

FECAP

Fundação Escola de Comércio Álvares Penteado

Projeto Educação de qualidade a todos

Requisitos da disciplina Modelagem de Software e Arquitetura de Sistemas

São Paulo

2024

Integrantes do Projeto e RA'S:

Adriano Xu Ming Hui RA:24025742

Rafael Budoya 23025462

# 1.Introdução

No contexto atual, a educação desempenha um papel fundamental na formação de indivíduos e na construção de sociedades mais justas e igualitárias. Contudo, o acesso a uma educação de qualidade ainda é um desafio enfrentado por muitos ao redor do mundo. Com o objetivo de democratizar o conhecimento e proporcionar oportunidades de aprendizado a todos, este projeto visa criar uma plataforma online que facilite o acesso a recursos educacionais de qualidade.

O site que estamos desenvolvendo será um espaço dedicado à educação inclusiva, permitindo que pessoas de diferentes origens e níveis de conhecimento explorem uma variedade de temas. Através de parcerias com fontes confiáveis e instituições educacionais, compartilharemos informações, cursos e materiais de aprendizado, garantindo que todos tenham a chance de participar ativamente de sua formação.

Além de oferecer conteúdos selecionados, o site promoverá um ambiente colaborativo onde os usuários poderão interagir, trocar experiências e compartilhar conhecimentos. Acreditamos que a educação não deve ter limites e que, ao conectar pessoas e recursos, podemos transformar a maneira como aprendemos e crescemos juntos.

Este projeto representa nosso compromisso com a construção de uma educação mais acessível, equitativa e enriquecedora, refletindo a convicção de que o conhecimento deve ser um direito de todos.

## 2. DOCUMENTO DE ABERTURA DO PROJETOS

### 2.1-Prefácio

Este documento destina-se a educadores, alunos, administradores de instituições de ensino, desenvolvedores de conteúdo educacional e interessados em promover a democratização do acesso ao conhecimento. Ele é uma fonte abrangente de informações sobre o projeto de criação de um site voltado para a educação de qualidade para todos, oferecendo insights sobre suas funcionalidades, objetivos e impacto potencial na comunidade educacional.

## 2.2-Glossário

**ONG (Organização Não Governamental):** Entidade sem fins lucrativos que promove a educação de qualidade e a inclusão social, oferecendo recursos e cursos acessíveis. A ONG visa empoderar comunidades e garantir que a educação esteja ao alcance de todos.

**Acessibilidade:** Capacidade de um site ou conteúdo digital ser utilizado por todas as pessoas, independentemente de suas habilidades ou limitações, garantindo que todos possam acessar e interagir com as informações.

**Colaboração:** Processo em que indivíduos ou grupos trabalham juntos para atingir um objetivo comum, compartilhando conhecimento, experiências e recursos.

**Conteúdo Educacional:** Informações, materiais ou recursos que têm como objetivo ensinar ou informar sobre um determinado assunto ou tema.

**Democratização da Educação:** Processo de tornar a educação mais acessível a todos, independentemente de fatores como classe social, localização geográfica ou nível de escolaridade.

**E-learning:** Forma de aprendizado que utiliza a tecnologia e a internet para oferecer cursos e conteúdos educativos, permitindo que os alunos aprendam de forma remota.

**Feedback:** Opiniões ou sugestões fornecidas por usuários ou participantes sobre um produto ou serviço, que ajudam na melhoria contínua da oferta.

## 2.3-Definição de requisitos de usuários

**Requisitos Funcionais:** Funcionalidades específicas que o site deve oferecer, como cadastro de usuários, acesso a conteúdos educacionais, sistema de feedback, busca de conteúdo e integração com redes sociais.

**Requisitos Não Funcionais:** Características do site que definem como ele deve operar, incluindo usabilidade, desempenho, segurança, compatibilidade, acessibilidade e manutenibilidade.

**Lei Geral de Proteção de Dados (LGPD):** Assegurar o tratamento adequado de dados pessoais.

**Desempenho:** O tempo de resposta do sistema não deve ultrapassar 3 segundos para ações de busca e carregamento de páginas.

**Segurança:** Os dados do usuário devem ser protegidos por criptografia, e o sistema deve aderir às diretrizes da LGPD para garantir a privacidade das informações pessoais.

**Escalabilidade:** O sistema deve suportar um aumento no número de usuários simultâneos sem perda significativa de desempenho.

