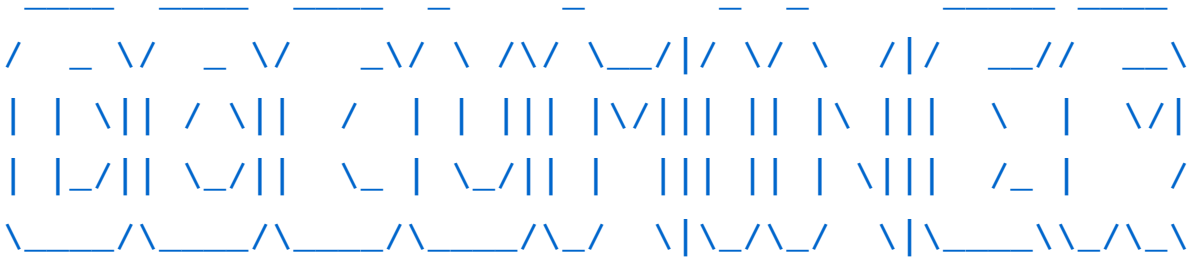


DocuMiner - Documentação



DocuMiner - Analise inteligente de documentos com IA.

Desenvolvido por: Kayki Ivan (Sh1ft)

Índice

- 1. Introdução
- 2. Objetivos
- 3. Arquitetura do Projeto
- 4. Tecnologias Utilizadas
- 5. Instalação e Execução
 - Linux
 - Windows
 - macOS
 - Android (Termux)
- 6. Funcionalidades
- 7. Considerações Finais
- 8. Referências

1. Introdução

O DocuMiner é uma ferramenta de linha de comando que utiliza inteligência artificial para analisar, resumir e responder perguntas sobre documentos em PDF. Ideal para estudantes, pesquisadores e profissionais que lidam com grandes volumes de informação, o DocuMiner transforma documentos complexos em insights acessíveis.

2. Objetivos

Geral

- Facilitar a análise e compreensão de documentos extensos por meio de IA.

DocuMiner - Documentação

Especificos

- Automatizar a conversao de PDFs para JSON.
- Permitir consultas interativas sobre o conteudo dos documentos.
- Suportar multiplos sistemas operacionais.
- Oferecer uma interface de terminal intuitiva.

3. Arquitetura do Projeto

```
DocuMiner/  
??? main.py  
??? core/  
? ??? processor.py  
? ??? prompt_builder.py  
??? converter/  
? ??? pdf_to_json.py  
??? ui/  
? ??? banner.py  
? ??? login.py  
? ??? terminal_ui.py  
??? data/  
? ??? jsons/  
??? pdfs/  
??? logs/  
??? user/  
??? requirements.txt  
??? config.sh  
??? config.ps1  
??? config_macos.sh  
??? config_termux.sh  
??? README.md
```

4. Tecnologias Utilizadas

- Python 3.11+
- Google Generative AI (gemini-1.5-flash)
- Rich: Para interfaces de terminal aprimoradas.
- PDFMiner: Extracao de texto de PDFs.
- OpenAI Whisper: Transcricao de audio (opcional).
- Tqdm: Barra de progresso.
- Python-dotenv: Gerenciamento de variaveis de ambiente.

5. Instalacao e Execucão

Linux (Debian, Arch e derivados)

```
git clone https://github.com/shlftx/DocuMiner.git
```

DocuMiner - Documentação

```
cd DocuMiner
chmod +x config.sh
./config.sh
```

No Windows

```
git clone https://github.com/shlftx/DocuMiner.git
cd DocuMiner
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope Process
.\config.ps1
```

No MacOS

```
git clone https://github.com/shlftx/DocuMiner.git
cd DocuMiner
chmod +x config_macos.sh
./config_macos.sh
```

No Android (Termux)

```
pkg update && pkg upgrade
pkg install git python ffmpeg
git clone https://github.com/shlftx/DocuMiner.git
cd DocuMiner
chmod +x config_termux.sh
./config_termux.sh
```

6. Funcionalidades

- Conversao de PDFs para JSON: Automatizada para facilitar o processamento.
- Consulta Interativa: Faça perguntas sobre o conteudo dos documentos.
- Resumos Inteligentes: Geracao de resumos concisos dos documentos.
- Suporte Multiplataforma: Compativel com Linux, Windows, macOS e Termux.
- Interface de Terminal Aprimorada: Utiliza a biblioteca Rich para uma melhor experiencia.

7. Consideracoes Finais

O DocuMiner e uma solucao poderosa para quem precisa extrair informacoes de grandes volumes de documentos. Com suporte a multiplas plataformas e uma interface intuitiva, torna a analise de documentos mais eficiente e acessivel.

8. Referencias

- Python Docs: <https://docs.python.org/3/>
- Gemini API: <https://ai.google.dev/>
- PyMuPDF: <https://pymupdf.readthedocs.io>
- Whisper (OpenAI): <https://github.com/openai/whisper>

DocuMiner - Documentação

- Projeto Termux: <https://termux.dev>

Desenvolvido com por Kayki Ivan (Sh1ft)

Importante para funcionamento do projeto:

Para que o DocuMiner funcione corretamente, voce precisa adicionar os arquivos PDF que contem as informacoes que vao treinar a IA na pasta 'pdfs/' dentro do diretorio do projeto.

Exemplo:

DocuMiner/

??? pdfs/

? ??? documento1.pdf

? ??? artigo_analise.pdf

? ??? ...

Depois de adicionar os PDFs, execute o processo de conversao e analise para que o sistema possa aprender e responder as suas perguntas.

E um passo simples, mas essencial para garantir o funcionamento correto do DocuMiner.