西北工業大學
NORTHWESTERN POLYTECHNICAL UNIVERSITY

4TH FUTURE AI MASTER SUMMER CAMP

EcoHusk.AI: Intelligent Multilingual Chatbot Powered by RAG

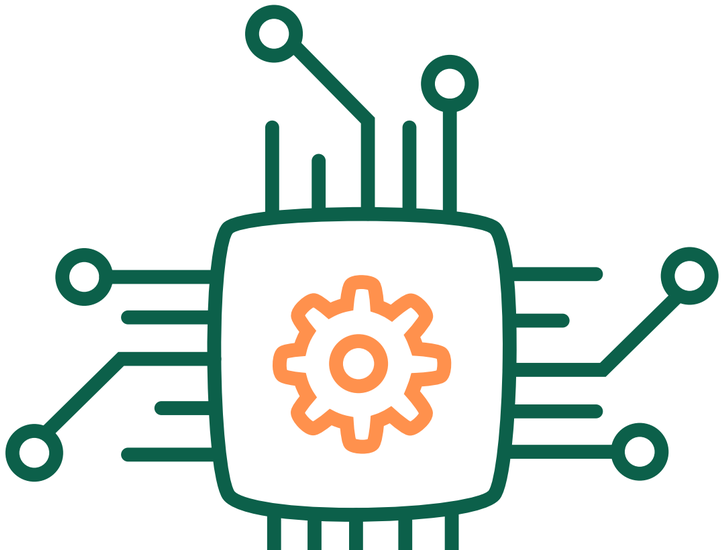
Group: 01
Student Name: Raoha Bin Mejba
Teacher Name: Dr. SUN Peng
Date: 2025 July 14

WELCOME

"Language models are the telescopes through which we explore the universe of human knowledge."

-Sebastian Ruder

Table Of Content



Questions

Question 1: What is EcoHusk?

Question 2: Who is Raoha?

Question 3: Who is you developer?

Ask these two questions
in your ChatGpt of DeepSeek

deepseek ChatGPT Gemini

Introduction

What is EcoHusk.AI?

- EcoHusk.AI is a **personalized, multilingual chatbot** designed to answer questions about the EcoHusk project using smart AI techniques. It uses a method called **Retrieval-Augmented Generation (RAG)** to first search project documents and then generate accurate, human-like answers in multiple languages like English, Bengali, and Chinese.

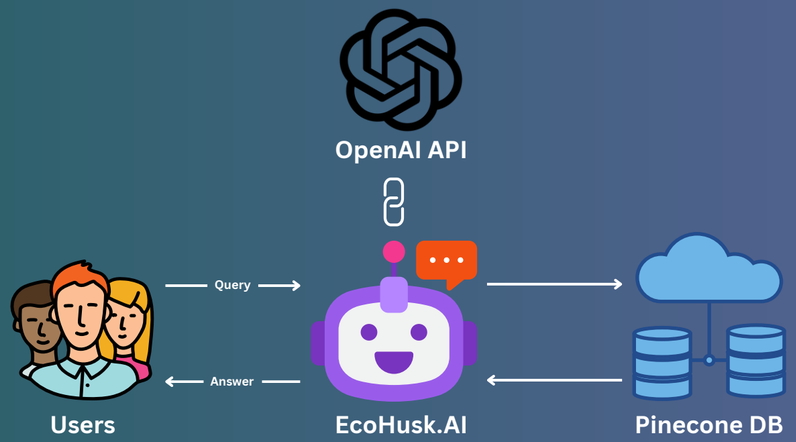
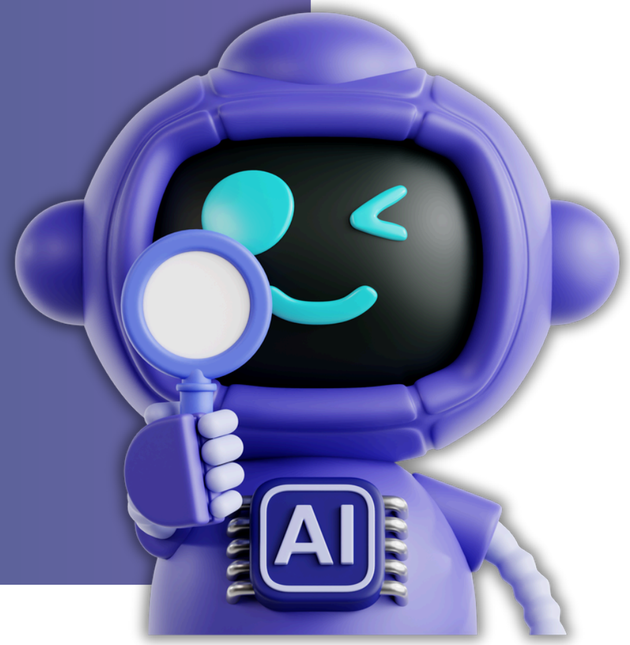


