

Executive Summary

In the digital age, platforms like Facebook increasingly serve as hotspots for fraudulent activities, with the elderly frequently falling victim. These activities range widely, from deception of personal information to money theft. Such scams can lead to significant financial losses and emotional turmoil for those impacted. To address this growing concern, the HoneyTrap AI chatbot was initiated.

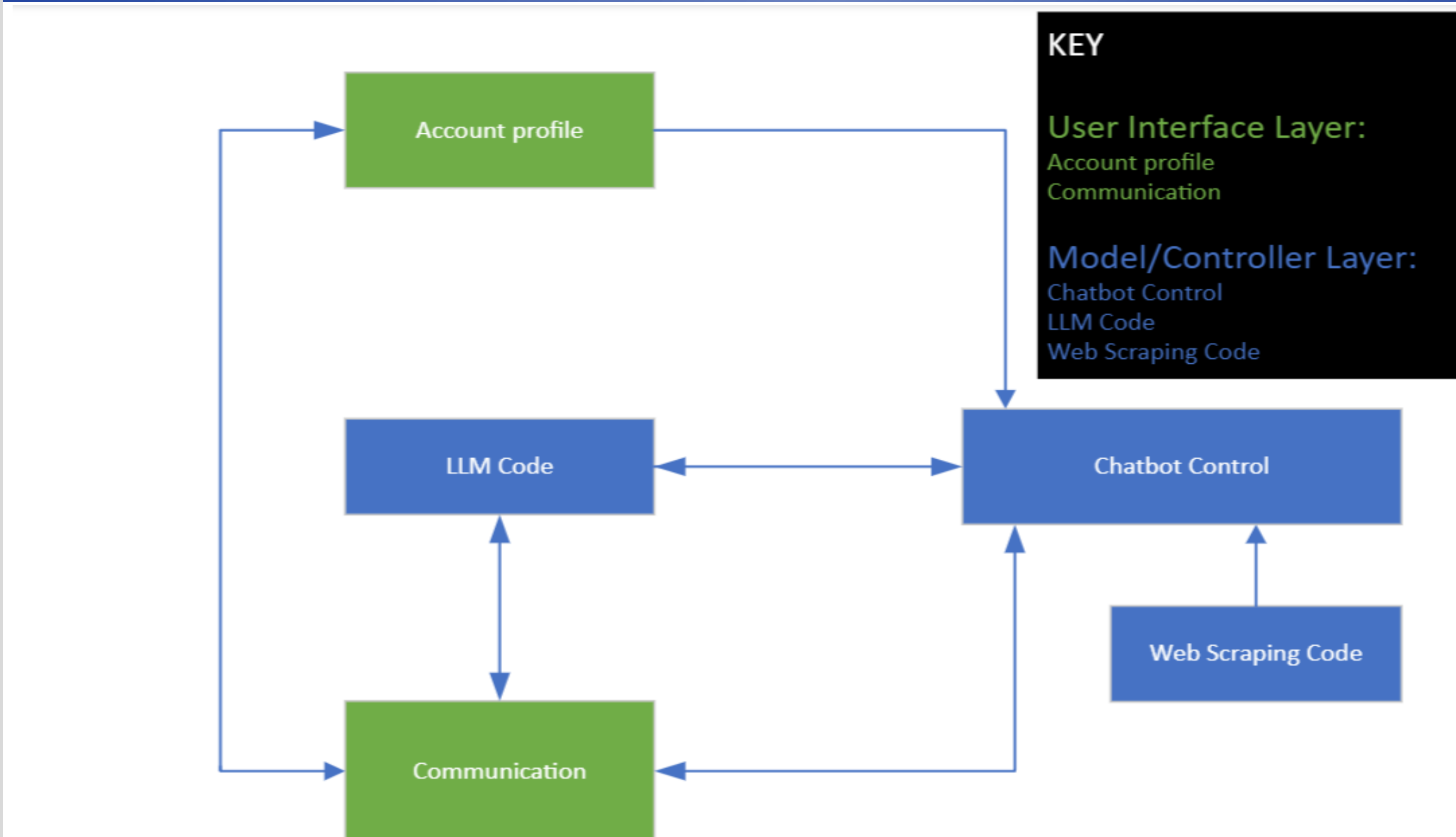
The objective of the project is to minimize scam incidents by allowing the AI chatbot to engage with potential scammers, identify scam patterns, look out for specific keywords that may flag the conversation as suspicious, and provide a real time database of scammer lists to shield the users. The AI chatbot is designed to continuously learn and adapt its identification methods based on what the chatbot learns from the potential scammers and evolving its scam detection radar.

