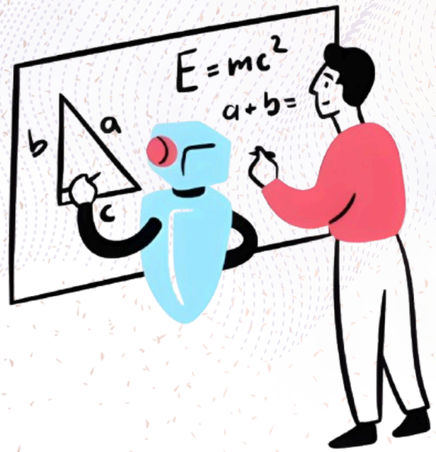
Proposal Presentation



Hybrid Documents RAG based Chatbot

Group - 3

- Maharun Afroz
- Sanjida Amin Nadia
- Tasnia Hossain
- Tahmina Mozumdar

Course Instructor

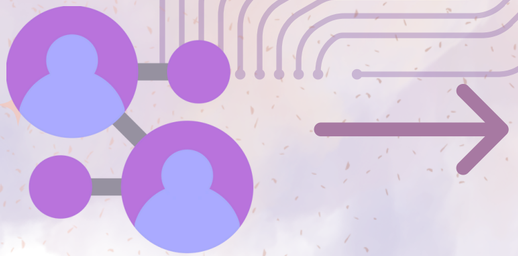
Dr. Mohammad Shifat-E-Rabbi
Associate Professor
Dept. of Computer Science

Problem Statement

A RAG-based chatbot allows users to interact with documents that state-of-the-art **models do not have prior knowledge of**. It enhances **security** by offering an offline mode, ensuring **data privacy**. Additionally, it enables seamless conversations across **multiple documents** simultaneously, improving efficiency and accessibility.

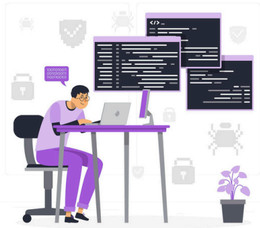
- Allows interaction with documents not covered by state-of-the-art models.
- Enhances privacy with an offline mode.
- Supports conversations across multiple documents for better efficiency.

System Flow:



