



Desarrollo y Programación de
Aplicaciones Avanzadas con Angular

Formación online

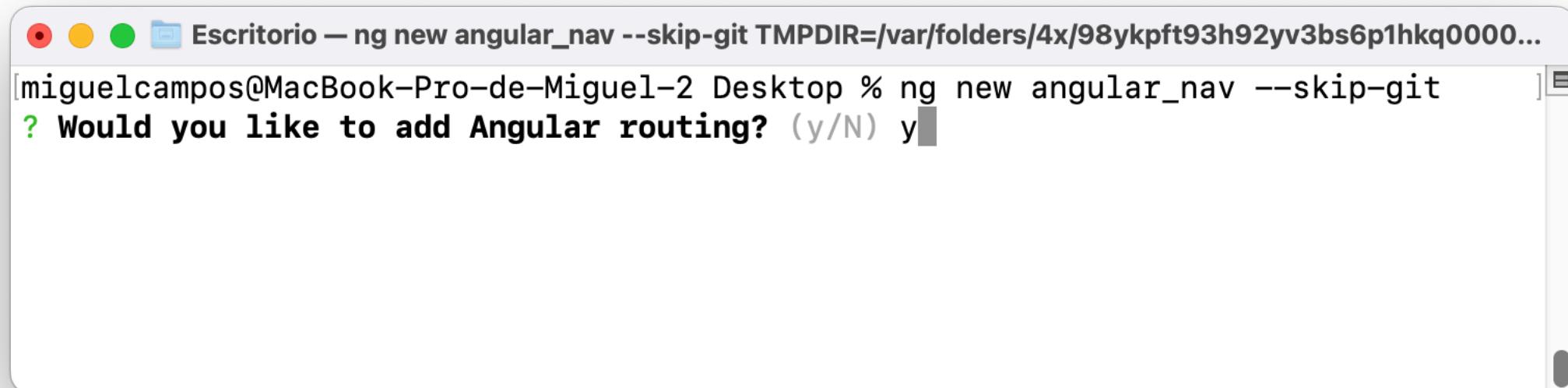


Módulo 8. Angular Routing

Módulo 8. Angular Routing

Navegación entre componentes en Angular

- Hasta ahora en la creación de un proyecto Angular, hemos rechazado la creación del módulo de Angular Routing, respondiendo a la siguiente pregunta que no. A partir de ahora lo vamos a generar para poder realizar la navegación entre componentes.



The screenshot shows a terminal window on a Mac OS X desktop. The title bar says "Escritorio — ng new angular_nav --skip-git TMPDIR=/var/folders/4x/98ykpft93h92yv3bs6p1hkq0000...". The command entered is "miguelcampos@MacBook-Pro-de-Miguel-2 Desktop % ng new angular_nav --skip-git". A question is displayed: "? Would you like to add Angular routing? (y/N) y". The "y" key is being typed, with the cursor visible at the end of the "y".

Módulo 8. Angular Routing

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { AppRoutingModule } from './app-routing.module'; // CLI imports
AppRoutingModule
import { AppComponent } from './app.component';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule // CLI adds AppRoutingModule to the AppModule's imports array
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

- El efecto que va a tener esta decisión es que vamos a disponer de un módulo importado en nuestro fichero **app.module.ts** que va a definir la navegación entre componentes.

Módulo 8. Angular Routing

```
import { NgModule } from '@angular/core';
import { Routes, RouterModule } from '@angular/router'; // CLI imports router

const routes: Routes = [];// sets up routes constant where you define your routes

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

Para que el módulo RouterModule que está siendo importado aquí pueda ser utilizado en otro módulo, es necesario que sea exportado.

Array de rutas, donde especificamos el mapeo de una URL a un componente

Asignación del array de routes al componente encargado de gestionar la navegación, para su inicialización

Módulo 8. Angular Routing

- El otro cambio importante es que debemos indicar el elemento `<router-outlet></router-outlet>` en el HTML del `app.component.html`.

The screenshot shows the Visual Studio Code interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help
- Title Bar:** app-routing.module.ts - my-first-routing-app - Visual Studio Code
- Explorer Sidebar:** Shows the project structure under 'MY-FIRST-ROUTING-APP': e2e, node_modules, src, and app. Inside app, files listed are: app-routing.module.ts (selected), app.component.css, app.component.html (highlighted with a red box), app.component.spec.ts, app.component.ts, and app.module.ts.
- Editor Area:** The 'app.component.html' file is open. The code contains several placeholder comments (523-529) and ends with the line: 534 <router-outlet></router-outlet>. This line is also highlighted with a red box and has a red arrow pointing from the 'app.component.html' entry in the Explorer sidebar.

¡El orden SÍ importa!

Route order

The order of routes is important because the [Router](#) uses a first-match wins strategy when matching routes, so more specific routes should be placed above less specific routes. List routes with a static path first, followed by an empty path route, which matches the default route. The [wildcard route](#) comes last because it matches every URL and the [Router](#) selects it only if no other routes match first.

Ruta por defecto

```
{ path: '**', component: <component-name> }
```

- Se suele definir como última opción del array de rutas.
- El doble asterisco ** indica que es la ruta “comodín”, en caso de que ninguna otra haya hecho match con la ruta en la que el usuario se encuentra.

```
const routes: Routes = [
  { path: 'first-component', component: FirstComponent },
  { path: 'second-component', component: SecondComponent },
  { path: '**', component: PageNotFoundComponent }, // Wildcard route for a 404 page
];
```

Redirecciones

```
const routes: Routes = [  
  { path: 'first-component', component: FirstComponent },  
  { path: 'second-component', component: SecondComponent },  
  { path: '', redirectTo: '/first-component', pathMatch: 'full' }, // redirect to  
`first-component`  
  { path: '**', component: PageNotFoundComponent }, // Wildcard route for a 404 page  
];
```

- Se puede definir una ruta que nos redireccione a otra de las que están definidas en el array de rutas.
- Importante el uso de la propiedad ***pathMatch: 'full'***, que indica que el “match” de la ruta debe ser completo.

Rutas con parámetros

- Podemos recibir una ruta con un parámetro dinámico en una URL, por ejemplo el *id* de un elemento.

```
const routes: Routes = [
  { path: 'personaje', component: PeopleListComponent },
  {
    path: 'personaje/detail/:id',
    component: PeopleDetailComponent,
  },
  {
    path: 'planeta/detail/:id',
    component: PlanetDetailComponent,
  },
  { path: '', component: IndexComponent },
  { path: '**', component: NotFoundComponent },
];
```

Módulo 8. Angular Routing

ENLACES

EJEMPLO 1: ruta sin *paso de parámetros*

```
<li routerLink="/personaje">Personaje</li>
```

EJEMPLO 2: ruta con *paso de parámetros*

```
<li [routerLink]="['/planeta/detail', idPlaneta]">Planeta Detail</li>
```

```
export class PlanetDetailComponent implements OnInit {  
  id!: string | null;  
  constructor(private route: ActivatedRoute) {}  
  
  ngOnInit(): void {  
    this.id = this.route.snapshot.paramMap.get('id');  
  }  
}
```



De esta forma rescatamos el parámetro id que se ha definido en la ruta