Una de las formas que encontre para contornar ese problema podemos hacer lo siguiente:

1) en el arquivo loginComponent, importamos NgZone

1. import { NgZone } from '@angular/core';

2) En el constructor:

1. constructor(
2. ...
3. private zone: NgZone ) {
4. }

3) en el arrachSigIn:

1. attachSigIn( element ) {
2. this.auth2.attachClickHandler( element, {}, ( googleUser ) => {
3. //let profile = googleUser.getBasicProfile();
4. let token = googleUser.getAuthResponse().id\_token;
6. //Ejecutamos dentro de una zona
7. this.zone.run( () => {
8. console.log('ESTOU DENTRO DE UMA ZONA')
9. this.usuarioService.loginGoogle( token )
10. .subscribe( isLogado => this.router.navigate( [ '/dashboard' ] ) );
11. } );
12. } );
13. }

import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

import { NgForm } from '@angular/forms';

import { UsuarioService } from '../services/service.index';

import { Usuario } from '../models/usuario.model';

import { NgZone } from '@angular/core';

declare function init\_plugins();

declare const gapi: any;

@Component({

selector: 'app-login',

templateUrl: './login.component.html',

styleUrls: ['./login.component.css']

})

export class LoginComponent implements OnInit {

email: string;

recuerdame: boolean = false;

auth2: any;

constructor(

public router: Router,

public \_usuarioService: UsuarioService,

private zone: NgZone

) { }

ngOnInit() {

init\_plugins();

this.googleInit();

this.email = localStorage.getItem('email') || '';

if ( this.email.length > 1 ) {

this.recuerdame = true;

}

}

googleInit() {

gapi.load('auth2', () => {

this.auth2 = gapi.auth2.init({

client\_id: '456438129700-kp358v82r68omji6rp6cv55v9qujfq7k.apps.googleusercontent.com',

cookiepolicy: 'single\_host\_origin',

scope: 'profile email'

});

this.attachSignin( document.getElementById('btnGoogle') );

});

}

attachSignin( element ) {

this.auth2.attachClickHandler( element, {}, (googleUser) => {

// let profile = googleUser.getbasicProfile();

const token = googleUser.getAuthResponse().id\_token;

this.zone.run( () => {

console.log('Estoy dentro de una zona');

this.\_usuarioService.loginGoogle( token )

.subscribe( isLogado => this.router.navigate( ['/dashboard']) );

});

});

}

ingresar( forma: NgForm) {

if ( forma.invalid ) {

return;

}

const usuario = new Usuario( null, forma.value.email, forma.value.password );

this.\_usuarioService.login( usuario, forma.value.recuerdame )

.subscribe( correcto => this.router.navigate(['/dashboard'] ));

// this.router.navigate([ '/dashboard']);

}

}

import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

import { NgForm } from '@angular/forms';

import { UsuarioService } from '../services/service.index';

import { Usuario } from '../models/usuario.model';

declare function init\_plugins();

declare const gapi: any;

@Component({

selector: 'app-login',

templateUrl: './login.component.html',

styleUrls: ['./login.component.css']

})

export class LoginComponent implements OnInit {

email: string;

recuerdame: boolean = false;

auth2: any;

constructor(

public router: Router,

public \_usuarioService: UsuarioService

) { }

ngOnInit() {

init\_plugins();

this.googleInit();

this.email = localStorage.getItem('email') || '';

if ( this.email.length > 1 ) {

this.recuerdame = true;

}

}

googleInit() {

gapi.load('auth2', () => {

this.auth2 = gapi.auth2.init({

client\_id: '456438129700-kp358v82r68omji6rp6cv55v9qujfq7k.apps.googleusercontent.com',

cookiepolicy: 'single\_host\_origin',

scope: 'profile email'

});

this.attachSignin( document.getElementById('btnGoogle') );

});

}

attachSignin( element ) {

this.auth2.attachClickHandler( element, {}, (googleUser) => {

// let profile = googleUser.getbasicProfile();

const token = googleUser.getAuthResponse().id\_token;

this.\_usuarioService.loginGoogle( token )

.subscribe( resp => {

console.log( resp );

});

});

}

ingresar( forma: NgForm) {

if ( forma.invalid ) {

return;

}

const usuario = new Usuario( null, forma.value.email, forma.value.password );

this.\_usuarioService.login( usuario, forma.value.recuerdame )

.subscribe( correcto => this.router.navigate(['/dashboard'] ));

// this.router.navigate([ '/dashboard']);

}

}