Simplified version of an online HelpDesk

Project for course: Computer networks laboratory

Course code:15Z510

Simplified implementation of an online HelpDesk System.

INTRODUCTION

Online HelpDesk System nowadays are much faster and efficeint. Most of them are equiped with automated replies, but we humans still prefer a person over a machine.

Online helpDesk systems can be implemented via Websockets whichfacilitates the communication between the client and the server whithout the need to refresh the page, i.e. real time application.

- So how different is it from a regular chat Application?

  Here, we have multiple clients and multiple server side users.

  When a client sends a request for a query session, the program checks for availability of server side useers.These users are assigned randomly based on their availability at that particular time. If all users are occupied with the request of the other clients, it will just display that its not possible to assign a user to that client and asks the client to try after a while.

PACKAGES USED

SOCKETIO – Handles all socket connections

FLASK – light web application Framework; handles the backend

SQLAlchemy – converts to RAW SQL commands

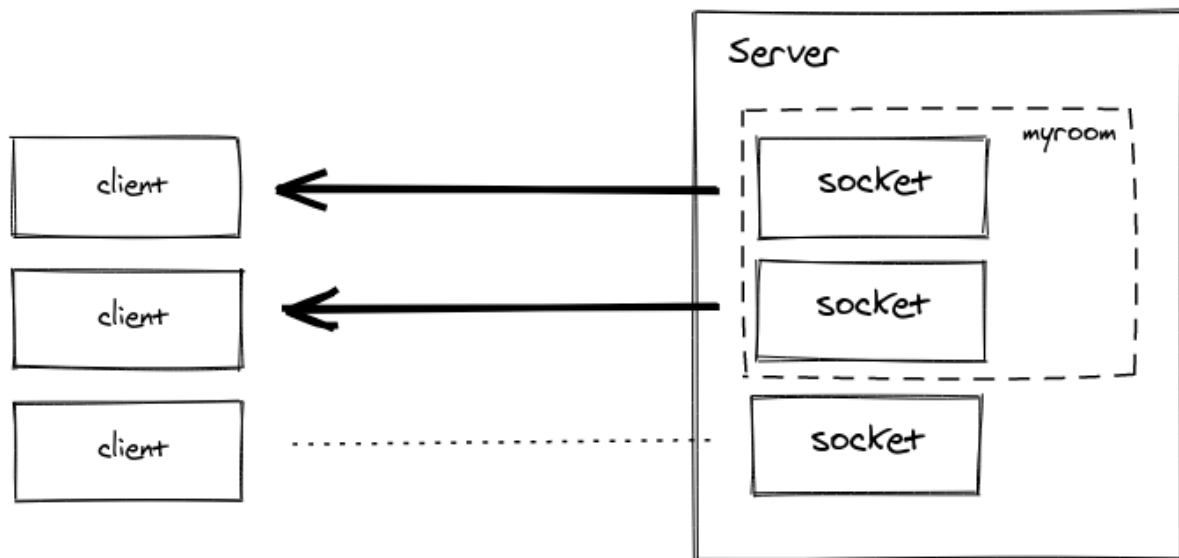HTML ,CSS , JS – Front End

PostgreSQL -Database

FUNCTIONALITIES

The functionalities provided in the actual application is as follows:

- Multiple server accounts/Instances
- Multiple customer
- Private chat based on Room concept in SocketIO
- Concept of Availability is invoked
- Ignore request functionality available for the users from the server side

ADVANTAGES

- No logging of data
- Close to instant pairing
- Not much user data is collected
- Can be extended to Discussion forums that can hold upto 250

Server

myroom

client

socket

client

socket

client

socket

Flask SQLAlchemy

socket.io

HTML 5

JS 5

CSS 3

Folder Structure

HELPDESK/

    - __pychache__/

    - static/

    - templates/

        -index.html

        -chat.html

        -server.html

        -serverchat.html

   - app.py

   - Pipfile

   - Pipfile.lock

   - config.cfg

Screenshots:

TECH SUPPORT - HELPDESK CLIENT REQUEST PANEL

SEND CLIENT REQUESTS

CLIENT 1	CLIENT 2	CLIENT 3	CLIENT 4
RoomID:1234	RoomID:2345	RoomID:3456	RoomID:7890

TECH SUPPORT - HELPDESK SOLUTION PANEL

ACCEPT/IGNORE CLIENT REQUESTS

SERVER REQUEST	SERVER REQUEST	SERVER REQUEST	SERVER REQUEST
Candidate: mani	Candidate: nagul	Candidate: rohith	Candidate: arun
No Client Assigned	No Client Assigned	No Client Assigned	No Client Assigned

ACCEPT/IGNORE CLIENT REQUESTS

SERVER REQUEST	SERVER REQUEST	SERVER REQUEST	SERVER REQUEST
Candidate: mani	Candidate: nagul	Candidate: rohith	Candidate: arun
No Client Assigned	No Client Assigned	No Client Assigned	💬
			IGNORE

SERVER REQUEST

Candidate: arun

💬

IGNORE

CLIENT

CHAT ROOM

<Chat Active!>
SERVER: Hello sir, how may I help you?
CLIENT: Hello, I have this query regarding a particular product
SERVER: Can you enter the product id, sir?
CLIENT: #437129078
SERVER: We will be right back with you sir.