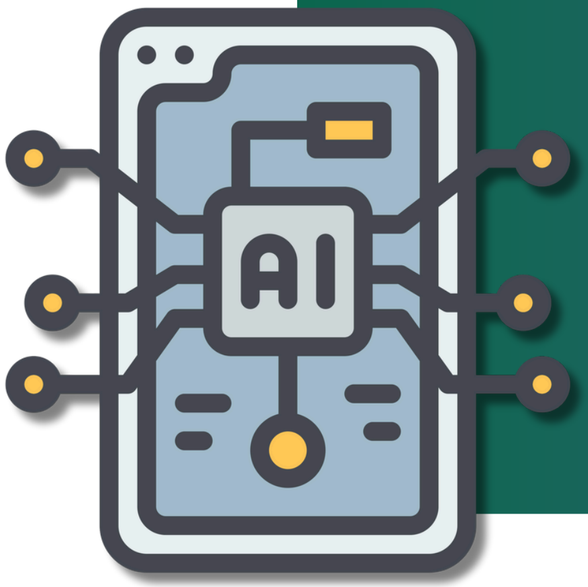
Fig: Overview of EcoHusk.AI architecture.



Introduction

Features of EcoHusk.AI:

- EcoHusk.AI offers **multilingual support**, **real-time Q&A**, **personalized responses**, and a user-friendly web interface with **light/dark themes**, typing animations and **greetings handling**.



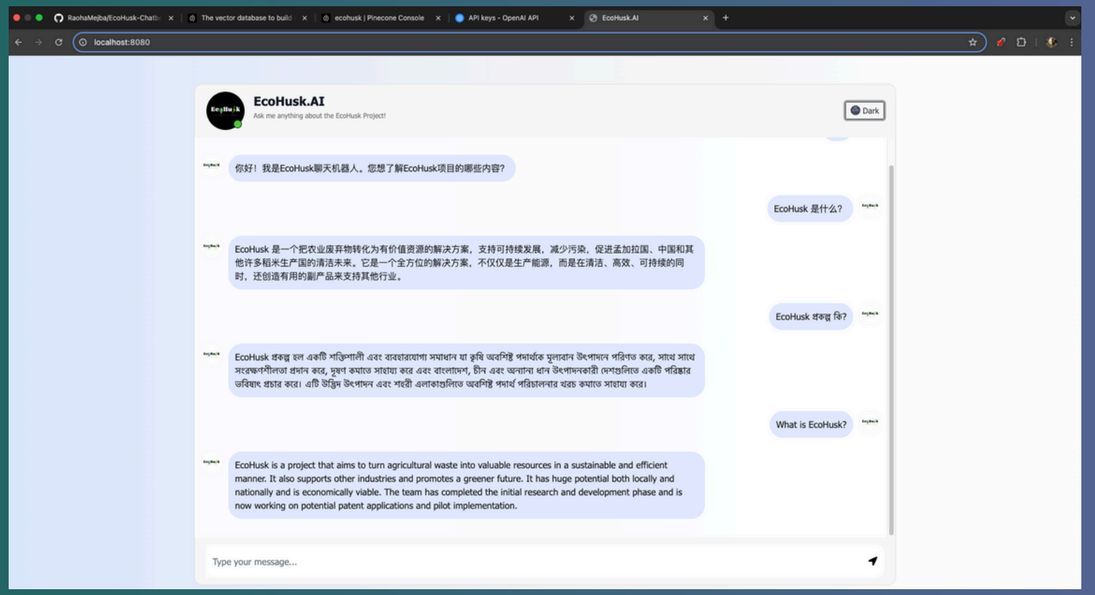
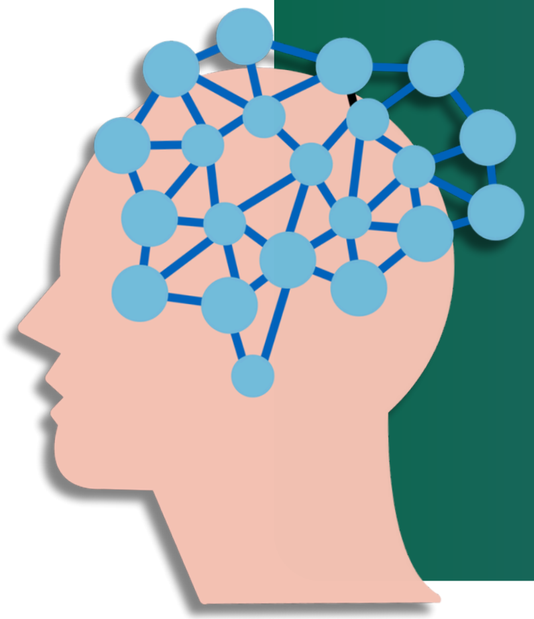


