Project Proposal: Intelligent Educational Assistant (RAG App)

1. Project Description

The Intelligent Educational Assistant is a Retrieval-Augmented Generation (RAG) application designed to help students quickly and accurately find answers to their academic questions using uploaded PDF educational materials such as lecture notes, textbooks, and research papers.

The system combines information retrieval and language generation to ensure that students receive factually correct and contextually relevant responses extracted directly from their study resources.

By leveraging Natural Language Processing (NLP), vector databases, and large language models (LLMs), the app enables students to:

- Upload their course PDFs.
- Ask questions in natural language.
- Receive concise, cited answers derived from the documents.

The platform aims to reduce study time, improve comprehension, and enhance accessibility to educational content.

2. Group Members and Roles

Name	Role / Responsibility
Wael Atalla	LLM and Generation
Ahmed El Shenawy	Backend RESTful APIs
Hanin Reda	Vector Database Indexing and Retrieval
Rahma Mostafa	Deployment and Infrastructure
Mai Mohammed	Frontend Development

3. Team Leader

Name: Mai

Role: Project Leader

Responsibilities: Supervising the project, coordinating tasks, ensuring timely milestone completion, and managing communication between sub-teams.

4. Objectives

- Main Goal:

To develop a smart educational assistant that utilizes Retrieval-Augmented Generation (RAG) to provide accurate, context-based answers to students' academic questions directly from PDF learning materials.

- Specific Objectives:

1. Build a reliable RAG pipeline integrating retriever and generator models.
2. Develop an intuitive web interface for PDF upload and interactive Q&A.
3. Optimize retrieval accuracy and response quality.
4. Deploy the system using cloud and containerization technologies.

5. Tools and Technologies

Category	Tools / Frameworks
Backend Framework	FastAPI
RAG Framework	LangChain
Language Models	LLM (Groq/ Llama)
Vector Database	pgvector
Programming Language	Python

6. Milestones and Deadlines

Phase	Task	Duration
Phase 1	Collect and preprocess educational data	1 week
Phase 2	Design RAG pipeline (Retriever + Generator)	1 weeks
Phase 3	Backend development (APIs, database integration)	1 weeks
Phase 4	Frontend UI development and integration	1 weeks
Phase 5	Deployment and documentation	1 week
Phase 6	Final review and	1 week

presentation

7. Expected Outcome

By the end of the project, the team will deliver a fully functional RAG-based educational assistant capable of:

- Handling PDF uploads and contextual question answering.
- Providing accurate, source-grounded responses.
- Offering a user-friendly web interface accessible to students and educators.