



Generative AI and ML

Capstone Project
edureka! a VERANDA
ENTERPRISE

Overview

The goal of this capstone project is to develop a Generative AI-powered application that enables users to query enterprise documents using autonomous AI agents. The system uses Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), and Agentic AI frameworks to retrieve relevant information, reason over it, and generate accurate, context-aware responses.

Project Description

This project aims to build an AI agent-based knowledge and decision support system. The application allows users to upload documents in multiple formats (PDF, TXT, CSV, Excel) and ask natural language questions. The system retrieves relevant content using a vector database and generates grounded responses using an LLM.

AI agents are used to plan the task, retrieve information, reason over the retrieved context, and validate the final output, demonstrating a full-fledged Generative AI and Agentic AI workflow.

Tasks for Learners

- 1. Set up the project foundation** – Initialize the project repository, environment configuration, and basic application structure for the Generative AI system.
- 2. Design the user interaction layer** – Create a simple interface or API that allows users to upload documents and ask natural language questions.
- 3. Implement document ingestion** – Enable uploading and processing of enterprise documents in multiple formats such as PDF, TXT, CSV, or Excel.
- 4. Prepare data for semantic search** – Convert processed document content into chunks suitable for embedding and retrieval.
- 5. Build a vector-based knowledge store** – Generate embeddings and store them in a vector database to support semantic similarity search.
- 6. Implement intelligent document retrieval** – Retrieve the most relevant document content based on user queries using similarity search.
- 7. Develop a Retrieval-Augmented Generation pipeline** – Combine retrieved document context with an LLM to generate accurate, grounded responses.
- 8. Implement agent-based reasoning** – Create one or more AI agents that plan, retrieve, reason, and generate responses using available tools.
- 9. Add reliability and safety controls** – Handle errors, validate inputs, and apply guardrails to reduce hallucinations and unsafe outputs.
- 10. Deploy and document the solution** – Deploy the application and provide documentation explaining the architecture, workflow, and limitations.

Submission Guidelines:

- Submit the complete source code and a documentation file in a Zip format.
- Documentation should explain system setup, architecture, agent roles, and deployment steps, along with limitations and challenges faced during development.