Enter Here...

Terminate

SERVER

CHAT ROOM

<Chat Active!>
SERVER: Hello sir, how may I help you?
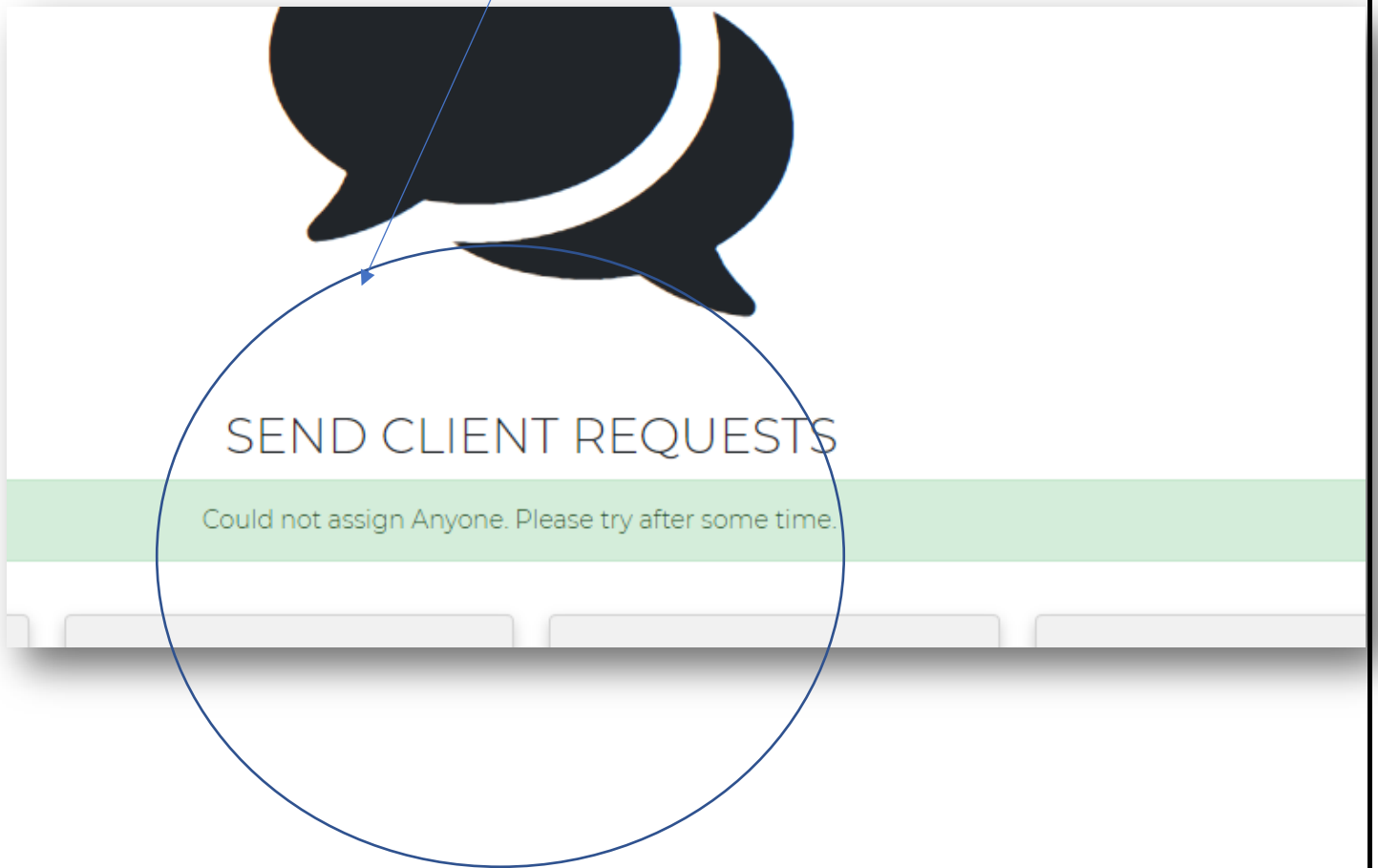CLIENT: Hello, I have this query regarding a particular product
SERVER: Can you enter the product id, sir?
CLIENT: #437129078
SERVER: We will be right back with you sir.

Enter Here...

Terminate

SEND CLIENT REQUESTS

Could not assign Anyone. Please try after some time.

Source code:

app.py: