

> Menu

# Document Portal Overview

Relevant source files

## Purpose and Scope

This document provides a high-level introduction to the Document Portal system, a comprehensive document processing platform that combines AI-powered analysis, comparison, and conversational capabilities. The system enables users to upload documents and perform various operations including metadata extraction, document comparison, and interactive question-answering through retrieval-augmented generation (RAG).

This overview covers the system's core capabilities, architecture, and technology stack. For detailed setup instructions, see [Getting Started](#). For in-depth architectural details, see [System Architecture](#). For deployment information, see [Deployment and Infrastructure](#).

## System Capabilities

The Document Portal provides four primary document processing capabilities:

Capability	Description	Primary Components
Document Analysis	Extract structured metadata from single documents using LLMs	<code>DocumentAnalyzer</code> , <code>DocHandler</code>
Document Comparison	Compare two documents and generate structured comparison results	<code>DocumentComparator</code> , <code>DocumentComparatorLLM</code>
Single Document Chat	Interactive Q&A with individual documents using RAG	<code>SingleDocIngestor</code> , <code>ConversationalRAG</code>
Multi Document Chat	Cross-document conversational AI with unified knowledge base	<code>DocumentIngestor</code> , <code>ConversationalRAG</code>

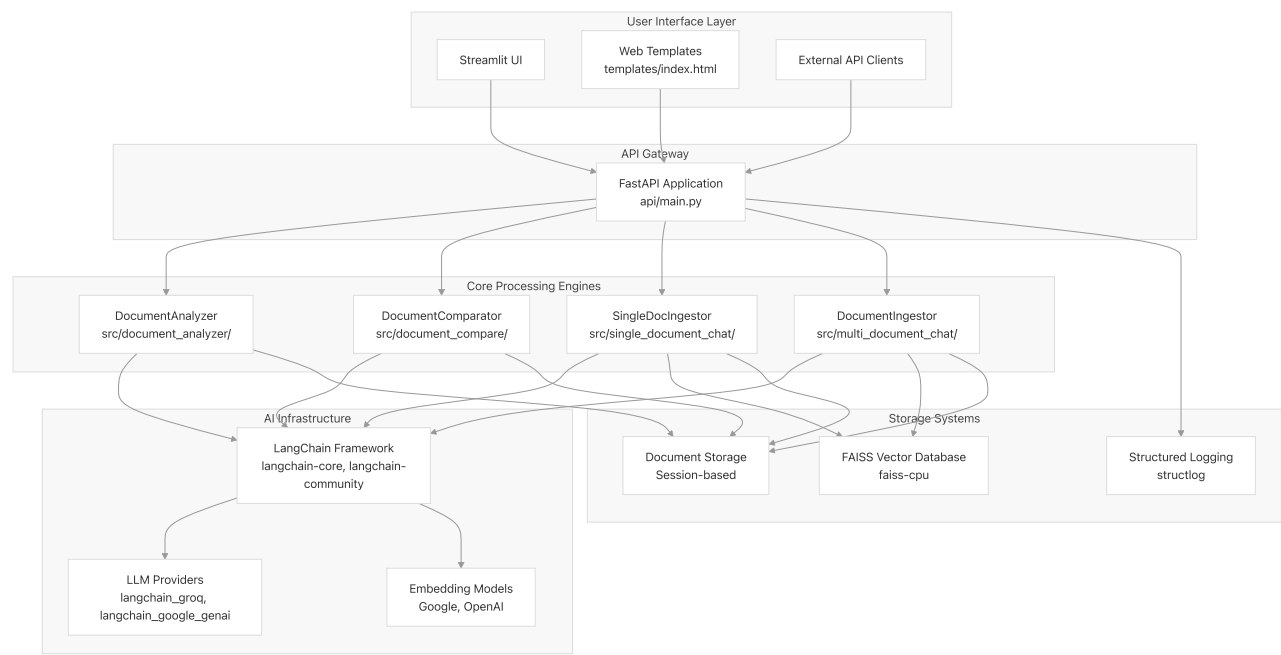
Ask Devin about sunnysavita10/document\_portal

