

Documentação do Projeto Hackathon FIAP X

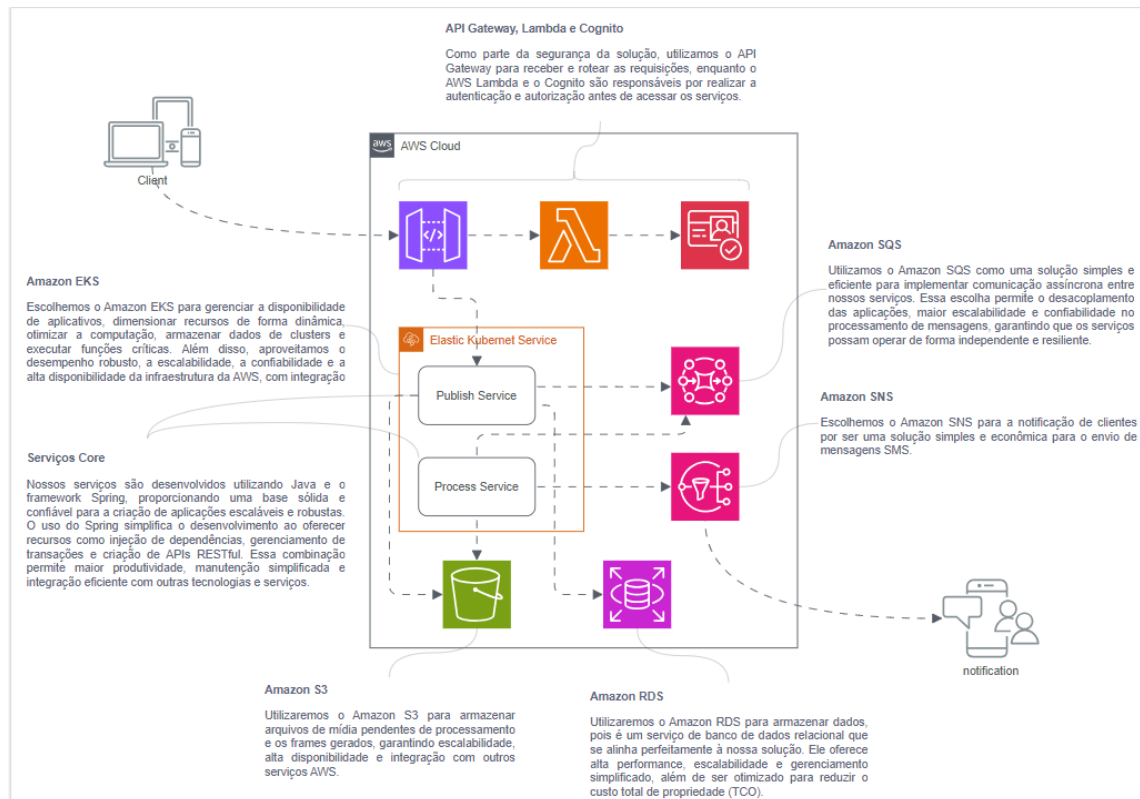
Introdução

A empresa FIAP X iniciou o desenvolvimento de um projeto para processamento de vídeos, que converte um vídeo enviado em imagens e retorna as imagens compactadas em um arquivo .zip. Este projeto atraiu o interesse de investidores, que agora desejam uma versão aprimorada do sistema, permitindo que os usuários enviem vídeos e façam o download dos arquivos processados.

Link do Event Storming - Processamento De Arquivos:

https://miro.com/app/board/uXjVKXjixqA=/?share_link_id=332305603387

Arquitetura do Sistema



O sistema foi projetado para garantir escalabilidade, segurança e desempenho, utilizando os seguintes componentes da AWS:

- **API Gateway, Lambda e Cognito:** O API Gateway recebe e roteia as requisições, enquanto o AWS Lambda executa funções serverless. O Cognito é responsável pela autenticação e autorização dos usuários.

