Non-Profit Quiz/Tutor Bot

Course Name : Datagami Skill Based Course

Institution Name: *Medicaps University*

Student Name(s) & Enrolment Number(s):

Sr no	Student Name	Enrolment Number
1	Devansh Gupta	EN22CS301325
2	Avantika Gupta	EN22CS301242
3	Chiran Vikram Shahi	EN22CS301301
4	Deepali Pastariya	EN22CS301318
5	Chetan Oswal	EN22CS301295
6	Bharat Rathore	EN22CS301260

Group Name: 11D3

Project Number: GAI-35

Industry Mentor Name: Mr. Suraj Nayak

University Mentor Name: Prof. Vineeta Rathore

Academic Year: 2025-26

Problem Statement & Objectives

1. Problem Statement

Non-profit organizations rely heavily on donor communication emails to raise funds, build emotional connections, and maintain long-term relationships with supporters. However, many organizations lack structured mechanisms to evaluate the effectiveness of their donor emails in terms of persuasion, emotional storytelling, call-to-action clarity, donor psychology alignment, and ethical messaging.

Currently:

- Email evaluation is subjective and manual.
- There is no AI-assisted feedback system tailored specifically for nonprofit communication.
- Organizations do not have interactive learning tools to improve fundraising communication strategies.

Therefore, there is a need for an AI-powered system that can:

- Extract content from donor emails,
- Analyze communication quality,
- Generate evaluation-based assessments,

The **Email Evaluation Tutor & Assessment Bot** addresses this gap using multimodal AI, vector embeddings, and Retrieval-Augmented Generation (RAG).

2. Project Objectives

Primary Objective

To build an AI-driven web-based system that evaluates donor email communication strategies through interactive assessments.

Functional Objectives

- Extract donor email text from uploaded screenshots.
- Convert extracted text into semantic embeddings.

Performance Objectives

- Maintain fast response time for quiz generation.
- Ensure accurate contextual retrieval using embeddings.
- Support multiple image uploads per session.

User Experience Goals

1. Simple web interface.
2. Interactive quiz flow.
3. Immediate feedback.

3. Scope of the Project

Included:

- Uploading donor email screenshots (PNG/JPG/JPEG).
- AI-based text extraction.
- Embedding generation using Sentence Transformers.
- Contextual retrieval via ChromaDB.
- AI-generated MCQ assessment.

Excluded:

- Advanced analytics dashboards.
- Multi-user authentication system.
- Cloud-based deployment architecture.

Constraints:

- Depends on Gemini API response time.
- Local vector database (runtime storage).

Proposed Solution

1. Key Features

- Multimodal AI-based email text extraction
- Retrieval-Augmented Generation (RAG) architecture
- Semantic embedding generation
- AI-driven MCQ generation in structured JSON
- Instant feedback with explanations

2.Overall Architecture / Workflow

High-Level Design (HLD)

The system follows a modular, layered architecture integrating AI models, embeddings, and a vector database.

1. Presentation Layer (Frontend)

- Built using Streamlit.
- Handles file uploads, quiz display, user interaction, and scoring.
- Uses session state to manage runtime quiz data.

2. AI Processing Layer

- Uses Gemini 2.5 Flash model.
- Performs:
    - Text extraction from images.
    - MCQ generation from contextual data.

3. Embedding Layer

- Uses SentenceTransformer (all-MiniLM-L6-v2).

- Converts extracted text into vector embeddings.
- Enables semantic similarity search.

4. Data Layer

- ChromaDB vector database.
- Stores:
    - Extracted email text
    - Embeddings
    - Unique document IDs

