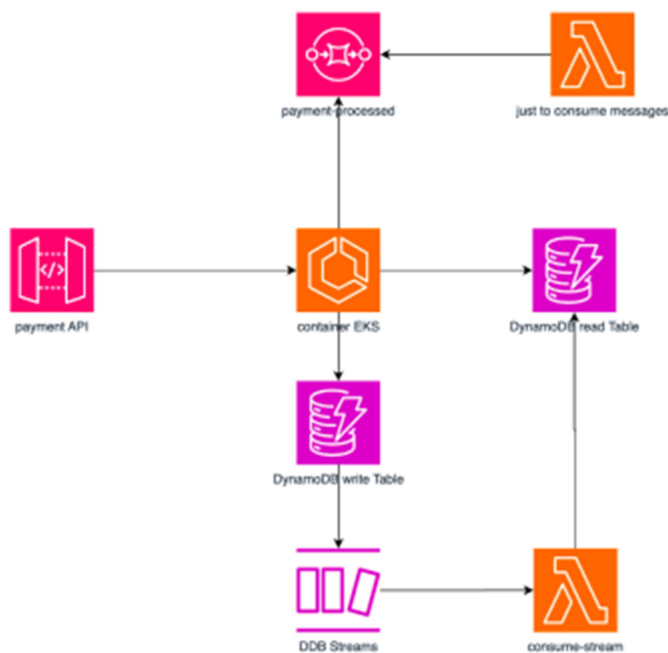


## Introdução

Será provido um desenho de arquitetura, uma descrição do resultado esperado e um conjunto de requisitos funcionais e não funcionais a serem considerados. Ao final, é esperado uma demo feita pelo candidato demonstrando o resultado obtido e todo o código fonte gerado deve fazer parte do entregável.

## Descrição

O candidato deve codificar o máximo que conseguir no tempo provido a seguinte arquitetura:



## Requisitos funcionais

A aplicação é uma API de pagamentos onde se pode fazer um pagamento e consultar pagamentos. O fluxo da aplicação é esse:

1. É feita uma chamada POST /transactions para o api gateway:

### POST /transactions

```
{
  accountID: uuid
  amount: float32
  type: credit
}
```

2. O Api gateway envia a request para um serviço rodando em container dentro de um serviço EKS.
3. O micro-serviço salva a transação em um banco Dynamodb(DDB) adicionando o campo status: in\_processing, e um campo transactionID: uuid, então envia uma mensagem para uma fila do Amazon SQS onde uma lambda irá apenas consumer o evento.
4. O DDB dispara via streams um evento que será consumido por um lambda
5. A Lambda consome o evento e salva na segunda tabela para leitura com o status: **processed**
6. Agora o api gateway pode via **GET /transactions/{transactionID}** a transação

### **Requisitos não funcionais**

1. A API deve ser capaz de processar 100 TPS
2. Deve ser necessário demonstrar o TPS alcançado e a latência em P99
3. Infraestrutura deve usar terraform
4. Todo desenvolvimento deve ser feito em Java

### **Entrega final de Demo**

Ao final, o candidato fará uma apresentação do progresso feito e deve entregar um único arquivo Zip com todo o código desenvolvido.

Para esse cenário, o prazo será de 2 dias.

O candidato deve informar quanto tempo (em horas) foi investido na codificação da solução e no dia da demo informar esse valor