**Pesquisa Personalizada:** O sistema permitirá que os usuários busquem conteúdos educacionais de forma intuitiva, com base em palavras-chave ou áreas de interesse.

**Recomendações de Conteúdo:** O sistema sugerirá materiais personalizados com base no histórico de uso e nas preferências do usuário.

## 2.4-Arquitetura do sistema

**2.4.1-Frontend:** Interface do usuário desenvolvida com React, que fornece uma experiência interativa e responsiva.

-**Componente de Pesquisa:** Permite que o usuário busque conteúdos específicos.

-**Componente de Recomendação:** Exibe conteúdos sugeridos com base no histórico do usuário.

- **Componente de Navegação:** Facilita o acesso às diferentes seções da plataforma.

**2.4.2-Backend:** API desenvolvida com Node.js, responsável por processar as requisições, aplicar regras de negócio e interagir com o banco de dados.

- **Módulo de Autenticação:** Gerencia o login e a segurança das sessões de usuário.

-**Módulo de Conteúdo:** Gerencia o acesso e a organização dos conteúdos educacionais.

**2.4.3-Banco de Dados:** SQLite, para armazenamento de dados relacionados aos usuários e conteúdos educacionais.

- **Tabela de Usuários:** Armazena informações como nome, e-mail e preferências.

**-Tabela de Conteúdos:** Contém os materiais educacionais, categorizados por tema e formato.

## 2.5-Especificação de requisitos do sistema

### **Cadastro de Usuário**

O sistema deve permitir que o usuário crie uma conta informando e-mail e senha.

#### **Login e Autenticação**

O sistema deve permitir que o usuário faça login com e-mail e senha.

### **Gerenciamento de Conteúdo**

O administrador do sistema deve poder adicionar, editar ou remover conteúdos educacionais.

Os conteúdos devem ser categorizados por tema e tipo (vídeo, artigo, etc.).

### **Desempenho**

O tempo de resposta para buscas deve ser inferior a 3 segundos em condições normais de operação.

O sistema deve suportar pelo menos 1000 usuários simultâneos sem degradação perceptível no desempenho.

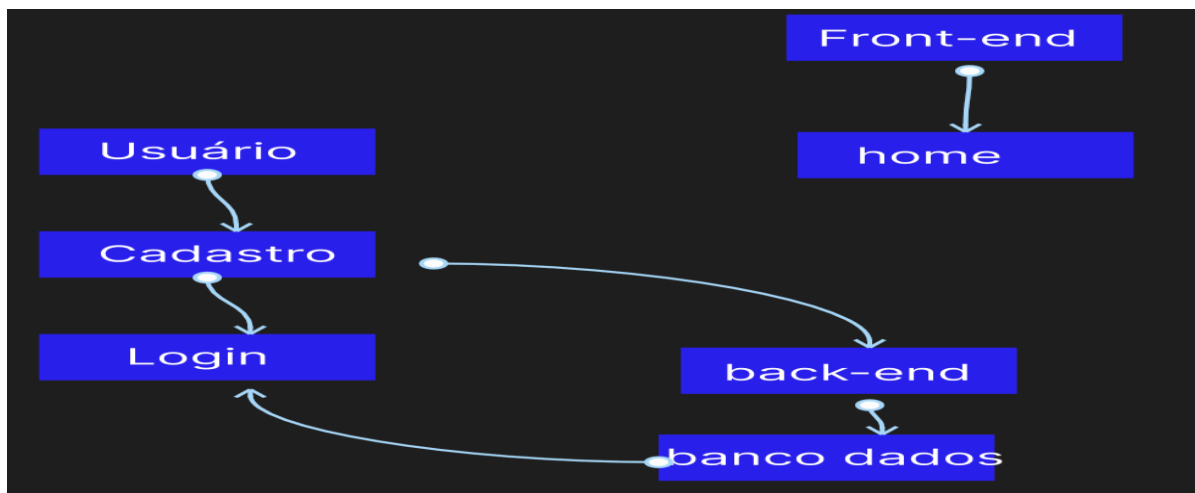
### **Segurança**

Os dados do usuário devem ser protegidos por criptografia (senha hashada com algoritmo bcrypt).

### **Compatibilidade**

O sistema deve funcionar corretamente nos navegadores mais usados: Chrome, Firefox, Safari e Edge.