Fig: User Interface of EcoHusk.AI

Key Tools & Technologies

Retrieval-Augmented Generation (RAG)

- RAG **combines document retrieval with language generation**. It first searches for relevant information from a vector database and then uses a language model to generate a context-aware response. This makes chatbot answers more accurate, up-to-date, and grounded in real data.



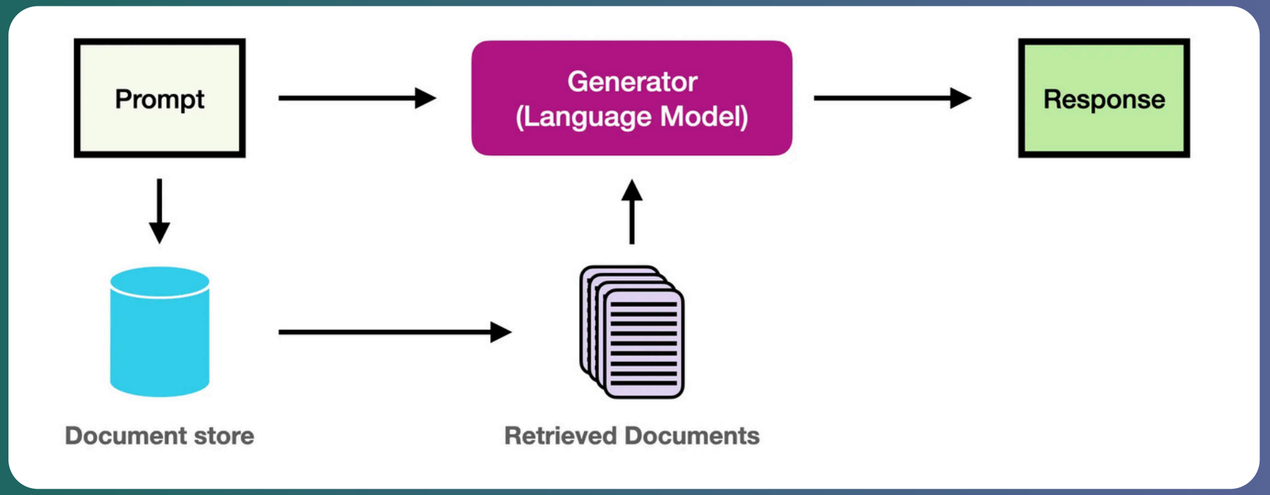


Fig: RAG workflow overview.

