

Vamos a construir un ejemplo de Angular 5 Hello world y explicar un poco a detalle las diferentes partes de este framework. Angular 5 supone un avance muy fuerte sobre Angular 1.x e incluye el uso de Typescript como lenguaje de referencia. El objetivo de Angular 5 es convertirse en el framework de referencia en el mundo de TypeScript y JavaScript.

Instalación de Angular 5

La instalación de Angular 5 es sencilla . En principio es suficiente con instalar la herramienta de Angular Cli a través de npm. ¿Para qué sirve esta herramienta? . Esta herramienta funciona a nivel de linea de comandos y permite automatizar las tareas más habituales de programación con Angular5.

```
npm install -g @angular/cli
```

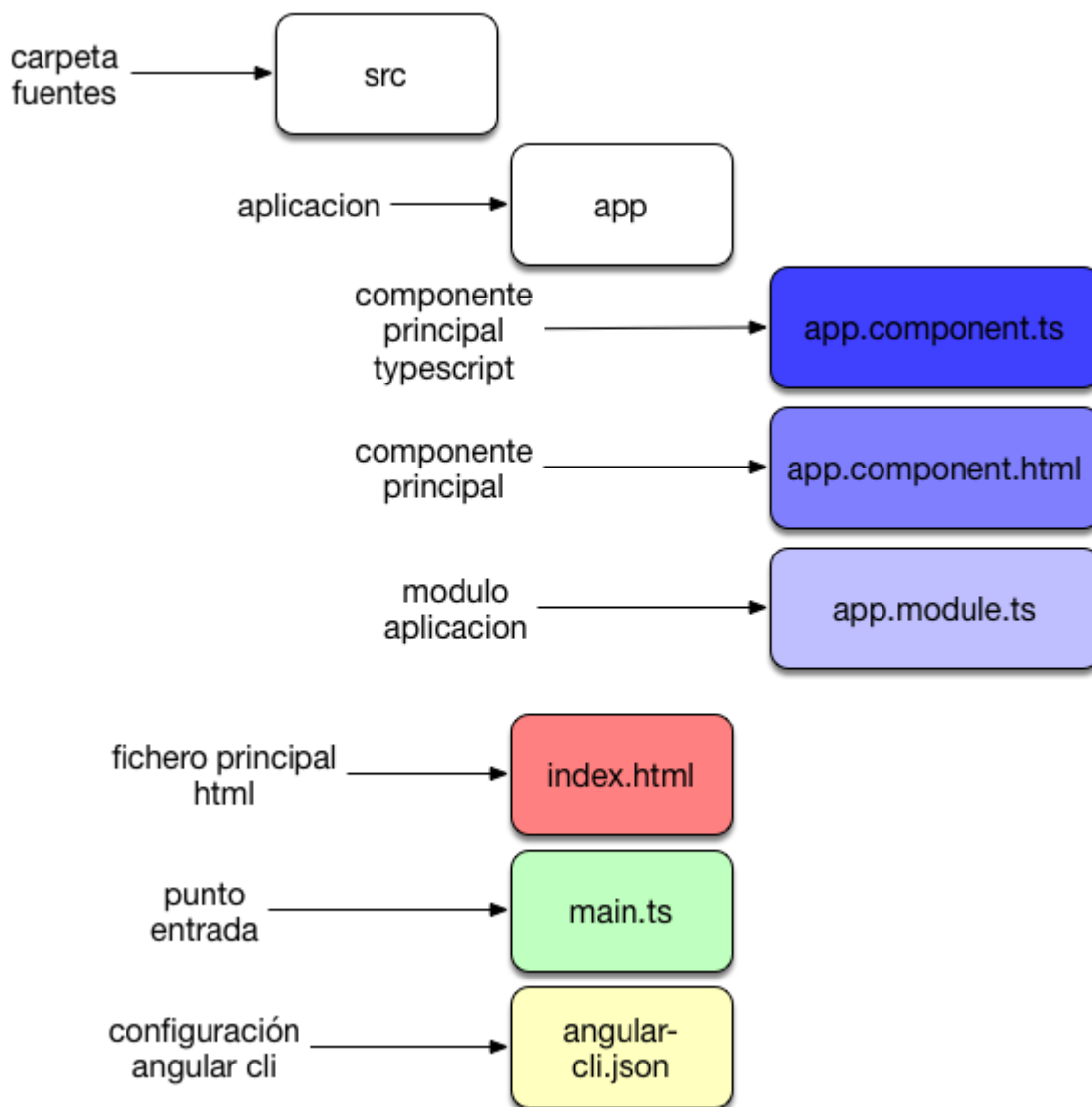
Instalaremos la herramienta a nivel global. Una vez instalada ya podemos hacer uso de ella y construir un nuevo proyecto desde la linea de comandos.

```
ng new aplicacion1
```

Se trata de un comando muy sencillo de ejecutar y que se encarga de crear nuestro proyecto de inicio. El gran problema que existe con Angular es que el proyecto de inicio contiene “muchos ficheros” . Esto hace que en un primer momento estemos muy muy perdidos sobre como empezar con Angular.



