

# Atelier 08 Angular :

## Authentification en utilisant une api REST

L'objectif de cet atelier est de consommer une api REST qui retourne un utilisateur et ses rôles enregistrés dans une BD MySql.

1. Ajouter le modèle Role et modifier le modèle User,

```
export class Role{  
  role:string ;  
}
```

```
import { Role } from './Role';  
  
export class User{  
  username:string ;  
  password: string ;  
  roles:Role[];  
}
```

2. Créer, dans AuthService la méthode ***getUserFromDB*** :

```
public loggedUser:string;  
public isLoggedIn: Boolean = false;  
public roles:Role[];  
  
apiURL: string = 'http://localhost:8081/users/login';  
  
getUserFromDB(username:string):Observable<User>  
{  
  const url = `${this.apiURL}/${username}`;  
  return this.http.get<User>(url);  
}
```

3. Appeler la méthode ***getUserFromDB*** à partir de login.comonent.ts :

```
onLoggedIn()  
{  
  this.authService.getUserFromDB(this.user.username).subscribe((usr:User) => {  
    if (usr.password==this.user.password)
```

```

        {
            this.authService.signIn(usr);
            this.router.navigate(['/']);
        }
        else
            this.erreur = 1;
    },(err) => console.log(err));
}

```

4. Dans AuthService, modifier la méthode **signIn** comme suit :

```

signIn(user :User){
    this.loggedUser = user.username;
    this.isloggedIn = true;
    this.roles = user.roles;
    localStorage.setItem('loggedUser',this.loggedUser);
    localStorage.setItem('isloggedIn',String(this.isloggedIn));
}

```

5. Dans AuthService, modifier les méthodes **isAdmin()** et **getRoles()** comme suit

```

isAdmin():Boolean{
    let admin: Boolean = false;
    if (!this.roles) //this.roles== undefined
        return false;

    this.roles.forEach((curRole) => {
        if(curRole.role == 'ADMIN') {
            admin = true;
        }
    });
    return admin;
}

```

```

getUserRoles(username :string){
    this.getUserFromDB(username).subscribe((user: User)=>{
        this.roles = user.roles;
    });
}

```