User

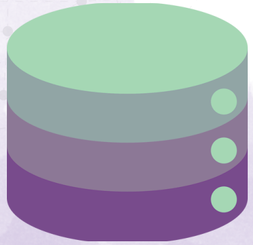
FrontEnd



Backend

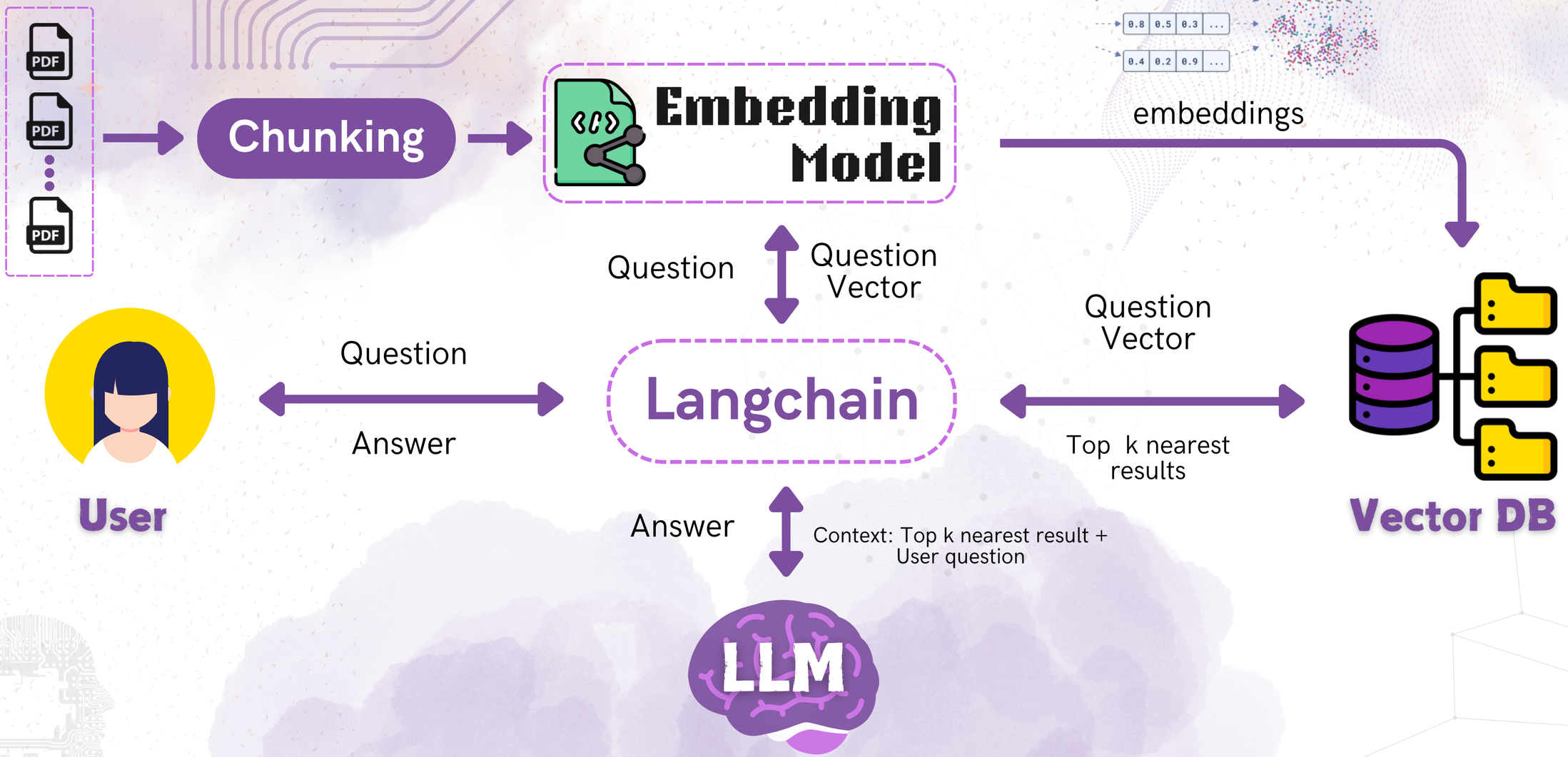
Database

Document Processing



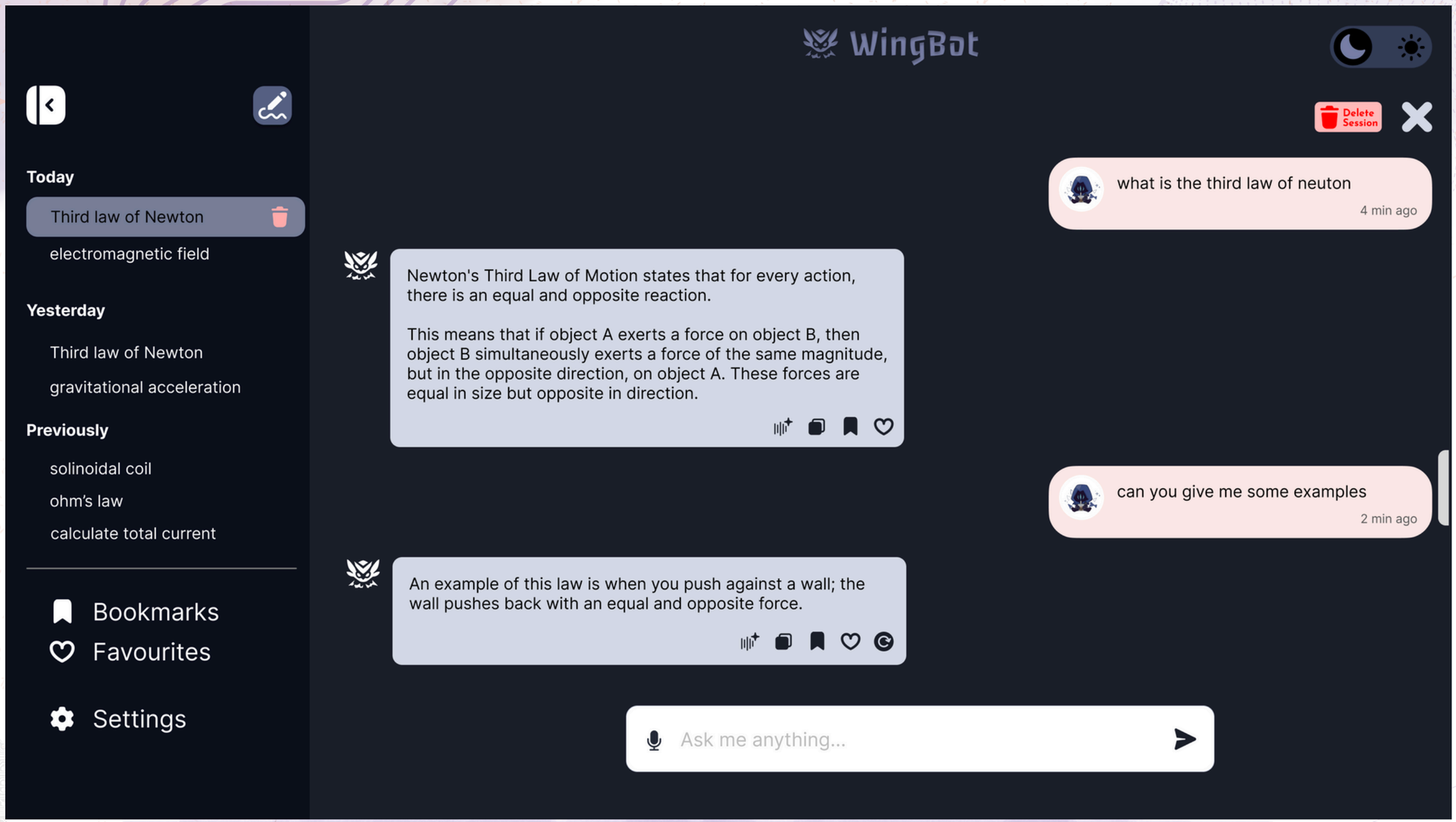
Ai Pipeline

RAG PIPELINE



UI

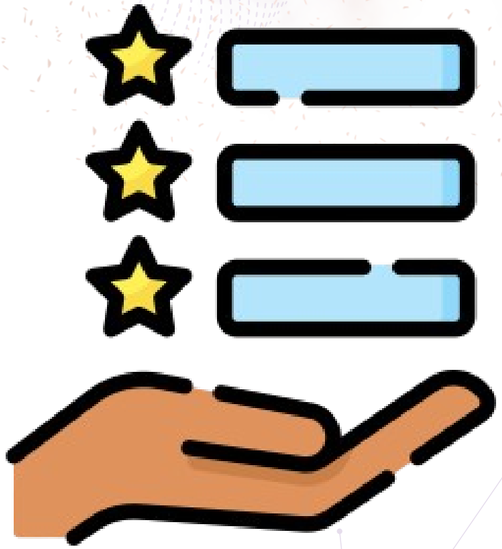
Proposed UI



Features

Key Features of the Hybrid Documents RAG-based Chatbot:

- **Hybrid Retrieval** – Supports both online (web search) and offline (local document search) for flexibility and privacy.
- **Efficient Document Processing** – Extracts text from PDFs, DOCX, and other formats using semantic and fixed-size chunking for better context.
- **Vector-Based Search** – Uses FAISS and embeddings for fast and accurate information retrieval.
- **Reranking for Accuracy** – Combines BM25, cross-encoders, and hybrid retrieval to improve search relevance.
- **Adaptive AI Models** – SLMs for speed, LLMs for complex queries, with dynamic selection based on query needs.
- **Interactive Conversational Search** – Refines responses with follow-up queries and learns from user feedback.
- **Privacy & Security** – Offline mode ensures sensitive data stays on-device, reducing cloud dependencies.
- **User-Friendly Interface** – Enables seamless document search with conversational interaction.



This chatbot enhances document-based search with AI-driven conversation, accuracy, and privacy.

Chatbot Testing Strategy

- **Unit Testing:** Test individual modules in isolation (FAISS, embeddings, LLM).
- **Integration Testing:** Verify component interaction (Embeddings → FAISS, Pinecone → LLM).
  - **Sequential & Parallel Integration:** Test components in sequence or parallel.
