***Nome do grupo***

**LDM Soluções**

***Participantes***

**Lenine De Maria (**[**lenine.de.maria@ldmsolucoes.com.br**](mailto:lenine.de.maria@ldmsolucoes.com.br)**)**

**Leonardo De Maria (**[**leonardodmail@gmail.com**](mailto:leonardodmail@gmail.com)**)**

***Público-alvo***

O público-alvo do sistema de FAQ para NF-e do SEFAZ-SP inclui contribuintes, contadores, profissionais de tecnologia da informação e demais usuários que precisam esclarecer dúvidas sobre a emissão, validação e fiscalização da Nota Fiscal Eletrônica. Essas pessoas buscam informações rápidas e confiáveis para garantir o cumprimento das obrigações fiscais de forma eficiente.

***Justificativa para o desenvolvimento do FAQ NF-e***

O acesso a informações sobre NF-e é complexo devido à dificuldade em interpretar as normas fiscais brasileiras, e a escassez de recursos confiáveis e acessíveis para a consulta torna ainda mais complicado o entendimento das NF-e.

***Agente de ia para respostas sobre nf-e***

[1. Proposta preliminar para a construção do FAQ NF-e 3](#_Toc204100548)

[2. Pitch Deck do projeto 5](#_Toc204100549)

[3 Elementos adicionais 6](#_Toc204100550)

# Proposta preliminar para a construção do *FAQ NF-e*

**CENÁRIO**

**1**. Usuário: Ponto de entrada do sistema, onde o usuário interage para tirar dúvidas sobre NFEs.

**2**. FastAPI (NFE API): A interface web principal do sistema, responsável por receber as requisições do usuário e retornar as respostas. Atua como o ponto de comunicação entre o usuário e o backend.

**3**. Query Processor (Processamento de Consultas): Componente central que gerencia o fluxo de uma consulta do usuário. Ele decide se a consulta pode ser respondida pela base de conhecimento local ou se precisa de um fallback para a LLM externa.

**4**. Knowledge Base (ChromaDB - Base de Conhecimento Local): O banco de dados vetorial que armazena os pares de perguntas e respostas sobre erros de NFE. É a primeira fonte de busca para as consultas do usuário.

**5**. Perplexity AI (LLM Fallback): Um modelo de linguagem grande (LLM) externo que é acionado quando a base de conhecimento local não consegue fornecer uma resposta satisfatória para a consulta do usuário. Atua como um recurso de fallback para expandir a capacidade de resposta.

**6**. Embeddings (Sentence Transformers - Embeddings Semânticos): Processo que converte o texto (tanto das consultas quanto dos documentos na base de conhecimento) em representações numéricas (vetores) que capturam o significado semântico, permitindo buscas por similaridade.

**7**. Data Ingestion Pipeline (Pipeline de Ingestão de Dados): Um subsistema complexo responsável por coletar, processar e carregar dados de diversas fontes para a Base de Conhecimento. Inclui as seguintes etapas:

7.1. Data Sources (Fontes de Dados): Representa as diversas origens de informação, como documentos oficiais da SEFAZ-SP, blogs, artigos, PDFs e websites especializados em NFE.

7.2. Document Crawler (Identificação & Download): Componente que varre as fontes de dados (sites) para identificar e baixar documentos relevantes (PDFs, DOCXs, etc.) para processamento local.

7.3. PDF Processor (PyPDF2): Ferramenta específica para extrair texto bruto de arquivos PDF. \* 7.4. DOCX Processor (python-docx): Ferramenta específica para extrair texto bruto de arquivos DOCX.

7.5. Web Scraper (BeautifulSoup/Scrapy): Ferramenta para extrair conteúdo textual diretamente de páginas web (blogs, artigos online).

7.6. Text Processor (Regex/NLTK): Ferramenta para processar e extrair informações de arquivos de texto genéricos, utilizando expressões regulares ou bibliotecas de PLN.

7.7. Data Converter (Pares Q&A): Processo que recebe o texto bruto extraído e o transforma em pares estruturados de perguntas e respostas, prontos para serem adicionados à base de conhecimento.

7.8. Data Ingestion (Script Python): O script final que orquestra o carregamento dos pares Q&A formatados para dentro do ChromaDB.

**8**. Response Formatter (Formatação de Respostas): Componente que recebe as respostas da Base de Conhecimento ou da Perplexity AI e as formata de maneira clara e concisa para serem apresentadas ao usuário via API.

**9**. Langchain Agent (Processamento Inteligente): Uma representação de um agente Langchain que poderia ser usado para um processamento de consulta mais complexo, utilizando ferramentas (Tools) para buscar informações ou executar ações. No protótipo atual, o Query Processor lida com a lógica de fallback, mas um agente Langchain poderia ser integrado para cenários mais avançados.

**10**. Tools (Ferramentas de Busca - Opcional/Alternativo): Ferramentas que um Langchain Agent poderia utilizar para interagir com sistemas externos ou realizar buscas específicas, complementando suas capacidades de raciocínio.

# 2. Pitch Deck do projeto

Enviado junto com este documento no formato .pptx contento oito lâminas com uma apresentação estimada em cinco minutos aproximadamente.

Nome do arquivo enviado por email para [challenges@i2a2.academy](mailto:challenges@i2a2.academy):

* LDM\_Soluções\_Pitch\_Deck\_FAQ\_NFE.pptx

# Elementos adicionais

Fluxo operacional:

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Json com perguntas e respostas:

* json

{

"pergunta": "Como emitir NF-e?",

"resposta": "Passo 1: Acessar o portal..."

}