|  |
| --- |
| import { Injectable } from '@angular/core'; |
|  | import { HubConnection } from '@aspnet/signalr'; |
|  | import \* as signalR from '@aspnet/signalr'; |
|  |  |
|  | @Injectable() |
|  | export class HubConnectionService { |
|  |  |
|  | msg: string; |
|  | unread: number = 1; |
|  | text: string; |
|  | messages: string[] = []; |
|  | // messages: string; |
|  | private \_hubConnection: HubConnection; |
|  |  |
|  | constructor() { |
|  | this.init(); |
|  | } |
|  |  |
|  | setconectid(sender) { |
|  | this.\_hubConnection.invoke('setconnectid', sender); |
|  | } |
|  |  |
|  | setstatus(sender) { |
|  | this.\_hubConnection.invoke('setstatus', sender); |
|  | } |
|  |  |
|  | senddirectmsg(recevierid, senderid, message, sender) { |
|  | this.\_hubConnection |
|  | .invoke('Send', recevierid, senderid, message, sender); |
|  | return this.messages; |
|  | } |
|  |  |
|  |  |
|  | private init() { |
|  |  |
|  |  |
|  | this.\_hubConnection = new signalR.HubConnectionBuilder() |
|  | .withUrl('http://localhost:5000/chat') |
|  | .configureLogging(signalR.LogLevel.Information).build(); |
|  |  |
|  | this.\_hubConnection |
|  | .start() |
|  | .then(() => console.log('Connection started!')) |
|  | .catch(err => console.log('Error while establishing connection :')); |
|  |  |
|  | this.\_hubConnection.on('setconnectid', (sender: string) => { }); |
|  |  |
|  | this.\_hubConnection.on('setstatus', (sender: string) => { }); |
|  |  |
|  | this.\_hubConnection.on('send', (receivedMessage: string, sender: string) => { |
|  | this.text = `${sender}:${receivedMessage}`; |
|  | this.messages.push(this.text); |
|  | }); |
|  | } |
|  | } |