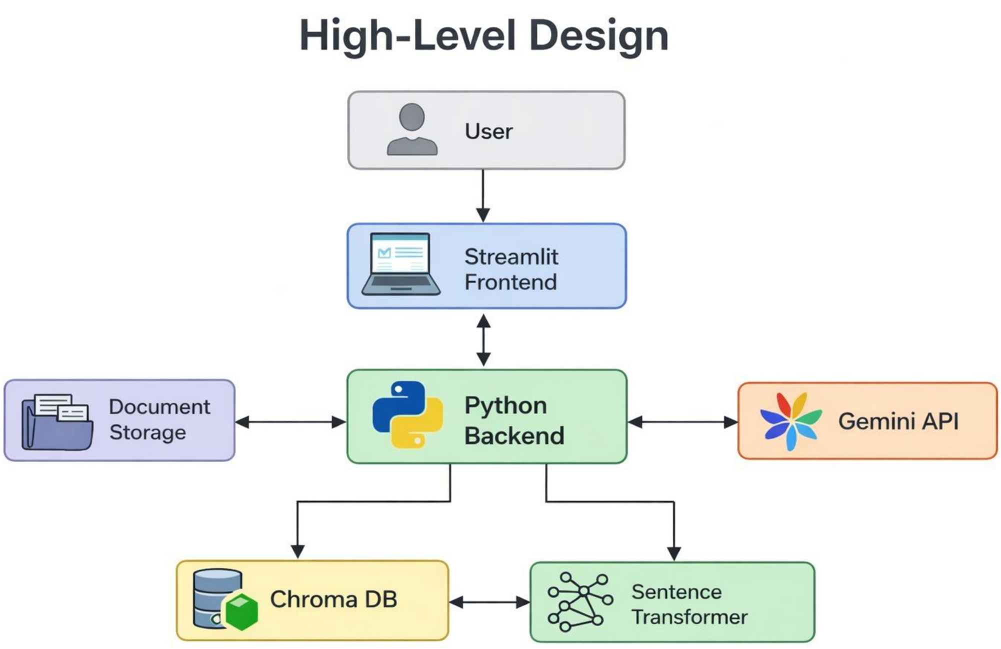


Figure 1-High level Design

Process Flow -

1. User uploads donor email screenshots.
2. Gemini extracts email text from images.
3. Extracted text is converted into embeddings.
4. Embeddings are stored in ChromaDB.

5. Query embedding retrieves relevant contextual emails.

6. Gemini generates deep evaluation MCQs in JSON format.

7. User answers questions.

8. System evaluates answers and provides feedback.

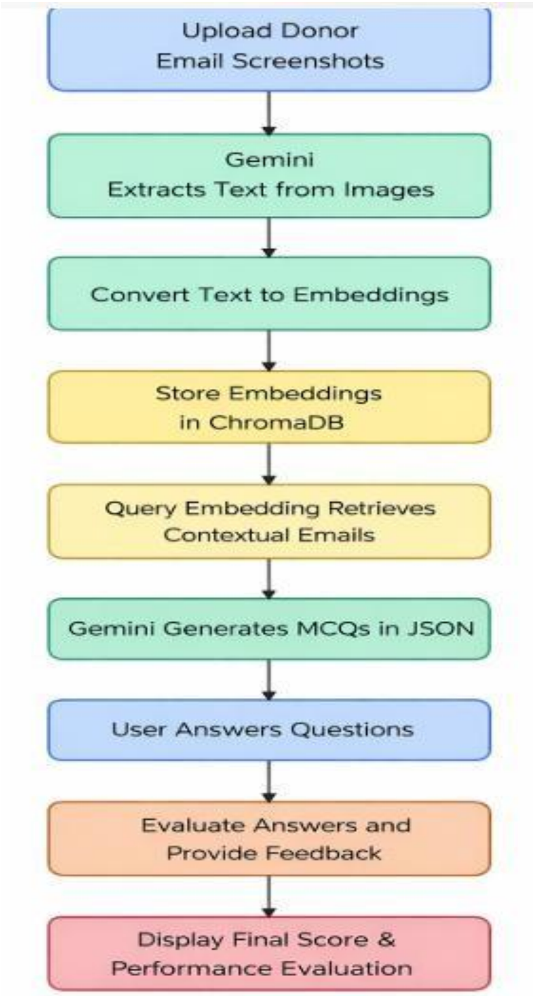


Figure 2- Process Design

Information Flow -

Image
→ AI Text Extraction
→ Extracted Text
→ Embedding Model
→ Vector Database

→ Retrieved Context
→ AI Question Generation

→ Quiz JSON
→ User Interaction
→ Answer Validation
→ Score Calculation

Components Design -

- **UI Component (Streamlit)** – User interaction and quiz display.
- **AI Component (Gemini API)** – Text extraction and question generation.
- **Embedding Component** – Semantic vector conversion.
- **Vector Database (ChromaDB)** – Context storage and retrieval.
- **Evaluation Engine** – Answer validation and scoring logic.

Key Design Considerations -

1. Modular architecture for scalability.
2. Retrieval-Augmented Generation for contextual accuracy.
3. JSON validation to handle AI output errors.
4. Secure API key storage.

