Agentic RAG System – System Design Document

1. System Overview

The system is a **local-first, agentic Retrieval-Augmented Generation (RAG) platform** designed to ingest heterogeneous documents, construct contextual knowledge representations, and answer user queries using a **multi-agent reasoning workflow**.

The architecture explicitly separates:

- **Data Processing (offline / pre-query)**
- **Agentic Reasoning & Retrieval (online / query-time)**

This separation is intentional to ensure scalability, debuggability, and extensibility.

---

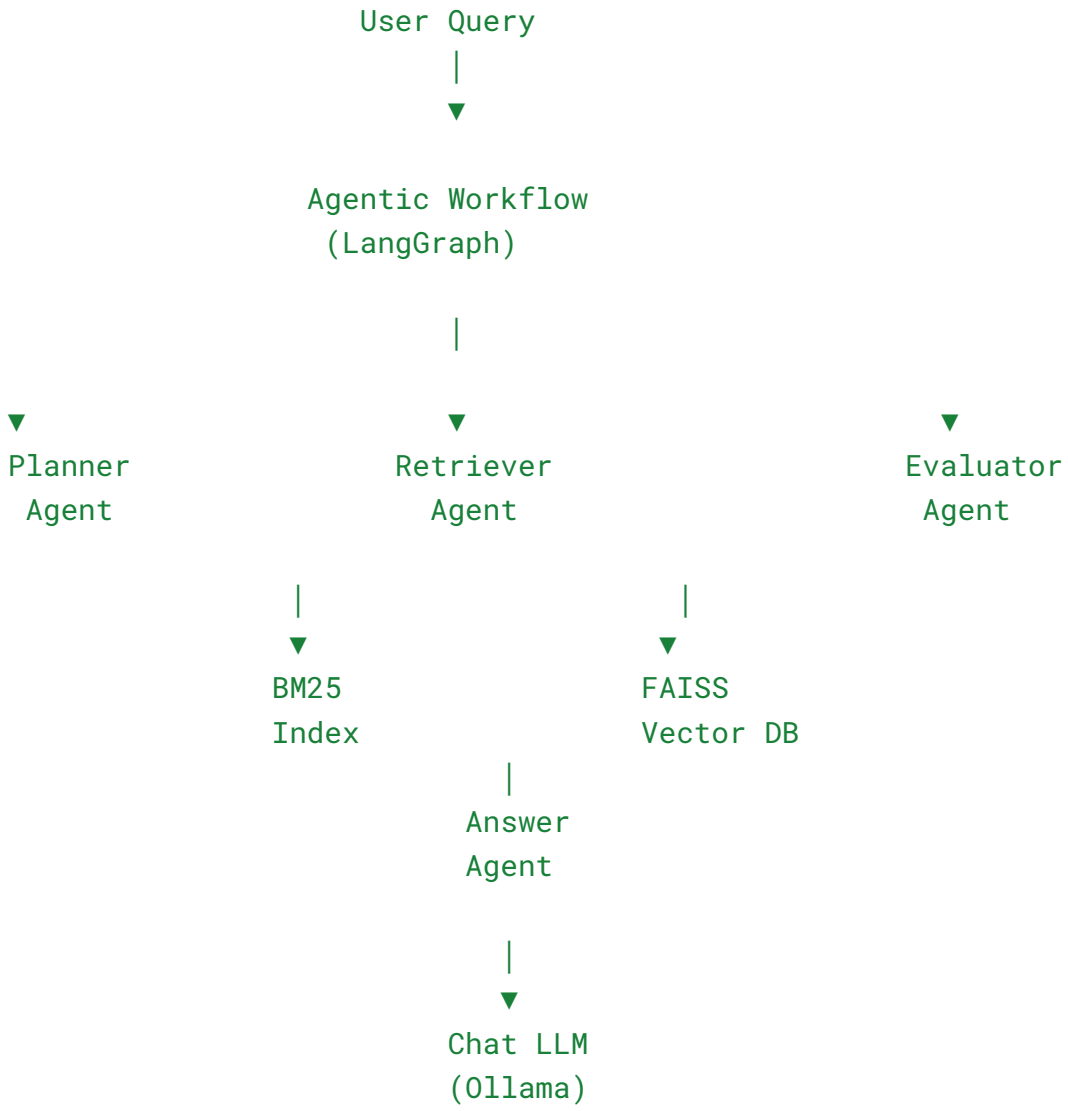
2. System Architecture

2.1 High-Level Architecture

Main Components

1. Document Ingestion & Processing Pipeline
2. Hybrid Retrieval Engine (BM25 + Vector Search)
3. Agentic Reasoning Layer (LangGraph)
4. LLM Abstraction Layer (Chat-based, local-first)
5. MCP (Model Context Protocol) Integration Layer