## 2.6-Modelos do sistema



## 2.7-evolução do sistema

**Segurança e Privacidade:** Atualizações frequentes serão essenciais para atender a padrões de segurança em constante evolução, protegendo os dados dos usuários

**Mudanças nas Necessidades do Usuário:** Novas funcionalidades podem ser adicionadas, como sistemas de aprendizado adaptativo, integração com plataformas de ensino online e suporte a novas línguas.

## 3.0-Requisitos do Sistema

### 3.1-Requisitos Funcionais de Software

Cadastro de usuário	
Função	Permite que usuários novos cadastrem no sistema
Descrição	O sistema deve fornecer um formulário de cadastro onde os usuários podem inserir suas informações pessoais para criar uma conta. Esse cadastro permitirá que eles acessem os conteúdos educacionais disponíveis.
Entradas	Nome completo, Endereço de e-mail, Senha, Confirmação de senha
Fonte	Dados inseridos pelos usuários através da interface do formulário de cadastro.
Saídas	Mensagem de confirmação de cadastro (se bem-sucedido).

	Mensagem de erro (se houver problemas, como e-mail já cadastrado ou senhas não coincidentes)
<b>Ação</b>	Quando o usuário preencher o formulário e clicar no botão de "Cadastrar", o sistema deverá validar os dados informados, armazená-los no banco de dados e fornecer uma resposta visual informando se o cadastro foi realizado com sucesso ou se ocorreram erros.
Acesso ao Conteúdo Educacional	
<b>Função</b>	Permitir que os usuários acessem os materiais educacionais disponíveis na plataforma
<b>Descrição</b>	O sistema deve disponibilizar uma interface onde os usuários autenticados possam visualizar e acessar diversos conteúdos educacionais, como textos, vídeos e cursos, organizados por categorias
<b>Entradas</b>	Credenciais de login (e-mail e senha) para autenticação do usuário. Seleção da categoria ou tipo de conteúdo desejado.
<b>Fonte</b>	Dados provenientes do banco de dados que armazenam informações sobre os conteúdos disponíveis e as credenciais dos usuários.
<b>Saídas</b>	Lista de conteúdos educacionais disponíveis, com títulos, descrições e links para acesso. Mensagem de erro (se o usuário não estiver autenticado ou se não houver conteúdo disponível).
<b>Ação</b>	Quando o usuário fizer login na plataforma e selecionar um conteúdo para visualizar, o sistema deverá verificar se o usuário está autenticado, recuperar as informações do conteúdo solicitado no banco de dados e apresentá-las na interface
Sistema de Login	
<b>Função</b>	Permitir logar
<b>Descrição</b>	Permite ter acessibilidade aos históricos de vídeos
<b>Entradas</b>	E-mail e senhas cadastradas
<b>Fonte</b>	Dados cadastrados do usuário



<b>Saídas</b>	Acesso ao pagina de vídeos
<b>Ação</b>	Inserir os dados cadastrados
Sistema de Feedback	
<b>Função</b>	Permitir que os usuários deixem feedback sobre os conteúdos ducacionais e a plataforma.
<b>Descrição</b>	O sistema deve fornecer uma interface onde os usuários possam avaliar e comentar sobre os materiais que acessaram, permitindo a coleta de opiniões e sugestões para melhorias.
<b>Entradas</b>	Avaliação (nota de 1 a 5 estrelas). Comentário (opcional). Identificação do conteúdo avaliado (ID do conteúdo).
<b>Fonte</b>	Dados inseridos pelos usuários através da interface de feedback.
<b>Saídas</b>	Mensagem de confirmação de envio de feedback (se bem-sucedido). Mensagem de erro (se houver problemas, como dados inválidos).
<b>Ação</b>	Quando o usuário preencher o formulário de feedback e clicar no botão de "Enviar", o sistema deve validar os dados, armazená-los no banco de dados associado ao conteúdo específico e fornecer uma resposta visual informando se o feedback foi enviado com sucesso ou se ocorreram erros

