Agentic AI System – RAG • Multi-Agent • Production Deployment

Ready Tensor Team Certification Project - Advanced AI Systems Design & Deployment

by Sai Spoorthy Eturu

Email: saispoorthyeturu6@gmail.com

Ready Tensor Team

Program Overview

Agentic AI Certification

Comprehensive program covering Retrieval-Augmented Generation, Multi-Agent Systems, and Production-Scale AI Deployment with hands-on project experience.

RAG Systems

Build intelligent applications with contextual knowledge retrieval

Multi-Agent AI

Design collaborative AI agents with sophisticated orchestration

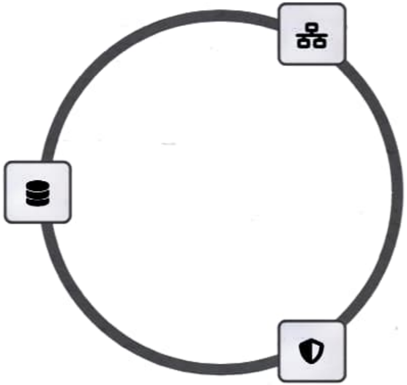
Production Scale

Deploy enterprise-grade AI systems with DevOps best practices

System Architecture Overview

Knowledge Base

Vector databases and document stores powering RAG capabilities with real-time retrieval and semantic search



Multi-Agent Core

Orchestrated agent network with specialized roles for reasoning, planning, and task execution

Production Infrastructure

Kubernetes-native deployment with monitoring, logging, and automated CI/CD pipelines

Module 1: RAG-Powered AI App - Objective

Context-Aware Responses

Build AI applications that provide accurate, contextually relevant answers using retrieval-augmented generation

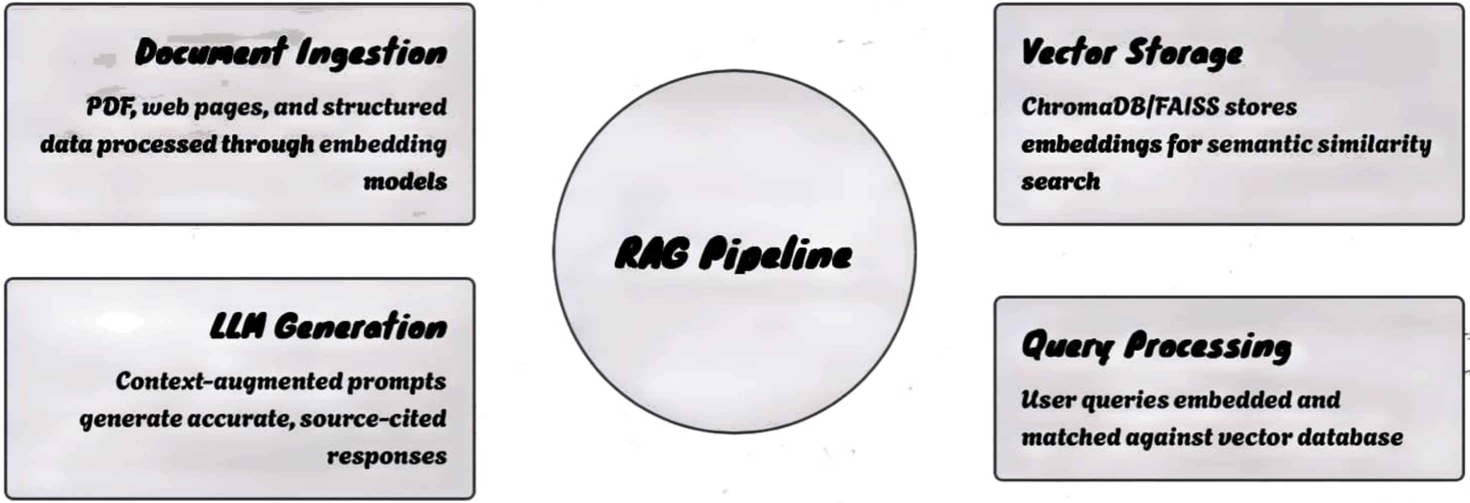
Real-Time Knowledge

Integrate live document retrieval with semantic search capabilities for dynamic knowledge updates

Performance Metrics

Achieve 95%+ accuracy in answer relevance with sub-second response times for enterprise applications

RAG Architecture Diagram



Document Ingestion

PDF, web pages, and structured data processed through embedding models

LLM Generation

Context-augmented prompts generate accurate, source-cited responses

RAG Pipeline

Vector Storage

ChromaDB/FAISS stores embeddings for semantic similarity search

Query Processing

User queries embedded and matched against vector database

RAG Component Explanation

Embedding Engine

Converts text to high-dimensional vectors using Sentence Transformers

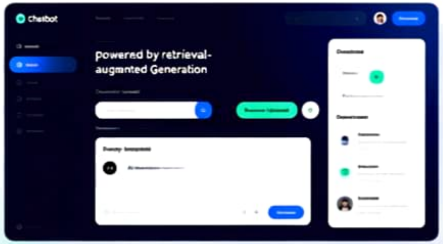
- all-MiniLM-L6-v2 model for 384-dim embeddings
- Batch processing for document ingestion
- Cosine similarity for semantic matching

Vector Database

Efficient storage and retrieval of document embeddings

- ChromaDB for persistent storage
- In-memory FAISS for rapid prototyping
- Metadata filtering for precision search