- **Amazon EKS:** Utilizado para orquestração de contêineres, garantindo escalabilidade e alta disponibilidade.
- **Amazon S3:** Armazena os arquivos de mídia gerados pelo processamento, garantindo durabilidade e disponibilidade.
- **Amazon RDS:** Banco de dados relacional para armazenamento de metadados, garantindo consistência e escalabilidade.
- **Amazon SQS:** Implementa comunicação assíncrona entre serviços, permitindo desacoplamento e resiliência.
- **Amazon SNS:** Utilizado para notificação de usuários via e-mail ou SMS.

Essas escolhas foram feitas para garantir que o sistema seja altamente escalável, seguro e resiliente, atendendo aos requisitos do projeto.

Estrutura dos Repositórios

Os repositórios do projeto seguem uma nomenclatura padronizada:

- Prefixo: hacka (indica que faz parte do projeto Hackathon FIAP X).
- Tipo:
 - msc: Microserviço.
 - infra: Infraestrutura.
 - lambda: Funções serverless.
 - rds: Banco de dados relacional.
- Descrição: Descreve o propósito do repositório.

Repositórios

- **Microserviços**
 - hacka-msc-mgmt-media: Gerenciamento de mídia.
 - hacka-msc-pcs-midia: Processamento de mídia.
- **Infraestrutura**
 - hacka-infra-eks: Gerenciamento de clusters Kubernetes (EKS).
 - hacka-infra-rds: Banco de dados relacional (RDS).
 - hacka-infra-api-gateway: Configuração de API Gateway.

- hacka-infra-cognito: Autenticação e autorização com Amazon Cognito.
 - **Funções Serverless**
 - hacka-lambda-authorized: Função Lambda para autorização.
-

Padrões de Qualidade e Governança

Integração Contínua e Entrega Contínua (CI/CD)

- Todos os projetos possuem pipelines automatizados de CI/CD configurados.
- As pipelines incluem:
 - Build e deploy automáticos.
 - Execução de testes automatizados.

Qualidade de Código

- Todas as APIs são integradas com o SonarQube.
- Meta de qualidade: 80% de cobertura de código.

Proteção de Branch

- A branch principal (main) é protegida e requer Pull Requests (PRs) para qualquer modificação.
-

Escalabilidade dos Microserviços e Uso do HPA

O sistema foi projetado para ser altamente escalável, utilizando Kubernetes para gerenciar os microserviços. Para garantir que o sistema possa lidar com variações na carga de trabalho, utilizamos o **Horizontal Pod Autoscaler (HPA)**. O HPA está configurado para escalar automaticamente os pods quando o uso da CPU atinge **80%**, garantindo que novos pods sejam criados para suportar a demanda sem comprometer o desempenho.

Justificativa do API Gateway e Fila de Processamento

Devido à escalabilidade dinâmica do sistema, o uso de um **API Gateway** e de um sistema de **filas de mensagens (SQS)** se torna essencial. O API Gateway:

- Garante um ponto único de entrada para as requisições.

- Facilita a autenticação e autorização.
- Controla o tráfego e distribui as requisições entre os serviços disponíveis.

A fila de mensagens (SQS) permite que as requisições sejam armazenadas e processadas de forma assíncrona, garantindo que nenhuma solicitação seja perdida em momentos de pico e permitindo que os serviços de processamento consumam mensagens conforme sua capacidade disponível.

Essa abordagem melhora a resiliência e a escalabilidade da aplicação, garantindo um sistema eficiente e confiável.

Escolha da Linguagem Java

Optamos pelo uso da linguagem Java devido ao amplo conhecimento da equipe de desenvolvimento, o que reduz a curva de aprendizado e melhora a produtividade. Além disso, Java oferece diversas vantagens, como:

- Robustez e estabilidade;
- Ampla comunidade de suporte;
- Facilidade na implementação de microsserviços com Spring Boot;
- Forte suporte a integração com serviços em nuvem.

Com a arquitetura modular adotada, no futuro será possível desenvolver novos microsserviços em qualquer linguagem, permitindo maior flexibilidade na evolução do sistema.