Sistema de Busca de Conteúdos	
<b>Função</b>	Permitir que os usuários realizem buscas por conteúdos educacionais disponíveis na plataforma.
<b>Descrição</b>	O sistema deve fornecer uma funcionalidade de busca que permita aos usuários pesquisar por palavras-chave, categorias

	ou tipos de conteúdo (como textos, vídeos e cursos) para encontrar facilmente os materiais desejados.
<b>Entradas</b>	Termo de busca (palavras-chave inseridas pelo usuário). Filtros adicionais (opcionais), como categorias ou tipo de conteúdo.
<b>Fonte</b>	Dados inseridos pelos usuários na barra de busca e informações armazenadas no banco de dados sobre os conteúdos disponíveis
<b>Saídas</b>	Lista de resultados da busca, apresentando títulos, descrições e links para os conteúdos que correspondem aos critérios de pesquisa. Mensagem de aviso (se nenhum resultado for encontrado).
<b>Ação</b>	Quando o usuário digitar um termo de busca e clicar no botão de "Buscar", o sistema deverá processar a solicitação, consultar o banco de dados para encontrar conteúdos correspondentes e exibir os resultados na interface
Registro de Vídeos Assistidos	
<b>Função</b>	Permitir que o sistema registre os vídeos assistidos pelos usuários e identifique quais vídeos foram completamente finalizados.
<b>Descrição</b>	O sistema deve acompanhar o progresso dos usuários em relação aos vídeos educacionais assistidos, armazenando informações sobre quais vídeos foram iniciados, quais foram assistidos até o final e o tempo assistido para cada um.
<b>Entradas</b>	Identificação do usuário (para autenticação). ID do vídeo assistido. Progresso do vídeo (percentual assistido)
<b>Fonte</b>	Dados gerados pela interação do usuário com o player de vídeo, que registra automaticamente o tempo assistido e o status de conclusão.
<b>Saídas</b>	Atualização do registro de vídeos assistidos, incluindo o status de cada vídeo (iniciado, finalizado, não assistido). Relatório que pode ser exibido ao usuário, mostrando uma lista dos vídeos assistidos e seu status.

<b>Ação</b>	Quando o usuário assiste a um vídeo, o sistema deve registrar o ID do vídeo e o progresso em tempo real. Ao finalizar o vídeo, o sistema deve atualizar o status para "finalizado" e armazenar essas informações no banco de dados, permitindo que o usuário consulte posteriormente seu histórico de vídeos assistidos.
-------------	--

### 3.2-Requisitos não funcionais de Software

<b>Desempenho do Sistema</b>	
<b>Função</b>	Garantir que o sistema mantenha uma performance adequada, mesmo com um grande número de usuários simultâneos.
<b>Descrição</b>	O sistema deve proporcionar tempos de resposta rápidos e estabilidade, independentemente da carga de usuários ou volume de dados processados. Isso inclui o tempo de carregamento de páginas, consultas ao banco de dados e interações com vídeos e outros recursos multimídia.
<b>Entradas</b>	Número de usuários simultâneos. Solicitações de dados (como carregar páginas, acessar conteúdos ou consultar o banco de dados)
<b>Fonte</b>	Dados gerados pelos usuários, como acessos simultâneos à plataforma. Consultas ao banco de dados. Requisições de carregamento de mídia.
<b>Saídas</b>	Tempo de carregamento de páginas inferior a 3 segundos. Respostas de consultas ao banco de dados com tempo inferior a 1 segundo. Sistema responsivo para até 500 usuários simultâneos.
<b>Ação</b>	O sistema deve monitorar o desempenho em tempo real, otimizando o uso de recursos (como cache, balanceamento de carga e otimização de banco de dados) para garantir tempos de resposta rápidos. Caso a capacidade do sistema seja atingida, medidas como otimização de infraestrutura devem ser adotadas para manter a experiência do usuário.

Segurança do Sistema	
<b>Função</b>	Proteger os dados dos usuários e garantir que o sistema esteja protegido contra acessos não autorizados.
<b>Descrição</b>	O sistema deve implementar medidas de segurança robustas para proteger as informações pessoais dos usuários, incluindo dados de autenticação, históricos de acesso e informações de pagamento. Isso deve incluir criptografia, autenticação forte e monitoramento de atividades suspeitas
<b>Entradas</b>	Credenciais de usuários (e-mail e senha). Informações de pagamento (número do cartão de crédito, dados pessoais).
<b>Fonte</b>	Dados inseridos pelos usuários durante o cadastro e transações. Logs de acesso e atividades do sistema.
<b>Saídas</b>	Mensagens de erro em caso de tentativas de acesso não autorizadas. Alertas para administradores sobre atividades suspeitas ou tentativas de invasão.
<b>Ação</b>	O sistema deve utilizar protocolos de segurança, como HTTPS, para todas as transações, e armazenar senhas de forma criptografada. Além disso, deve realizar autenticação de dois fatores para os usuários ao acessar áreas sensíveis, e monitorar continuamente o sistema para detectar e responder rapidamente a qualquer tentativa de acesso não autorizado.
Usabilidade do Sistema	
<b>Função</b>	Garantir que a interface do usuário seja intuitiva e fácil de navegar.
<b>Descrição</b>	O sistema deve proporcionar uma experiência de usuário fluida e agradável, permitindo que os usuários naveguem facilmente entre as seções da plataforma, encontrem conteúdos e utilizem funcionalidades sem necessidade de treinamento prévio.
<b>Entradas</b>	Interações do usuário com a interface (cliques, seleções, navegação).

