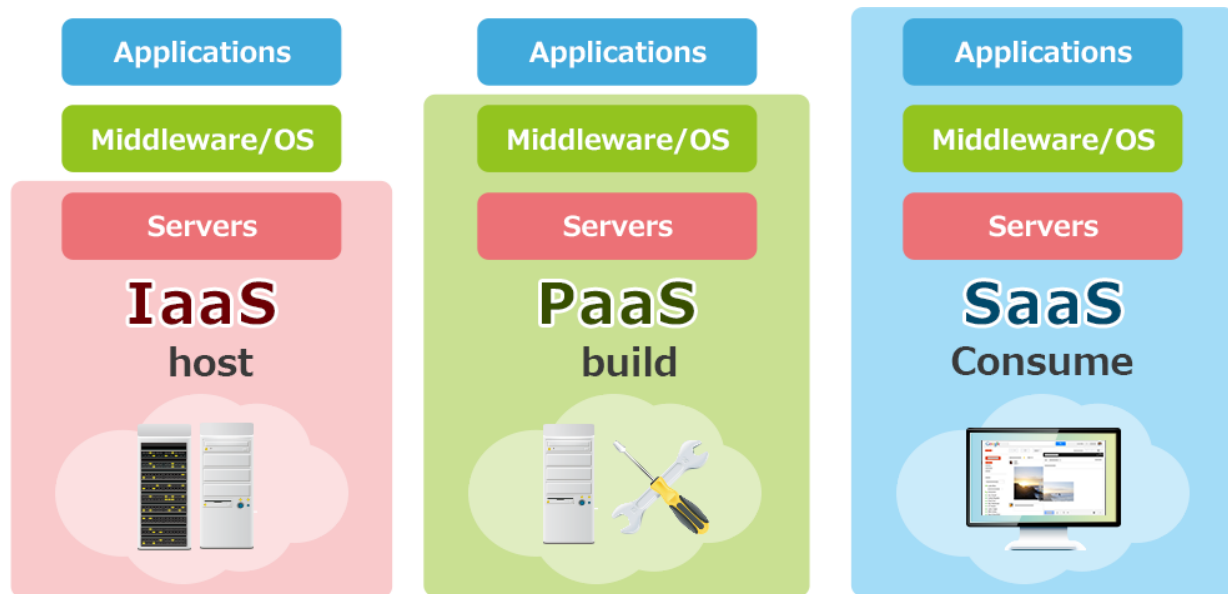


Figura: 1, Tipos de Disponibilidade

IaaS(Infraestrutura como serviço)

Infraestrutura como Serviço (IaaS) é um dos três modelos de serviço fundamentais da computação em nuvem (cloud computing). No caso específico do IaaS, o recurso de computação é fornecido especificamente com relação ao hardware virtualizado, em outras palavras, a infraestrutura de computação.

A definição inclui ofertas como espaço virtual de servidores, conexões de rede, largura de banda, endereços IP e balanceadores de carga. Fisicamente, o conjunto de recursos de hardware é proveniente de diversos servidores e redes normalmente distribuídos em vários centros de dados, sendo o provedor de nuvem responsável pela manutenção destes.

Ao cliente, por outro lado, é dado acesso aos componentes virtualizados a fim de construir as suas próprias plataformas de TI.

PaaS(Plataforma como serviço)

Esse modelo de negócio oferece às empresas um ambiente de desenvolvimento de aplicações, bem como um ambiente de execução para entregá-las. Com isso, os desenvolvedores/engenheiros ganham um framework/apis para a criação ou modificação de apps. Trazendo isso de maneira simples, é como se o desenvolvedor pudesse criar Macros no Excel, Criar Sites no Word, criar dashboards e painéis no powerpoint. Ele usa seus conhecimentos, mas tira proveito de tecnologias que já estão à disposição. Por conta disso, o PaaS poupa dinheiro para empresas que precisam criar, testar e implementar aplicações inteiras e complexas de maneira rápida e simples.

SaaS(Software como serviço)

A sigla significa Software as a Service, ou Software como Serviço. Esse modelo de negócio de TI é o mais popular e amadurecido e estável. Em parte, por ser também o formato que mais tem contato com o consumidor final como Netflix e outros softwares que são serviços hoje em dia. Outro exemplo que ilustra bem isso é o Google Docs onde estou fazendo este documento agora.

Interface com Industria 4.0 (Estudo sobre Balena Cloud)

Atualmente existem muitas formas de usar essa sopa de (IaaS, SaaS, PaaS), dentro do contexto de IoT. Nesse ponto, o Balena Cloud, pode nos dar um excelente ponto de como usar essa sopa de letrinhas para agregar valor ao negócio, pensando em escalabilidade e gerenciamento.

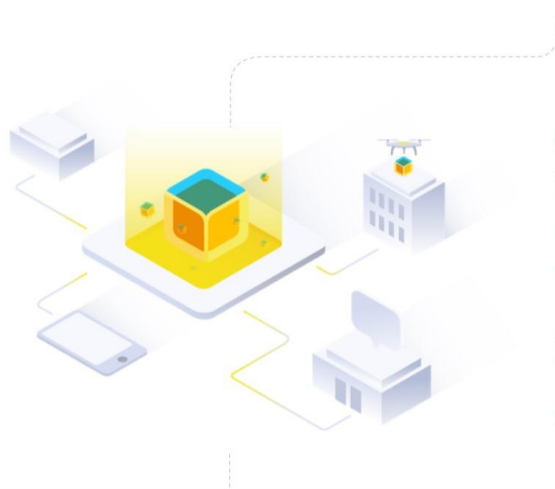


Figura 2: Fluxo de Operação Balena Cloud

O Balena nos permite gerenciar todas as etapas de entrega de um processo complexo, mas de forma simples e objetiva, passando pela etapa de desenvolvimento de software embarcado com inteligência artificial, coleta de dados em sensores e gerenciamento de automação inteligente. Quando um desenvolvedor faz um “push” de sua aplicação, os servidores do balena gerenciam esse processo de forma automática por projeto, de forma que cada desenvolvedor responsável possa manter e gerenciar sua cadeia de iots, a aplicação entra em uma fluxo que build/constrói uma imagem de , e internamente o balena com sua api gerencia pra qual iot a versão do software deve ser atualizada.

Ao olhar o contexto temos software as service, na primeira camada, servindo de interface com desenvolvedores para entrega contínua e melhoria contínua. Na Segunda camada, temos infra como serviço, visto que durante o processo de configuração com iot, os desenvolvedores definem o tipo de hardware utilizado para estrutura do software a ser desenvolvido e integrado na estrutura on cloud e finalmente o último é mais importante ponto de destaque é de plataforma como um serviço, o balena fornece uma interface web que permite gerenciar nossos

IoT's dispostos por projetos e realocar eles de forma performática e ágil para mitigar desperdícios.



IMPLANTAÇÕES DE CONTÊINERES

Tecnologia de contêiner Docker para atualizações seguras, protegidas e confiáveis.

Envie atualizações com um único comando enquanto seus dispositivos permanecem online e totalmente funcionais. Arquitetura de microsserviços? Oferecemos suporte a aplicativos de vários contêineres.

✓ Atualizações fáceis

✓ Implementações seguras de tijolos

✓ Cargas úteis leves

Conclusão

Com base nas necessidades do usuário e requisitos do sistema atual, foi levantado que para resolver e fazer esse processo de migração, levando em conta os dados de uso de recurso dos usuários tanto pela aplicação como pelo consumo de banco de dados. Conclui-se que usar a aws como plataforma de infra foi a melhor solução e mais ágil para esse tipo de migração e integração.