**Hướng dẫn: ngx-cookie-service (angular) vs cookie-parser (nodejs)**

***Phần Font-end:***

**Nạp thư viện ngx-cookie-service và viết hàm set get và delete Token**

**Tạo guard service để kiểm tra người dùng tại frontend : canActive**

**Tạo bẫy Oninit ở app.module để khi có đăng nhập không vào được trang login**

**Khi Req đến server cần gửi token: authorization để xác nhận, dùng HttpIntercepter xây dựng như 1 service để thêm authorization vào trong header request**

***Phần Backend:***

**Tạo AuthorHelper.js có nhiệm vụ kiểm tra token khi request từ Front-end và dùng như phần middleware (Dùng passport và passport.strategy để khỏi viết)**

**Để lấy thông tin user từ token, Front end xây dựng hàm lấy phần payload từ token: token.split(‘.’)[1] và giải mã: JSON.parse(window.atob(payload))**

**Triển khai socket.io: gồm 2 phần 1 phía frontend và 1 phía backend**

**Backend:**

**Frontend: vào polyfills.ts bổ sung (window as any).global = window; để giải quyết lỗi global is not define trong angular**

1. Nạp ngx-cookie-service và khai báo trong app.module: gồm 2 phần

Phần 1: Nạp thư viện npm ngx-cookie-service, khai báo ngx-cookie-service vào app.module

Phần 2: Tạo file service: token.service, và nạp cookieService để có các hỗ trợ set, get, delele

Tạo service services/token.service và Viết hàm SetToken, GetToken, DeleteToken

2. Tạo services/token.service với 3 chức năng SetToken, GetToken, DeleteToken

SetToken cho signup và login

DeleteToken cho signout

GetToken cho các chứng thực web

3. Tạo service guard

Ng g guard service services/auth --no-spec với chức năng khi có token thì báo true, không thì đẩy về trang gốc và báo false.

Đưa guard vào CanActive trong routing của streams-routing

4. Giải quyết vấn đề sau khi login thì không về trang login được nữa với xử lý trong app.component