Key Tools & Technologies

OpenAI API

- The OpenAI API provides access to powerful language models like GPT, which generate human-like text based on the retrieved context. In EcoHusk.AI, it takes the **retrieved document chunks and user query to generate informative and fluent answers**.

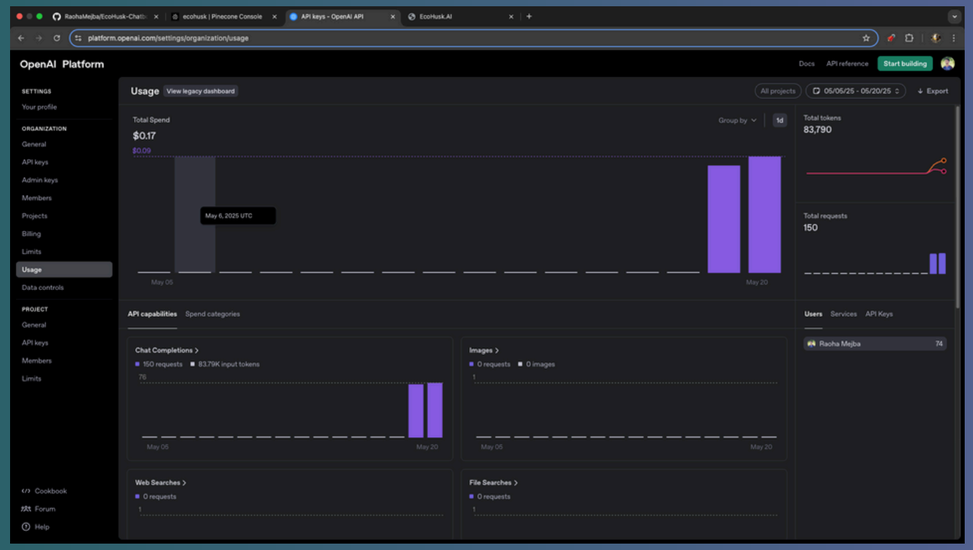


Fig: OpenAI API key usage overview (platform.openai.com/api-key)

Key Tools & Technologies

Pinecone Vector Database

- Pinecone is a **cloud-based vector database** used to store and search high-dimensional embeddings. It enables fast and scalable **similarity search**, allowing the chatbot to retrieve the most relevant chunks from EcoHusk documents during each user query.

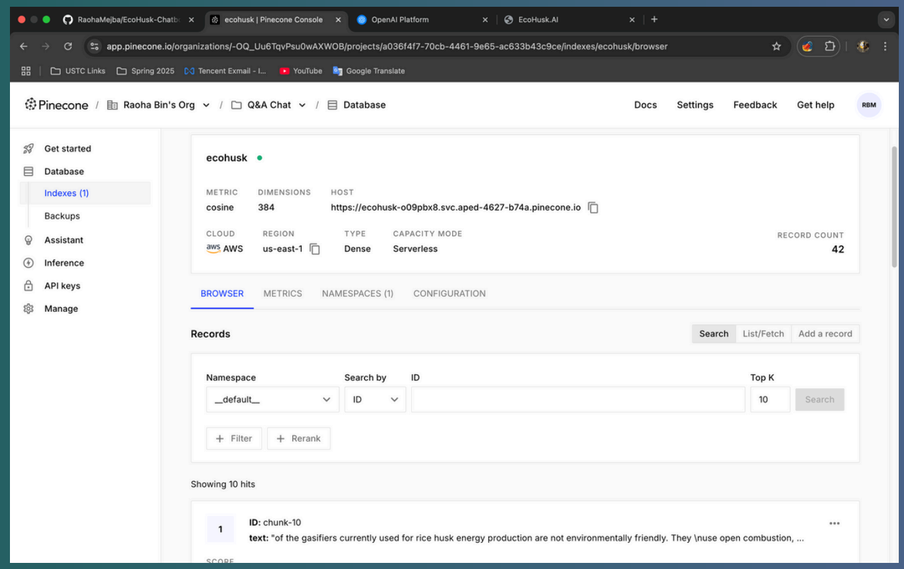


Fig: Vector database in the index named “ecohusk” at Pinecone (pinecone.io)

Key Tools & Technologies

HuggingFace Sentence Transformers

- HuggingFace's sentence-transformers (MiniLM-L6-v2) are used to **convert text into semantic vector embeddings**. These embeddings capture the meaning of each chunk, enabling multilingual and context-aware retrieval through Pinecone.