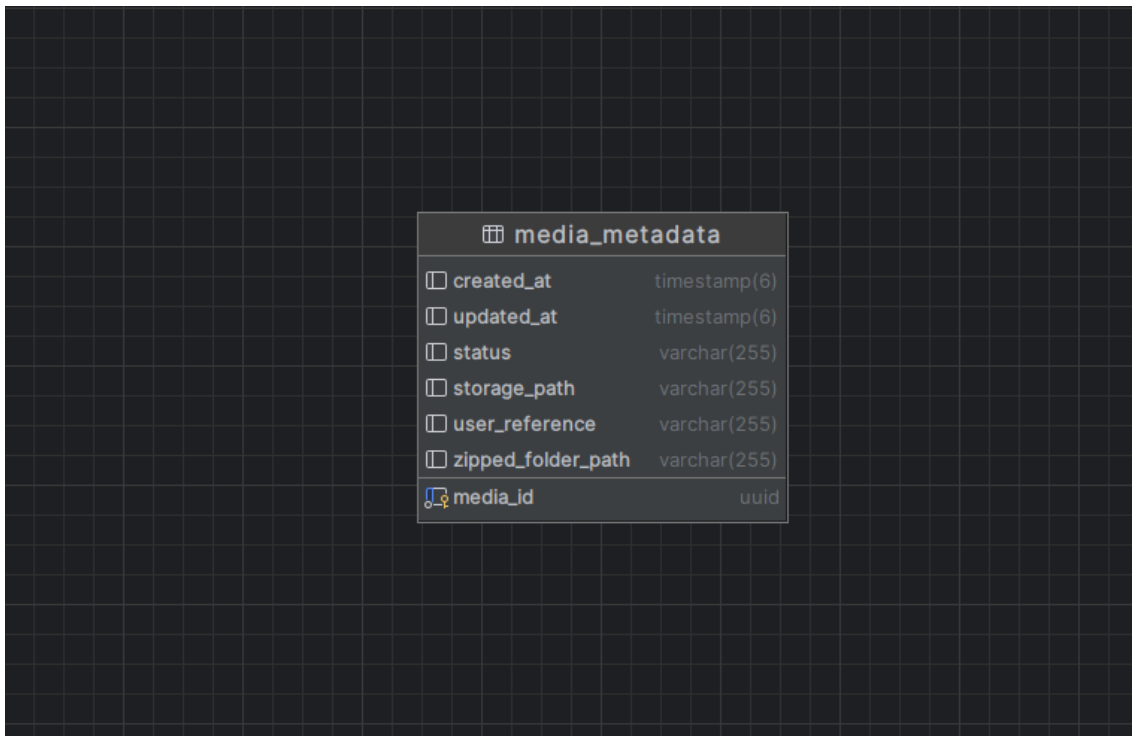
Link de Download no S3

Por questões de segurança, os links de download gerados para os arquivos armazenados no Amazon S3 possuem uma validade de **5 minutos**. Esse tempo limitado reduz riscos de acesso indevido e garante que apenas usuários autenticados possam baixar os arquivos dentro do período determinado.

A geração desses links assinados é feita dinamicamente, sempre que um usuário solicita um download, garantindo maior controle sobre o acesso aos arquivos.

Estrutura de Dados Armazenados

Os metadados das mídias processadas são armazenados no banco de dados utilizando a seguinte estrutura:



media_metadata	
created_at	timestamp(6)
updated_at	timestamp(6)
status	varchar(255)
storage_path	varchar(255)
user_reference	varchar(255)
zipped_folder_path	varchar(255)
media_id	uuid

Explicação dos Campos:

- **mediaId:** Identificador único da mídia.
- **storagePath:** Caminho onde o arquivo original está armazenado no S3.
- **status:** Status do processamento da mídia.
- **userReference:** Referência ao usuário que enviou o vídeo.
- **zippedFolderPath:** Caminho para a pasta compactada contendo as imagens extraídas.

Esses dados permitem que o sistema rastreie e gerencie o processamento das mídias, garantindo integridade e rastreabilidade.

Estimativa de Custo

A estimativa de custo foi realizada utilizando a calculadora da AWS. Os principais componentes incluídos na estimativa são:

- **Amazon EKS** para orquestração de contêineres.
- **Amazon RDS** para armazenamento de metadados.
- **Amazon S3** para armazenamento de arquivos de mídia.
- **Amazon SQS e SNS** para comunicação e notificações.

O detalhamento completo dos custos pode ser consultado na captura de tela da calculadora AWS.

