Table of Contents

[Basics 2](#_Toc118581839)

[Current Resources with Changes 2](#_Toc118581840)

[New Resources 2](#_Toc118581841)

[npm install / ng serve 2](#_Toc118581842)

[/core/services/signalr 2](#_Toc118581843)

[Signalr-group.interface.ts 2](#_Toc118581844)

[presence-hub.service.ts 2](#_Toc118581845)

[createHubConnection 2](#_Toc118581846)

[message-hub.service.ts 3](#_Toc118581847)

[createHubConnection 3](#_Toc118581848)

[Components 3](#_Toc118581849)

[/site/members/member-detail 3](#_Toc118581850)

[member-detail.component.ts 3](#_Toc118581851)

[/site/members/member-messages 3](#_Toc118581852)

[member-messages.component.html 3](#_Toc118581853)

[member-messages.component.css 3](#_Toc118581854)

# Basics

|  |  |
| --- | --- |
| WorkingFolder | Here still working with “Site-06-SignalR” and issue   * ng serve |

# Current Resources with Changes

1. /site/members/member-messages
2. /core/services/signalr/presence-hub.service
3. /core/services/signalr/message-hub.service.ts
4. /site/members/member-detail

# New Resources

1. /core/services/signalr/signalr-group.interface

# npm install / ng serve

Follow basics and then run “npm install” and “ng serve” commands.

# /core/services/signalr

## Signalr-group.interface.ts

create two interfaces in the signalr-group.interface.ts file

export interface SignalRGroup{

  groupName: string;

  connections: SignalRConnection[];

}

export interface SignalRConnection{

  connectionId: string;

  userName: string;

  userId: number;

}

## presence-hub.service.ts

### createHubConnection

Inject router via the constructor and then add a new notification for NewMessageReceived

    //message notification when the user is online but not on the same page

    //also provide a click to go to the messages

    this.hubConnection.on('NewMessageReceived', (user: userDto) => {

      this.toastrService.info(`${user.displayName} has sent you a new message!`)

        .onTap

        .pipe(take(1))

        .subscribe(() => {

          this.router.navigateByUrl(`/members/detail/${user.guId}/${user.userName}?tab=3`)

        })

        ;

    })

For UserIsOnline fire the current user that has gone online

    //listen for events UserIsOnline and UserIsOffline. both return userName

    this.hubConnection.on('UserIsOnline', userName => {

      this.toastrService.info(userName + ' has connected');

      //remove the user from the usernames

      this.onlineUsers$.pipe(take(1)).subscribe(usernames => {

        this.onlineUsersSource.next([...usernames, userName]);

      });

    });

For the UserIsOffline remove the user from the list and fire

    this.hubConnection.on('UserIsOffline', userName => {

      this.toastrService.warning(userName + ' has disconnected');

      //remove the user from online users and fire

      this, this.onlineUsers$.pipe(take(1)).subscribe(usernames => {

        this.onlineUsersSource.next([...usernames.filter(x => x !== userName)]);

      });

    });

## message-hub.service.ts

### createHubConnection

Add the UpdatedGroup event

    this.hubConnection.on("UpdatedGroup", (group: SignalRGroup) => {

      if (group.connections.some(x => x.userName == otherUserName)) {

        this.messageThread$.pipe(take(1)).subscribe(messages => {

          if (!messages) return;

          messages.forEach(message => {

            if (!message.dateMessageRead)

              message.dateMessageRead = new Date(Date.now());

          });

          this.messageThreadSource.next([...messages]);

        });

      }

    });

# Components

## /site/members/member-detail

### member-detail.component.ts

Add the following line in the constructor.

    //do not use routeReusestrategy. When the user is on detail page for one user and received a new message toast for user two then

    //the page will not refresh the content

    this.router.routeReuseStrategy.shouldReuseRoute = () => false;

## /site/members/member-messages

### member-messages.component.html

Apply the class to display the text in a different color that belongs to the logged in user

<p [ngClass]="{currentUser: message.senderId !== userId}">{{ message.messageContent }}</p>

### member-messages.component.css

add the style

.chat li p.currentUser{

  color: #FF8C00;

}