2.2 Architecture Diagram (Textual Representation)



---

3. Agentic Workflow Design

The system uses **LangGraph** to implement a deterministic, inspectable multi-agent workflow.

3.1 Agents and Responsibilities

1. Planner Agent

- Analyzes the user query
- Determines query type (factual / exploratory)

- Generates search variants
- Decides whether retrieval is needed

Why it exists:
Avoids naïve single-shot retrieval and improves recall.

---

2. Retriever Agent

- Executes **hybrid retrieval**
    - Sparse search: BM25
    - Dense search: FAISS (cosine similarity)
- Normalizes and combines scores
- Deduplicates results

Score fusion strategy

```
combined_score = 0.3 * BM25 + 0.7 * Vector
```

This bias toward semantic similarity improves robustness for poorly phrased queries.

---

3. Evaluator Agent

- Assesses retrieved context quality
- Detects missing information
- Decides next action:
    - Answer
    - Expand query
    - Retry retrieval

Key role:
Prevents hallucinated answers by blocking low-quality context.

---

4. Answerer Agent

- Synthesizes final answer
- Grounds responses in retrieved sources
- Explicitly acknowledges context limitations

3.2 Control Flow

1. Planner → Retriever
2. Retriever → Evaluator
3. Evaluator decides:
    - `answer`
    - `expand_query` (loop back to Retriever)
4. Answerer → END

Iteration count is capped to avoid infinite loops.

---

4. Context Construction Strategy

4.1 Document Chunking

- Sentence-aware chunking
- Fixed window with overlap
- Preserves semantic continuity

Rationale:
Token-based chunking breaks meaning; sentence-based chunking does not.

4.2 Context Assembly

- Top-K ranked chunks (default: 5)
- Ordered by combined relevance score
- Annotated with metadata (source, page, score)

Example context block:

```
[Source 1] Page 2 — AI_Engineer_Assignment.pdf
[Source 2] Page 3 — AI_Engineer_Assignment.pdf
```

4.3 Hallucination Mitigation

- Evaluator blocks answer if context score < threshold
- Answerer instructed to acknowledge missing info explicitly
- No answer is generated without retrieval evidence

---

5. Technology Choices & Rationale

5.1 Core Stack

Component	Technology	Rationale
Embeddings	sentence-transformers / MPNet	High semantic quality, open-source
Vector DB	FAISS	Fast, local, zero infra
Sparse Search	Custom BM25	Transparent scoring
Agent Graph	LangGraph	Deterministic multi-agent flow

LLM	Ollama (Chat-based)	Free, local, privacy-first
File Access	MCP	Decoupled context acquisition

5.2 Why Local-First?

- No vendor lock-in
- No data leakage
- Predictable latency
- Offline-friendly

---

6. Key Design Decisions

Decision 1: Hybrid Retrieval over Pure Vector

Why:
Vector search alone fails on keyword-heavy or structured queries.

Decision 2: Agentic Control Instead of Single Prompt

Why:

Complex questions require:

- Planning
- Evaluation
- Iteration

Single-shot prompting does none of this reliably.

Decision 3: Chat-Based LLM Interface

Why:
Completion-style APIs break when adding:

- Memory
- Tools
- System roles
- MCP

This system is future-proof by design.

Decision 4: Explicit Failure Handling

Why:
Silent failures create fake confidence. This system fails loudly and safely.

---

7. Limitations

Be honest in your PDF. Reviewers respect this.

7.1 Current Limitations

- No long-term conversational memory
- Confidence score is retrieval-based, not calibrated
- No re-ranking LLM stage