

Blog sobre Java EE

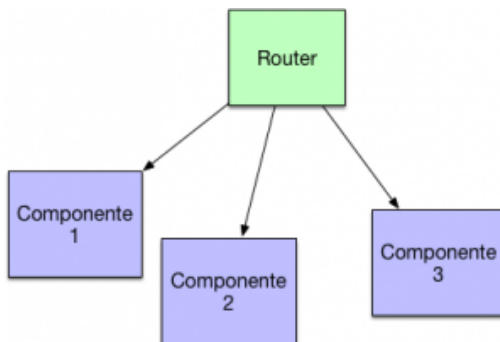
Estás aquí: [Inicio](#)/[Arquitectura](#)/Angular router y su configuración

Angular router y su configuración

3 mayo, 2018 por [Cecilio Álvarez Caules](#) — [Deja un comentario](#)

AHORA CAIGO.

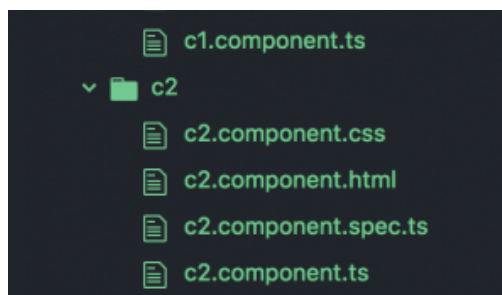
Vamos a introducir el concepto de **Angular Router** y como realizar su configuración para que las cosas nos queden organizadas. Angular Router se encarga de decidir que componentes de Angular se muestran en cada momento.



Vamos a ver un ejemplo sencillo creando dos componentes c1 y c2 , para ello ejecutamos **angular-cli**:

ng generate component c1

ng generate component c2



Vamos a cambiar las plantillas para que cada uno tenga un mensaje en castellano:

```
1 | este es el componente 1
```

```
1 | este es el componente 2
```

Angular Router

Es momento de configurar Angular Router , para ello el primer paso es modificar el fichero **app.module.ts** y añadir la configuración propia del router.

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

El siguiente paso **es definir un conjunto de rutas válido:**

```
1 | const rutas: Routes = [  
2 |   { path: 'c1', component: C1Component },  
3 |   { path: 'c2', component: C2Component },  
4 |   { path: '', redirectTo: '/c1', pathMatch: 'full' },  
5 | ];
```

Angular y Rutas

En este caso solamente hemos declarado dos rutas cada una de ellas carga un componente.

Ademas hemos añadido una ruta vacía **que realiza un redirect y nos carga el componente1** . Nos queda configurar el Router para que realice tareas de debugging cada vez que solicitemos una ruta eso se realiza a nivel de @NgModule

```
1 | @NgModule({  
2 |   declarations: [  
3 |     AppComponent,  
4 |     C1Component,  
5 |     C2Component  
6 |   ],  
7 |   imports: [  
8 |     BrowserModule,  
9 |     RouterModule.forRoot()
```

el fichero de **app.component.html** para que no cargue por defecto ningún componente y se apoye en el router en este caso hay que utilizar las etiquetas de **“router-outlet” en la página.**

```
1 | <router-outlet></router-outlet>
```

Es momento de probar el router , utilizamos ng serve y arrancamos la aplicación:

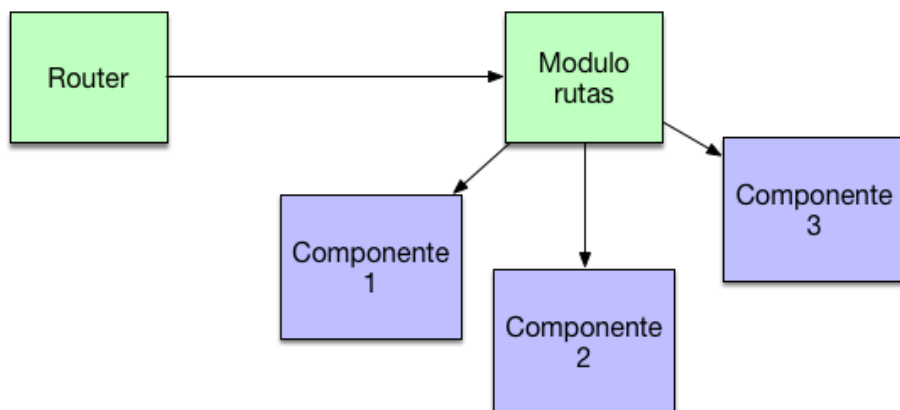


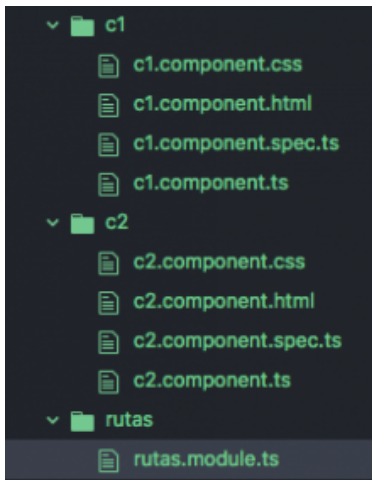
El componente c1 carga sin problemas , vamos a ver el segundo:



