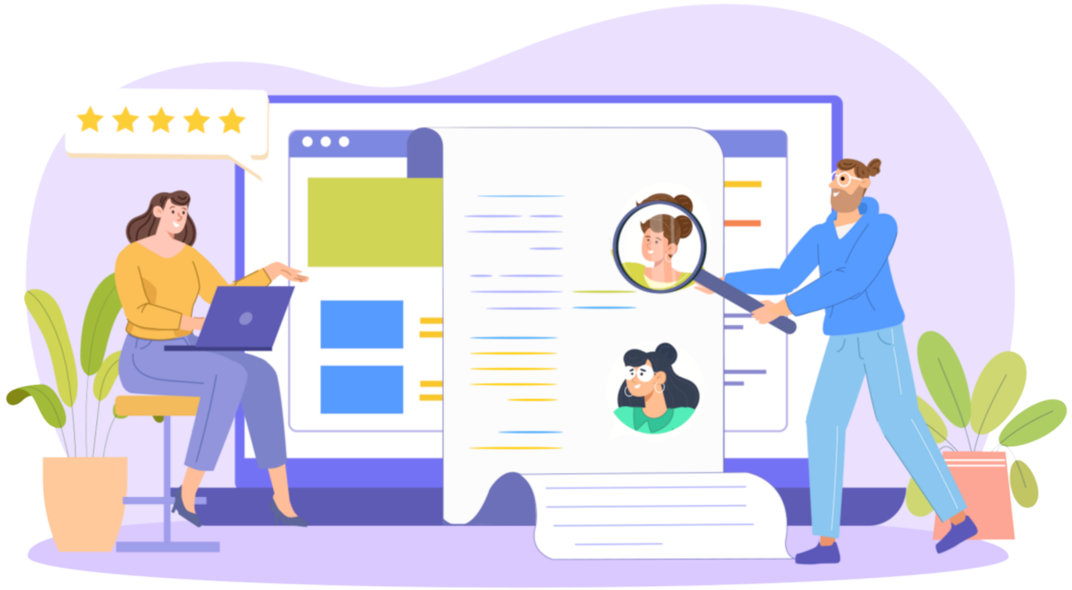
Chisphung

June 19, 2025

Curriculum Vitae extractor with Langchain

974731534 Linh Xuan, Thu Duc, TP.Ho Chi Minh www.linkedin.com/in/chisphung



Introduction

The Problem: Recruiters handle thousands of CVs in unstructured formats—manual review is time-

--> How to automatically extract the structured data from CVs?

System overview

PDF Loader

Splitter

JSON file

Embed -ding model

Vector data- base

GenAi chain

Prompt: Find a candidate who .....

