

The Title of the Project: **SyncSphere**

This project is one of my notable projects.

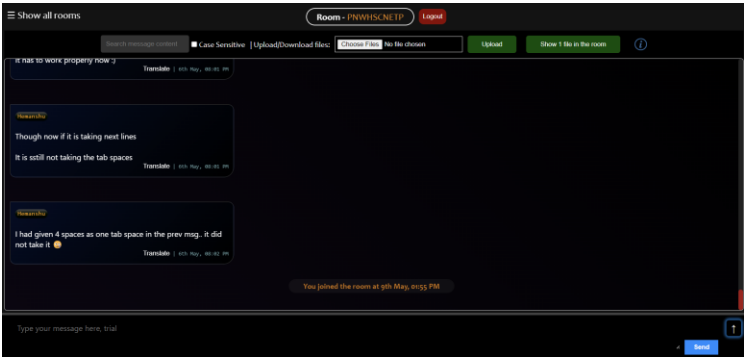
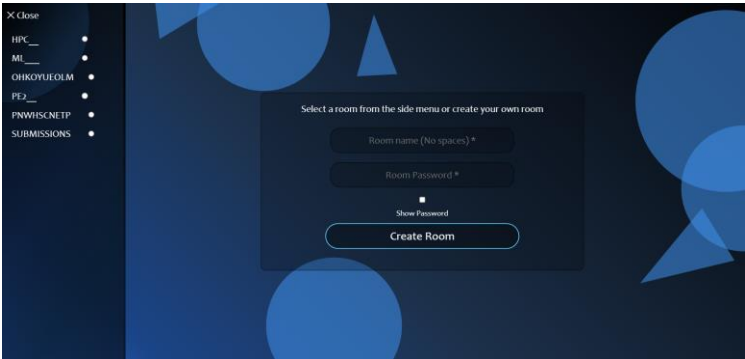
SyncSphere is a LAN-hosted chat application designed to enhance privacy and communication within an organization. This project reflects my skills in Python programming, cybersecurity, networking, and web development. This PDF contains the images from the current version of the project dated: 09/05/2024.

The project is undergoing changes to polish the feel to use, the security and to eliminate bugs from the project, addition of further features is also a work in progress. The functionality won't be changing.



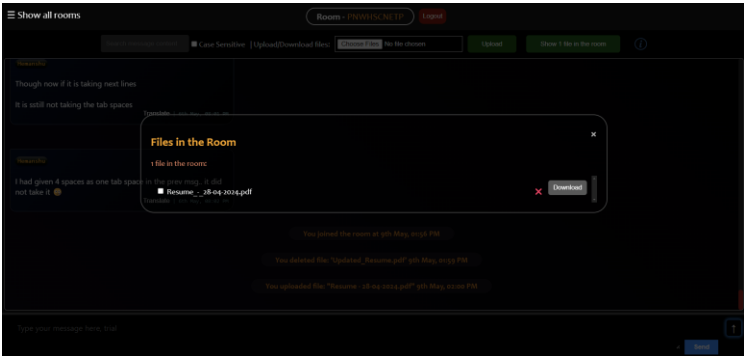
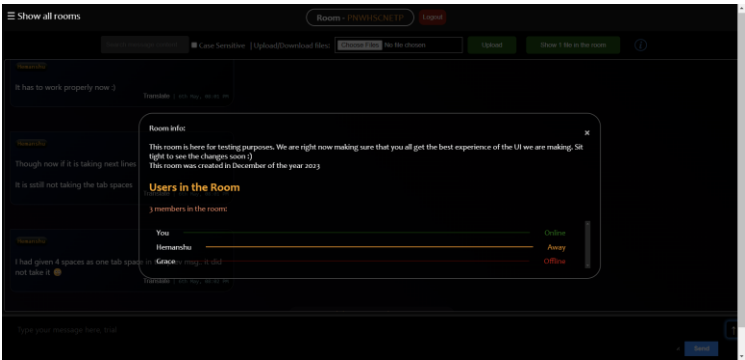
Landing Page of the website

