

| Fichier                | Classe                 |
|------------------------|------------------------|
| Server/server.js       |                        |
| client/src/app/service | Authentication.service |
|                        | Auth.interceptor       |
|                        | Auth.guard             |
|                        | utilisateur.service    |

client/src/app/login

login Component

|                      |                 |
|----------------------|-----------------|
| client/src/app/fiche | Ficge Component |
|----------------------|-----------------|

| Variables  | Nom                             |
|--|---------------------------------|
|  | app.get('/api/getEtudiants')    |
|  | app.post('/api/login')          |
|  | verifyToken(json,String,Number) |
| CurrentUserSubject ( Behaviour Subject)<br>CurrentUser ( Observable) | CurrentUserSubject()            |
|  | IsAuthenticated()               |
|  | login(String,String)            |
|  | logout()                        |
|  | intercept(HttpRequest)          |
|  | can-activate()                  |
|  | getAllEtudiants()               |
|  | getUser()                       |
| enseignant (Enseignant)  | ngOnInit()                      |
|  | submit()                        |
| etudiantsList (Etudiant[])<br>etudiants (Etudiant)                   | ngOnInit()                      |
|  | getEtudiants()                  |
|  | getInfoCompte()                 |
|  | logout()                        |

| Description   | Paramètre  |  |  |  |
|---|--|--|--|--|
| Si Token ok = Retourne la liste des étudiants<br>Si Token pas ok = forbidden                              | Fonction <b>verifyToken</b>  |  |  |  |
| Login<br>Si log ok = génère Token/retourne Token  | Mail (String)<br>Mot de passe(String)                                      |  |  |  |
| Recupère header, le transforme en string,<br>et le split pour récupérer la chaîne Token                   | Docs -> log de BD (Json)<br>Secret (String)<br>ExpireTimer (String/number) |  |  |  |
| Récupérer la valeur CurrentUserSubject  |  |  |  |  |
| Vérifie si le token est expiré  |  |  |  |  |
| Appelle <b>app.post('/api/login')</b><br>Stock le token si success<br>Redirige vers la fiche              | email (String)<br>mdp (String)   |  |  |  |
| Supprime le Token du local storage<br>Redirige vers Login   |  |  |  |  |
| Intercept TOUT les échanges http<br>Prend le Token et paramètre le header SI<br>token existe              | HttpRequest(HttpRequest)   |  |  |  |
| Sécurise les routes en vérifiant la<br>connexion.<br>Si authentifié -> OK<br>Sinon -> redirige vers Login |  |  |  |  |
| Set header requête pour autorisation<br>Retourne une liste d'étudiant                                     |  |  |  |  |
| Set header requête pour autorisation<br>Retourne info utilisateur connecté                                |  |  |  |  |
| Initialise l'objet Enseignant   |  |  |  |  |
| Appelle la fonction login de l'auth-service   |  |  |  |  |
| Initialise l'objet Etudiant<br>lance la fonction getEtudiants()   |  |  |  |  |
| Appelle <b>utilisateurService.getAllEtudiant()</b><br>Si ok > Ajoute résultat dans la liste               |  |  |  |  |
| appelle utilisateurService.getUser  |  |  |  |  |
| appelle <b>authenticationService.logout()</b>   |  |  |  |  |

[illegible]