	Feedback do usuário sobre a experiência de uso.
<b>Fonte</b>	Dados de uso coletados a partir de ferramentas de análise de comportamento do usuário. Questionários ou formulários de feedback fornecidos aos usuários.
<b>Saídas</b>	Relatórios de usabilidade que identifiquem áreas de melhoria. Aumento na taxa de retenção de usuários e diminuição das taxas de abandono durante a navegação.
<b>Ação</b>	O sistema deve ser projetado seguindo diretrizes de design centradas no usuário, com elementos de interface (como botões, menus e formulários) organizados de maneira lógica e consistente. Além disso, deve oferecer suporte acessível e fácil de entender, como tutoriais e dicas contextuais, para ajudar os usuários em sua interação com a plataforma.
Compatibilidade do Sistema	
<b>Função</b>	Assegurar que o sistema funcione corretamente em diferentes dispositivos e navegadores.
<b>Descrição</b>	O sistema deve ser projetado para ser acessível e funcional em uma ampla variedade de plataformas, incluindo desktop, tablets e dispositivos móveis, bem como em diferentes navegadores (Chrome, Firefox, Safari, Edge) e sistemas operacionais (Windows, macOS, iOS, Android).
<b>Entradas</b>	Dispositivo e navegador utilizados pelo usuário. Resolução da tela.
<b>Fonte</b>	Dados de acesso que informam sobre os dispositivos e navegadores utilizados pelos usuários. Testes de compatibilidade realizados durante o desenvolvimento.
<b>Saídas</b>	Interface que se adapta responsivamente a diferentes tamanhos de tela. Funcionalidades que operam de maneira consistente em todos os navegadores suportados.

<b>Ação</b>	O sistema deve ser testado em várias plataformas e navegadores durante o desenvolvimento para garantir que todas as funcionalidades operem de maneira eficaz. Isso deve incluir a implementação de design responsivo e a utilização de tecnologias web padrão para garantir a compatibilidade com as versões mais comuns de navegadores.
Escalabilidade do Sistema	
<b>Função</b>	Permitir que o sistema se adapte a um aumento no número de usuários e na quantidade de dados sem perda de desempenho.
<b>Descrição</b>	O sistema deve ser projetado para escalar horizontalmente e verticalmente, permitindo que novos recursos e servidores sejam adicionados conforme a demanda aumenta. Isso garante que a plataforma continue a funcionar de maneira eficiente mesmo durante picos de acesso.
<b>Entradas</b>	Aumento no número de usuários simultâneos. Crescimento na quantidade de conteúdo armazenado (ex: vídeos, artigos).
<b>Fonte</b>	Dados de uso que indicam padrões de acesso e crescimento da base de usuários. Análise de desempenho sob diferentes cargas de trabalho.
<b>Saídas</b>	Capacidade de suportar um aumento de 100% no tráfego sem degradar a performance. Relatórios sobre o uso de recursos e necessidades futuras de escalabilidade.
<b>Ação</b>	O sistema deve utilizar arquiteturas de microserviços e serviços em nuvem que permitem a adição fácil de recursos conforme necessário. Além disso, deve incluir monitoramento de desempenho em tempo real para identificar quando a escalabilidade é necessária e automatizar o provisionamento de recursos.
Manutenibilidade do Sistema	
<b>Função</b>	Facilitar a manutenção e atualização do sistema ao longo do tempo.

<b>Descrição</b>	O sistema deve ser projetado de maneira modular, permitindo que as atualizações, correções de bugs e adições de novos recursos sejam realizadas com facilidade e sem impactar negativamente as funcionalidades existentes.
<b>Entradas</b>	Código-fonte do sistema. Documentação técnica e de usuário.
<b>Fonte</b>	Estruturas de código e padrões de design utilizados durante o desenvolvimento. Feedback da equipe de desenvolvimento sobre a facilidade de implementação de mudanças.
<b>Saídas</b>	Tempo reduzido para implementar atualizações e correções. Documentação clara que suporte a equipe de manutenção na realização de alterações.
<b>Ação</b>	O sistema deve seguir princípios de design de software e utilizar práticas de codificação limpa. A documentação deve ser mantida atualizada e deve incluir informações sobre a arquitetura do sistema, fluxos de trabalho e instruções de uso, permitindo que novos desenvolvedores entendam rapidamente o sistema.

