# Software Documentation for RAG Chatbot using LangChain, Mistral 7B, and Pinecone

## Table of Contents

1. Introduction

2. System Architecture

3. Technology Stack

4. Features and Functionalities

5. Data Ingestion and Preprocessing

6. System Design

7. Conclusion

## 1. ****Introduction****

### 1.1 Purpose of the Document

This document serves as a comprehensive guide to understanding, developing, and deploying a Retrieval-Augmented Generation (RAG) chatbot using the LangChain framework, the Mistral 7B language model, and Pinecone as the vector store for efficient indexing and retrieval. The document is intended for developers, researchers, and project stakeholders involved in the project.

### 1.2 Project Overview

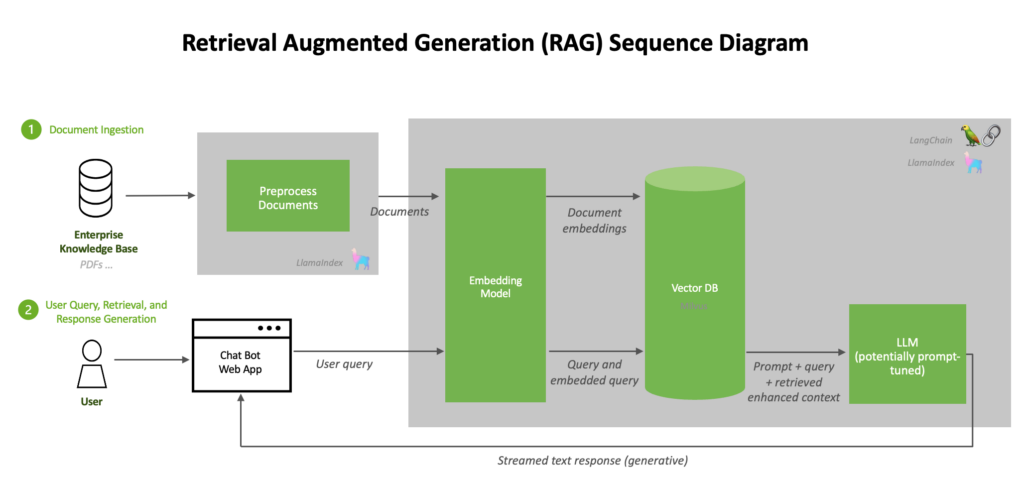
The RAG chatbot is designed to answer queries related to a predefined set of data—specifically, the academic courses and curriculum of the Department of Computer Engineering. The chatbot leverages the capabilities of the Mistral 7B model for generating responses and Pinecone for storing and retrieving indexed data, providing accurate and context-aware answers.

### 1.3 Scope

The scope of this document includes an explanation of the system architecture, technology stack, implementation details, deployment, and testing processes involved in building the chatbot.

## 2. ****System Architecture****

### 2.1 Architectural Diagram



Source: Nvidia

### 2.2 Components Overview

* **LangChain**: A framework for connecting language models with various sources of data to build a RAG-based system.
* **Mistral 7B**: A lightweight and efficient language model for generating natural language responses based on retrieved context.
* **Pinecone**: A vector database service used for indexing, storing, and retrieving embeddings of course-related data.

## 3. ****Technology Stack****

### 3.1 LangChain

LangChain is used to create and manage the pipeline for connecting the language model to the indexed data stored in Pinecone.

### 3.2 Mistral 7B

Mistral 7B is a 7-billion parameter model known for its efficiency and performance. It serves as the core model for understanding queries and generating appropriate responses based on the context retrieved from the Pinecone index.

### 3.3 Pinecone

Pinecone is a vector database that efficiently stores and manages vector embeddings for fast and accurate retrieval during query processing.

## 4. ****Features and Functionalities****

### 4.1 Document Indexing

* Extracts key information from course content.
* Converts text data into vector embeddings using the Mistral 7B model.
* Stores embeddings in Pinecone for efficient retrieval.

### 4.2 Query Processing

* Takes user input and converts it into a vector representation.
* Searches the Pinecone index for the most relevant document embeddings.
* Retrieves and ranks results based on similarity to the input query.

### 4.3 Response Generation

* Uses the retrieved context to generate a coherent and contextually accurate response.
* Provides additional information or references to related courses, if applicable.

## 5. ****Data Ingestion and Preprocessing****

### 5.1 Document Structure

The data includes course descriptions, course codes, unit values, and lecture hours, stored in a structured JSON format.

### 5.2 Data Preprocessing Techniques

* Tokenization and cleaning of text data.
* Conversion to lower case and removal of stop words.
* Splitting long documents into smaller chunks for better indexing and retrieval.

## 6. ****System Design****

### 6.1 Data Flow Design

1. **Data Ingestion**: Raw course data is preprocessed and converted into vector embeddings.
2. **Indexing**: The embeddings are stored in Pinecone, where each document is associated with its course code and content.
3. **Query Processing**: User queries are embedded and matched with the indexed vectors in Pinecone.
4. **Response Generation**: Relevant course information is passed to Mistral 7B for generating the final response.

### 6.2 Indexing and Retrieval Process

* **Indexing**: Each course content is represented as an embedding using Mistral 7B and stored in Pinecone.
* **Retrieval**: User queries are transformed into embeddings and matched against the indexed content using cosine similarity or other distance metrics.

## 7. ****Conclusion****

The RAG chatbot provides an interactive solution for querying academic course information using LangChain, Mistral 7B, and Pinecone. It effectively combines retrieval and generation capabilities to deliver accurate and context-aware responses to user queries. This documentation outlines the implementation and deployment process, providing a robust foundation for further development and optimization.