Chat chain

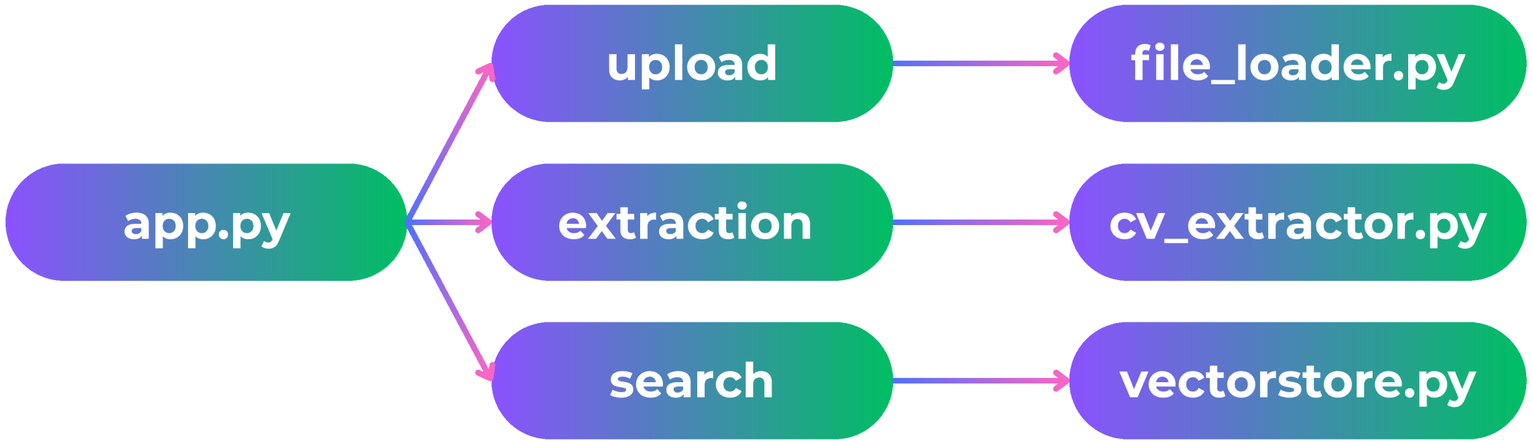
Search result

Directory structure

```
CV_EXTRACTOR_LANGCHAIN/
├── chat_histories/
│   └── .gitkeep
├── data_source/
│   └── generative_ai/
│       ├── download.sh
│       └── PhungMinhChiCV.pdf
├── docker/
│   └── Dockerfile
├── src/
│   ├── base/
│   │   ├── llm_model.py
│   │   └── output_parser.py
│   ├── chat/
│   │   ├── history.py
│   │   └── main.py
│   ├── rag/
│   │   ├──
cv_extractor.py
│   │   ├── file_loader.py
│   │   ├── main.py
│   │   ├──
offline_rag.py
│   │   ├── utils.py
│   │   └──
vectorstore.py
│   ├── app.py
│   └── streamlit.py
├── .dockerignore
├── .gitignore
├── docker-
compose.yml
├── README.md
└── requirements.txt
```

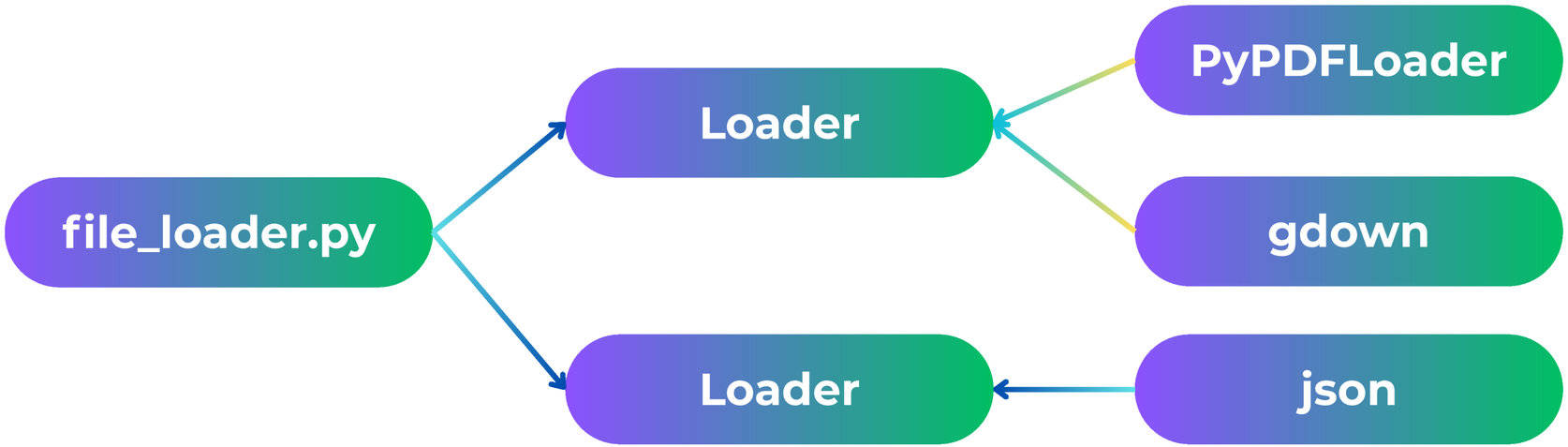
Solution architecture

- **API Layer**: FastAPI server in **app.py** handles **upload, extraction, search, and QA**.
- **Processing Flow:**
  - **Uploaded CVs** are **loaded** and **chunked**.
  - **LLM parses** the content i**nto structured JSON**.
  - **Extracted information** is **embedded** and **indexed**.
- **Storage:** FAISS vector store enables similarity search over CV chunks.