Room Selector

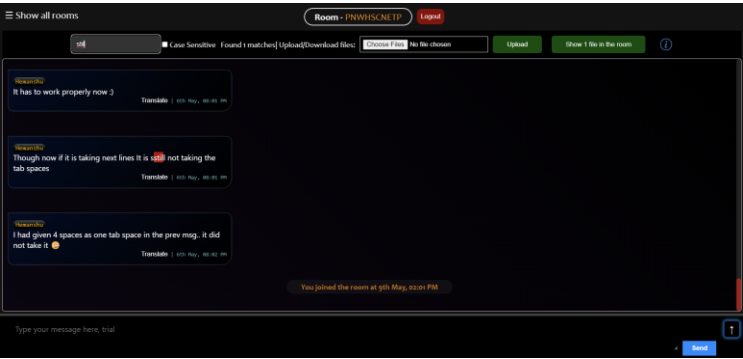


Inside Room

Room Information Modal

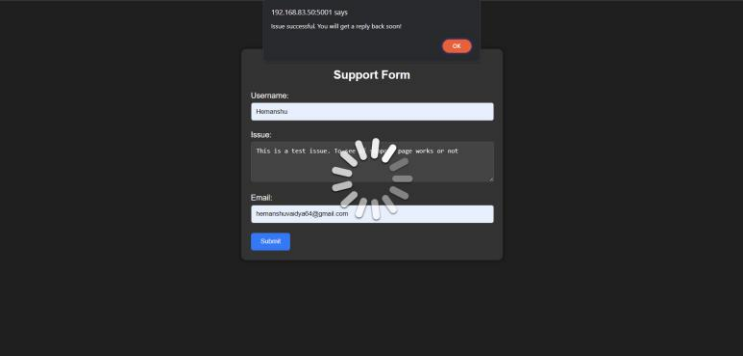
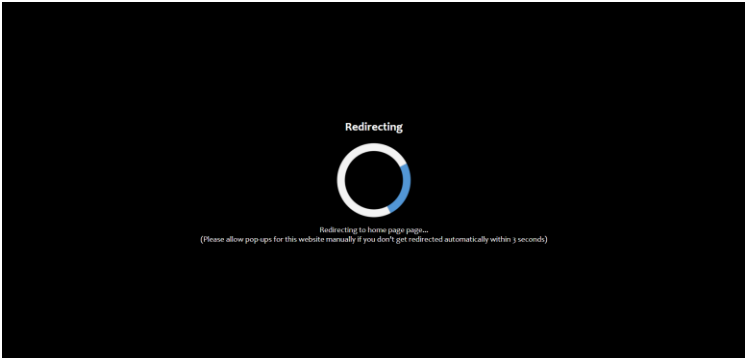


Files Modal



Message Searching


Redirections




Support Form

Regarding your latest issue report Inbox x

User wanted some help! External Inbox x

 noreply.lanchatting@gmail.com
to me ▾

 noreply.lanchatting@gmail.com
to me ▾

Hello SyncSphere user 'Hemanshu',
We have reported your issue
=====

This is a test issue. To see if support page works or not
=====

Our developers are looking into the issue and will revert back on this email soon.
We are extremely sorry for any inconvenience caused.
Keep this report ticket handy:
MAGHYVERZA
Regards,
SyncSphere.

Hello SyncSphere Developers!
The user 'Hemanshu' needs your help. They have this issue with our system:
=====

This is a test issue. To see if support page works or not
=====

They have left their email: hemanshuvaidya64@gmail.com .

Support Email to developers
(on the right) and to the user
related to (on the left)

```

111 ##### SELECT ROOM HANDLER #####
112 @app.route('/rooms')
113 > def chat_room():...
124
125 @app.route('/password_checker_room', methods=['POST'])
126 > def check_r_pass():...
135
136 > def send_join_event(data, flag=0):...
171
172 ##### SELECT ROOM HANDLER #####
173
174
175 ##### USER ACTIVITY STATUS #####
176
177 @app.route('/get_partis', methods=['POST'])
178 > def get_partis():...
197
198 @app.route('/back_online', methods=['POST'])
199 > def set_online():...
207
208 > def set_offline(username, room):...
215
216 @app.route('/heartbeat', methods=['POST'])
217 > def handle_heartbeat():...
229
230 @app.route('/going_away', methods=['POST'])
231 > def handle_away():...
241
242 ##### USER ACTIVITY STATUS #####
243
244 ##### INSIDE ROOM HANDLER #####
245 @app.route("/room/<room_>", methods=['POST', 'GET'])
246 > def room_(room):...
269
270 @socketio.on('redirect_to_room')
271 > def redirect_to_room(data):...
277

```

Sample code for the server created using Flask Framework in Python

This work is also published on IJIRCCE (A journal where we had published our paper). The link for the same can be opened with this hyperlink:

[Click here for Published Research Paper](#)
[Click here for Publication Certificate](#)