
Secure Fast Chat: Client Program

Khushang Singla \and Mridul Agarwal \and Arhaan Ahmad

Oct 27, 2022

CONTENTS:

1	Client_Side	1
1.1	Message module	1
1.2	app module	2
2	Indices and tables	5
	Python Module Index	7
	Index	9

CLIENT_SIDE

1.1 Message module

class Message.**Message** (*conn_socket, task, request*)

Bases: object

This is the class to handle Encryption of messages. The format in which the message is sent to server is determined in this class

Parameters

- **task** (*str*) – Task to be done. It can have the values signup, login, send_message
- **socket** (*socket.socket*) – The socket used for connection with Server
- **request_content** (*dict*) – Content to include in the request to send to server
- **_data_to_send** (*bytes*) – Contains the data to send to the server
- **_recvd_msg** (*bytes*) – Content recieved from server is stored here

Constructor Object

Parameters

- **conn_socket** (*socket.socket*) – Socket which has a connection with server
- **task** (*str*) – Task to do. It can have values: login, signup, send_message
- **request** (*str*) – Content to send to server

_send_data_to_server ()

Function to send the string to the server. It sends content of _send_data_to_server to the server

_recv_data_from_server (*size*)

Function to recv data from server. Stores the bytes recieved in a variable named _recvd_msg.

Parameters size (*int*) – Length of content to recieve from server

_json_encode (*obj, encoding*)

Function to encode dictionary to bytes object

Parameters

- **obj** (*dict*) – dictionary to encode
- **encoding** (*str*) – Encoding to use

Returns Encoded obj

Return type bytes

`_hash_password (passwd)`

Function to salt and hash the password before sending to server

Parameters `passwd (str)` – Password to be hashed

Returns Transformed Password

Return type string

`_create_login_request ()`

The jsonheader has the following keys: | byteorder, request, content-length, content-encoding. The value for request is 'login' | The content has user id and password separated using "

Returns Message to send to server directly for login

Return type bytes

`_create_signup_request ()`

The jsonheader has the following keys: | byteorder, request, content-length, content-encoding. The value for request is 'signup' | The content has user id and password separated using "

Returns Message to send to server directly for login

Return type bytes

`_login ()`

Function to help login into the system. This function sends the login details to the server | The function expects to receive a response of size 2 from server which gives 0 if invalid id/password and 1 if successful login and 2 for any other case

Returns Response from server converted to int

Return type int

`_signup ()`

Function to help signup to make new account. This function sends the new user details to the server | The function expects to receive a response of size 2 from server which gives 0 if username already taken and 1 if successful login and 2 for any other case

Returns Response from server converted to int

Return type int

`processTask ()`

Processes the task to do

Returns Returns int to represent result of the process. The details of return values are given in the corresponding functions handling the actions.

Return type int

1.2 app module

`app.login (sock=None)`

Function to help user log in to the app

Returns Socket with which user is connected to server

Return type socket.socket

`app.signup (sock=None)`

Function to help user make new account

Returns Socket with which user is connected to server

Return type socket.socket

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

a

app, [2](#)

m

Message, [1](#)

Symbols

`_create_login_request()` (*Message.Message method*), 2
`_create_signup_request()` (*Message.Message method*), 2
`_hash_password()` (*Message.Message method*), 1
`_json_encode()` (*Message.Message method*), 1
`_login()` (*Message.Message method*), 2
`_recv_data_from_server()` (*Message.Message method*), 1
`_send_data_to_server()` (*Message.Message method*), 1
`_signup()` (*Message.Message method*), 2

A

`app`
 module, 2

L

`login()` (*in module app*), 2

M

`Message`
 module, 1
`Message` (*class in Message*), 1
`module`
 `app`, 2
 `Message`, 1

P

`processTask()` (*Message.Message method*), 2

S

`signup()` (*in module app*), 2