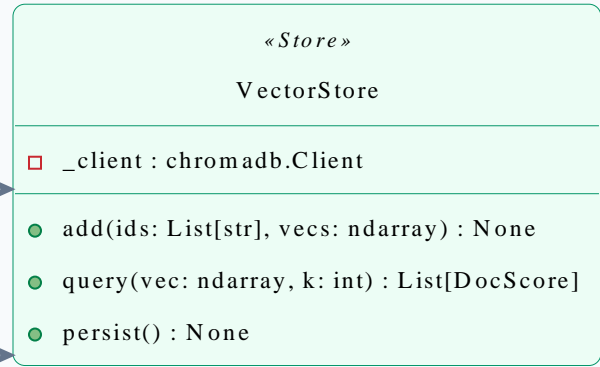
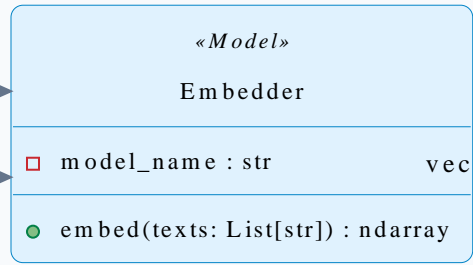
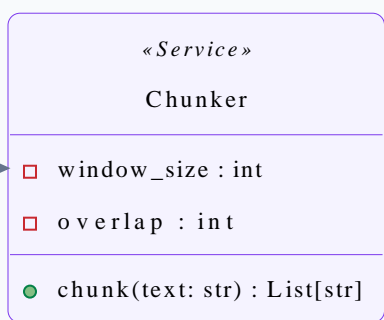
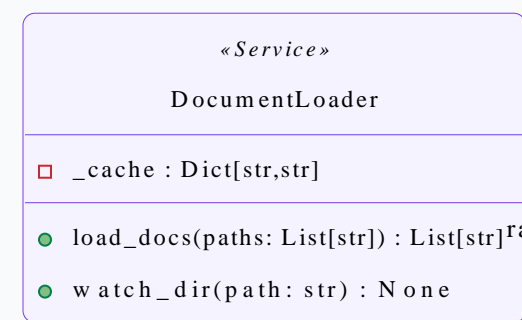


RAG - TinyLlama (Full Stack Overview)

1) Ingestion / Memory



sends raw text

chunks

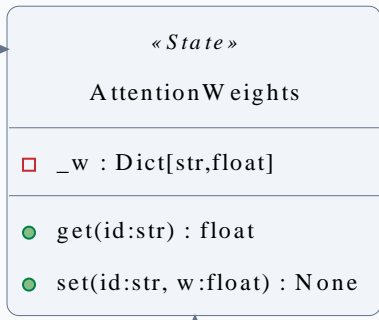
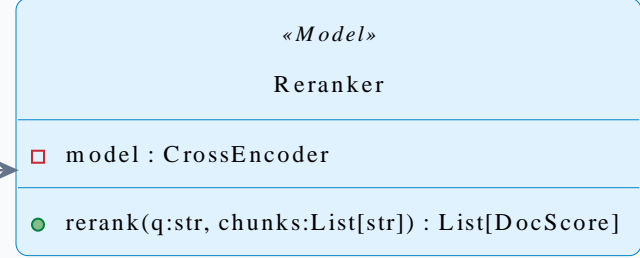
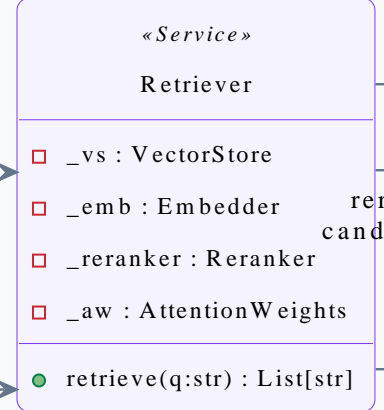
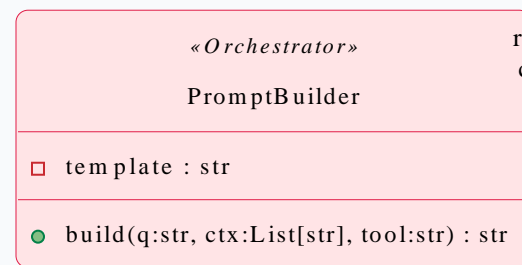
vecs + ids

query embedding

top-k candidates

2) Retrieval & Ranking

4) Prompt Orchestration



receives context

rerank candidates

applies weights

retrieve context

receives TOOL_OUTPUT

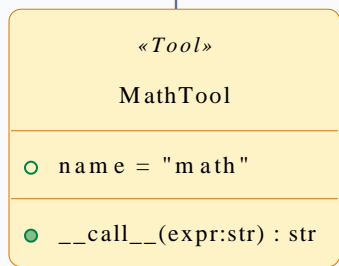
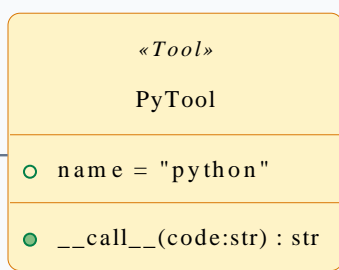
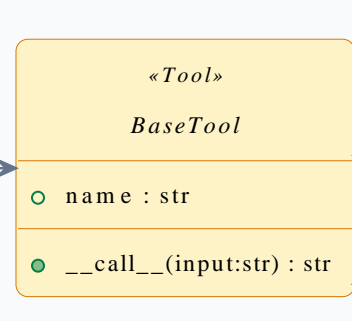
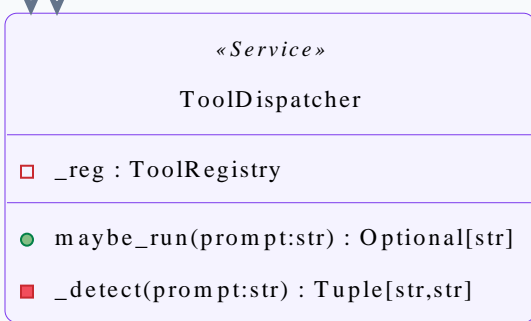
update weights

