**Document Search Bot**

**Introduction**

The Document Search Bot is an intelligent assistant designed to streamline document retrieval and information retrieval processes. By leveraging natural language processing and a question-answering model, the bot aims to enhance user experience in accessing information within various document formats.

**Objectives**

The primary objectives of this PoC are to:

Demonstrate the feasibility of the Document Search Bot.

Showcase the key functionalities, including document upload, deletion, and user queries.

Validate the effectiveness of the question-answering model in providing relevant responses.

**Prerequisite**

* Install Python: Install Python on your local machine.
* Install a code editor: Install any code editor for running your code (e.g., Visual Studio Code).

To install the required packages, you can run the following commands in your terminal or command prompt:

pip install streamlit

pip install PyPDF2

pip install langchain

pip install pandas

pip install python-pptx

pip install python-docx

pip install python-dotenv

**Data Detective Bot📑**

This script, built with Streamlit, serves as a document search bot. It enables administrators to upload and delete various document types (PDF, TXT, DOCX, XLSX, PPTX). Both Admin and User can utilize a question-answering model for document search. Admins have additional capabilities for document management, while normal users can perform search queries exclusively.

* **Obtaining API Key**: Sign up for an account on the OpenAI platform and generate an API key from the dashboard.
* This key is used for authentication and authorization to access OpenAI's services, such as their language models and other APIs. It allows the application to communicate with the OpenAI platform and make use of its capabilities, such as natural language processing.
* **Using API Key**: In your application code, set the API\_KEY variable to the generated API key to authenticate requests to the OpenAI API.
* The code is implemented in app.py file.
* To set up the environment variables, you need to create a file named **.env** in the same directory as your script and define the required variables.

**OPENAI\_API\_KEY = your\_openai\_api\_key\_here**

**function and class in the code**

* **SessionState class:** Holds user session information like document folders and login status.
* **process\_documents function:** Extracts text from uploaded documents (PDF, TXT, DOCX, XLSX, PPTX) and creates a searchable index.
* **answer\_query function:** Takes your question, searches through the document index, and uses an AI model to answer based on relevant documents.
* **delete\_files function:** Helps admins remove unwanted documents from their uploaded files.
* **exponential\_backoff\_retry function:** Handles errors by retrying a function with increasing wait times to avoid overwhelming servers.
* **display\_assistant function:** Creates the AI assistant chat on the side, allowing you to ask questions and see answers based on your documents.
* **main function:** Initializes the app, handles user roles (admin/normal), document upload, and question answering functionality.

**Running the Application**

* Open the terminal or command prompt.
* Navigate to the project directory.
* Run the Streamlit app:

C:\genai> streamlit run app.py

* We get

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501

Network URL: <http://172.27.96.53:8501>

* So, now it will directly redirect to browser to access

**How to Use the Data Detective Bot📑:**

* **Accessing the Bot:**

Open your web browser and navigate to [Bot URL].

The Document Search Bot interface will appear.

* **User Type Selection:**

Choose User Role : Admin or Normal User.

1. **Admin Authentication:**

If you are accessing as an Admin, enter the username "admin" and password "admin@123" for authentication.

**Document Upload (Admin Only):**

After authentication, you can upload documents by dragging or browsing files from your local machine.

Supported document types: PDF, TXT, DOCX, XLSX, PPTX.

**Asking Queries:**

Type your query in the input field on the right side.

Click the "Request" button to request information.

**Assistant Bot 🤖:**

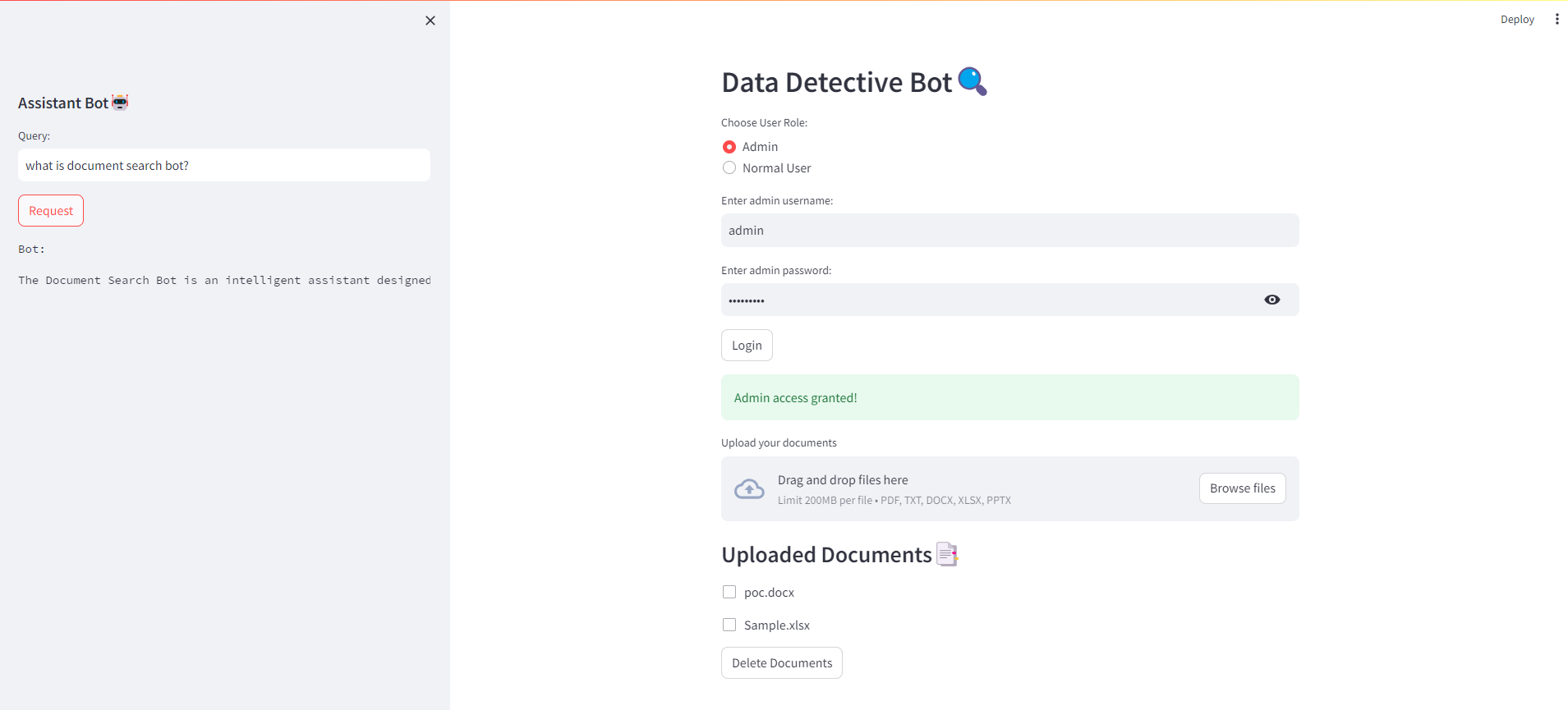
The AI Assistant will respond to your query, and the conversation will appear on the right side.

