

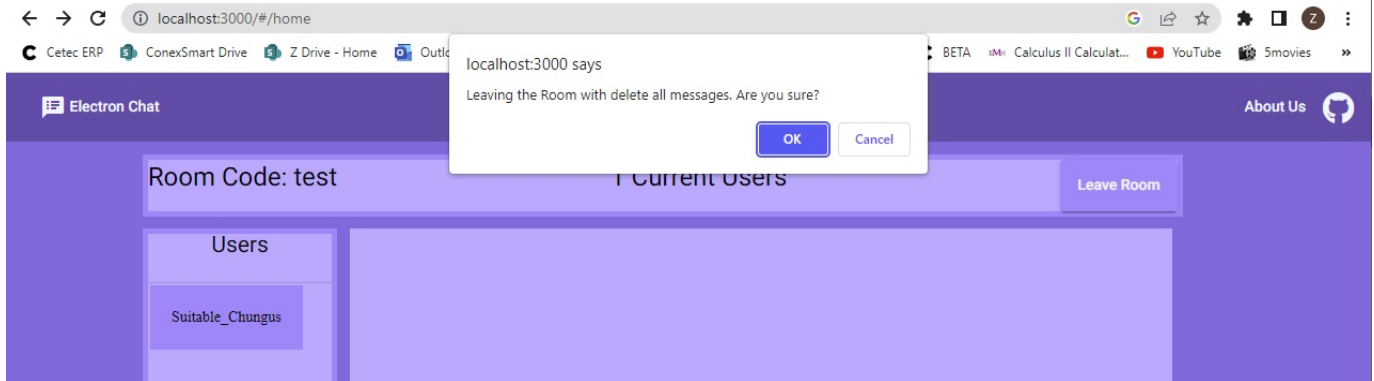
Annotation 12

Title: Are You Sure Message

Description: This artifact is a collection of screenshots of source code that displays a pop up "Are you sure?" message when the user attempts to leave an ElectronChat room. The artifact also contains a screenshot of the pop up message itself. The source code responsible for the pop up message was developed by me. It was important that we developed a pop up message warning users that the messages are not persistent and are lost when exiting the room. Messages being lost when the user leaves the room is a key feature of the privacy chat, but we didn't want users to be punished for not knowing how the app operates. This exercise is good example of developing for the stakeholder and creating a user friendly environment.

Reflection: As a result of this exercise I gained insight into the users mindset, taking into consideration possible nuances that can lead to unusable environments. I also found the Angular CanDeactivate utilized to create pop up very extensive and designed to protect users from activating routes without intent. This is very important when key data is involved, in my case the ElectronChat app is designed to lose the messages when exiting the room, but most users are used to persistent messaging applications which can result in accidental misuse. This is very common in cases where users have to fill out forms with data and the page is accidentally refreshed mid way, this can result in irritating user experience that can be avoided by implementing guards.

Artifact 12



```
103     /**
104      * Confirms if user wants to leave room
105      * @returns {string} window close confirmation */
106     canExit() {
107         return confirm('Leaving the Room with delete all messages. Are you sure?')
108     }
```

```
5     @Injectable({
6         providedIn: 'root'
7     })
8     export class CanDeactivateGuard implements CanDeactivate<RoomComponent> {
9
10         /**
11          *
12          * @param component
13          * @param currentRoute
14          * @param currentState
15          * @param nextState
16          * @returns check if component can exit
17          */
18         canDeactivate(component: RoomComponent, currentRoute: ActivatedRouteSnapshot,
19             currentState: RouterStateSnapshot, nextState: RouterStateSnapshot) {
20             return component.canExit();
21         }
22     }
```