

# RAG Interativo - Análise de Documentos

Sistema de análise inteligente de documentos usando RAG (Retrieval-Augmented Generation) com interface Streamlit.

## Funcionalidades

- Upload de ZIP com múltiplos documentos
- Suporte para PDF, DOCX, PPTX, TXT, MD
- Interface minimalista e intuitiva
- Chat interativo com IA
- Processamento automático com barra de progresso
- Histórico de conversas

## Formatos Suportados

- **PDF** - Documentos Adobe PDF
- **DOCX** - Documentos Microsoft Word
- **PPTX** - Apresentações Microsoft PowerPoint
- **TXT** - Arquivos de texto simples
- **MD** - Arquivos Markdown

## Configuração para GitHub + Streamlit Cloud

### 1. Preparar Repositório GitHub

```
# Criar novo repositório no GitHub  
# Fazer upload dos arquivos:  
# - app.py  
# - requirements.txt  
# - .streamlit/config.toml  
# - README.md
```

### 2. Deploy no Streamlit Cloud

1. Acesse [share.streamlit.io](https://share.streamlit.io)
2. Conecte sua conta GitHub
3. Selecione o repositório criado

4. Configure:
5. **Main file path:** app.py
6. **Python version:** 3.11
7. Clique em "Deploy"

### 3. Configurar Variáveis de Ambiente (Opcional)

Se quiser pré-configurar a API Key:

1. No Streamlit Cloud, vá em "Settings" > "Secrets"
2. Adicione:

```
OPENAI_API_KEY = "sua-api-key-aqui"
```

## ♂ Execução Local

### Pré-requisitos

- Python 3.8+
- OpenAI API Key

### Instalação

```
# Clonar repositório
git clone https://github.com/seu-usuario/seu-repositorio.git
cd seu-repositorio

# Instalar dependências
pip install -r requirements.txt

# Executar aplicação
streamlit run app.py
```

## Como Usar

### 1. Configuração Inicial

- Ao abrir a aplicação, insira sua OpenAI API Key
- A chave não é salva e precisa ser inserida a cada sessão

## 2. Upload de Documentos

- Prepare um arquivo ZIP com seus documentos
- Faça upload do ZIP na interface
- Aguarde o processamento automático

## 3. Chat Interativo

- Digite perguntas sobre seus documentos
- Receba respostas baseadas no conteúdo
- Visualize o histórico de conversas

## Exemplos de Perguntas

- "Faça um resumo de todos os documentos"
- "Quais são os pontos principais do documento X?"
- "Compare as informações entre os arquivos"
- "Extraia todas as datas mencionadas"
- "Liste os processos descritos nos documentos"

## Segurança

- API Keys não são armazenadas
- Documentos são processados temporariamente
- Dados não são persistidos entre sessões
- Processamento local no Streamlit Cloud

## Tecnologias Utilizadas

- **Streamlit** - Interface web
- **LlamaIndex** - Framework RAG
- **OpenAI GPT-3.5** - Modelo de linguagem
- **HuggingFace** - Embeddings
- **Python-docx** - Processamento DOCX
- **Python-pptx** - Processamento PPTX
- **PyPDF** - Processamento PDF

# Estrutura do Projeto

```
projeto/
├── app.py          # Aplicação principal
├── requirements.txt # Dependências Python
├── .streamlit/
│   ├── config.toml # Configuração Streamlit
│   └── README.md   # Documentação
```

## Solução de Problemas

### Erro de API Key

- Verifique se a API Key está correta
- Confirme que começa com "sk-"
- Verifique saldo na conta OpenAI

### Erro no Upload

- Confirme que o arquivo é um ZIP válido
- Verifique se contém documentos nos formatos suportados
- Limite de 200MB por arquivo no Streamlit Cloud

### Erro de Processamento

- Verifique se os documentos não estão corrompidos
- Alguns PDFs protegidos podem não funcionar
- Documentos muito grandes podem causar timeout

## Suporte

Para problemas ou sugestões:

1. Verifique a seção de solução de problemas
2. Consulte a documentação do Streamlit
3. Verifique logs no Streamlit Cloud

## Licença

Este projeto é de uso livre para fins educacionais e comerciais.

---

Desenvolvido com ❤ usando Streamlit e LlamaIndex