



Bài 12: Angular Guards - Bảo vệ Route trong Angular

1. Giới thiệu về Angular Guards

Angular Guards là một cơ chế giúp kiểm soát quyền truy cập vào các route trong ứng dụng Angular. Chúng giúp bảo vệ các route khỏi những người dùng không đủ quyền hoặc chưa đăng nhập.

Có 5 loại Angular Guards chính:

- **CanActivate** – Kiểm tra quyền truy cập vào một route.
- **CanActivateChild** – Kiểm tra quyền truy cập vào các route con.
- **CanDeactivate** – Kiểm tra xem người dùng có thể rời khỏi trang hiện tại không.
- **Resolve** – Tiền xử lý dữ liệu trước khi vào route.
- **CanLoad** – Ngăn người dùng tải module nếu không có quyền.

2. Cài đặt một dự án Angular mới (Nếu chưa có)

Nếu bạn chưa có dự án Angular, hãy tạo một dự án mới:

```
sh
```

```
ng new angular-guards-demo  
cd angular-guards-demo  
ng serve
```

3. Tạo AuthGuard để kiểm tra người dùng đăng nhập

Trong ứng dụng thực tế, chúng ta sẽ có một dịch vụ xác thực (**AuthService**). Ở đây, chúng ta giả lập việc kiểm tra đăng nhập.

3.1. Tạo **AuthService**

Chạy lệnh:

```
sh
```

```
ng generate service services/auth
```

Sau đó, mở file **auth.service.ts** và thêm nội dung sau:

typescript

```
import { Injectable } from '@angular/core';

@Injectable({
  providedIn: 'root'
})
export class AuthService {
  private isLoggedIn = false; // Giả lập trạng thái đăng nhập

  login() {
    this.isLoggedIn = true;
  }

  logout() {
    this.isLoggedIn = false;
  }

  isAuthenticated(): boolean {
    return this.isLoggedIn;
  }
}
```

3.2. Tạo **AuthGuard**

Chạy lệnh:

```
sh
```

```
ng generate guard guards/auth
```

Sau đó, mở file **auth.guard.ts** và chỉnh sửa như sau:

typescript

```
import { Injectable } from '@angular/core';
import { CanActivate, ActivatedRouteSnapshot, RouterStateSnapshot, Router } from '@angular/router';
import { AuthService } from '../services/auth.service';

@Injectable({
  providedIn: 'root'
})
export class AuthGuard implements CanActivate {
  constructor(private authService: AuthService, private router: Router) {}

  canActivate(route: ActivatedRouteSnapshot, state: RouterStateSnapshot):
  boolean {
    if (this.authService.isAuthenticated()) {
      return true;
    } else {
      this.router.navigate(['/login']); // Chuyển hướng đến trang đăng nhập
      nếu chưa đăng nhập
      return false;
    }
  }
}
```

4. Tạo Component để kiểm tra AuthGuard

4.1. Tạo các Component cần thiết

sh

```
ng generate component pages/home
ng generate component pages/dashboard
ng generate component pages/login
```

4.2. Cập nhật nội dung các Component

HomeComponent (**home.component.html**):

```
html
```

```
<h1>Trang chủ</h1>
<a routerLink="/dashboard">Đi đến Dashboard</a>
<a routerLink="/login">Đăng nhập</a>
```

DashboardComponent (**dashboard.component.html**):

```
html
```

```
<h1>Dashboard (Chỉ truy cập khi đăng nhập)</h1>
<button (click)="logout()">Đăng xuất</button>
```

DashboardComponent (**dashboard.component.ts**):

typescript

```
import { Component } from '@angular/core';
import { AuthService } from '../../services/auth.service';
import { Router } from '@angular/router';

@Component({
  selector: 'app-dashboard',
  templateUrl: './dashboard.component.html',
  styleUrls: ['./dashboard.component.css']
})
export class DashboardComponent {
  constructor(private authService: AuthService, private router: Router) {}

  logout() {
    this.authService.logout();
    this.router.navigate(['/']);
  }
}
```

LoginComponent (**login.component.html**):

html

```
<h1>Đăng nhập</h1>
<button (click)="login()">Đăng nhập</button>
```

LoginComponent (**login.component.ts**):

typescript

```
import { Component } from '@angular/core';
import { AuthService } from '../../services/auth.service';
import { Router } from '@angular/router';

@Component({
  selector: 'app-login',
  templateUrl: './login.component.html',
  styleUrls: ['./login.component.css']
})
export class LoginComponent {
  constructor(private authService: AuthService, private router: Router) {}

  login() {
    this.authService.login();
    this.router.navigate(['/dashboard']);
  }
}
```

5. Cấu hình Routing và Thêm AuthGuard vào Route

Mở file `app-routing.module.ts` và cập nhật như sau:

typescript

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { HomeComponent } from '../pages/home/home.component';
import { DashboardComponent } from '../pages/dashboard/dashboard.component';
import { LoginComponent } from '../pages/login/login.component';
import { AuthGuard } from '../guards/auth.guard';

const routes: Routes = [
  { path: '', component: HomeComponent },
  { path: 'dashboard', component: DashboardComponent, canActivate:
[AuthGuard] },
  { path: 'login', component: LoginComponent }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```

6. Chạy thử ứng dụng

Khởi động lại server:

```
sh
```

```
ng serve
```

Mở trình duyệt và thử các bước sau:

1. Truy cập **/dashboard** → Bị chuyển hướng đến **/login** (vì chưa đăng nhập).
2. Nhấn **"Đăng nhập"** → Chuyển đến Dashboard.
3. Nhấn **"Đăng xuất"** → Quay lại trang chủ và không thể vào Dashboard nữa.

7. Tổng kết

- **AuthGuard** kiểm tra người dùng có đăng nhập hay chưa.
- Nếu chưa đăng nhập, người dùng bị chuyển hướng đến trang đăng nhập.
- Sau khi đăng nhập, người dùng có thể truy cập Dashboard.
- Khi đăng xuất, trạng thái đăng nhập bị xóa, ngăn người dùng quay lại Dashboard.

Bạn có câu hỏi nào về bài học này không? 🚀