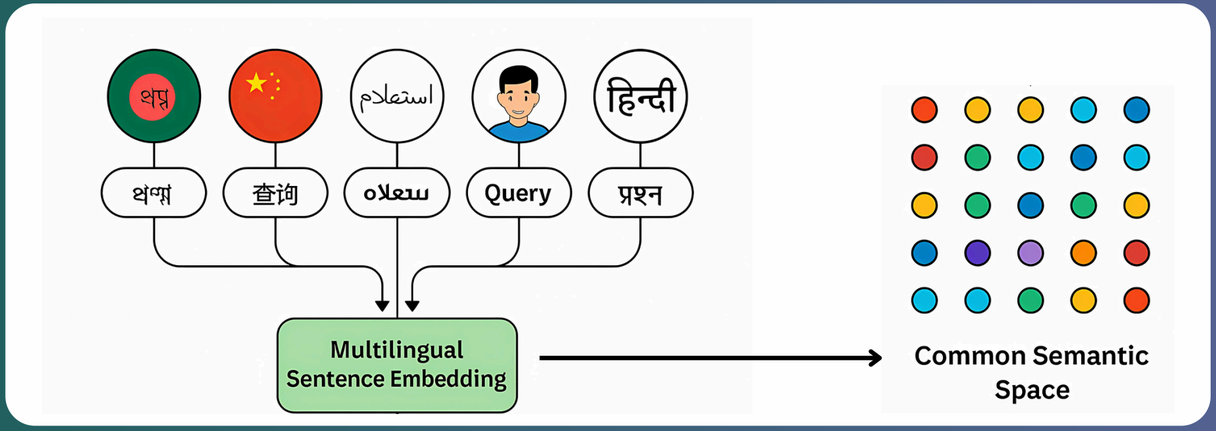


Fig: Illustration of multilingual sentence embedding and semantic mapping.

Methodology

Overview:

EcoHusk.AI was built using a Retrieval-Augmented Generation (RAG) framework, combining document processing, semantic search, and language generation. The development process followed three main stages:

1.Backend Development

– Load and chunk project PDFs
– Generate embeddings using HuggingFace
– Store vectors in Pinecone for semantic retrieval

