const express = require('express');

const app = express();

const http = require('http');

const server = http.createServer(app);

const { Server } = require("socket.io");

const io = new Server(server);

var count =0;

io.on('connection',  function(socket) {

  console.log('a user connected');

  count++;

  io.emit('usercnt',count);

  io.emit('user\_join', function()

  {

  });

  socket.on('disconnect',function()

  {

    console.log("user disconnected")

    count--;

    io.emit('usercnt',count);

  })

  socket.on('connection', (socket) => {

    socket.on('chat message', (msg) => {

      console.log('message: ' + msg);

  });

  });

})

app.get('/', (req, res) => {

  res.sendFile(\_\_dirname + '/index.html');

});

server.listen(3000, () => {

  console.log('listening on \*:3000');

});

<!DOCTYPE html>

<html>

    <head>

        <title>Real-time Chat App</title>

        <meta name="viewport" content="width=device-width,minimum-scale=1,initial-scale=1">

        <style>

            body {

                margin: 0;

                font-family: sans-serif;

            }

            form {

                position: fixed;

                bottom: 0;

                left: 0;

                width: 100%;

                display: flex;

                box-sizing: border-box;

                padding: 0.25rem;

            }

            form input {

                border: 0;

                padding: 0.5rem;

                width: 100%;

                outline: 0;

                margin-right: 0.5rem;

                border-radius: 0.25rem;

                background: #ccc;

            }

            form button {

                width: 6rem;

                background-color: #1b8c00;

                color: white;

                border: none;

                padding: 0.5rem;

                cursor: pointer;

                border-radius: 0.25rem;

                text-transform: uppercase;

            }

            form button:hover {

                background-color: #166d01;

            }

            .messages {

                margin: 0;

                padding: 0;

                margin-bottom: 3rem;

            }

            .messages li {

                padding: 0.5rem;

            }

            .messages li:nth-child(odd) {

                background: #eee;

            }

        </style>

    </head>

    <body>

        <ul class="messages"></ul>

        <form>

            <input type="text" class="input" autocomplete="off" autofocus />

            <button>Send</button>

        </form>

        <script src="/socket.io/socket.io.js"></script>

        <script>

            const form = document.querySelector("form");

            const input = document.querySelector(".input");

            const messages = document.querySelector(".messages");

            const username = prompt("Please enter a nickname: ", "");

            const socket = io();

            socket.on('usercnt', function(msg)

            {

              console.log("updated")

              document.getElementById("count").innerHTML=msg;

            })

            form.addEventListener("submit", function(event) {

                event.preventDefault();

                addMessage(username + ": " + input.value);

                socket.emit("chat\_message", {

                    message: input.value

                });

                 input.value = "";

                return false;

            }, false);

            socket.on("chat\_message", function(data) {

                addMessage(data.username + ": " + data.message);

            });

            addMessage("You have joined the chat as '" + username  + "'.");

            // socket.emit("user\_join", username);

            function addMessage(message) {

                const li = document.createElement("li");

                li.innerHTML = message;

                messages.appendChild(li);

                window.scrollTo(0, document.body.scrollHeight);

            }

        </script>

  <span id="count"> 0 </span> Users online

  </body>

</html>