

# Assignment Problem Statement: AI Agent for QA on Shared Documents

## Overview

Develop an AI-powered agent capable of answering questions based on shared documents. This project will test your ability to integrate natural language processing (NLP) techniques with a user-friendly interface, enabling users to interact with the system efficiently.

## Objectives

- **Develop an AI Agent:** Build a robust AI system that can parse and understand shared documents, then provide accurate answers to user queries.
- **Implement a Simple UI:** Create a user interface using frameworks such as Streamlit or Chainlit that allows users to upload documents and interact with the AI agent.
- **Document the Code:** Provide clear documentation and a comprehensive code walkthrough that explains your design decisions, architecture, and implementation details.

## Requirements

### 1. Codebase

- **Core Functionality:**
  - Implement the logic for reading, processing, and extracting information from shared documents.
  - Integrate a question-answering mechanism that leverages AI/NLP models to generate accurate responses.
- **User Interface:**
  - Build a simple, intuitive UI using Streamlit, Chainlit, or a similar framework.
  - Ensure the UI allows users to:
    - Upload documents.
    - Input questions.
    - View answers generated by the AI agent.
- **Code Quality:**
  - Follow best coding practices.
  - Ensure the code is modular, well-organized, and easy to understand.

## 2. Documentation & Code Walkthrough

- **Comprehensive Documentation:**
  - Provide clear instructions on how to set up and run the application.
  - Include a description of the overall system architecture and the individual components.
- **Code Walkthrough:**
  - Prepare a step-by-step guide that explains the main parts of the codebase.
  - Highlight key functions, algorithms, and integrations with the chosen UI framework.
  - Discuss any challenges encountered and how they were addressed.

## Deliverables

- **Source Code:**
  - A fully functional codebase hosted on a public repository (e.g., GitHub) with clear commit history.
- **Documentation:**
  - A README file containing setup instructions, usage guidelines, and an overview of the system architecture.
  - Additional documentation or a separate walkthrough guide detailing the code structure and important modules.
- **Demonstration:**
  - A short video or live demo showing the application in action, emphasizing both the UI interaction and the AI's QA functionality.
  -