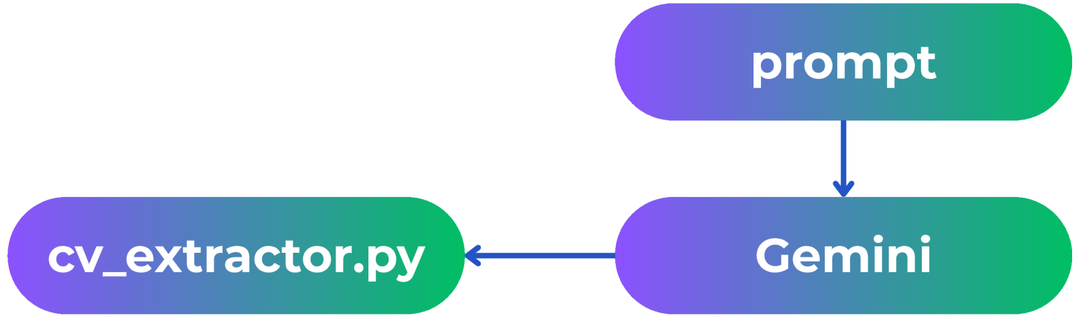
Module breakdown

- **file_loader.py:** Reads PDFs/HTML, splits into chunks.
- **cv_extractor.py:** Uses LLM to extract fields (e.g., name, skills) from each chunk.
- **vectorstore.py:** Creates and queries the FAISS index.
- **app.py:** Connects all modules, defines REST endpoints.



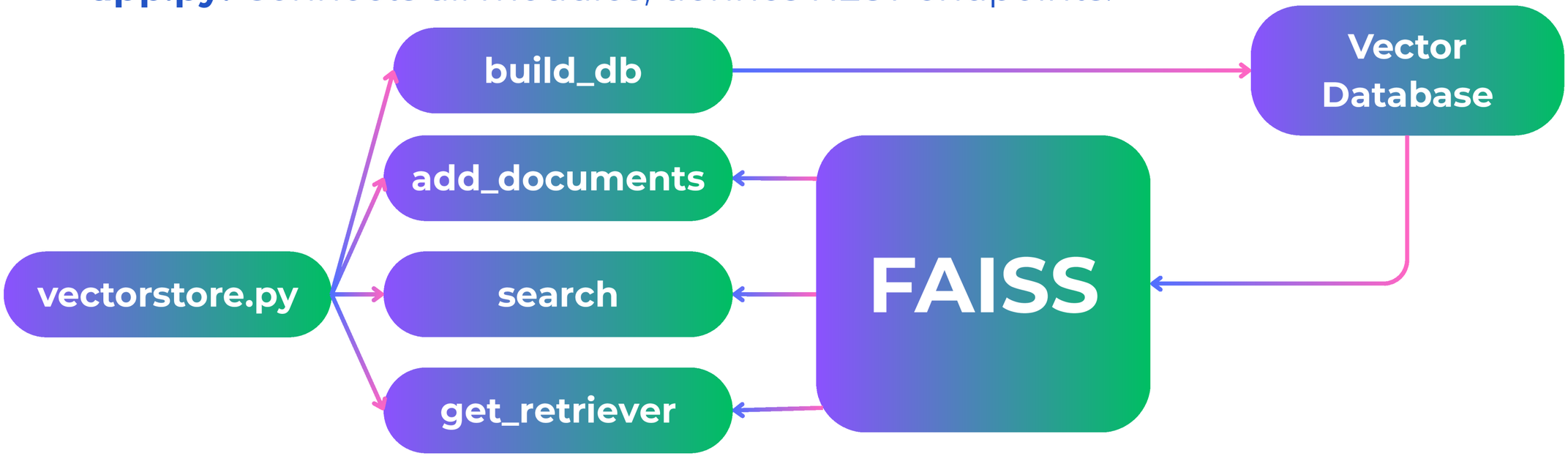
Module breakdown

- **file_loader.py:** Reads PDFs/HTML, splits into chunks.
- **cv_extractor.py:** Uses LLM to extract fields (e.g., name, skills) from each chunk.
- **vectorstore.py:** Creates and queries the FAISS index.
- **app.py:** Connects all modules, defines REST endpoints.