**Document Deletion (Admin Only):**

As an Admin, you can delete documents.

Select the checkboxes next to the documents you want to delete.

Click the "Delete Documents" button.



1. **user Authentication:**

If you are accessing as a Normal user, you can see the list of documents that are uploaded.

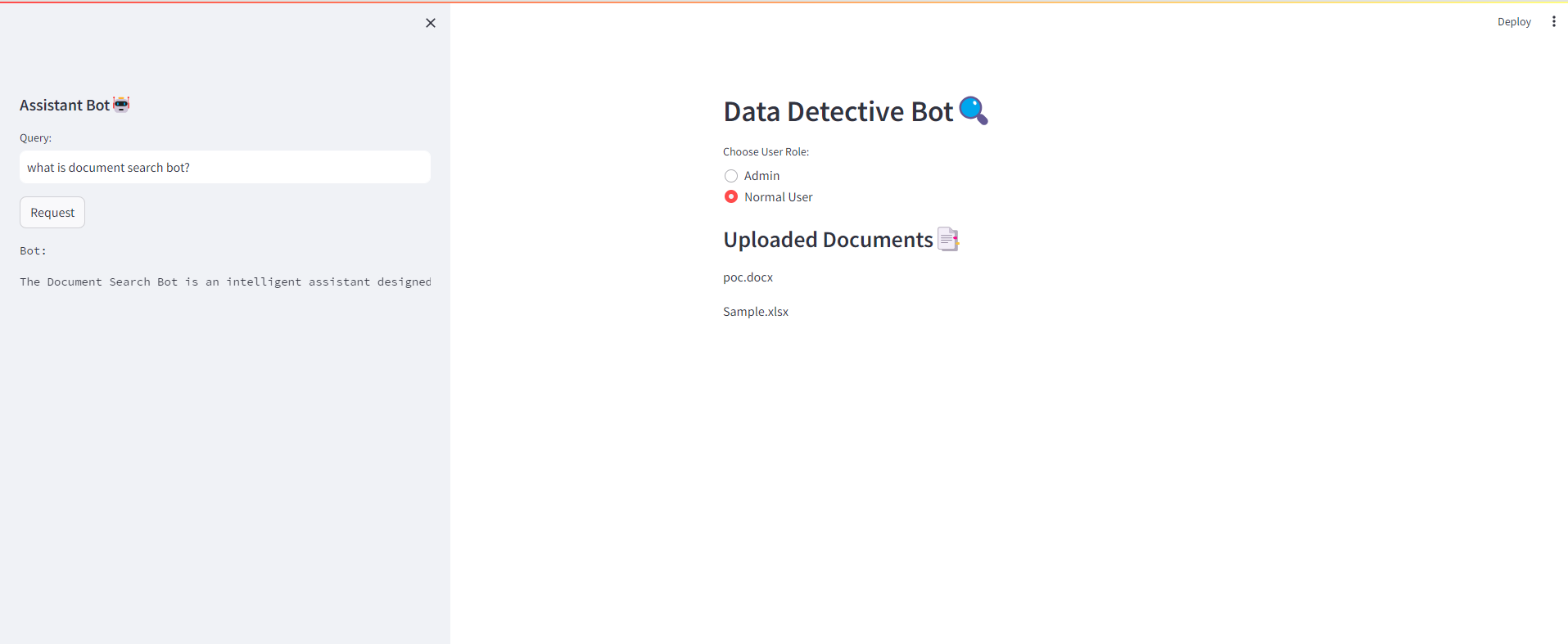
**Asking Queries:**

Type your query in the input field on the right side.

Click the "Request" button to request information.

**Assistant Bot 🤖:**

The AI Assistant will respond to your query, and the conversation will appear on the right side.



**Test Summary:**

The Document Search Bot was tested for functionalities including document upload, deletion, query processing, error handling, and document retrieval. Various scenarios were covered to ensure reliability and robustness.

**Conclusion**

The PoC demonstrates the successful implementation of the Document Search Bot, showcasing its key features and functionalities. Initial findings indicate a promising solution for enhancing document search and retrieval processes.