



Grupo: MOORE-290

I2A2_AGENTES_INTELIGENTES_PROJETO_FINAL

PARTICIPANTE DO GRUPO:

Nome	E-mail
Monica	monica01.elias@gmail.com

RELATÓRIO TÉCNICO DE PROVA DE CONCEITO (POC)

AGENTE DE EXTRAÇÃO E ESTRUTURAÇÃO DE DADOS FISCAIS

1 INTRODUÇÃO

Este documento apresenta o Relatório Técnico da Prova de Conceito (PoC) para o desenvolvimento de um agente de Inteligência Artificial (IA) generativa, focado na **extração e estruturação automática de dados de documentos fiscais** no formato XML (Nota Fiscal Eletrônica - Nfe). O projeto visa eliminar o input manual de dados, aumentando a eficiência e a acurácia dos processos fiscais e financeiros.

2 DETALHAMENTO DO TEMA ESCOLHIDO

2.1 PÚBLICO-ALVO

A solução é primariamente destinada a profissionais que lidam com grandes volumes de dados fiscais:

- **Analistas Fiscais e Contadores:** Para conciliação fiscal e fechamento de balanços.
- **Setor de Compras (Procurement):** Para validação de faturas e agilidade no ciclo de pagamento.
- **Equipes de Tecnologia (TI):** Para integração dos dados estruturados diretamente em sistemas de ERP (Enterprise Resource Planning) ou bases de dados.

2.2 JUSTIFICATIVA E VALOR AGREGADO

O processamento tradicional de documentos fiscais envolve a análise manual de arquivos XML complexos ou a transcrição de dados para sistemas, etapas que são intrinsecamente lentas e sujeitas a erros humanos.

Desafio Atual	Valor Agregado da Solução
Erros de Digitação	Acurácia de 100% na extração dos campos definidos, garantida pela validação Pydantic.
Lentidão Operacional	Processamento em segundos por documento, acelerando a conciliação e o ciclo de pagamento.
Dados Não Estruturados (XML)	Saída em JSON , pronta para integração via API em qualquer sistema moderno, eliminando a necessidade de parsers XML complexos.

3 DETALHAMENTO DO DESENVOLVIMENTO

3.1 ARQUITETURA DA SOLUÇÃO

O agente foi construído com uma arquitetura de *Chain* (Cadeia) utilizando a biblioteca **LangChain** para orquestrar o fluxo de trabalho entre o pré-processamento, o Large Language Model (LLM) e a validação de saída.

O fluxo de operação é dividido em três fases principais:

- 1. Entrada e Pré-processamento:** O código em Python recebe o arquivo (.xml ou .zip), decodifica o conteúdo e, em caso de ZIP, itera sobre cada arquivo fiscal.

- 2. Motor de Extração:** O modelo **Gemini 2.5 Flash** é acionado como um agente especializado em dados fiscais.

3. Validação e Saída: O **PydanticOutputParser** impõe que o resultado da extração seja um objeto JSON válido, aderente ao esquema fiscal definido.

3.2 MODELAGEM DOS DADOS (PYDANTIC)

Para garantir a integridade e a tipagem correta dos dados, foram utilizados dois modelos Pydantic, que definem estritamente o formato de saída do agente.

Objeto Python (Model)	Campo	Tipo	Função
NotaFiscalEstruturada	numero_nota	string	Número de identificação da NF.
	data_emissao	string	Data de emissão no formato DD/MM/AAAA
	cnpj_emitente	string	CNPJ da empresa que emitiu a nota.
	valor_total_nota	float	Valor total da transação.
	itens	list[ItemFiscal]	Lista de produtos extraídos.
ItemFiscal	descricao	string	Nome do produto/serviço.
	quantidade	float	Quantidade numérica.
	cfop	string	Código Fiscal de Operações e Prestações (4 dígitos).

3.3 FUNCIONALIDADES PRINCIPAIS

Função	Descrição
processar_arquivo_uploaded()	Lida com o upload do Google Colab, identifica o tipo de arquivo (.xml ou .zip) e realiza a decodificação de bytes para string.
extrair_dados_documento()	Contém a lógica de invocação do LLMChain . Envia o conteúdo do documento para o Gemini, recebe o JSON de volta e o valida via Pydantic. Esta função garante a acurácia dos dados.
Autenticação	Utiliza os.environ e serdata.get('GEMINI_API_KEY') para uma autenticação segura e robusta, passando a chave explicitamente ao construtor ChatGoogleGenerativeAI para evitar falhas de conexão (RefreshError).

3.4 OPERAÇÃO

A solução é operada em ambiente Colab, necessitando apenas da chave de API Gemini configurada e da execução sequencial das células de código. O ponto de interação se dá com o *widget files.upload()*, onde o usuário carrega o documento fiscal.

4 RESULTADOS DA PROVA DE CONCEITO

O teste de validação final foi realizado em um arquivo teste XML, resultando em uma extração 100% correta e validada pelo Pydantic, confirmando a eficácia do agente.

4.1 RESUMO DA EXTRAÇÃO

Exemplo:

Campo	Valor Extraído	Status
Arquivo Fonte	Cópia de NFe... (3).xml (nome do arquivo)	OK
NF/Nº	353057	OK
Data de Emissão:	04/07/2025	OK
CNPJ Emitente	92718354000194	OK
Valor Total	R\$ 29125.34	OK
Itens Extraídos	12	OK

5 CONCLUSÃO

A Prova de Conceito demonstrou que, a integração do Gemini 2.5 Flash, com o Pydantic e o LangChain, oferece uma solução de extração fiscal altamente precisa, automatizando uma tarefa de alto volume e eliminando os erros típicos do processo manual.

5 REFERÊNCIAS

5.1 FERRAMENTAS UTILIZADAS

- Google Gemini API (via langchain-google-genai)
- LangChain
- Pydantic
- Google Colaboratory (Ambiente de Execução)
- Python 3.12+
- Arquivo teste, Nota Fiscal Eletrônica, Lote 1. Disponível em:
<https://drive.google.com/drive/folders/174QH2jpLJAjr74rkq54a48cMPLkN2CNG>

5.1 REPOSITÓRIO DO PROJETO

Disponível em: <https://github.com/monica-sabag/Projeto-Final-Artefatos>