Module breakdown

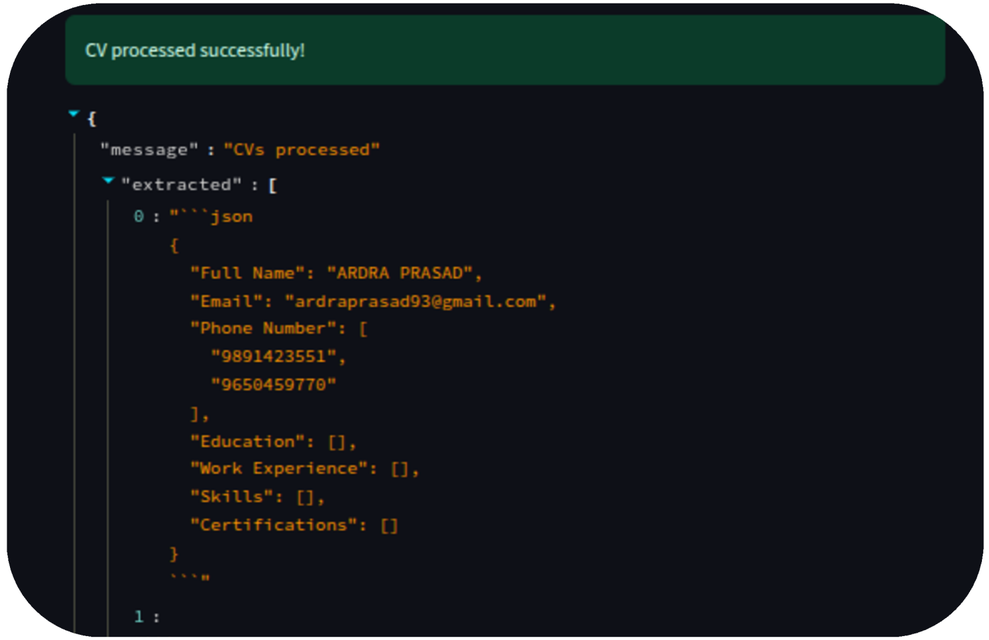
- **file_loader.py:** Reads PDFs/HTML, splits into chunks.
- **cv_extractor.py:** Uses LLM to extract fields (e.g., name, skills) from each chunk.
- **vectorstore.py:** Creates and queries the FAISS index.
- **app.py:** Connects all modules, defines REST endpoints.



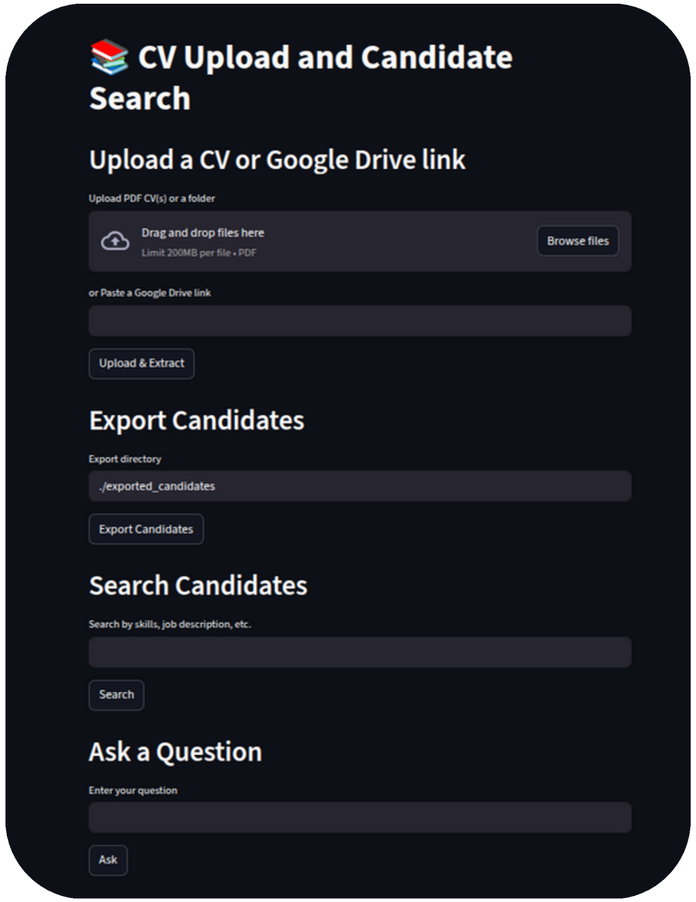
Technologies Used

- **Backend:** Python 3.11, FastAPI
- **LLM Orchestration:** LangChain, Google Generative AI
- **Search:** FAISS for embedding-based similarity queries
- **Front-End:**
  - Streamlit UI for uploading/searching/exporting results
  - Langchain playground for searching
- **Other:** PyPDFLoader