  - **Data Flow Testing:** Ensure correct data transformation.
  - **Interface Testing:** Validate data handling and error management.
- **End-to-End (E2E) Testing:** Simulate real-world interactions with the chatbot.
- **Functional Testing:**
  - **NLU Testing:** Test entity and intent recognition.
  - **NLP Testing:** Evaluate response accuracy and paraphrasing quality.
- **Load Testing:** Simulate high traffic to assess performance.
- **A/B Testing:** Compare chatbot variations for optimization.
- **User Acceptance Testing (UAT):** Gather user feedback before deployment.

Unit Testing

A/B Testing

NLP Testing

Chatbot Performance Metrics

- Accuracy Metrics:
  - **BLEU / ROUGE:** Measure n-gram overlap for response quality.
  - **Exact Match:** Percentage of perfectly matched responses.
  - **Hit Rate:** Checks if relevant documents are retrieved.
- Latency & Efficiency:
  - **Response Time:** Time to generate a response.
  - **Throughput:** Requests handled without performance drop.
  - **Query Retrieval Time:** FAISS document retrieval speed.
- Other Metrics:
  - **Bias Assessment:** Detects unintended biases.
  - **Response Variation:** Avoids repetitive answers.
  - **Error Handling:** Tests responses to unknown inputs.
  - **Security:** Checks for data leaks and vulnerabilities.

Dataset

Data is essential for **evaluating** and **tracking improvements** in our project.

It helps us assess performance, identify weaknesses, and refine our approach. We will create a dataset of **1,000 samples** to ensure thorough evaluation and optimization, leading to a more accurate and reliable system.

THANK YOU