Name: Harshan M V  
Roll No: 22PD14

Chosen question: Personal AI Communication Assistant

General Summary:

All the 3 projects involve the use of APIs to fetch the live data.

* For Gmail I have used google cloud API to fetch live data and Streamlit is used for the front end. The email classification is done using Naïve Bayes and rule-based categorization. The replies are generated using the FLAN-T5. Sample categorization of the email is done and we also get input from the user to check if its correct else the user input is used to train the model. For the summarization I have used reges based model. We also implement temperature based sampling for diverse responses. The replies are encoded using the Base64 and send via Gmail API. We avoid memory overload using Hugging Face cache. The user can also opt in to add new label like follow up with will update correspondingly in the follow ups as well.
* For Slack, Streamlit is used to connect Slack for retrieving messages and process them for key insights. It includes functionalities like message summarization, actionable message extraction, and smart search. It uses slack\_sdk.WebClient to connect to Slack. We use NLP based summarization which tokenizes messages into sentences. It performs smart search using keyword-based matches.
* For WhatsApp, We have given the user for Webhooks option as well. twilio library facilitates the integration with Twilio's API, enabling sending and receiving WhatsApp messages. flask is employed to create a local web server for handling incoming webhook requests from Twilio, ensuring real-time message processing. threading and socket are used to run the Flask app in a separate thread and manage network connections, ensuring the main Streamlit app remains responsive. re and nltk are used for text processing, including tokenization, stop word removal, and sentiment analysis, enabling the bot to understand and respond to user messages effectively.

Running the code:  
For WhatsApp:

1. Install the required dependencies (the libraries)
2. **Twilio Setup:** Create a Twilio account and obtain your Account SID, Auth Token, and a Twilio phone number with WhatsApp capabilities.
3. Ngrok Setup: Install ngrok and ensure it's in your system's PATH.
4. Configure Twilio: In the Streamlit app's sidebar, enter your Twilio credentials. Then, start ngrok. Copy the provided ngrok URL and paste it into your Twilio WhatsApp Sandbox configuration as the webhook URL.
5. **Upload chat data:** To use the chat analysis features, upload a WhatsApp chat export file.

For Slack:

1. Install the libraries of streamlit, nltk, transformers and torch
2. Ensure there exists a D drive
3. Provide the slack token
4. Ensure that the slack app has been authorized the required permission to fetch the messages. The slack app must be installed in the required webspace.

For Gmail

1. Create a Google cloud account and slect the free version.
2. Copy the API and store it as a .json file in the same directory as that of the .py file and rename it as credentials.py
3. Authorize appropriate permissions.