

# Corrigé

P1C6 - Structurez et sécurisez votre API avec NestJS



1. **Créer le guard** : Dans un fichier `roles.guard.ts`, créez le guard suivant :

```
TypeScript ▾ | 📄 ...  
  
import {Injectable, CanActivate, ExecutionContext }from '@nestjs/common';  
import {Observable }from 'rxjs';  
  
@Injectable()  
export class RolesGuard implements CanActivate {  
  canActivate(context:ExecutionContext):boolean {  
    return true;  
  }  
}
```

Ici, la méthode `canActivate()` retourne `true`. Cela signifie que toutes les requêtes sont autorisées.

2. **Appliquer le guard à une route** - Dans le fichier `app.controller.ts`, importez le guard et appliquez-le à une route avec le décorateur `@UseGuards()` :

```
import {Controller,Get,UseGuards }from '@nestjs/common';
import {AppService }from './app.service';
import {RolesGuard }from './roles.guard';

@Controller()
export class AppController {
  constructor(private readonly appService: AppService) {}

  @UseGuards(RolesGuard)
  @Get('projects')
  getProjects() {
    return this.appService.getProjects();
  }
}
```

Ici, le guard protège la route `GET /projects`.

**3. Tester le comportement avec `true`** - Lancez votre serveur, puis effectuez une requête vers :

```
GET http://localhost:3000/projects|
```

Comme le guard retourne `true`, la requête est autorisée. Vous obtenez donc une réponse normale avec un statut **200**.

**4. Modifier le guard pour bloquer l'accès** - Modifiez maintenant le fichier `roles.guard.ts` pour retourner `false` :

```
import {Injectable, CanActivate, ExecutionContext } from '@nestjs/common';
import {Observable } from 'rxjs';

@Injectable()
export class RolesGuard implements CanActivate {
  canActivate(context:ExecutionContext):boolean {
    return false;
  }
}
```

Cette fois, le guard refuse toutes les requêtes.

**5. Tester le comportement avec `false`** - Relancez la même requête :

```
GET http://localhost:3000/projects
```

Cette fois, l'accès à la route est bloqué. NestJS renvoie une réponse avec un statut 403 Forbidden.