**1. Introdução**

A FIAP X, empresa inovadora no setor de tecnologia, busca aprimorar sua plataforma de processamento de imagens, expandindo suas funcionalidades para atender à crescente demanda do mercado. O projeto inicial, que processava vídeos e gerava arquivos .zip com as imagens extraídas, demonstrou grande potencial e despertou o interesse de investidores.

Esta especificação detalha o escopo do projeto de desenvolvimento da nova plataforma, que permitirá aos usuários enviar vídeos e receber arquivos .zip contendo as imagens processadas, oferecendo uma experiência mais completa.

**2. Regra de negócio**

A plataforma FIAP X, através da plataforma “Frame Extractor” processará vídeos enviados pelos usuários, extraindo frames em intervalos definidos e gerando arquivos .zip contendo as imagens resultantes. O sistema aceitará apenas vídeos nos formatos .mp4, .mov e .avi, com tamanho máximo de 1GB e com um intervalo maior que 0. Os arquivos .zip gerados serão armazenados na nuvem e um link de download será disponibilizado ao usuário. A plataforma garantirá a segurança dos dados dos usuários e dos vídeos processados, através de mecanismos de criptografia e controle de acesso.

É imprescindível nesse projeto, que os frames gerados sejam armazenados de maneira segura e com alta disponibilidade para os usuários, sendo um requisito essencial para a FIAP X. No item 5 desta documentação, temos as especificações de tecnologias para atender essa demanda.

Em concordância com a FIAP X, a primeira versão desse projeto tende a ser numa menor escala, para verificar a adesão dos usuários, mas ainda sendo capaz de suprimir altas requisições com escalonamento flexível (tópico 5.6) e algumas melhorias futuras a serem feitas (tópico 9)

**3. Domain-Driven Design (DDD)**

Abordagem para o desenvolvimento de software que coloca o foco principal no domínio do problema, ou seja, na área de negócio que o software está destinado a resolver.

**3.1 Linguagem Ubíqua (Dicionário)**

**FIAP X:** Nome da empresa inovadora no setor de tecnologia

**Frame Extractor:** Nome da plataforma de extração dos frames

**Usuário administrador:** Uma conta exclusiva com privilégios de acesso no sistema

**Usuário cliente:** Usuários que irão se cadastrar na plataforma

**Perfil:** perfil associado ao usuário, que define acessos no sistema

**Login:** Texto para identificação e autenticação do usuário, sendo único do sistema

**E-mail:** E-mail válido para recuperação de senha e notificações do sistema, sendo único no sistema

**Frames:** imagens extraídas do vídeo e salvar em um arquivo .zip

**3.1.1 Dicionário para desenvolvedores**

Pela experiência da equipe de desenvolvimento em desenvolver o código todo em inglês, considerar o seguinte de/para dos termos:

**Usuário:** User (modelo que representa os usuários do sistema)

**Perfil:** role (determina se o usuário é administrador ou cliente)

**E-mail:** email

**Login:** username

**3.2 Domínio**

Sendo a extração dos frames o coração da aplicação, entende-se uma maior demanda da equipe, para garantir as conformidades do que foi pedido.

Segue o Domain Storytelling do Domínio da aplicação:

Diagrama

O conteúdo gerado por IA pode estar incorreto.

**3.3 Subdomínio de suporte: Autenticação e gestão de usuários**

O gerenciamento de usuários e autenticação será realizada pela própria aplicação Frame Extractor. Para facilitar a visualização e criar uma diferenciação entre o usuário administrador e o usuário cliente, foi criado dois Domain Storytelling da autenticação e gestão de usuários:

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Diagrama

O conteúdo gerado por IA pode estar incorreto.

**3.4 Event Storming**

Link para acessar: <https://miro.com/app/board/uXjVLh61E0Q=/?share_link_id=540764023452>

**4. Requisitos**

**4.1. Requisitos Funcionais**

**Envio de vídeos**: A plataforma deve suportar o envio de vídeos de diferentes formatos e tamanhos.

**Processamento de vídeos**: A plataforma deve processar os vídeos, extraindo os frames em intervalos definidos pelo usuário ou automaticamente.

**Geração de arquivos .zip**: A plataforma deve gerar arquivos .zip contendo as imagens extraídas, fornecendo para o usuário um link de download e salvando o arquivo na nuvem.

**Gerenciamento de usuários**: A plataforma deve incluir funcionalidades de cadastro, login e gerenciamento de usuários, com diferentes níveis de acesso.

**4.2. Requisitos Não Funcionais**

**Desempenho**: A plataforma deve processar os vídeos e gerar os arquivos .zip de forma rápida e eficiente, sendo capaz de processar mais de um vídeo ao mesmo tempo

**Escalabilidade**: O sistema deve estar em uma arquitetura que o permita ser escalado horizontalmente

**Segurança**: A plataforma deve garantir a segurança dos dados dos usuários e dos vídeos enviados, salvando as informações sensíveis com criptografia.

**Disponibilidade**: A plataforma deve ter disponibilidade de 99%

**Manutenibilidade**: Código modular e bem documentado para facilitar futuras atualizações; Cobertura de testes; CI/CD;

**5. Especificações técnicas para execução do projeto**

A seguir segue o detalhamento das tecnologias escolhidas para solução do sistema Frame Extractor

**5.1 Linguagem de Programação e Framework**

Uso da linguagem Python na versão 3.12.8, por ser versátil e amplamente utilizada no desenvolvimento web, com uma vasta comunidade e diversas bibliotecas. Principal framework utilizado: FastAPI. Framework moderno e de alta performance para desenvolvimento de APIs RESTful em Python. Baseado em padrões ASGI (Asynchronous Server Gateway Interface), proporcionando alta concorrência e escalabilidade.