3. Tools & Technologies Used

Frontend

- Streamlit

AI Model

- Gemini 2.5 Flash (Multimodal + Text generation)

Embedding Model

- SentenceTransformer (all-MiniLM-L6-v2)

Vector Database

- ChromaDB

Programming Language

- Python

Supporting Libraries

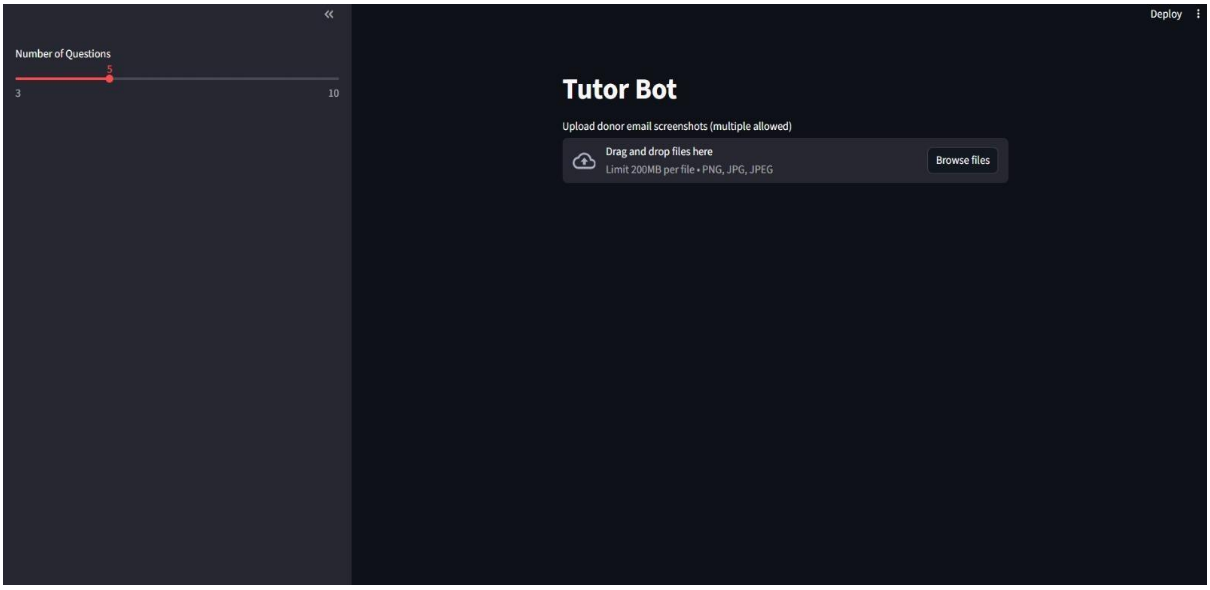
- PIL (Image processing)
- dotenv (Environment variables)

