**Next-version upgrade options**

**1) Ingestion & chunking**

* **PDF support** (keep single-folder model): add pymupdf (a.k.a. fitz) for high-quality text extraction.
* **Smarter chunking**: sentence/paragraph-aware chunks with overlap (e.g., ~250–350 words, 15–20% overlap) to preserve context.
* **Source metadata**: include page numbers and per-doc IDs in each chunk.

**2) Retrieval quality**

* **Reranking**: retrieve top 20 via vectors, then **re-rank with the LLM** (cheap, short prompts) or a cross-encoder for better precision.
* **Multi-query expansion**: automatically rewrite the user’s question into 3–5 paraphrases, retrieve for each, and union results.
* **De-duplication**: remove near-duplicate chunks before prompting.

**3) Answer quality & safety**

* **Strict citation mode**: force inline citations like “[DocA p.3, chunk #7]” next to claims.
* **Quote-and-analyze**: include short verbatim quotes (bounded tokens) plus your analysis.
* **Grounding guardrails**: in strict mode, refuse claims not supported by excerpts.

**4) Output & reporting**

* **Sectioned reports**: auto-generate Executive Summary, Findings, Evidence, Recommendations.
* **Templates**: basic .docx styles (headings, list bullets, page breaks, TOC).
* **Appendix builder**: include full cited excerpts with page refs.

**5) Performance & ops**

* **Index persistence v2**: cache embeddings per-file (per-hash JSON) so adding a *new* file doesn’t re-embed everything.
* **Batching + rate limits**: embed in batches; graceful backoff on API 429s.
* **Config file**: config.yaml to centralize defaults (chunk size, model, top\_k, mode).
* **Logging**: add logging with INFO/DEBUG toggles; write timings (ingest, embed, search, LLM).
* **Tests**: minimal unit tests for loader, chunker, and manifest logic (pytest).

**6) UX & interfaces**

* **CLI polish**: subcommands (ingest, ask, report) via argparse/typer.
* **Simple UI**: lightweight Streamlit app for drag-and-drop docs and Q&A.
* **Saved Q&A runs**: write prompts/answers to JSONL for auditability.

**7) Models & costs**

* **Model switcher**: easy toggle between gpt-4o, gpt-4o-mini, or cheaper models for retrieval/rerank.
* **Cost control**: heuristics to cap prompt size (e.g., trim to N tokens, prioritize highest-scoring chunks).

**Suggested milestone plan (fastest impact → nice-to-have)**

1. **PDF support + smarter chunking (with overlap) + page metadata**
2. **Strict citation mode** (inline citations tied to chunks/pages)
3. **Reranking + multi-query expansion** for better retrieval precision
4. **Sectioned, styled reports** with an evidence appendix
5. **Per-file embedding cache + logging + config.yaml**
6. **CLI subcommands** (ingest/ask/report)