

Back end system For:

1. PDF/TXT file upload → chunk → embed → store (Pinecone)
2. Question → retrieve chunks → generate answer
3. Logs metadata (e.g., chunk strategy, embedding model) into a database
4. Compare chunking/embedding strategies based on retrieval quality & latency

MVC Architecture

MVC = Separation of concerns

- *Model* → Database schema and logic
- *View* → FastAPI endpoints
- *Controller* → Business logic (chunking, embedding, storing, retrieval)

1. Model (Database Layer)

Component	Description
ChunkMetadata	Stores each chunk's ID, text, file name, chunking strategy, embedding model, etc.
FileMetadata	Stores file-level info like original filename, upload time, number of chunks
DB Used	Relational DB (PostgreSQL, MySQL, SQLite) or NoSQL (MongoDB, Firestore)

2. View (API Layer)

Endpoint	Description
POST /upload/	Upload .pdf or .txt, chunk & embed, store in Pinecone & DB
POST /ask/	Accepts a question, retrieves top-k chunks, responds with generated answer
Optional GET /compare/	Endpoint for comparing performance of strategies (optional UI/reporting layer)

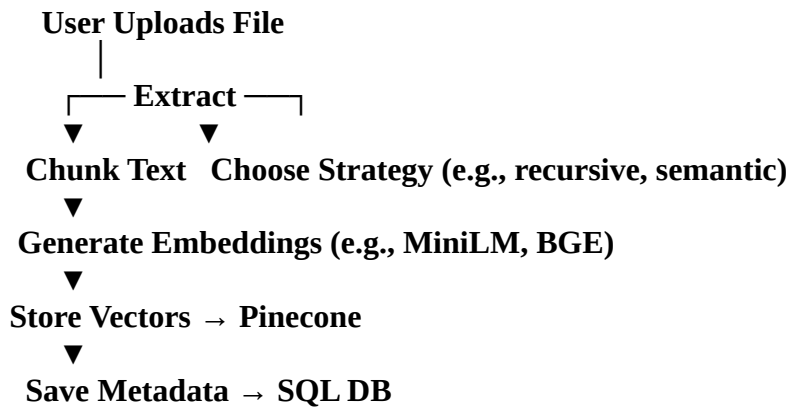
3. Controller (Business Logic Layer)

Component	Description
extract_text()	Handles PDF or TXT parsing
chunk_text()	Supports multiple chunking methods (recursive, semantic, custom)
generate_embeddings()	Uses Sentence Transformer or Hugging Face model
store_in_pinecone()	Stores embeddings as vectors in Pinecone
save_metadata()	Saves file/chunk info in relational DB

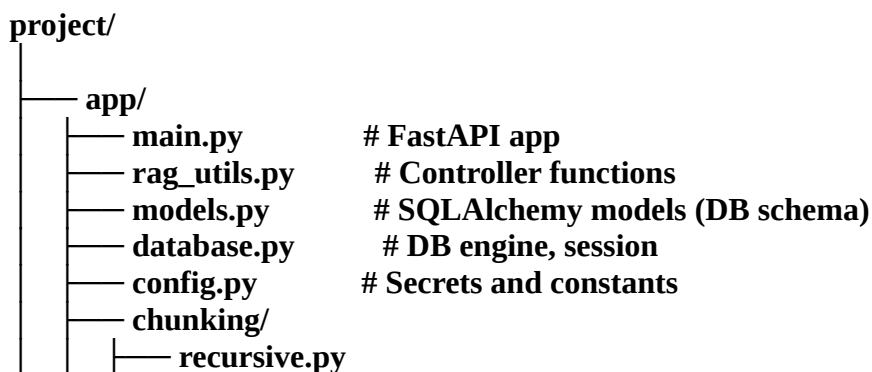
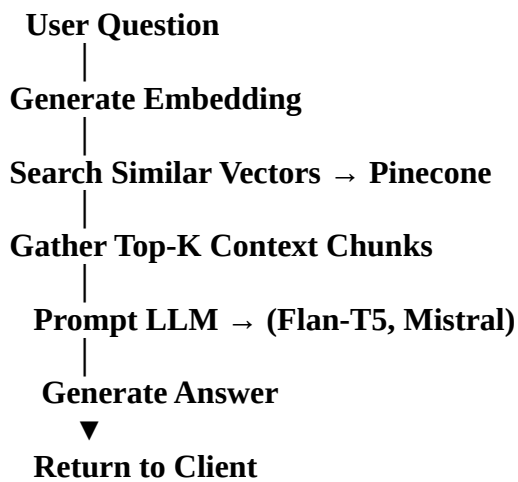
Component	Description
<code>retrieve_and_answer()</code>	Retrieves similar chunks and generates answer using LLM
<code>compare_strategies()</code>	Compares chunking/embedding methods based on accuracy/latency

Logic Breakdown

1. Upload Pipeline (/upload/)



2. Ask Pipeline (/ask/)



```
| | └─ semantic.py
| | └─ custom.py
|
```

```
└─ uploads/
└─ requirements.txt
└─ README.md
```

Code

```
sudo apt update
sudo apt install curl
```

```
curl -X 'POST' \
  'http://127.0.0.1:8000/ask/' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/x-www-form-urlencoded' \
  -d 'question=What happened after World War II?'
```