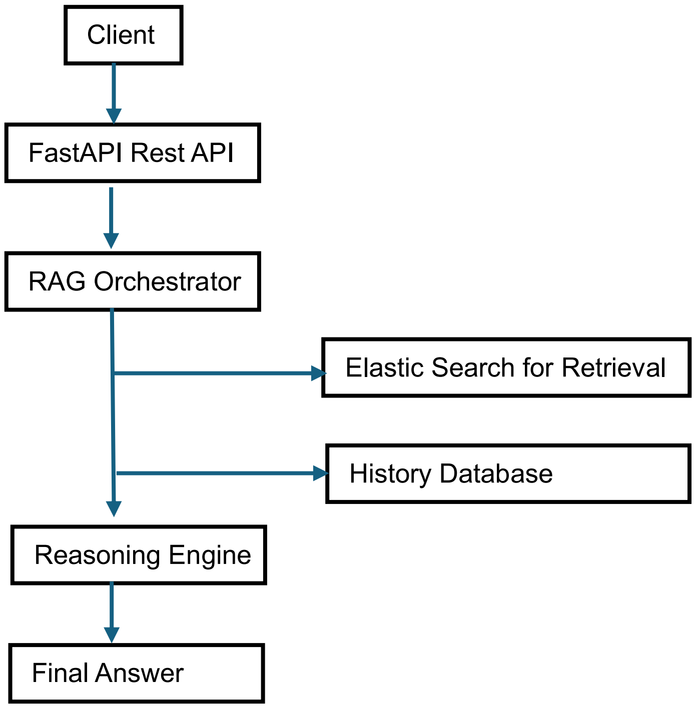
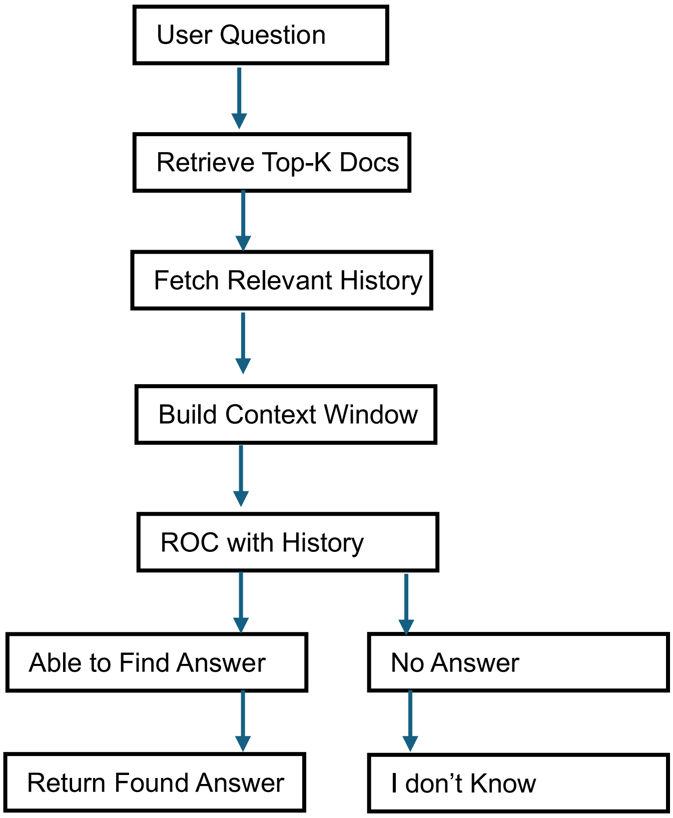
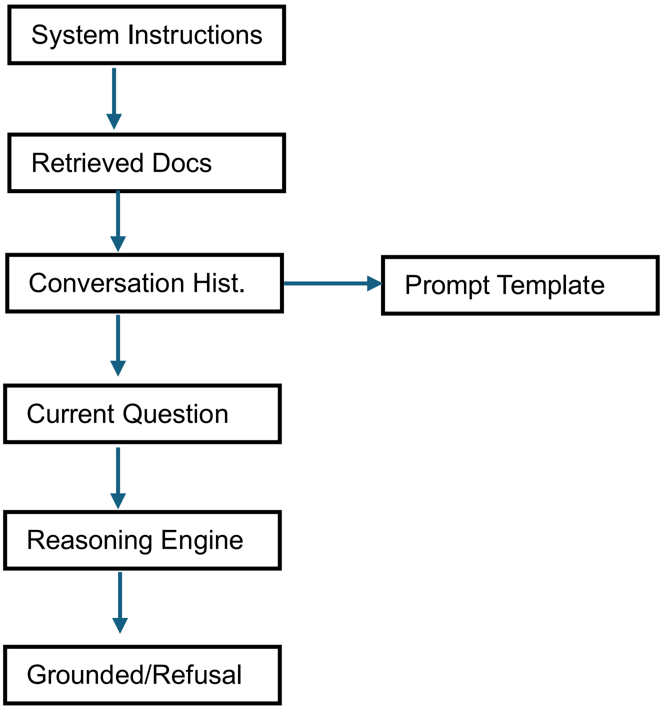
## System Architecture Diagram



