# Agentic Rag Based Study Assistant For Devops And Mlops Concept

**K Mutheeswari[1], R.B.Vishnu[2], M.K. Vishwaraj[3] ,M.Srisivaraman[4]**
[1, 2, 3, 4] Dept of CSE
[1, 2, 3, 4]K.L.N.College of Engineering, Sivagangai,India

***Abstract-*** *DevOps and MLOps have become essential skills, but students often struggle to navigate large amounts of unstructured learning material such as documentation, blogs, and tutorials. This project proposes an Agentic RAG-based Study Assistant that helps learners understand DevOps and MLOps concepts through domain-grounded question answering and personalized study support. The system first builds a knowledge base by collecting and chunking trusted DevOps/MLOps resources, then converts them into embeddings and stores them in a vector database using a Retrieval-Augmented Generation (RAG) pipeline. On top of this, multiple AI agents are orchestrated: a retrieval agent that selects relevant content, a planner agent that creates topic-wise study plans, and a tutor agent that provides explanations, summaries, and practice questions. By constraining responses to retrieved, curated material, the assistant aims to reduce hallucinations and improve factual accuracy for educational us.*

## I. INTRODUCTION

The rapid growth of DevOps and MLOps has transformed the way modern software systems are developed, deployed, and maintained. However, students and beginners often face significant challenges in learning these domains due to the vast amount of unstructured information available across documentation, blogs, and tutorials. This scattered learning environment makes it difficult to identify reliable resources, understand core concepts, and follow a structured learning path.

To address these challenges, this project proposes an Agentic Retrieval-Augmented Generation (RAG)-based Study Assistant designed to support learners with domain-specific guidance in DevOps and MLOps. The system builds a curated knowledge base from trusted sources, converts the content into embeddings, and stores it in a vector database for efficient retrieval. By leveraging a multi-agent architectureincluding retrieval, planner, and tutor agentsthe assistant provides accurate, context-aware answers, personalized study plans, and interactive learning support. This approach not only enhances understanding but also reduces misinformation by grounding responses in verified content, thereby improving the overall learning experience.

## II. LITERATURE SURVEY

The literature on Retrieval-Augmented Generation (RAG) and intelligent learning systems highlights the growing need for structured and reliable knowledge access in complex domains like DevOps and MLOps. Traditional learning approaches rely heavily on static resources such as documentation and tutorials, which often lack personalization and structured guidance. With the advancement of Natural Language Processing (NLP) and Large Language Models (LLMs), several studies have explored automated question answering and knowledge retrieval systems to improve learning efficiency.

Recent research emphasizes the importance of RAG frameworks, which combine information retrieval with generative models to enhance factual accuracy and contextual relevance. Studies show that RAG systems address the major limitation of LLMshallucinationby grounding responses in external knowledge sources. The core components of RAG include retrieval, augmentation, and generation, which work together to produce more reliable outputs compared to standalone language models . Furthermore, systematic reviews indicate that RAG has significantly improved performance in knowledge-intensive tasks such as question answering, summarization, and educational assistance.

## III. EXISTING METHODOLOGY

The proposed system follows a structured methodology to design and implement an Agentic RAG-based Study Assistant for DevOps and MLOps learning. Initially, relevant and trusted learning resources such as documentation, blogs, and tutorials are collected and pre-processed. The collected data is then segmented into smaller chunks to ensure efficient retrieval and better contextual understanding. These chunks are converted into vector representations using embedding models and stored in a vector database to enable semantic search.

The system utilizes a Retrieval-Augmented Generation (RAG) pipeline where a retrieval agent fetches the most relevant content based on user queries. A planner agent analyses the learner's needs and generates a structured study plan, while a tutor agent provides explanations, summaries, and practice questions. These agents work collaboratively to deliver accurate and context-aware responses by grounding outputs in retrieved knowledge. This multi-agent approach ensures personalized learning, reduces hallucinations, and enhances the overall effectiveness of the study assistant.

## IV. PROPOSED METHODOLOGY

The proposed system introduces an Agentic Retrieval-Augmented Generation (RAG)-based approach to overcome the limitations of existing methods. It begins by collecting and preprocessing trusted DevOps and MLOps resources, which are then converted into embeddings and stored in a vector database for semantic retrieval. When a user submits a query, the system retrieves relevant information using similarity search and passes it through a multi-agent architecture. The retrieval agent selects the most relevant content, the planner agent generates personalized study plans, and the tutor agent provides explanations, summaries, and practice questions. The RAG pipeline ensures that responses are generated based only on retrieved knowledge, reducing hallucinations and improving accuracy. This approach provides structured, personalized, and context-aware learning support, making it more effective than traditional methods.

