CMP2204 Term Project

The project is about creating a Peer-to-peer chat Application using UDP and TCP ports with the steps having Service_Announcer, Peer_Discovery, Chat_Responder and Chat_Initiator.
The Application is developed in MacOS.
The pycryptodome library is used in this code and it needs to be installed before running the code.

The flow of the program:

1)Running Service_Announcer.py, Uses a UDP socket to get the local IP address. Just asks for the username when the code is run and stores it inside users.json with the IP address and time stamp.

2)Running Peer_Discovery.py, reads the users.json file and creates an array to store user statuses and checks user status every 8 seconds, displays the status of all the users that are stored and the last seen date if offline.

3)Running Chat_Responder.py, chat responder works as a server in this code, displays a message stating that the server is started and waiting for connections. Then waits for Chat_Initiator to be activated for forming any connections. When the connection is formed, works as the initial user and communicates with the user on the Chat_Initiator.py end, secured or unsecured according to the Selection of the second user. Also stores the chat information in a text file called chat_history.txt.

4)Running Chat_Initiator.py, chat initiator works as a client in this code, displays the users who are online and asks which user you would like to chat with, then it asks for users username(also gets the IP address) to form a connection with Chat_Responder.py. When these are set, it gives the user an option to securely chat or not by encryption. If the encryption is chosen, it encrpyts and then decrypts the messages on the chat using pycryptodome library. The messages sent from Chat_Initiator.py are also stored in the text file called chat_history.txt.

After all steps are applied, the peers are ready to chat anytime!