Resumo da estimativa

Custo inicial
0,00 USD

Custo mensal
509,08 USD

Custo total de 12 meses
6.108,96 USD
Inclui um custo inicial

Conceitos básicos da AWS

Comença a usar gratuitamente

Entre em contato com a equipe de vendas

My Estimate

Localizar recursos

Duplicar

Excluir

Mover para

Criar grupo

Adicionar suporte

Adicionar serviço

<input type="checkbox"/>	Nome do serviço	Status	Custo inicial	Custo mensal	Descrição	Região	Resumo da configuração
<input type="checkbox"/>	Amazon API Gateway	-	0,00 USD	0,00 USD	-	Leste dos EUA (Ohio)	Unidades de solicitações da API HTTP (milhões)
<input type="checkbox"/>	Amazon Cognito	-	0,00 USD	0,00 USD	-	Leste dos EUA (Ohio)	Taxa de otimização para solicitações de token
<input type="checkbox"/>	AWS Lambda	-	0,00 USD	0,00 USD	-	Leste dos EUA (Ohio)	Arquitetura (x86), Arquitetura (ARM), Modo de L...
<input type="checkbox"/>	Amazon RDS for PostgreSQL	-	0,00 USD	133,08 USD	-	Leste dos EUA (Ohio)	Quantidade de armazenamento (20 GB, Volu...
<input type="checkbox"/>	Amazon Simple Storage Service (S3)	-	0,00 USD	230,00 USD	-	Leste dos EUA (Ohio)	Armazenamento S3 Standard (10000 GB por ...
<input type="checkbox"/>	Amazon EC2	-	0,00 USD	146,00 USD	-	Leste dos EUA (Ohio)	Número de clusters do EC2 (2), Número de nód...

Estimativa detalhada

Nome	Grupo	Região	Custo inicial	Custo mensal
Amazon API Gateway	Nenhum grupo aplicado	US East (Ohio)	0,00 USD	0,00 USD
Status: -				
Descrição:				
Resumo da configuração: Unidades de solicitações da API HTTP (milhões), Tamanho médio de cada solicitação (34 KB), Unidades de solicitação da API REST (milhões), Tamanho da memória do cache (GB) (Nenhum), Unidades de mensagens WebSocket (milhares), Tamanho médio da mensagem (32 KB)				
Amazon Cognito	Nenhum grupo aplicado	US East (Ohio)	0,00 USD	0,00 USD
Status: -				
Descrição:				
Resumo da configuração: Taxa de otimização para solicitações de token (0), Taxa de otimização para clientes de aplicativo (0), Recursos avançados de segurança (Habilitada)				
AWS Lambda	Nenhum grupo aplicado	US East (Ohio)	0,00 USD	0,00 USD
Status: -				
Descrição:				
Resumo da configuração: Arquitetura (x86), Arquitetura (x86), Modo de invocação (Em buffer), Quantidade de armazenamento temporário alocada (512 MB), Número de solicitações (10000 por mês)				

Amazon RDS for PostgreSQL	Nenhum grupo aplicado	US East (Ohio)	0,00 USD	133,08 USD
---------------------------	-----------------------	----------------	----------	------------

Status: -

Descrição:

Resumo da configuração: Quantidade de armazenamento (20 GB), Volume de armazenamento (SSD de uso geral (gp2)), Nós (1), Tipo de instância (db.t2.medium), Utilização (somente sob demanda) (100 %Utilized/Month), Opção de implantação (Multi-AZ), Modelo de preço (OnDemand)

Amazon Simple Storage Service (S3)	Nenhum grupo aplicado	US East (Ohio)	0,00 USD	230,00 USD
------------------------------------	-----------------------	----------------	----------	------------

Status: -

Descrição:

Resumo da configuração: Armazenamento S3 Standard (10000 GB por mês), GET, SELECT e todas as outras solicitações do S3 Standard (10000)

Amazon EKS	Nenhum grupo aplicado	US East (Ohio)	0,00 USD	146,00 USD
------------	-----------------------	----------------	----------	------------

Status: -

Descrição:

Resumo da configuração: Número de clusters do EKS (2), Número de nós híbridos (1 por mês)