## RAG & Agentic Flow



## Prompt Design with History Management



Conversation history is truncated to make sure the context window. FAQ’s and history turns are included to avoid prompt bloatings.

## History Management Strategy

The system maintains conversational continuity while preventing context explosion. History is treated as an auxiliary signal, not a source of truth.

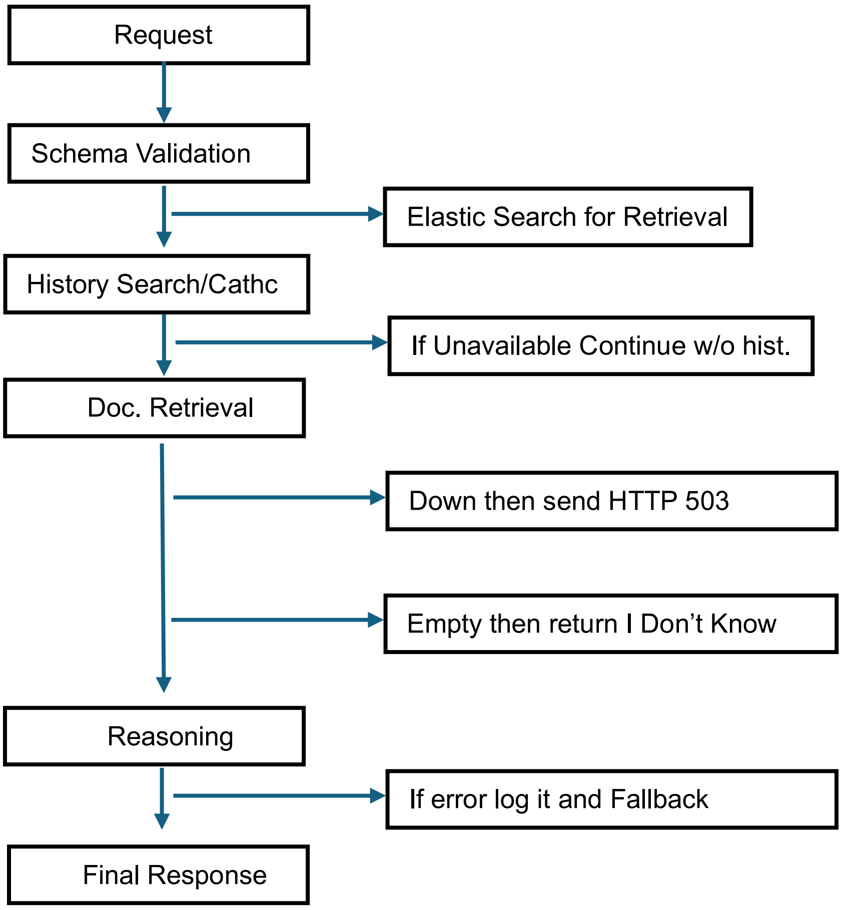
- Key design decisions:
- Store conversation turns externally
- Summarize older turns into compact memory
- Include only last N turns or relevant history
- Never allow history to override retrieved documents

In a conversational RAG system, history exists to preserve conversational continuity, not to act as a knowledge source. The system must remember what the user asked before and how the conversation evolved, while ensuring that all factual answers remain grounded in retrieved documents.

