

■ Milestone 1 – File Upload + Chatbox (Streamlit Project)

1. Introduction

The goal of this project is to build a Streamlit-based web app that allows users to:

- Upload files via a colorful, styled file uploader.
- Interact with a chatbox for simple Q&A.;

This lays the foundation for later integration with LLMs (like Ollama or OpenAI GPT) to enable intelligent file-based Q&A.;

2. Methodology

a) Tools & Technologies

- Python 3.10+ – Programming language
- Streamlit – Web app framework for interactive UI
- HTML/CSS – Styling for the file upload widget

b) Implementation Steps

- UI Setup – Used `st.set_page_config` for app title & favicon. Added custom CSS styling for a colorful file uploader.
- File Upload Feature – Implemented with `st.file_uploader`. Supports .jpg, .png, .pdf, .txt. Displays confirmation after upload.
- Chatbox Feature – Used `st.chat_input`. Stored messages in `session_state`. Implemented a basic echo bot for responses.

c) Workflow

- User uploads file → File is stored in memory.
- User interacts via chatbox → Messages + Bot replies displayed.
- All chat history maintained in session state.

3. Outcomes

- Successfully created colorful file uploader.
- Added chatbox UI for user interaction.
- Application runs locally using `streamlit run app.py`.

4. UI Screenshots

Screenshots should include:

- File uploader interface
- Chatbox interface with messages

5. Future Scope

- Connect chatbox to Ollama / OpenAI GPT.
- Extract text from uploaded PDFs using pypdf.
- Add support for multiple files.
- Deploy app on Streamlit Cloud.

6. Conclusion

This milestone demonstrates the ability to design a Streamlit UI with file upload and chatbox. The project is modular, allowing easy integration with AI models in later stages.