Deep Research



# High-Level Architecture

## System Components Overview

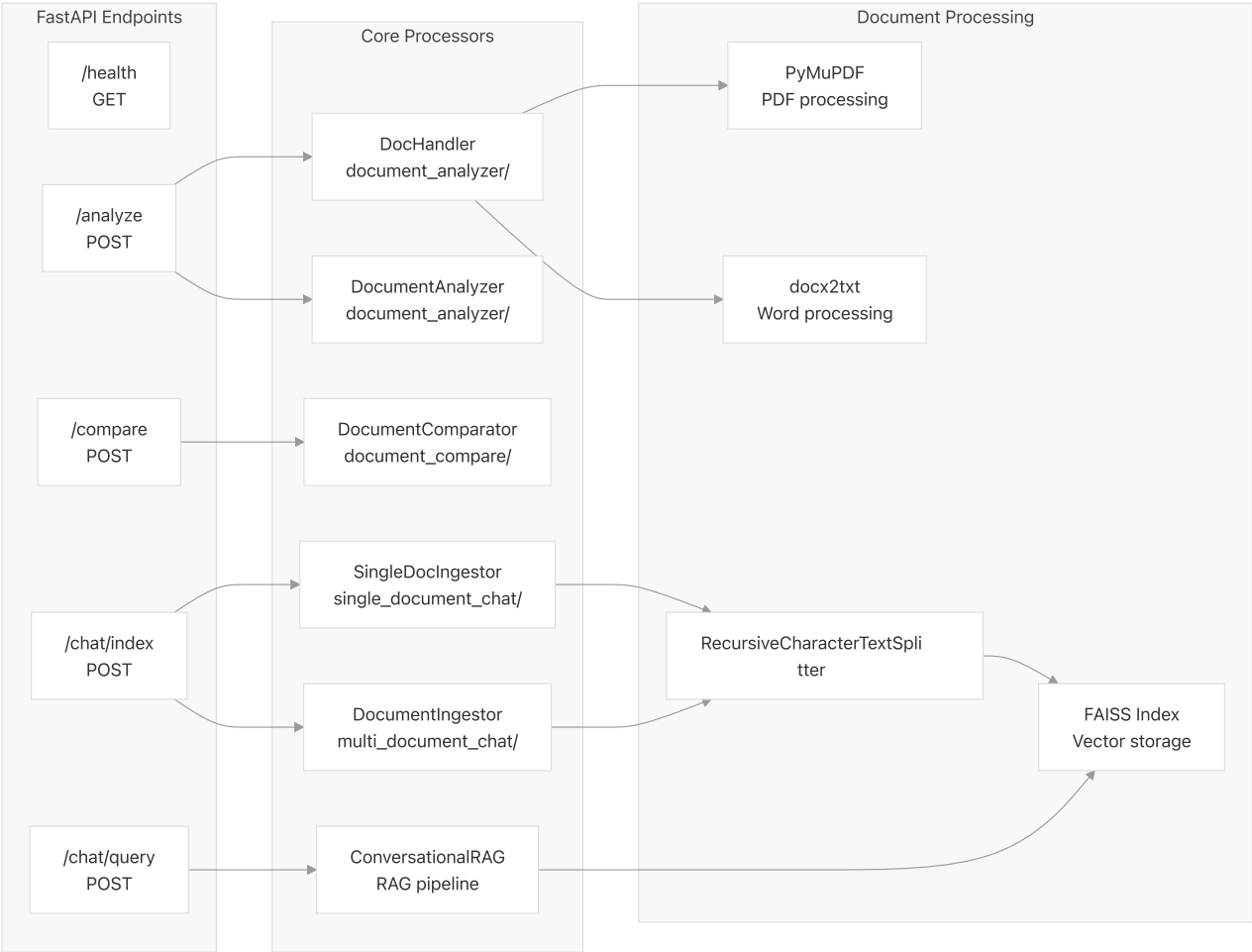


Sources: requirements.txt | 1-22

## API Endpoints and Processing Flow

Ask Devin about sunnysavita10/document\_portal

Deep Research



Sources: requirements.txt | 9–15

## Technology Stack

The Document Portal is built on a modern AI-first technology stack designed for document processing and conversational AI:

### Core Framework Dependencies

Component	Library	Version	Purpose
AI Framework	langchain	0.3.27	Core LLM orchestration
API Backend	fastapi	0.116.1	REST API services
Vector Storage	faiss-cpu	1.11.0	Similarity search
Document Processing	PyMuPDF	1.26.3	PDF text extraction

Ask Devin about sunnysavita10/document\_portal

Deep Research ☐

## LLM Provider Integrations

The system supports multiple LLM providers through dedicated LangChain integrations:

- **Groq:** Fast inference via `langchain-groq` (0.3.6)
- **Google GenAI:** Gemini models via `langchain-google-genai` (2.1.8)
- **Extensible:** Architecture supports additional providers

## Document Format Support

Format	Library	Capability
PDF	PyMuPDF , pypdf	Text extraction, metadata
DOCX	docx2txt	Word document processing
TXT	Built-in	Plain text handling

Sources: requirements.txt | 1-22    README.md | 41-57

## Key System Components

### Document Processing Pipeline

The system implements a modular document processing pipeline with four distinct workflows:

1. **Analysis Pipeline:** Single document metadata extraction using LLM-powered analysis
2. **Comparison Pipeline:** Structured comparison of two documents with LLM evaluation
3. **RAG Pipeline:** Retrieval-augmented generation for conversational document interaction
4. **Multi-Document RAG:** Cross-document knowledge synthesis and querying

### AI Integration Layer

The AI integration layer provides:

- **Model Management:** Configurable LLM and embedding model selection
- **Prompt Management:** Centralized prompt templates and registry
- **Chain Orchestration:** LangChain Expression Language (LCEL) pipeline composition
- **Output Parsing:** Structured output generation with error recovery

Ask Devin about sunnysavita10/document\_portal

- **File Management:** Temporary document storage with automatic cleanup
- **Logging:** Structured logging with component-level traceability

Sources: requirements.txt | 1-22

## Development and Deployment

The Document Portal supports multiple deployment scenarios:

### Local Development

- Conda/pip environment management
- Development server with hot reload
- Integrated testing framework with **pytest**

Ask Devin about sunnysavita10/document\_portal

Deep Research