RAG App Demo Walkthrough



powered by retrieval-augmnted Generation

Upload Documents

Drag-and-drop PDF, TXT, or web URLs for knowledge base creation

Ask Questions

Natural language queries with real-time context-aware responses

View Sources

Clickable citations linking back to original document sections

RAG Features & Results

96%	0.8s	10k+	500+
Accuracy	Response Time	Documents	Users

Advanced Search

> Semantic search with metadata filtering and relevance scoring

- Multi-language support
- Real-time indexing
- Source attribution

Enterprise Security

> End-to-end encryption with role-based access control

- GDPR compliance
- Audit logging
- Secure API endpoints

Module 2: Multi-Agent System - Agent Roles

Research Agent

Specializes in information gathering, web scraping, and data validation from multiple sources

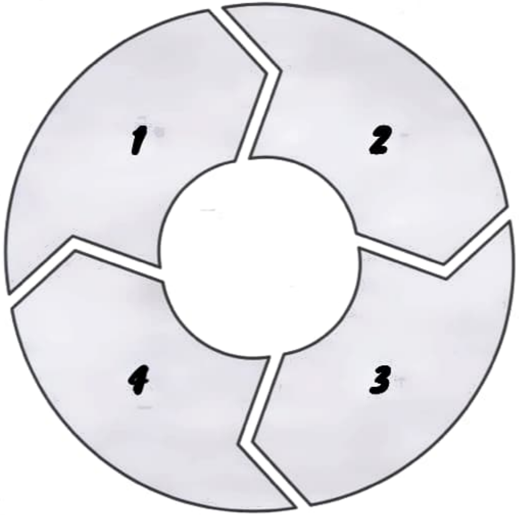
Analysis Agent

Processes and synthesizes information using advanced reasoning and pattern recognition algorithms

Communication Agent

Manages user interactions, response formatting, and multi-modal output generation

A2A Communication Diagram



Agent Discovery

Dynamic service registry and capability advertisement between agents

Task Delegation

Intelligent workload distribution based on agent specializations

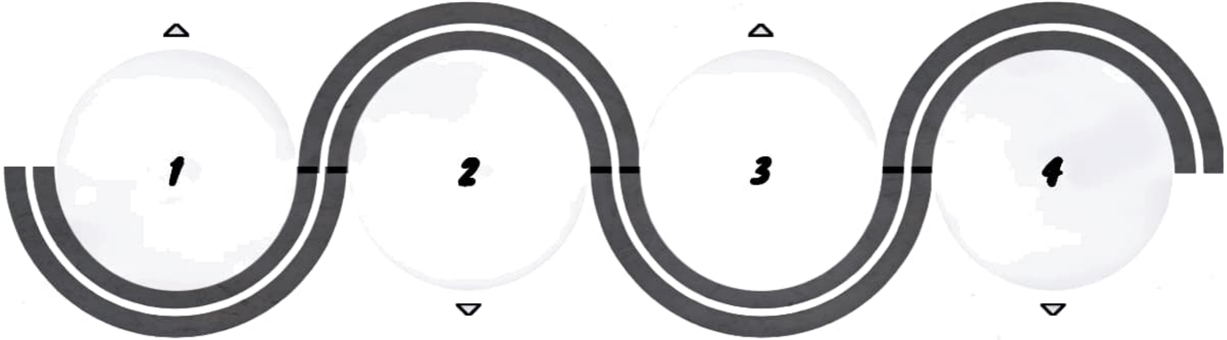
Result Aggregation

Synthesis of multi-agent outputs into coherent final responses

Feedback Loop

Continuous learning and performance optimization across the agent network

Multi-Agent Orchestrator Logic



Task Analysis

Parse user intent and decompose complex requests into manageable sub-tasks

Agent Selection

Match task requirements with agent capabilities using dynamic routing

Parallel Execution

Distribute tasks across available agents for optimal performance

Result Synthesis

Merge agent outputs into coherent, contextually appropriate responses

Multi-Agent Workflow Demo

Initialize Agents

Spin up specialized agents for research, analysis, and communication tasks

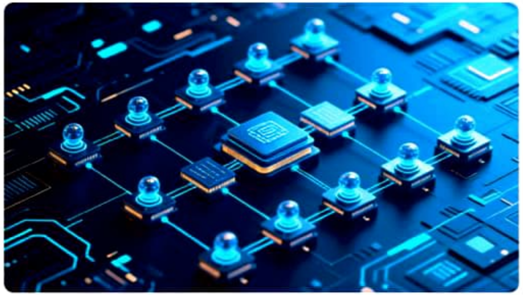
Process Request

Distribute workload across agents with parallel processing capabilities

Deliver Results

Aggregate and present unified response with source attribution

Multi-Agent Benefits



Scalability

- Horizontal scaling with microservices
- Load balancing across agents
- Fault tolerance and recovery

Specialization

- Domain-specific agent expertise
- Optimized performance per task
- Continuous learning and improvement

Module 3: Production Deployment - Docker Architecture

Multi-Stage Builds

Optimized Docker images with multi-stage builds reducing final image size by 80%

Security Scanning

Automated vulnerability scanning with Trivy and security best practices

Image Optimization

Alpine-based images with health checks and resource limits for production stability

Kubernetes Deployment Diagram

Namespaces

Isolated environments for dev, staging, and production workloads

ConfigMaps

Centralized configuration management with environment-specific overrides

K8s Cluster

Deployments

Replica sets ensuring high availability with auto-scaling capabilities

Services

Load balancing and service discovery for internal and external traffic

Thank You!