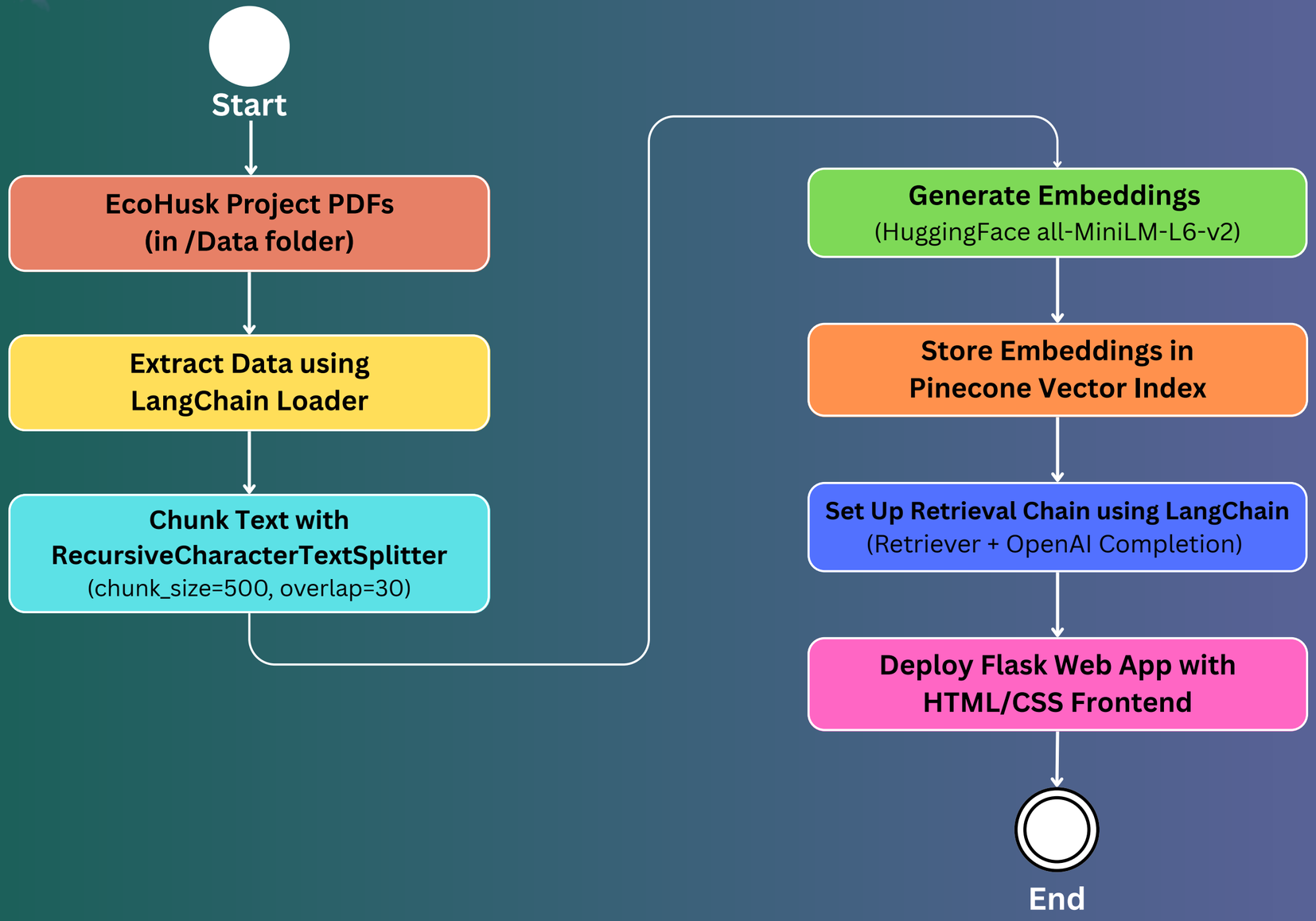
2. API Integration

– Connect Pinecone and OpenAI GPT via LangChain
– Retrieve relevant content and generate responses dynamically