Sample Hist Data Model

```
{
  session_id,
  turn_index,
  user_question,
  system_answer,
  timestamp
}
```

## Failure Handling Strategy Diagram


Request
Schema Validation
Elastic Search for Retrieval
History Search/Cathc
If Unavailable Continue w/o hist.
Doc. Retrieval
Down then send HTTP 503
Empty then return I Don’t Know
Reasoning
If error log it and Fallback
Final Response

## Evaluation Process with Metrics

Multi and Single turn accuracy is considered

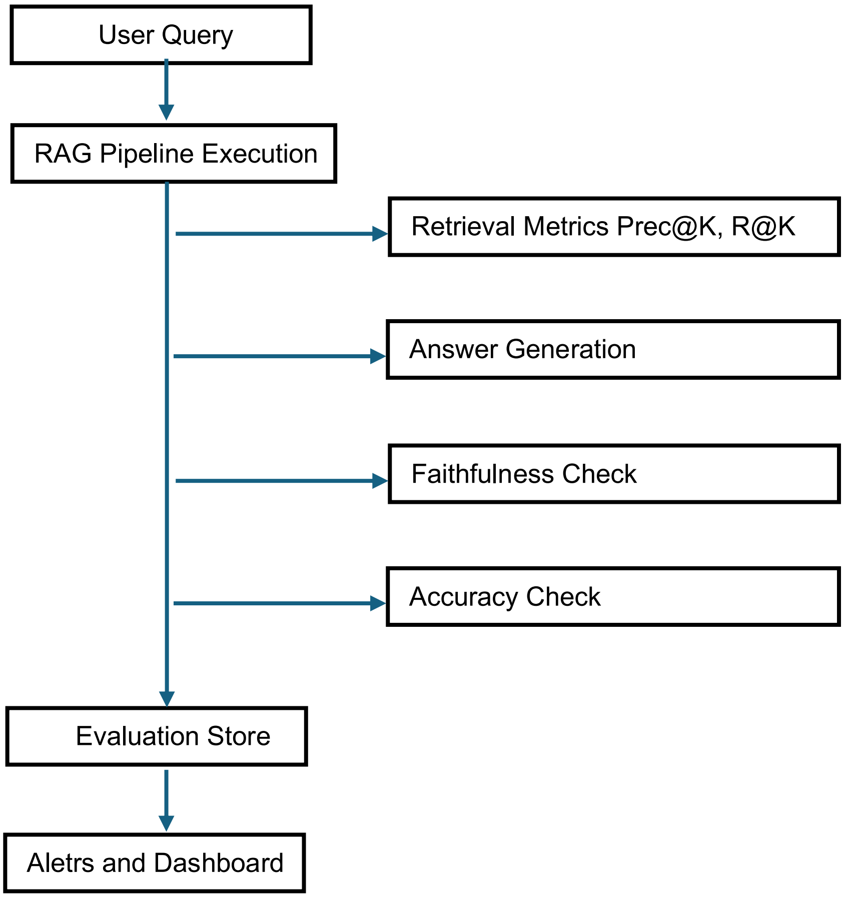
Mterics:

- Retrieval Precision@K and Recall@k
- Accuracy answers (Single and Multi)
- Context Fairness (Docs > hist.)
- Hallucination rate
- Impact due to latency of Hist. Inclusion

Evaluation is treated as a continuous lifecycle rather than a one-time offline activity. The strategy covers offline benchmarking, online monitoring, and regression detection across retrieval, reasoning, and multi-turn behavior.

Evaluation signals are logged asynchronously to avoid impacting user-facing latency. Alerts are triggered on metric degradation or hallucination spikes.

## Model Evaluation Workflow Diagram



User Query

RAG Pipeline Execution

Retrieval Metrics Prec@K, R@K

Answer Generation

Faithfulness Check

Accuracy Check

Evaluation Store

Aletrs and Dashboard

Evaluation gates are integrated into CI/CD pipelines:

- Model or prompt changes must pass offline benchmarks
- Regression tests ensure no drop in faithfulness or retrieval quality
- Rollbacks are triggered on evaluation failures