# **Prova de Conceito (POC) — Recuperação de Informações em Bulas de Medicamentos com LangChain e ChromaDB**

## **1. Objetivo**

Esta POC tem como objetivo demonstrar o uso integrado do **LangChain** e do **ChromaDB** para recuperação e geração de respostas baseadas em bulas de medicamentos. A ideia central é permitir que o usuário realize consultas semânticas (ex: “Quais são os efeitos colaterais do medicamento Dormec?”) e obtenha respostas resumidas, contextualizadas e fundamentadas no conteúdo da bula.

## **2. Funcionalidades Principais do LangChain**

O **LangChain** foi utilizado para construir o pipeline de processamento e resposta. Ele é responsável por:

1. **Carregamento do documento da bula**;
2. **Divisão em chunks (trechos menores)** para facilitar a vetorização e recuperação;
3. **Vetorização dos textos** utilizando embeddings pré-treinados (no caso, SentenceTransformer);
4. **Recuperação de trechos relevantes** do documento com base em uma *query* do usuário (ex: “efeitos colaterais”, “posologia”);
5. **Geração de respostas com LLM (Gemini AI)**, que resume e explica as informações recuperadas;
6. **Retorno da resposta final** acompanhada dos trechos de origem e de um **nível de confiança** na resposta.

## **3. Papel do ChromaDB**

O **ChromaDB** é o banco de dados vetorial utilizado para armazenar e consultar embeddings (vetores numéricos) dos trechos de texto das bulas.

Cada vetor é armazenado junto com **metadados** (ex: nome do medicamento, seção da bula, página, etc.).  
 Quando o usuário faz uma pergunta, o sistema converte a query em um embedding e busca no ChromaDB os **trechos mais semanticamente similares**.

### **Vantagens observadas:**

* Busca vetorial extremamente rápida;
* Suporte nativo a metadados;
* Integração simples com LangChain;
* Ideal para testes rápidos e prototipagem.

## **4. Comparativo: ChromaDB x Qdrant**

| **Situação** | **Banco recomendado** | **Motivo** |
| --- | --- | --- |
| Muitos documentos (dezenas de milhares ou mais) | **Qdrant** | Maior escalabilidade e confiabilidade |
| Aplicações em produção (API/servidor) | **Qdrant** | Suporte a persistência e buscas híbridas |
| Necessidade de filtros complexos e metadados | **Qdrant** | Melhor estrutura para filtros avançados |
| Protótipos e testes rápidos (POC) | **ChromaDB** | Simplicidade, velocidade e integração direta |
| Poucos documentos (centenas ou poucos milhares) | **ChromaDB** | Ideal para experimentação local |

## **5. Arquitetura Implementada**

[Arquivo da Bula]

↓

[Divisão em Chunks]

↓

[SentenceTransformer Embedding]

↓

[ChromaDB (armazenamento vetorial)]

↓

[Query do Usuário]

↓

[Recuperação dos Top-k Trechos do ChromaDB]

↓

[Gemini AI]

↓

[Resposta Resumida + Trecho + Confiança]

## **6. Detalhes de Implementação**

* **Modelo de Embedding:** SentenceTransformer  
   (Utilizado por não haver suporte atual ao embedding do Gemini AI e ausência de chave da OpenAI).
* **Base de Dados:** Bulas estruturadas e divididas por medicamento (ex: Dormec, Aciclovir).
* **LLM Utilizado:** Gemini AI.
* **Vetorização e busca:** Realizadas via ChromaDB.

## **7. Resultados e Observações**

### **Desempenho**

* Consulta via RAG: **15,78s**
* Consulta sem RAG: **26s** → O uso do RAG (com ChromaDB) proporcionou **respostas mais rápidas**.

### **Limitações Atuais**

* Em alguns casos, o modelo **não localizou o medicamento** mesmo ele estando na base — isso indica a necessidade de **pré-processamento mais robusto** e **melhor tratamento dos chunks** (especialmente nas seções que indicam o nome do medicamento).
* O RAG precisa ser aprimorado para garantir que o contexto do medicamento seja mantido em cada chunk.

## **8. Considerações Finais**

O uso combinado de **LangChain + ChromaDB + Gemini AI** mostrou-se eficaz para construir uma arquitetura de **RAG (Retrieval-Augmented Generation)** funcional e rápida para bulas de medicamentos.

Entretanto, observou-se que:

* O **ChromaDB** é significativamente mais rápido,
* Mas o **sucesso da recuperação** depende fortemente de um **bom pré-processamento dos textos** (definição de chunks consistentes e informativos).