**Author**: Jeremy Vuong

**Course**: CSC 138-03

**Professor**: Jun Dai

**Date**: 04/08/2023

**Socket Programming Assignment 2 – Mail Client**

Text

Description automatically generated

Graphical user interface, text, application, email, Teams

Description automatically generated

Text

Description automatically generated

from socket import \*

msg = "\r\nI love computer networks!"

endmsg = "\r\n.\r\n"

# Choose a mail server (e.g. Google mail server) and call it mailserver

mailserver = "smtp.csus.edu"

mailserver\_port = 25

# Create socket called clientSocket and establish a TCP connection with mailserver

clientSocket = socket(AF\_INET, SOCK\_STREAM)

clientSocket.connect((mailserver, mailserver\_port))

recv = clientSocket.recv(1024)

print(recv)

if recv[:3] != b'220':

print('220 reply not received from server.')

# Send HELO command and print server response.

helloCommand = 'HELO Alice\r\n'

clientSocket.send(helloCommand.encode())

recv1 = clientSocket.recv(1024)

print(recv1)

if recv1[:3] != b'250':

print('250 reply not received from server.')

# Send MAIL FROM command and print server response.

mailFromCommand = 'MAIL FROM: <jeremyvuong@csus.edu>\r\n'

clientSocket.send(mailFromCommand.encode())

recv2 = clientSocket.recv(1024)

print(recv2)

if recv2[:3] != b'250':

print('250 reply not received from server.')

# Send RCPT TO command and print server response.

rcptToCommand = 'RCPT TO: <jeremyvuong@csus.edu>\r\n'

clientSocket.send(rcptToCommand.encode())

recv3 = clientSocket.recv(1024)

print(recv3)

if recv3[:3] != b'250':

print('250 reply not received from server.')

# Send DATA command and print server response.

dataCommand = 'DATA\r\n'

clientSocket.send(dataCommand.encode())

recv4 = clientSocket.recv(1024)

print(recv4)

if recv4[:3] != b'354':

print('354 reply not received from server.')

# Send message data.

clientSocket.send(msg.encode())

# Message ends with a single period.

clientSocket.send(endmsg.encode())

recv5 = clientSocket.recv(1024)

print(recv5)

if recv5[:3] != b'250':

print('250 reply not received from server.')

# Send QUIT command and get server response.

quitCommand = 'QUIT\r\n'

clientSocket.send(quitCommand.encode())

recv6 = clientSocket.recv(1024)

print(recv6)

if recv6[:3] != b'221':

print('221 reply not received from server.')

# Close the client socket

clientSocket.close()