

ANNEX 2 - AUTORITZACIÓ EN FUNCIÓ DE L'AUTENTICACIÓ

1- Executa `php artisan make:middleware AdminAuth` dins **empresa**.

2- El codi del Middleware `app/Http/Middleware/AdminAuth.php` serà:

```
<?php

namespace App\Http\Middleware;

use Closure;
use Illuminate\Http\Request;
use Symfony\Component\HttpFoundation\Response;

class AdminAuth
{
    /**
     * Handle an incoming request.
     *
     * @param  \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Response)  $next
     */
    public function handle(Request $request, Closure $next): Response
    {
        if (Auth()->user()->role != "admin"){
            return redirect('dashboard-basic')->with('error','Error: usuari validat amb el role basic');
        }
        return $next($request);
    }
}
```

Aquest codi comprova si l'usuari té el role d'admin o no, i permet o no gestionar la petició realitzada. En cas de no permetre gestionar la petició, es redirecciona cap a la vista **dashboard-basic** i s'enviarà un missatge d'error que es pot visualitzar modificant **dashboard-basic.blade.php**.

3- Modificarem **dashboard-basic.blade.php** per aprofitar el missatge d'error. Fes que el seu contingut sigui aquest ara:

```
1 <x-app-layout>
2     <x-slot name="header">
3         <h2 class="font-semibold text-xl text-gray-800 leading-tight">
4             {{ __('Dashboard dels usuaris tipus bàsic') }}
5         </h2>
6     </x-slot>
7
8     <div class="py-12">
9         <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
10            <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
11                <div class="p-6 bg-white border-b border-gray-200">
12                    <a href="{{ url('trebs/index_basic') }}">Treballadors: Visualització bàsica</a>
13                </div>
14            </div>
15        </div>
16
17        <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
18            @if (session('error'))
19                <div class="alert alert-success">
20                    {{ session('error') }}
21                </div>
22            @endif
23        </div>
24 </x-app-layout>
```

Bàsicament, hem afegit les línies **17** a **23** al codi existent.

4- Registra el nou middleware per fer-ho disponible dins del projecte. Dins **bootstrap/app.php** el nou middleware s'ha de registrar a la secció →**withMiddleware** afegint:

```
$middleware->alias([
    'adminAuth' => \App\Http\Middleware\AdminAuth::class,
]);
```

5- Afegeix la ruta al Dashboard (i altres més endavant si ho creus necessari) dins d'un grup de middleware a **routes/web.php** tenint en compte que el grup hauria d'estar dins del grup de rutes controlades pel middleware **auth**. Per exemple:

```
Route::group(['middleware' => 'auth'], function() {
    Route::group(['middleware' => 'adminAuth'], function() {
        Route::get('/dashboard', function () {
            return view('dashboard');
        }->name('dashboard'));
    });

    //Route::get('/dashboard', function () {
    // return view('dashboard');
    //})->name('dashboard');

    Route::get('/dashboard-basic', function () {
        return view('dashboard-basic');
    }-> name('dashboard-basic');

    Route::get('trebs/index_basic', 'App\Http\Controllers\ControladorTreballador@index_basic')->name('trebs.index_basic');
    Route::get('trebs/show_basic/{tid}', 'App\Http\Controllers\ControladorTreballador@show_basic')->name('trebs.show_basic');

    Route::resource('trebs', ControladorTreballador::class);
});
```

Ara bé, hauràs d'analitzar si cal afegir més rutes dins del grup **adminAuth** perquè aquest codi només protegeix l'accés al dashboard d'administrador.