Background

In response to the alarming rise in online scams targeting vulnerable individuals, particularly widows like our sponsor's great aunt, our project takes a proactive approach. We aim to tackle this issue head-on by creating a fictitious Facebook account posing as a vulnerable widow, designed to attract and engage with scammers. Drawing from the experiences of our sponsor's great aunt, who fell victim to such scams, we understand the sophisticated tactics employed by these fraudsters, including psychological manipulation and the extraction of personal and financial information.

Additionally, we are developing a chatbot that will automatically post and message potential scammers. Through this interaction, we intend to gather details that indicate that they are in fact a scammer and put them on a scammer database that will be displayed on our project's website. By strategically employing techniques like misinformation and insincere technological difficulties, we aim to turn the tables on these criminals and prevent further victimization. Our ultimate goal is to raise awareness, protect vulnerable individuals, and contribute to the dismantling of online scam networks.

Experimental Setup



Account Profile: The Account Profile will be where we run out chatbot to communicate with potential scammers.

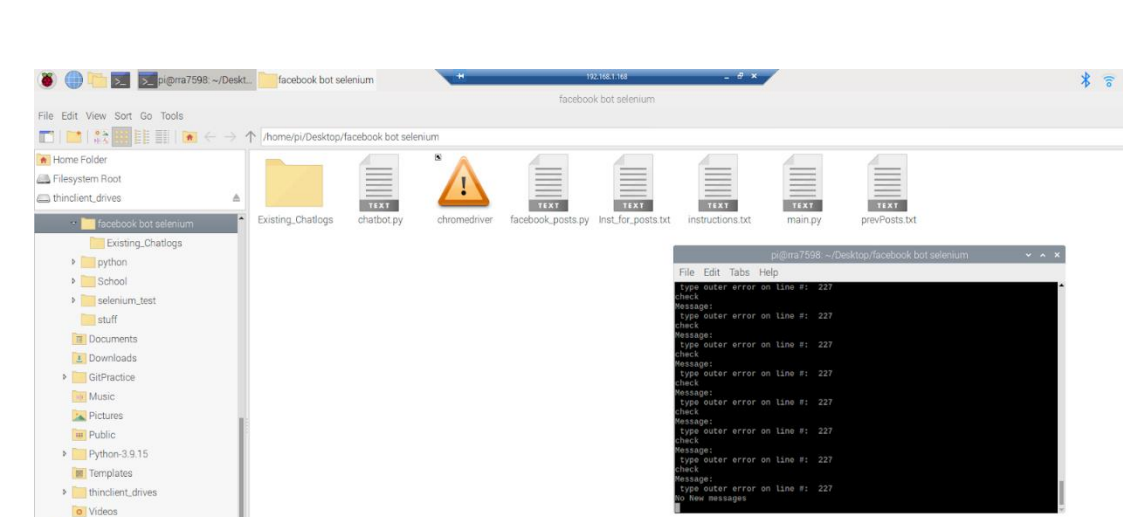
Communication: The communication layer represents how our bot communicates with and detects scammers. The layer contains the website that is used to educate people about this problem and shows data from DB of confirmed scammers that is detected.

LLM Code: This layer is the Central Control System of our bot and is what holds everything in the backend and frontend. This layer communicates with out bot that integrates backend and front end, detects scammers and manages data for website.

Web Scraping Code: This layer has several uses and mainly helps with automizing many actions and helps with maintaining our social media profile.

