

# CLICORD : Chat App in GOLANG

---

## Team Members :

---

- P K Navin Shrinivas [PES2UG20CS237]
- Mohamed Ayaan [PES2UG20CS200]
- Mukund Deepak [PES2UG20CS206]

## Abstract of project

---

The aim was to build a entire CLI focused chat app using golang, but not just any simple chat app. We implemented the following feature on top of basic chat system :

- Emojis support (Yes, emojis in terminal!)
- File transfer
- Lobby based chat system
- Stateless websockets system

We surely wanted to do more on this project such as :

- End to end encryption
- TUI [We had quit nasty bugs on CLI ]
- Work out an authentication system for one on one chatting

- Ofc a fast as hell backend for storing messages.

But given the time frame of just ~5hrs for this project, all the following could not be done.

## Output screenshots :

---

### Server :

We have a global server hosting out server program, using Dynamic DNS and Little port forwarding we made our server accessible from [catss.me:8080](https://catss.me:8080) [its just a laptop so its down most of the time ], anyways, the screenshot :

```
serveruser@serveruser-machine:~/CLICORD/server
    active_conn  map[*websocket.Conn]int
[screen is terminating]
[serveruser@serveruser-machine server]$ screen -r
There is no screen to be resumed.
[serveruser@serveruser-machine server]$ screen -r
There is no screen to be resumed.
[serveruser@serveruser-machine server]$ sudo go run *.go
2022/04/13 18:26:21 Starting server...
2022/04/13 18:26:21 listen tcp 0.0.0.0:8080: bind: address already in use
exit status 1
[serveruser@serveruser-machine server]$ ls
go.mod go.sum lobby_funcs.go ping_funcs.go server server.go
[serveruser@serveruser-machine server]$ nvim server.go
[serveruser@serveruser-machine server]$ sudo go run *.go
[sudo] password for serveruser:
2022/04/13 18:31:48 Starting server...
{"Msg_content":"hello 🐱:happy:", "File_content":null, "Username":"","Is_file":false, "File_name":"","File_type":""}
2022/04/13 18:32:51 Read error : websocket: close 1006 (abnormal closure): unexpected EOF
```

### Client :

```
go
1: go x +
user2 : hello 😊:happy:
user1 : [2] server.png
Enter message to send : /file ../server.png
```

```
go
1: go x +
user2 : hello 😊:happy:
user1 : [2] server.png
Enter message to send : /store 2
Downloading file 2
File written to home dir!!
Enter message to send :
```

## The project :

---

We have this project up on github and open for contributions at all times : [here](#)

As for this submission, a modified version [only the port and server link] has been submitted in zip files such that it can be run on localhost.