The system continuously analyzes user interactions, queries, and learning progress to refine recommendations and improve response quality over time. By leveraging feedback mechanisms and context-aware retrieval, the assistant can adjust the difficulty level of explanations, suggest relevant topics, and provide targeted practice questions. Additionally, the modular multi-agent design allows seamless integration of new data sources, advanced embedding models, or improved agents without affecting the overall system. This ensures that the study assistant remains scalable, up-to-date, and capable of delivering a more interactive and personalized learning experience for users in evolving domains like DevOps and MLops

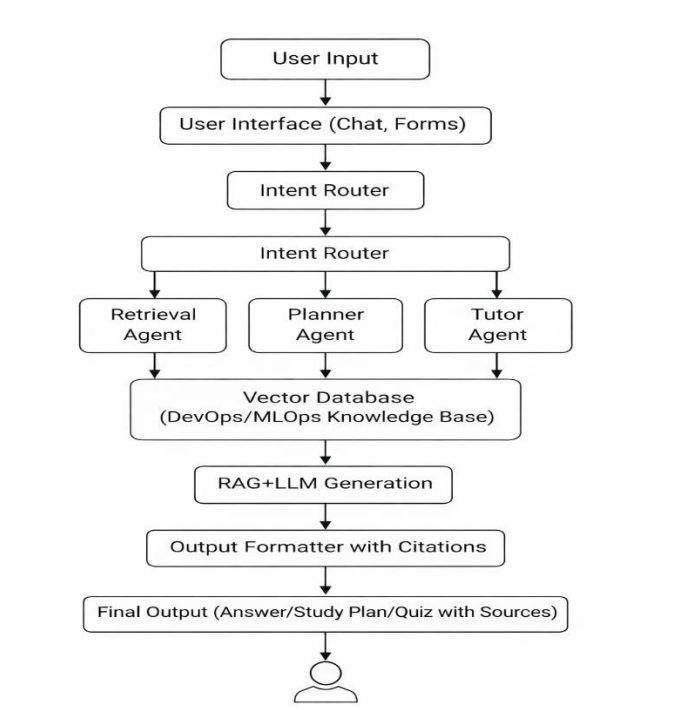


**Figure 1 : Dataflow diagram of our proposed module**

## V. SYSTEM PROCESS FLOW

The system process flow begins with the user providing input in the form of a query or request through the user interface, which includes chat or form-based interaction. This input is first handled by an intent router, which identifies the purpose of the user's request, such as whether it is for explanation, study planning, or practice questions. The processed query is then passed to a central routing layer that distributes the task among specialized agents:

**1. User Query Input**

The process is initiated when the user submits a query through the system interface, which may include requests for concept explanations, study plans, or practice assessments in the domains of DevOps and MLOps

**2. Query Processing and Intent Identification**

The input query is pre-processed and analysed using an intent recognition module to determine the user's objective. The query is then transformed into a structured representation suitable for further processing within the system.

**3. Semantic Retrieval using Embeddings**

The processed query is converted into vector embeddings using a pre-trained embedding model. These embeddings are matched against a vector database containing domain-specific knowledge. A similarity search mechanism retrieves the most relevant content based on semantic relevance.

4.**Multi-Agent Orchestration and Knowledge Processing**

The retrieved information is processed through a multi-agent architecture. The retrieval agent selects and refines relevant context, the planner agent generates structured and personalized learning paths, and the tutor agent produces detailed explanations, summaries, and assessment content. This collaborative mechanism ensures efficient handling of diverse user requirements.

**5.RAG-Based Response Generation and Output Presentation**

The refined content is passed to the Retrieval-Augmented Generation (RAG) module, where a large language model generates context-aware and accurate responses grounded in retrieved knowledge. The output is then formatted with appropriate structure and references, and presented to the user through the interface, ensuring clarity, reliability, and enhanced learning experience.

## VI. RESULT AND DISCUSSIONS

The results of the proposed Agentic RAG-Based Study Assistant demonstrate its effectiveness in providing accurate and context-aware learning support for DevOps and MLOps topics. The system successfully retrieves relevant information from the knowledge base and generates meaningful responses, including explanations, summaries, and practice questions. By grounding responses in retrieved content, the assistant significantly reduces hallucinations and ensures that the information provided is reliable and aligned with trusted sources. Additionally, the multi-agent approach enhances the learning experience by offering structured study plans and personalized guidance based on user queries.

In discussion, the system highlights the advantages of combining Retrieval-Augmented Generation with agent-based architecture for educational applications. The modular design improves scalability and allows easy integration of additional features or learning domains in the future. However, the performance of the system depends on the quality of the collected data and the effectiveness of the embedding model. Continuous updates to the knowledge base and optimization of retrieval strategies can further improve accuracy and relevance. Overall, the proposed system provides an efficient and intelligent solution for navigating complex and unstructured learning materials in DevOps and MLOps

## VII. CONCLUSION

In conclusion, the proposed Agentic RAG-Based Study Assistant provides an effective solution to the challenges faced by learners in understanding DevOps and MLOps concepts from unstructured resources. By integrating a Retrieval-Augmented Generation pipeline with a multi-agent architecture, the system delivers accurate, context-aware, and personalized learning support. It not only improves knowledge accessibility but also reduces misinformation by grounding responses in trusted content.

Furthermore, the modular design of the system ensures scalability and flexibility for future enhancements, such as expanding to other domains or improving agent capabilities. Overall, this approach enhances the learning experience by offering structured guidance, interactive support, and reliable information, making it a valuable tool for students and self-learners in technical fields.

## REFERENCES

[1] Lewis, P., Perez, E., Piktus, A., et al. (2020). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. Advances in Neural Information Processing Systems (NeurIPS).

[2] Reimers, N., &Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. Proceedings of EMNLP

[3] Brown, T. B., Mann, B., Ryder, N., et al. (2020). *Language Models are Few-Shot Learners*. Advances in Neural Information Processing Systems (NeurIPS).

[4] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). *Attention Is All You Need*. Advances in Neural Information Processing Systems (NeurIPS).

[5] Merkel, D. (2014). *Docker: Lightweight Linux Containers for Consistent Development and Deployment*. Linux Journal.

[6] Turnbull, J. (2014). *The Docker Book: Containerization is the New Virtualization*.