build prompt

generate answer

optional local tools

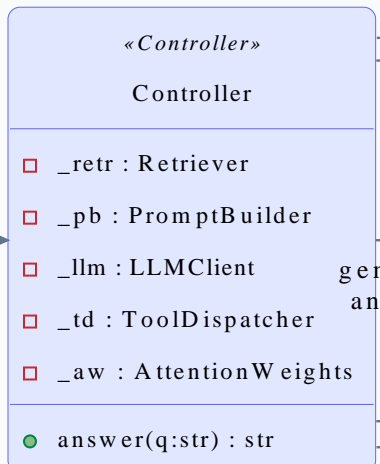
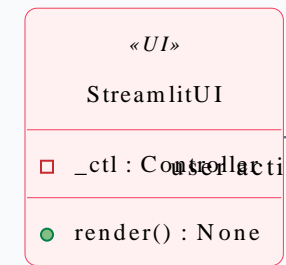
3) Local Tooling



lookup

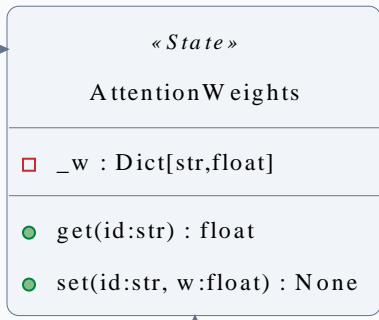
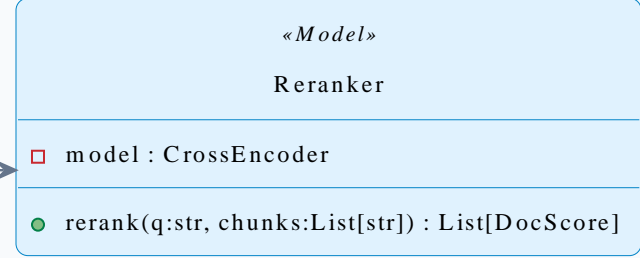
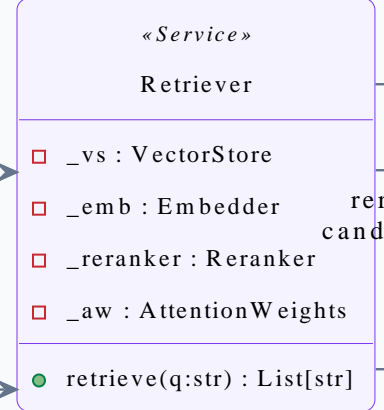
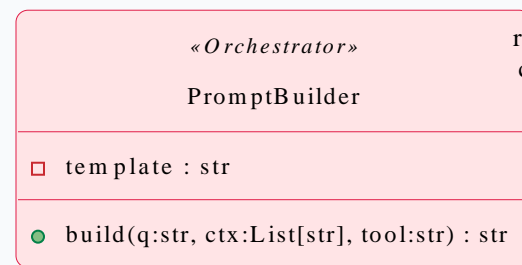
manages

5) Application Layer



receives user actions

4) Prompt Orchestration



receives context

rerank candidates

applies weights

retrieve context

receives TOOL_OUTPUT

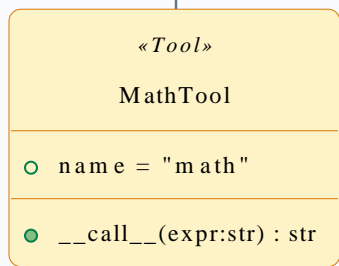
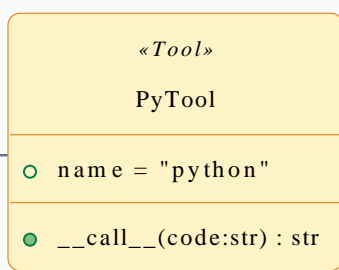
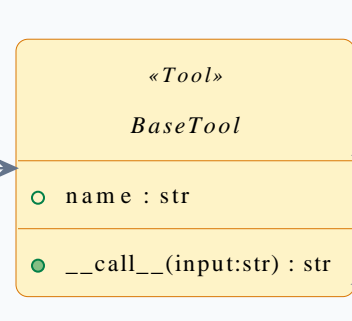
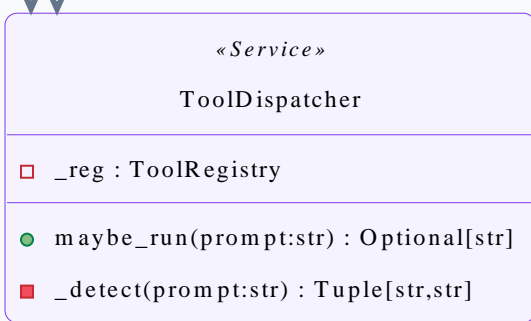
update weights

build prompt

generate answer

optional local tools

3) Local Tooling



lookup

manages

Legend (Stereotypes)

- «Service» Processing/Workers
- «Model» ML Models
- «Store» Persistence/Stores
- «Tool» Executable Tools
- «Registry» Tool Management
- «Orchestrator» Prompt/Flow
- «State» Weights/State
- «Client» External LLM Client
- «Controller» App Controller
- «UI» User Interface