



Desarrollo y programación de Aplicaciones Avanzadas Angular

10 Angular Routing y aplicaciones
single page

CORE
networks



Declaración de constante de rutas e implementación de módulo

Archivo app-routing.module.ts

```
import { Routes, RouterModule } from '@angular/router';

...
const routes: Routes = [
    {path:'nombre', component: ClaseComponent, opciones...}
];
...
imports: [RouterModule.forRoot(routes)],
...
```

Uso de routerLink

Template de carga:

```
<router-outlet></router-outlet>
```

Navegación desde elementos HMTL:

```
routerLink="/rutacomponente"
```

Clases activas para menús:

```
[routerLinkActiveOptions]="{{exact:true}} routerLinkActive="activo"
```

Uso de datos

En la ruta:

```
{..., data: {clave: valor}...}
```

Para recuperar el valor en el componente

```
import { ActivatedRoute } from '@angular/router';
... private route: ActivatedRoute ...
this.route.data.subscribe(data => //);
```

Uso de parámetros

En la ruta:

```
{path: "ruta/:parametro"}
```

En routerLink:

```
routerLink="/ruta/parametro"
```

Para recuperar el parámetro de la ruta:

```
this.route.snapshot.params['parametro'];
```

Navegación programática

En el componente:

```
import { Router } from '@angular/router';
...
private router: Router
...
this.router.navigate(['/ruta']);
```

Protección de rutas

Se implementa a través de un archivo generado con Angular CLI:

```
ng generate guard nombre
```

La sintaxis de archivo es:

```
canActivate( next: ActivatedRouteSnapshot,  
            state: RouterStateSnapshot): boolean {  
    // condición true o false;  
}
```

Módulos adicionales

Los módulos adicionales se generan con Angular CLI

```
ng generate module nombre
```

Para añadir componentes a módulos distintos del raíz:

```
ng g c nombremodulo/directorio-opcional/nombrecomponente
```

Lazy loading

En el módulo raíz:

```
{path: 'modulo',  
loadChildren: 'src/app/.../modulo.module#ModuloModule'},
```

En los módulos adicionales:

```
{path: "", component: ComponenteComponent},  
  
RouterModule.forChild(rutas)
```

Módulo compartido

En el módulo compartido:

```
declarations: [DirectivaDirective, PipePipe, ComponenteComponent ],  
exports: [DirectivaDirective, PipePipe, ComponenteComponent ]
```

En los módulos que importan:

```
imports: [ ModulocompartidoModule ],
```