Angular Router y Modulos

En muchas ocasiones es preferible definir el sistema de enrutado **de forma independiente y no cargarlo todo en el app.module.ts.**





Es en este módulo en el cual vamos a ubicar toda la funcionalidad del sistema de rutas:

```
1  import { NgModule } from '@angular/core';
2  import { CommonModule } from '@angular/common';
3  import { RouterModule, Routes } from '@angular/router';
4  import { C1Component } from '../c1/c1.component';
5  import { C2Component } from '../c2/c2.component';
6  const rutas: Routes = [
7    { path: 'c1', component: C1Component },
8    { path: 'c2', component: C2Component },
9    { path: '', redirectTo: '/c1', pathMatch: 'full' },
10 ];
11
12 @NgModule({
13   imports: [
14     CommonModule,
15     RouterModule.forRoot(
16       rutas,
17       { enableTracing: true }
18     )
19   ],
20   declarations: [],
21   exports :[
22     RouterModule
23   ],
24 })
25 export class RutasModule {
26
27 }
```

En este caso hemos trasladado toda la funcionalidad del enrutado a nuestro nuevo módulo y hemos solicitado que **se exporte el RouterModule**.

```
1  exports :[ RouterModule ]
```

```
9  @NgModule({
10    declarations: [
11      AppComponent,
12      C1Component,
13      C2Component
14    ],
15    imports: [
16      BrowserModule,
17      RutasModule,
18    ],
19    providers: [],
20    bootstrap: [AppComponent]
21  })
22  export class AppModule { }
```

Hemos exportado el sistema de rutas a un módulo independiente y ahora todo queda más ordenado:

Otros artículos relacionados

[JAVA SE](#) [SPRING](#) [JAVA EE](#) [JAVASCRIPT](#) [FRAMEWORKS JS](#) [ARQUITECTURA](#) [MIS LIBROS](#) [MIS CURSOS](#)

1. [Angular Modules y el uso de servicios](#)
2. [Angular ngFor la directiva y sus opciones](#)
3. [Angular async pipe y observables](#)
4. [Angular](#)



PDF



in



0

COMPARTIR



Archivada en: [Arquitectura](#), [Enterprise Design Patterns](#)

Leave a Reply

Be the First to Comment!

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra [política de cookies](#), pinche el enlace para mayor información.

[plugin cookies](#)

ACEPTAR

[✉ Subscribe ▼](#)

BUSCAR

**Cupón
Descuento
50%
PACKJAVACORE**

Mis Cursos de Java Gratuitos

Java Herencia



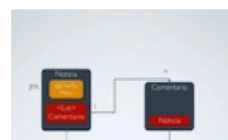
Java JDBC



Servlets

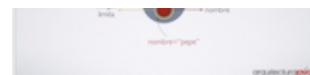


Introduccion JPA



Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra [política de cookies](#), pinche el enlace para mayor información.

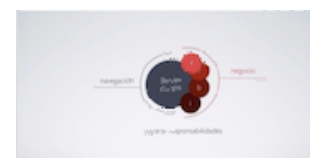
[plugin cookies](#)[ACEPTAR](#)



Java API Core



Java Web



Pack Java Core



Arquitectura Java Solida con Spring



POPULAR

[Spring REST Client con RestTemplates](#)

[Nuevo Curso:Arquitectura Java Sólida con Spring 4.3 y Anotaciones](#)

[Arquitecturas REST y sus niveles](#)

[Angular 5 Hello World y su funcionamiento](#)

[Java 9 Modules y el concepto de modularidad](#)

[El concepto de Java Annotations y su funcionamiento](#)

CONTACTO

contacto@arquitecturajava.com

LO MAS LEIDO

[¿Qué es Spring Boot?](#)

[Java Constructores this\(\) y super\(\)](#)

[Usando Java Session en aplicaciones web](#)

[Angular router y su configuración](#)

[Java Iterator vs ForEach](#)

[Introducción a Servicios REST](#)

[¿Cuales son las certificaciones Java?](#)

[Java Override y encapsulación](#)

[Ejemplo de Java Singleton \(Patrones y ClassLoaders\)](#)

[¿Qué es Gradle?](#)

[REST JSON y Java](#)

[Usando el patron factory](#)

[Ejemplo de JPA , Introducción \(I\)](#)

[Uso de Java Generics \(I\)](#)

[¿Qué es un Microservicio?](#)

[Comparando java == vs equals](#)

[Mis Libros](#)

[Arquitecturas REST y sus niveles](#)

[Spring MVC Configuración \(I\)](#)

[Java 8 Functional Interfaces y sus tipos](#)