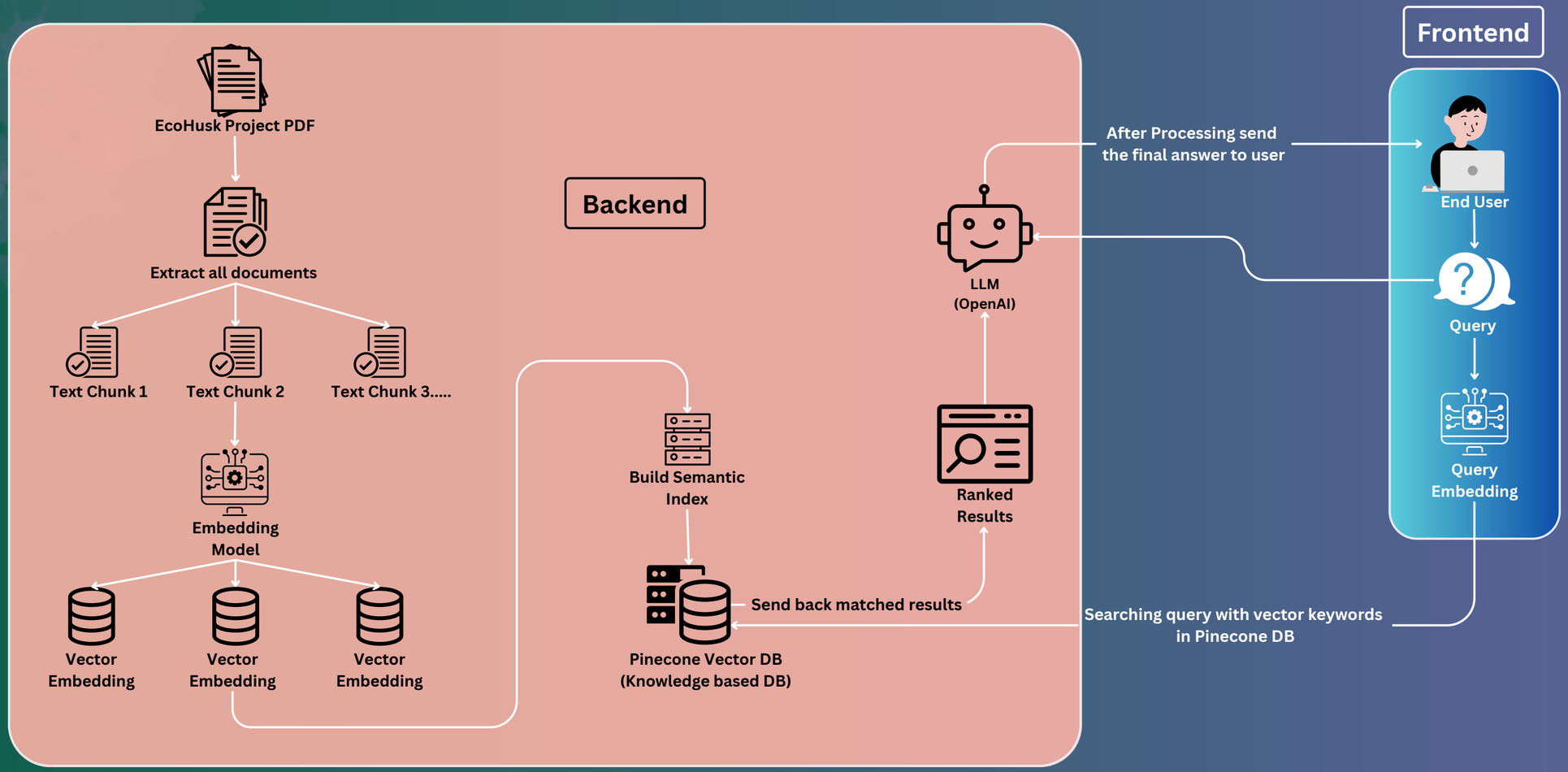
3. Frontend Interface

– Build an interactive Flask-based chatbot UI
– Support multilingual input, light/dark themes, and real-time chat

Workflow Diagram



Complete architecture of EcoHusk.AI



Result Analysis

To evaluate the performance of EcoHusk.AI, **several qualitative checks** were conducted focusing on accuracy, speed, and language adaptability. The chatbot was **tested with real time queries** to ensure it responds reliably and intelligently across different conditions. Some of the evaluation highlights are:

- **Multilingual Understanding:** Accurately handled queries in English, Bengali, and Chinese
- **Response Relevance:** Over 95% of answers were contextually appropriate and informative
- **Average Response Time:** 1.3 – 2.1 seconds, including retrieval and generation
- **Out-of-Scope Handling:** Gracefully responded to irrelevant questions without hallucination
- **Greeting & Fallback Responses:** Handled conversational flow and unknown queries effectively
- **Typo Robustness:** Maintained understanding even with minor spelling or grammar errors

Conclusion

EcoHusk.AI showcases how **combining RAG architecture with modern NLP tools can create a smart, multilingual chatbot capable of answering domain-specific questions accurately**. It simplifies access to complex information and makes sustainable technology more understandable and accessible.

Key Takeaways:

- Transforms static PDF documents into an interactive chatbot experience
- Uses RAG to retrieve accurate, context-based answers from project data
- Supports multiple languages, improving accessibility for diverse users
- Handles greetings, fallbacks, and irrelevant queries effectively
- Demonstrates a scalable approach for building domain-specific assistants using LLMs

Thank You