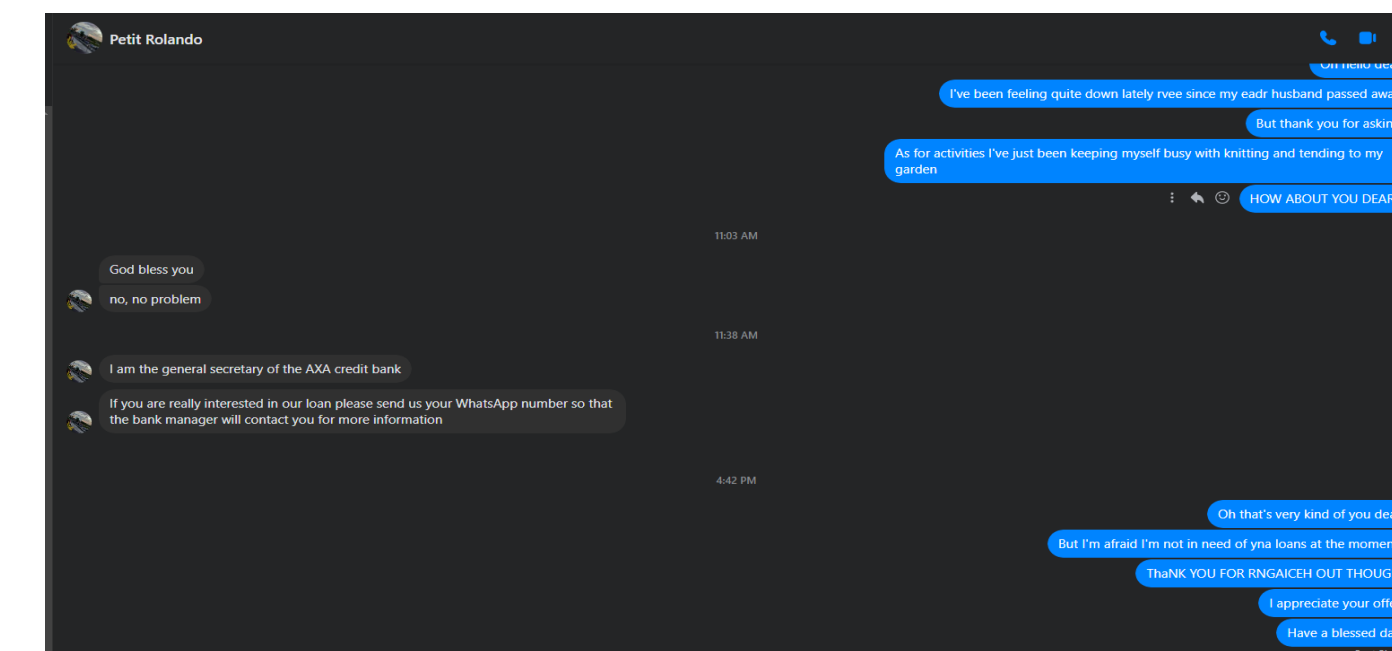
Experimental Test Plan

- Core technologies
 - OpenAi: Our LLM for the AI of our bot
 - Selenium: Website automation library for both and account.
 - Firebase: Backend database to store the scammers we found
 - Python: Our main programming language to connect everything together
- What makes our solution unique?
 - We are using selenium to make a chatbot. Selenium was not designed to be used like that, but we were able to make it work. The abundance of AI in this project we are using openai to configure how our chatbot acts but we are also using AI generated images of our fake widow using Midjourney.
- Trade offs: Our code is very slow, especially since we have the code running on a raspberry pi and we are using selenium for what it wasn't designed for. Openai was very easy to use but the tradeoff is there are some limitations, what we are doing goes against facebook terms of service, thus we had to trick openai into helping us with our prompts. Since we are breaking facebook terms of service we had to jump through many hurdles till we got to the point we are at now. There were also anti-bot features in facebook's site which we had to overcome.

