

Documento de Visão		
Nome do Projeto: Sistema de Recomendação de Conteúdo		
Autores: Vitor Hugo Dias Santos, Ítalo Leal Lana Santos		
Fornecedor(es) de Requisitos		
Nome	E-mail	Cargo ou Função - Empresa
Ítalo Leal	italoleallanasantos@gmail.com	Gestor, Dev
Vitor Hugo Dias Santos	vhugosantos2025@gmail.com	Dev
Descrição do Problema		
<p>Atualmente, plataformas que possuem grandes volumes de dados (produtos/mercadorias) enfrentam dificuldade em conectar usuários a conteúdos relevantes. A busca tradicional por palavras-chave é limitada e não captura o contexto semântico, gerando experiências de usuário frustrantes e baixo engajamento com o catálogo disponível.</p>		
Descrição Geral da Solução (Escopo)		
<p>O sistema será uma solução de backend focada em Inteligência Artificial, composta por dois módulos principais.</p> <ol style="list-style-type: none"> 1. API Principal (Java): Responsável por expor os endpoints públicos, receber as requisições dos clientes, gerenciar regras de negócio e orquestrar a comunicação com o serviço de IA. 2. Motor de Recomendação (Python + ML): Um microserviço que processa textos utilizando <i>Sentence Transformers</i> para gerar <i>embeddings</i> (vetores semânticos). Utilizará busca vetorial (FAISS ou Pinecone) para identificar itens similares (clustering) e retornar recomendações baseadas em contexto, não apenas em palavras exatas. 		
Fora do Escopo		
<p>Interface Gráfica (Frontend): O projeto foca na API e no motor de processamento. Não haverá desenvolvimento de telas para usuário final (apenas documentação da API e painel simples de admin se necessário).</p> <p>Processamento de Áudio/Vídeo: O sistema aceitará apenas <i>texto</i> como entrada para recomendação.</p> <p>Gestão de Pagamentos: Não haverá integração com sistemas financeiros.</p> <p>Criação de Conteúdo: O sistema recomenda conteúdos existentes, não gera novos textos (não é um LLM gerativo como GPT).</p>		
Usuários		
<p>Cliente (Sistema Externo): Aplicação terceira (e-commerce, blog, intranet) que consome a API para exibir recomendações.</p> <p>Administrador de Dados: Responsável por carregar os <i>datasets</i> (artigos, produtos) e treinar/atualizar o índice vetorial.</p> <p>Desenvolvedor: Utiliza a documentação da API para integrar a solução.</p>		

Requisitos Funcionais			
ID	Descrição do Requisito Usuário + Verbo + Predicado	Prioridade	Complexidade
R01	Cliente envia texto para análise	Alta	Baixa
R02	Sistema retorna lista de itens similares	Alta	Alta
R03	Administrador cadastra novos itens no catálogo.	Alta	Média
R04	Sistema gera embeddings (vetores) do novo item	Alta	Alta
R05	Administrador atualiza índice de busca vetorial	Alta	Alta
R06	Cliente filtra recomendação por categoria	Média	Média
R07	Administrador remove itens do catálogo	Média	Baixa
R08	Administrador visualiza métricas de acurácia	Baixa	Média
R09	Cliente consulta status da API (Health Check)	Baixa	Baixa
R10	Sistema armazena logs de requisições	Média	Baixa
Requisitos Não Funcionais			
ID	Descrição do Requisito	Prioridade	Complexidade
RN01	Tempo de resposta para uma “recomendação” deve ser inferior a 200ms	Alta	Alta
RN02	O sistema de busca vetorial deve suportar, no mínimo, 10.000 itens indexados	Alta	Média
RN03	A API deve ser desenvolvida em Java (Spring Boot ou Quarks)	Alta	Média
RN04	O microserviço de ML deve ser desenvolvido em Python	Alta	Média
RN05	A comunicação entre Java e Python deve ocorrer via HTTP(REST)	Média	Alta
RN06	O sistema deve estar em conformidade com a LGPD (anonimização de logs)	Alta	Média
Técnica(s) de Elicitação Utilizada(s)			
<p>Auto-entrevista Brainstorming</p> <p>Extensão web-scraping para pegar dados de sites como Amazon, etc</p>			