#### 4.0-Casos de uso

##### **Cadastro de usuário**

**Ator Principal:** Usuário

**Descrição:** O usuário deseja se cadastrar na plataforma para acessar conteúdos educacionais.

**Pré-condições:** O usuário não deve ter uma conta existente.

##### **Fluxo Principal:**

- O usuário acessa a página de cadastro.
- O sistema exibe um formulário de cadastro.
- O usuário preenche os campos obrigatórios ( e-mail, senha).
- O usuário clica no botão "Cadastrar".
- O sistema valida os dados e cria uma nova conta.

O usuário recebe uma mensagem de sucesso e é redirecionado para a página de login.

**Pós-condições:** O usuário é cadastrado no sistema e pode fazer login.

Assistir a um vídeo

**Ator Principal:** Usuário

**Descrição:** O usuário deseja assistir a um vídeo educacional na plataforma.

**Pré-condições:** O usuário deve estar autenticado na plataforma.

**Fluxo Principal:**

- O usuário navega até a seção de vídeos.
- O sistema exibe uma lista de vídeos disponíveis.
- O usuário seleciona um vídeo da lista.
- O sistema direciona para o youtube onde situa o vídeo

**Fornecer feedbacks**

**Ator Principal:** Usuário

**Descrição:** O usuário deseja fornecer feedback sobre um vídeo assistido.

**Pré-condições:** O usuário deve ter assistido a um vídeo.

**Fluxo Principal:**

- O usuário acessa a página do vídeo assistido.
- O sistema exibe uma opção para fornecer feedback.
- O usuário escolhe uma classificação (1 a 5 estrelas) e opcionalmente escreve um comentário.
- O usuário clica no botão "Enviar Feedback".
- O sistema valida os dados e registra o feedback no banco de dados.
- O sistema exibe uma mensagem de confirmação.

**Pós-condições:** O feedback do usuário é armazenado e pode ser utilizado para melhorar os conteúdos da plataforma.



## 5.0-Arquitetura do sistema

### **Camada de Apresentação (Frontend):**

**Descrição:** Esta camada é responsável pela interação do usuário com o sistema. Inclui a interface do usuário, que deve ser responsiva e acessível em diferentes dispositivos (desktop, tablets, smartphones).

**Tecnologias:** HTML, CSS, JavaScript, frameworks como React.

**Funcionalidades:**

- Exibição de conteúdos (textos, vídeos, imagens).

- Formulários para cadastro, login e feedback.

- Navegação intuitiva entre seções do site.

### **Camada de Lógica de Negócio (Backend):**

**Descrição:** Esta camada processa as solicitações dos usuários e executa a lógica de negócios do sistema. É responsável pela autenticação de usuários, manipulação de dados e gerenciamento de sessões.

**Tecnologias:** Linguagens de programação como Python, Node.js, Java ou PHP; frameworks como Express, Django ou Spring.

**Funcionalidades:**

- Gerenciamento de usuários (cadastro, login, logout).

- Processamento de vídeos e registros de visualização.

- Armazenamento e recuperação de feedback dos usuários.

### **Camada de Persistência de Dados (Banco de Dados):**

**Descrição:** Esta camada é responsável por armazenar e gerenciar os dados do sistema, incluindo informações de usuários, vídeos e feedback.

**Tecnologias:** Sistemas de gerenciamento de banco de dados como MySQL

**Funcionalidades:**

- Armazenamento seguro de dados de usuários e senhas (criptografadas).

- Estruturação de dados sobre vídeos e seus metadados.

- Registro do histórico de acesso e feedback dos usuários.

**Camada de Serviços Externos (APIs):**

**Descrição:** Esta camada permite a integração com serviços externos, como plataformas de pagamento, serviços de autenticação e APIs de análise.

**Funcionalidades:**

- Integração com serviços de pagamento para doações.

- Utilização de APIs para envio de e-mails de confirmação.

- Acesso a serviços de análise para monitorar o uso e desempenho do sistema.