Demo



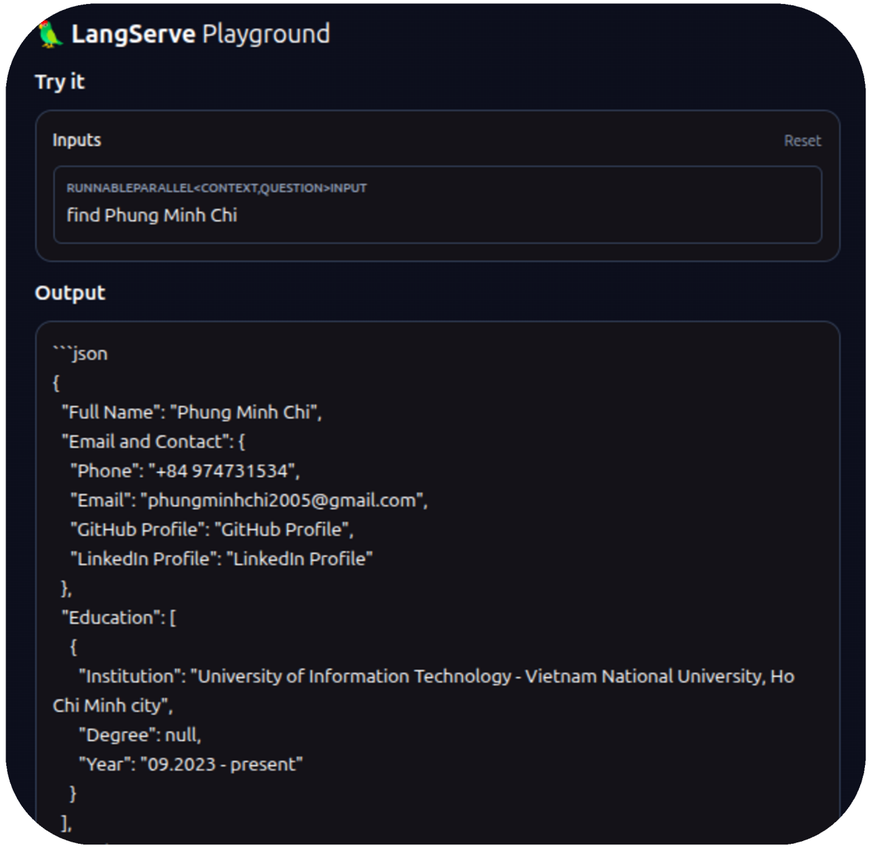
extraction result



streamlit

Demo

exported json



LangServe Playground

Future work

- Improve **retrieval accuracy**, the current model is **still facing hallucination** with some inaccurate information
- The current model **will be overloaded** if there are **over 100 CVs imported,** consider **batching strategies** or **async processing.**
- Explore **cloud deployment**
- Improve the **frontend site**
- Add **admin dashboard** for **reviewing** and **correcting** parsed results

References

1. *AI VIET NAM – AI COURSE 2024 Tutorial: RAG with LangChain for PDF Question Answering*
2. *Langchain Services - AIO, https://github.com/ThuanNaN/Langchain_Services/*
3. *Build a Retrieval Augmented Generation (RAG) App, https://python.langchain.com/docs/tutorials/rag/*
4. *ChatGoogleGenerativeAI, https://python.langchain.com/docs/integrations/chat/google_generative_ai/*
5. *PyPDFLoader, https://python.langchain.com/docs/integrations/document_loaders/pypdfloader/*
6. *langchain_community.document_loaders.pdf.PyPDFLoader, https://api.python.langchain.com/en/latest/document_loaders/langchain_community.document_loaders.pdf.PyPDFLoader.html*

Thank You

During the project, I needed to balance preparing for my final exams with working on the project. I could not dedicate full-time effort to it, so some parts may be imperfect. However, I truly appreciate the opportunity to learn about the related technologies.

Linh Xuan, Thu Duc, Tp. Ho Chi Minh

0974731534

www.linkedin.com/in/chisphung