**5.2 Banco de Dados**

Tecnologia escolhida: Amazon DynamoDB.

Banco de dados NoSQL totalmente gerenciado, escalável e altamente disponível.

**5.3 Armazenamento de Arquivos**

Tecnologia escolhida: Amazon S3

Serviço de armazenamento de objetos altamente escalável, durável e seguro.

Atende a demanda de armazenar arquivos .zip, pois oferece:

Durabilidade: Os dados são replicados em múltiplas zonas de disponibilidade, garantindo alta durabilidade.

Escalabilidade: Pode armazenar qualquer quantidade de dados.

Acesso rápido: Permite o acesso a objetos de qualquer lugar do mundo com baixa latência.

Gerenciamento de versões: Permite armazenar múltiplas versões de um mesmo objeto, facilitando a recuperação de dados.

**5.4 Envio de E-mail**

Tecnologia escolhida: Amazon Simple Email Service (SES)

É um serviço de e-mail da AWS que oferece as seguintes vantagens:

Infraestrutura robusta: O SES utiliza a infraestrutura global da AWS, garantindo alta disponibilidade e escalabilidade para o envio de e-mails, mesmo em grandes volumes.

Reputação de envio: A Amazon trabalha para manter uma boa reputação de envio, o que ajuda a garantir que seus e-mails cheguem à caixa de entrada dos destinatários e não sejam marcados como spam.

Integração fácil: O SES se integra facilmente com outras ferramentas e serviços da AWS, como o Amazon SNS para notificações e o Amazon S3 para armazenamento de anexos.

Personalização: Você pode personalizar seus e-mails com HTML e usar tags para adicionar informações dinâmicas, como o nome do destinatário.

**5.4 Cobertura de Testes**

Tecnologia escolhida: pytest

É um framework poderoso e amplamente utilizado para testes em Python. Ele serve para simplificar a criação e execução de testes, além de oferecer diversos recursos que facilitam o processo de desenvolvimento e garantem a qualidade do código

**5.5 Análise estática do código**

Tecnologia escolhida: SonarQube

Atua como um sistema de análise estática, vasculhando o código-fonte em busca de **bugs, vulnerabilidades e "code smells"**, que são trechos de código que, embora funcionem, podem indicar problemas futuros ou práticas inadequadas.

**5.6 Escalonamento**

Tecnologia escolhida: Kubernetes

A escalabilidade é um dos pilares do Kubernetes, permitindo que suas aplicações se adaptem dinamicamente às demandas do mercado. Seja para lidar com um pico de acessos inesperado ou para otimizar o uso de recursos em períodos de menor movimento, o Kubernetes oferece ferramentas poderosas para escalar seus aplicativos de forma eficiente e automatizada. Padrão utilizado no projeto Escalonamento Horizontal: consiste em aumentar ou diminuir o número de réplicas de um pod (grupo de contêineres) em execução.

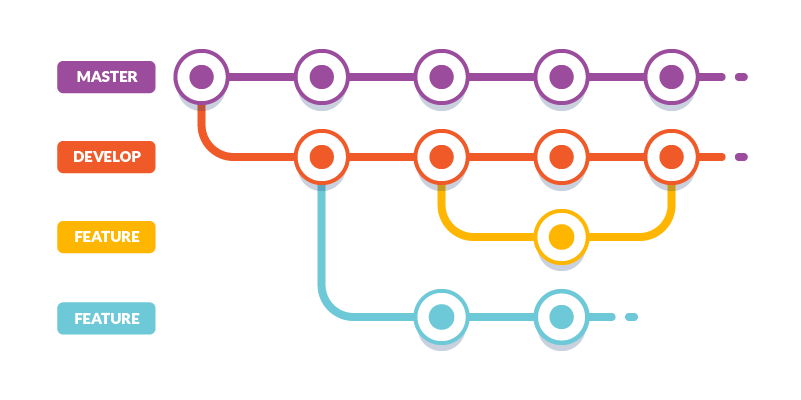
**6. Arquitetura escolhida**

**Arquitetura monolítica:** A escolha da arquitetura monolítica para aplicações de escopo limitado é uma decisão estratégica que visa otimizar o desenvolvimento, implantação e manutenção do sistema. Ao optar por essa abordagem, a equipe de desenvolvimento prioriza a simplicidade, eficiência e adequação ao escopo do projeto, garantindo um resultado satisfatório com menor investimento e complexidade.

**7. Padrões de projeto**

**Arquitetura hexagonal:** A Arquitetura Hexagonal, também conhecida como Arquitetura de Portas e Adaptadores, é um padrão de design que visa criar sistemas de software mais flexíveis, testáveis e fáceis de manter. Ela se baseia na separação de responsabilidades, isolando o núcleo da aplicação (a lógica de negócios) de suas dependências externas (bancos de dados, interfaces de usuário, etc.).

**Versionamento:** Uso da ferramenta Git, seguindo o modelo git flow.



**8. Desenho da aplicação**

Interface gráfica do usuário, Diagrama, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

**9. Melhorias futuras**

Foi alinhado com a empresa Fiap X, de ampliar a capacidade de processamento dos vídeos, permitindo o envio de arquivos com até 4GB, mediante uma assinatura premium.

Traduções para o sistema poder funcionar em diversos países além do Brasil.

**10. Testes executados**

Última cobertura de testes antes de entregar o projeto e status do SonarQube:  
Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.