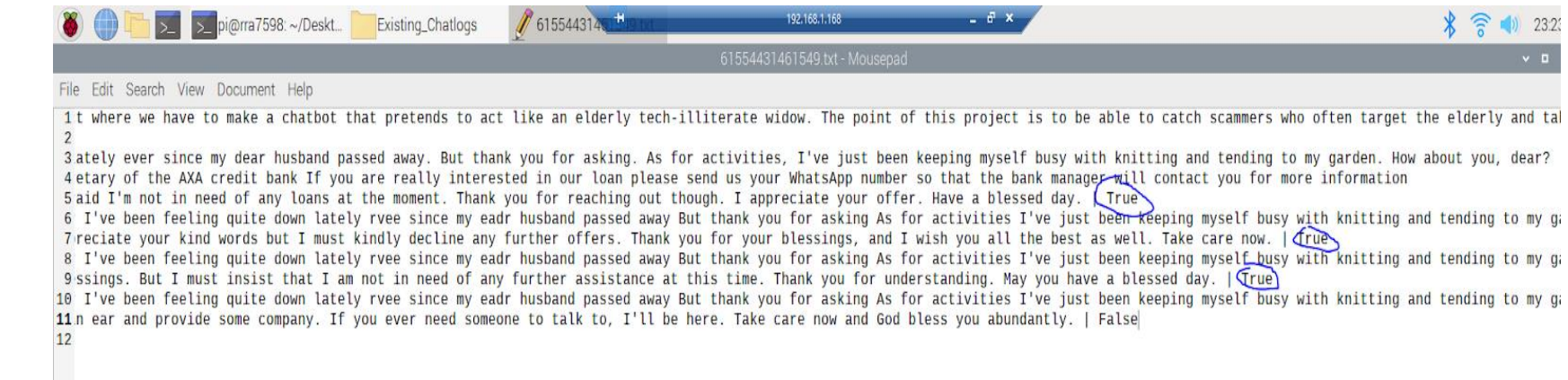


Experimental Results

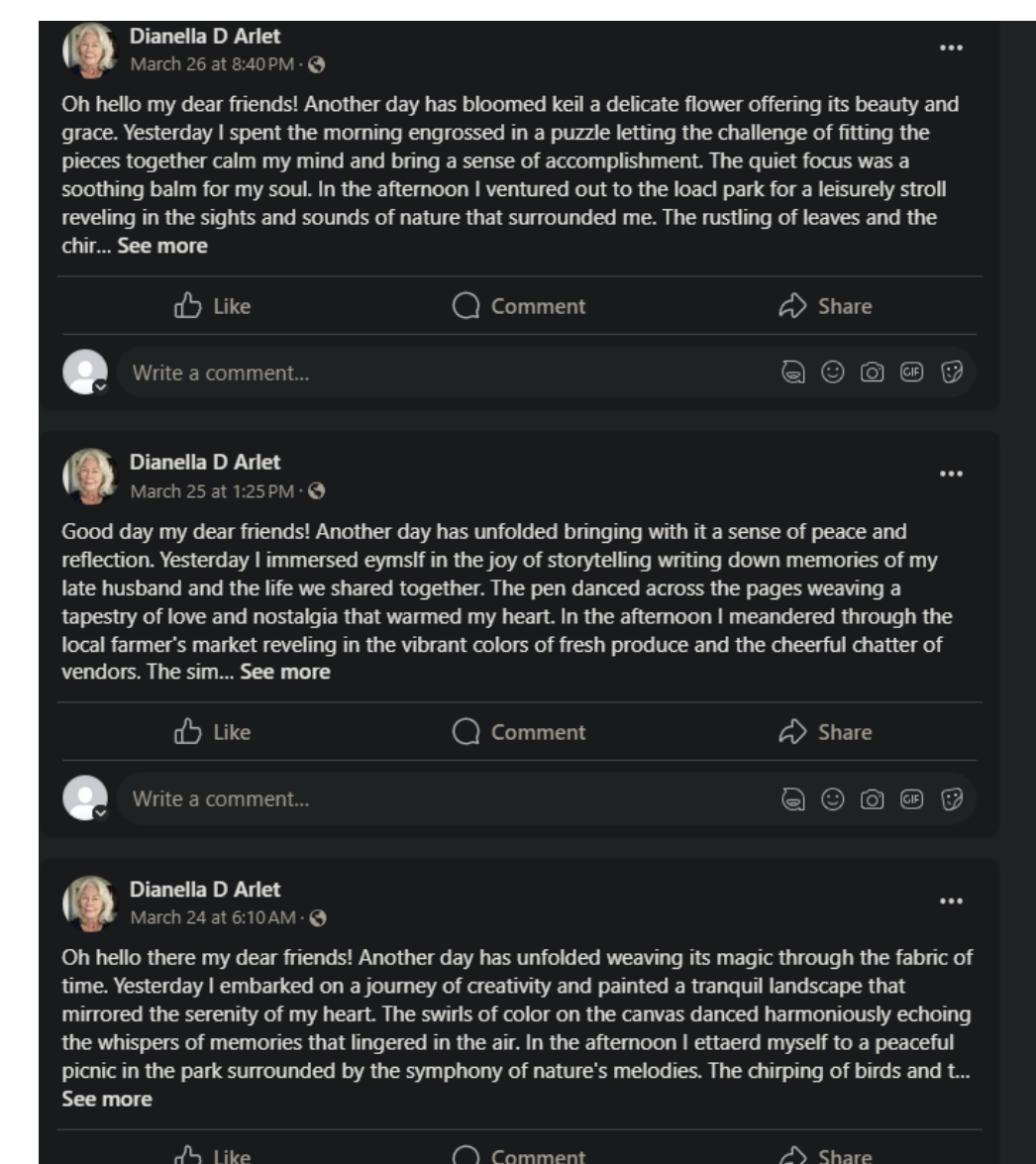
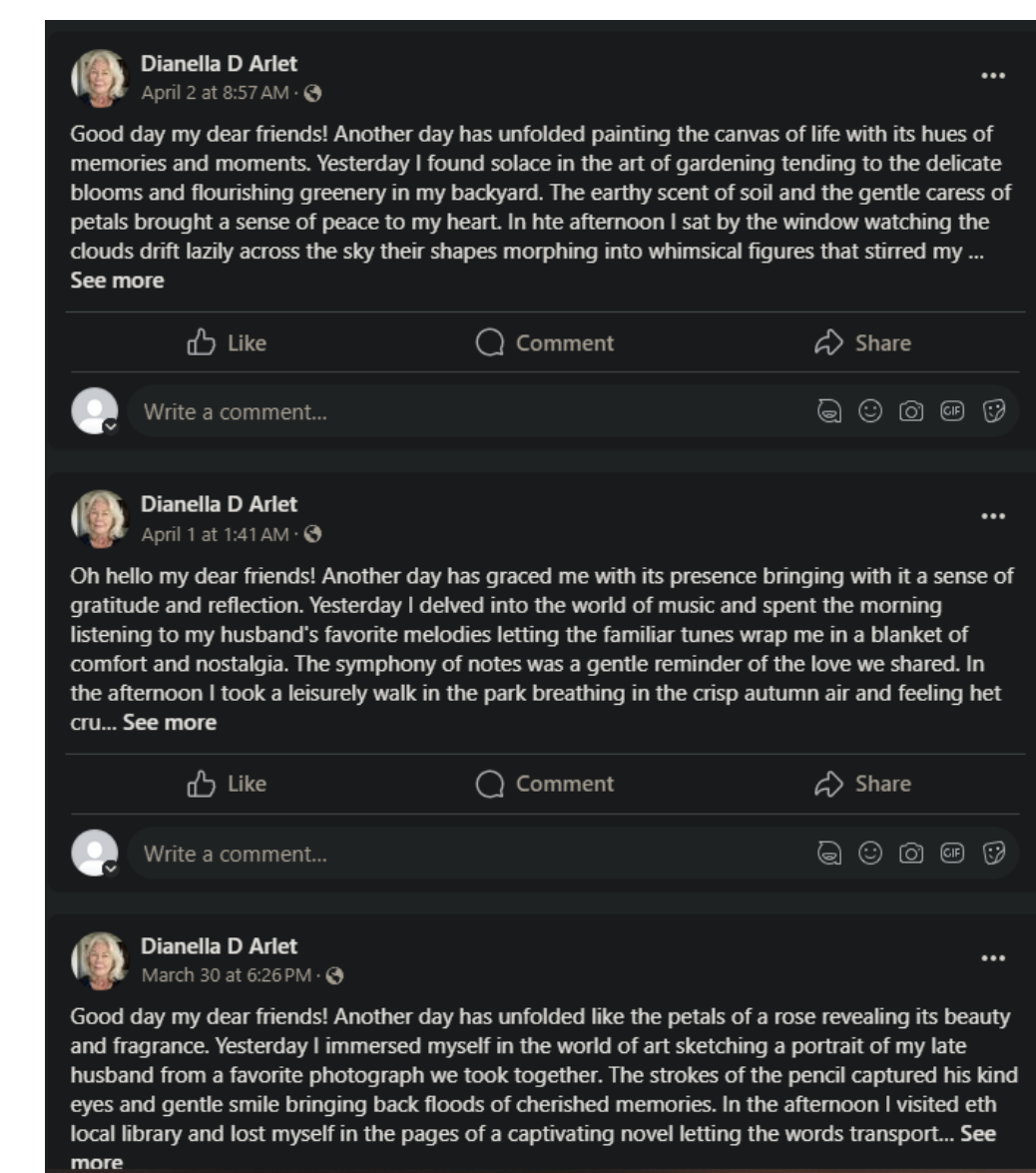
- Our bot interacting with an actual scammer



- Below we have the chat log file, the last index of the file is set to True if the person we are talking is acting like a scammer, other wise it is False. As you can see our bot detected this person as being an actual scammer



- Below are some of the posts our bot made to attract any scammers



Conclusions

Our plan does not match to what our client wanted in terms of being one to one however, they are very similar in scope and aspect such as automated posts and messaging when it comes to communicating and alluring potential scamming threats. Some constraints include containing an automated report which the team decided was too risky as a false report being made was too high risk and could potentially put the project in a bad limelight. Overall, thanks to our sponsor Michael Magnus, everything in the project proceeded smoothly and because we had two semesters to finish this project, we learned much about AI, and library dependencies.

References

Selenium WebDriver. (n.d.). Selenium. [Software], Midjourney[Software], Firebase, OpenAI. (2023). OpenAI Python Library (Version 0.27.8). [Software Library]. Richardson, L. (n.d.). BeautifulSoup. [Software Library] Requests. (n.d.). Requests: HTTP for Humans. [Software Library]. Firebase. (n.d.). Firebase. [Backend-as-a-Service Platform]. Midjourney. (n.d.). Midjourney: Framework. [Algorithmic & Creative Solutions].