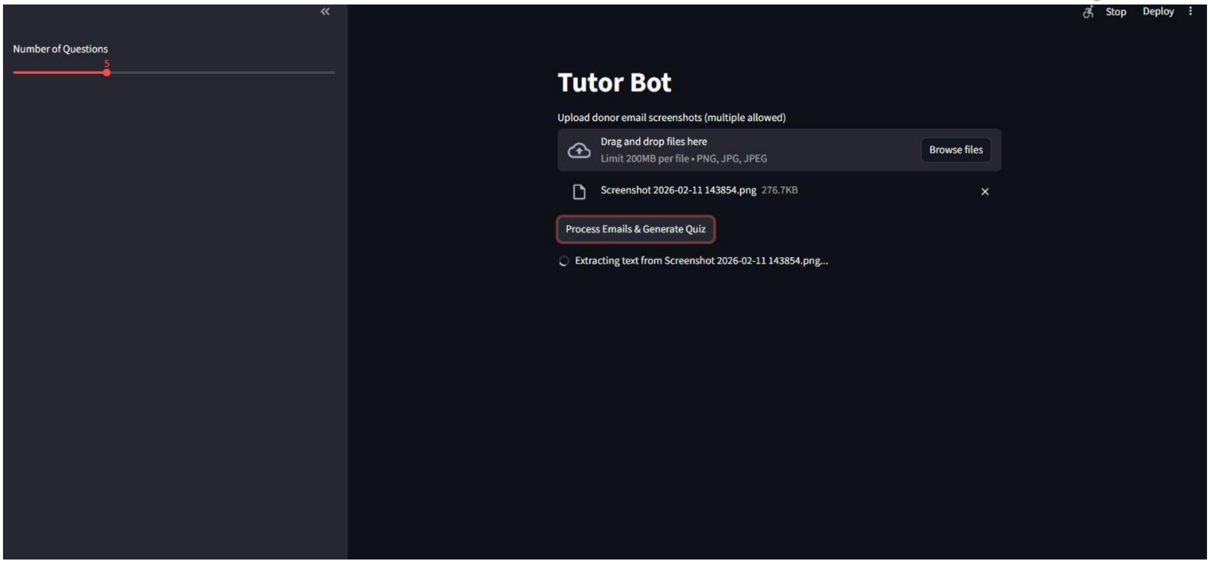
Results & Output

1. Screenshots / Outputs

1. Email Upload Interface

Users can upload multiple donor email screenshots.

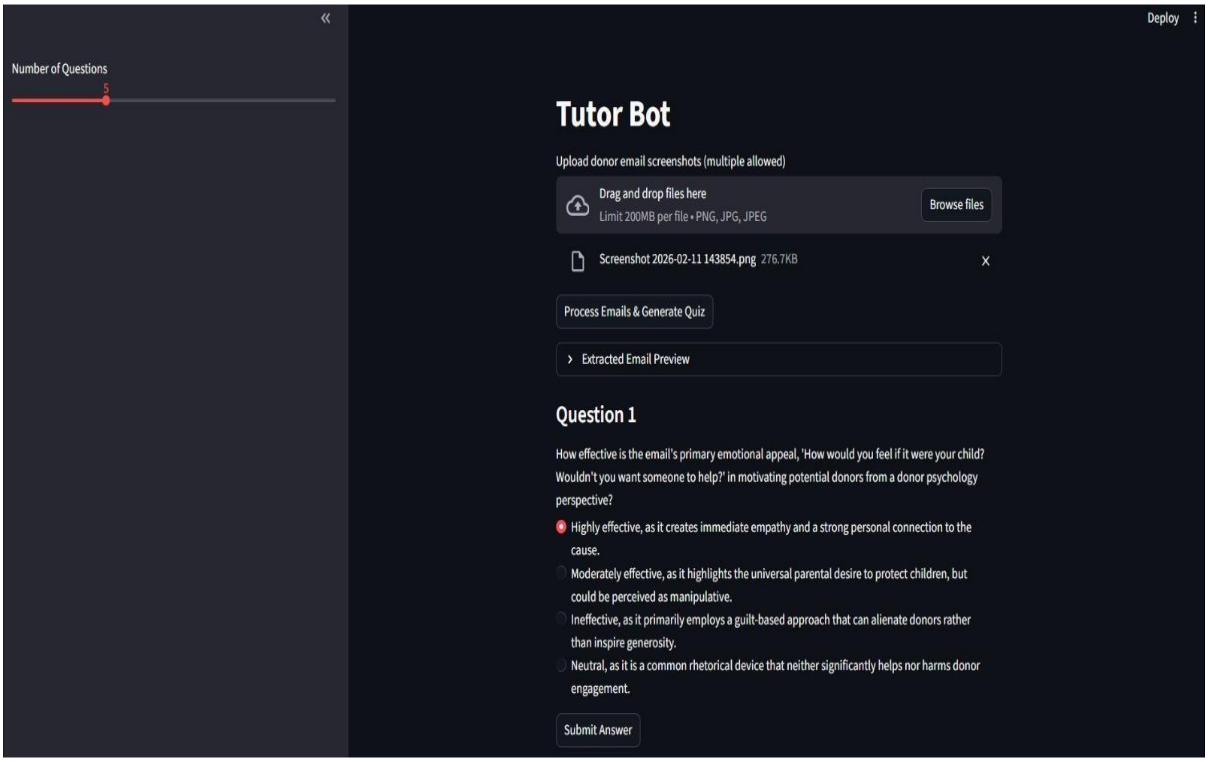




.

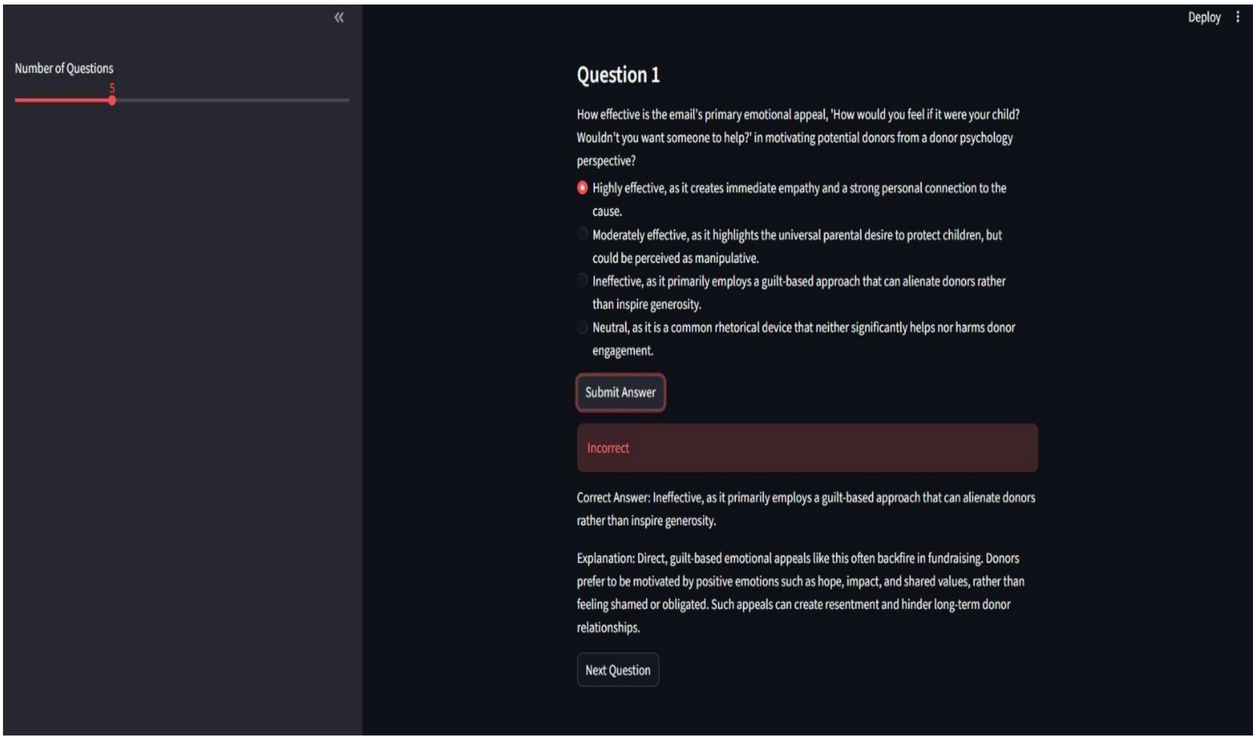
3. Quiz Interface

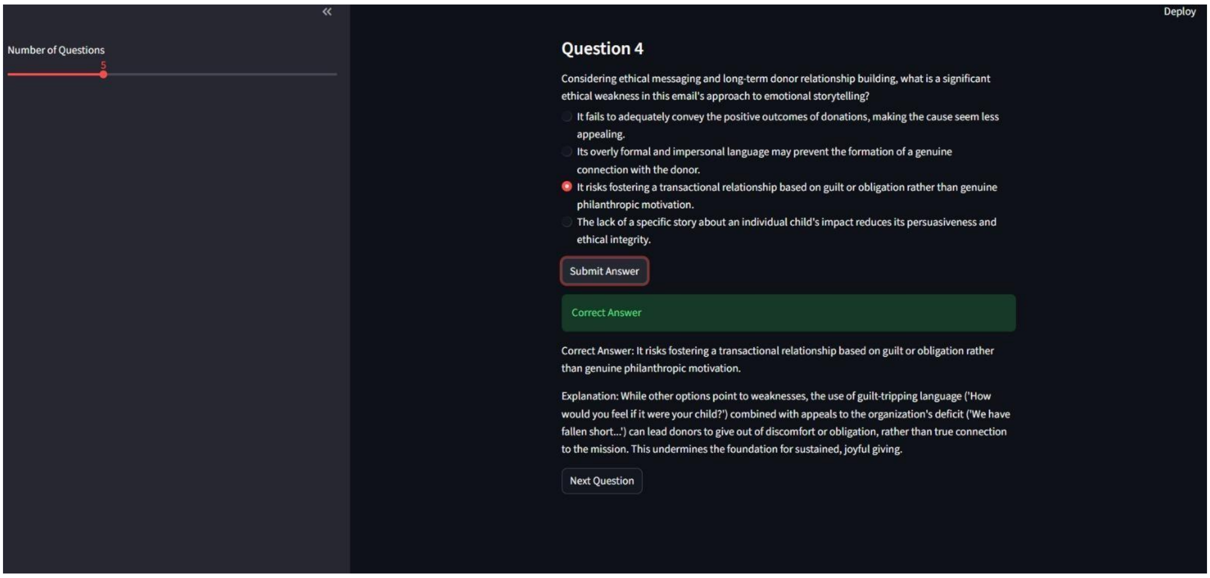
- Displays generated MCQs.
- Multiple-choice selection.
- Submit answer functionality.



4. Feedback Screen

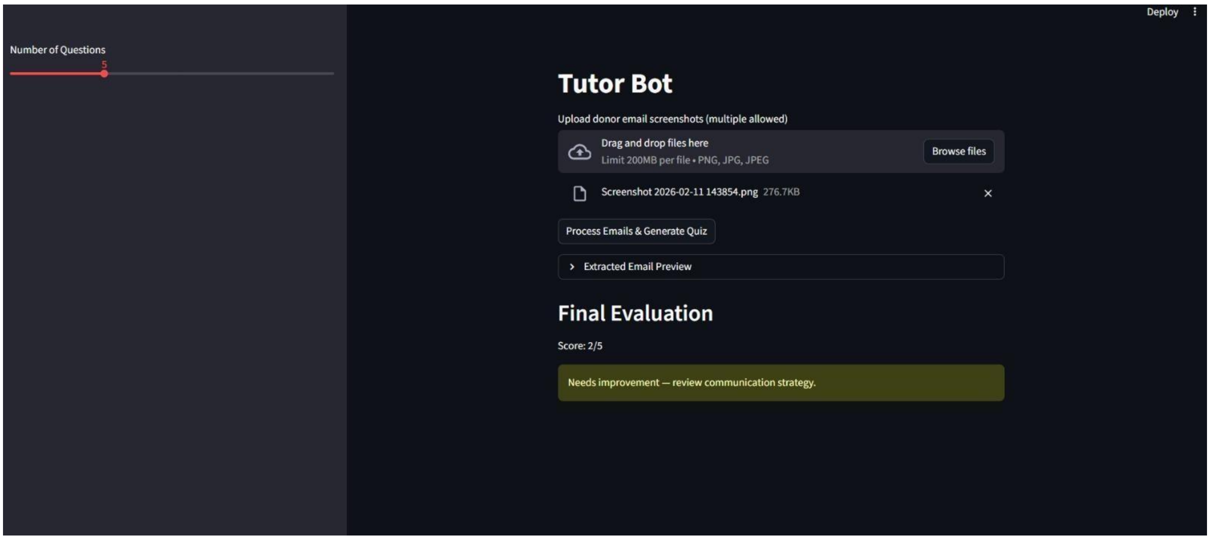
- Indicates correct or incorrect response.
- Displays correct answer.
- Provides explanation.





5. Final Evaluation Screen

- Displays total score.
- Provides qualitative feedback:
    - Excellent
    - Good but needs improvement



2. Reports / Models

- AI-generated MCQs in structured JSON format.
- Semantic embedding vectors stored in ChromaDB.
- Retrieval-based contextual question generation.

3. Key Outcomes

- Successfully implemented Retrieval-Augmented Generation (RAG).
- Achieved contextual quiz generation using embeddings.
- Implemented multimodal AI for OCR-like extraction.
- Ensured secure API key handling.

Conclusion

The Email Evaluation Tutor & Assessment Bot successfully transforms static donor email screenshots into an intelligent evaluation and learning system. By integrating multimodal AI, semantic embeddings, and vector-based retrieval, the system enables structured analysis of nonprofit communication strategies.

The project demonstrates:

- Practical implementation of Retrieval-Augmented Generation (RAG).
- Integration of AI models into real-world workflows.
- Use of vector databases for contextual reasoning.

Key learning outcomes include:

- System architecture design.
- AI model integration.
- Embedding-based retrieval.
- Session state management.

Future Scope & Enhancements

One major enhancement would be migrating from a local runtime vector database to a cloudbased persistent vector storage system, allowing long-term data retention and multi-user scalability. This would enable organizations to build historical communication intelligence over time.

Further enhancements may include:

- Integration with real email platforms (e.g., Gmail API) for live email evaluation.
- Advanced analytics dashboard with visual insights into persuasion strength and CTA clarity.
- Caching mechanisms to reduce repeated API calls and improve response time.
- Multi-user authentication and role-based access control.