¿Para que sirven todos estos ficheros? . Bueno no vamos a identificar uno a uno porque es un poco locura pero si vamos a hablar de los ficheros fundamentales.

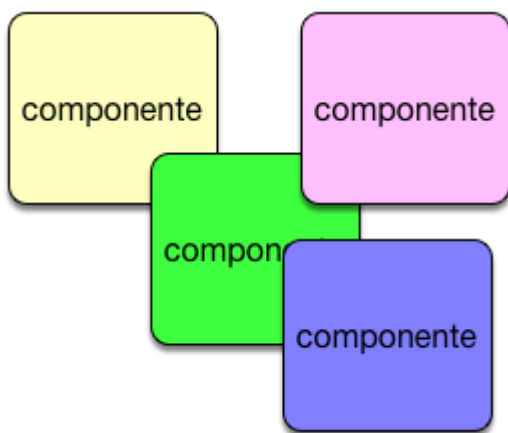


El primer fichero que destaca es el de index.html. Este es el fichero de entrada a la aplicación y contiene muy poco código . El código más importante que contiene este ubicado en el body.

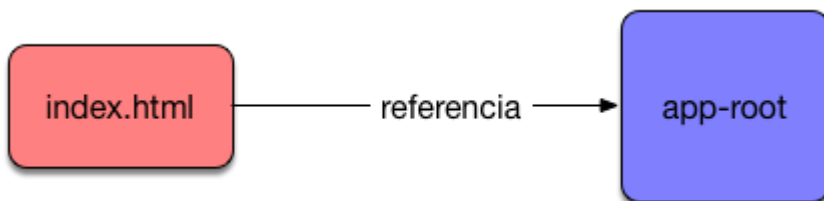
```
<body>
<app-root></app-root>
```

</body>

Como podemos ver el body dispone de una etiqueta especial `<app-root></app-root>` esta etiqueta es la encargada de definir el componente principal de Angular 5. Angular es un framework que tiene un fuerte enfoque de construcción de componentes visuales con TypeScript.



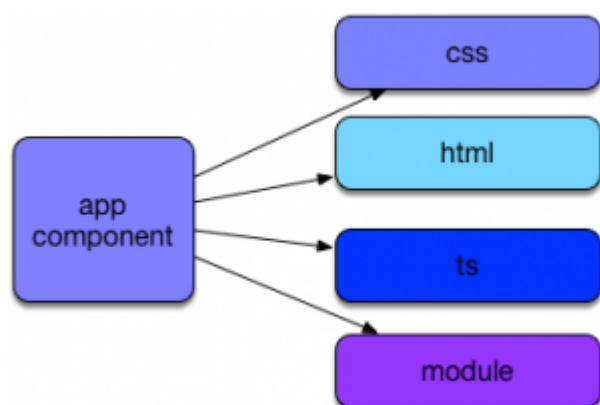
Así pues ya tenemos la primera parte, el fichero de Index referencia al componente raiz.



¿Donde tenemos el componente Raiz? .El componente raiz se encuentra ubicado en la carpeta app y se denomina a nivel de fichero app.component.



Esto es algo curioso en un primer momento porque pensamos que debería llamarme app-root. :



Angular divide el componente en varias partes

app.component.css: Define el estilo del componente

app.component.html: Define la plantilla del componente

app.component.ts: Define la funcionalidad de TypeScript del componente.

app.module.ts: Define el módulo al que pertenece el componente.

Es en el fichero app.component.ts en donde Angular enlaza la etiqueta <app-root> con el nombre del componente:

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

```
@Component({
  selector: 'app-root',
```

```
templateUrl: './app.component.html',  
styleUrls: ['./app.component.css']  
})  
export class AppComponent {  
  title = 'app';  
}
```

El fichero main.ts

Ya tenemos bastante claro como se relacionan los componentes y la página index. Ahora bien ¿para que sirve el fichero main.ts?. Este fichero es el encargado de definir que módulo es el de arranque.

```
import { enableProdMode } from '@angular/core';  
import { platformBrowserDynamic } from '@angular/platform-browser-  
dynamic';  
  
import { AppModule } from './app/app.module';  
import { environment } from './environments/environment';  
  
if (environment.production) {  
  enableProdMode();  
}  
  
platformBrowserDynamic().bootstrapModule(AppModule)  
  .catch(err =&gt; console.log(err));
```

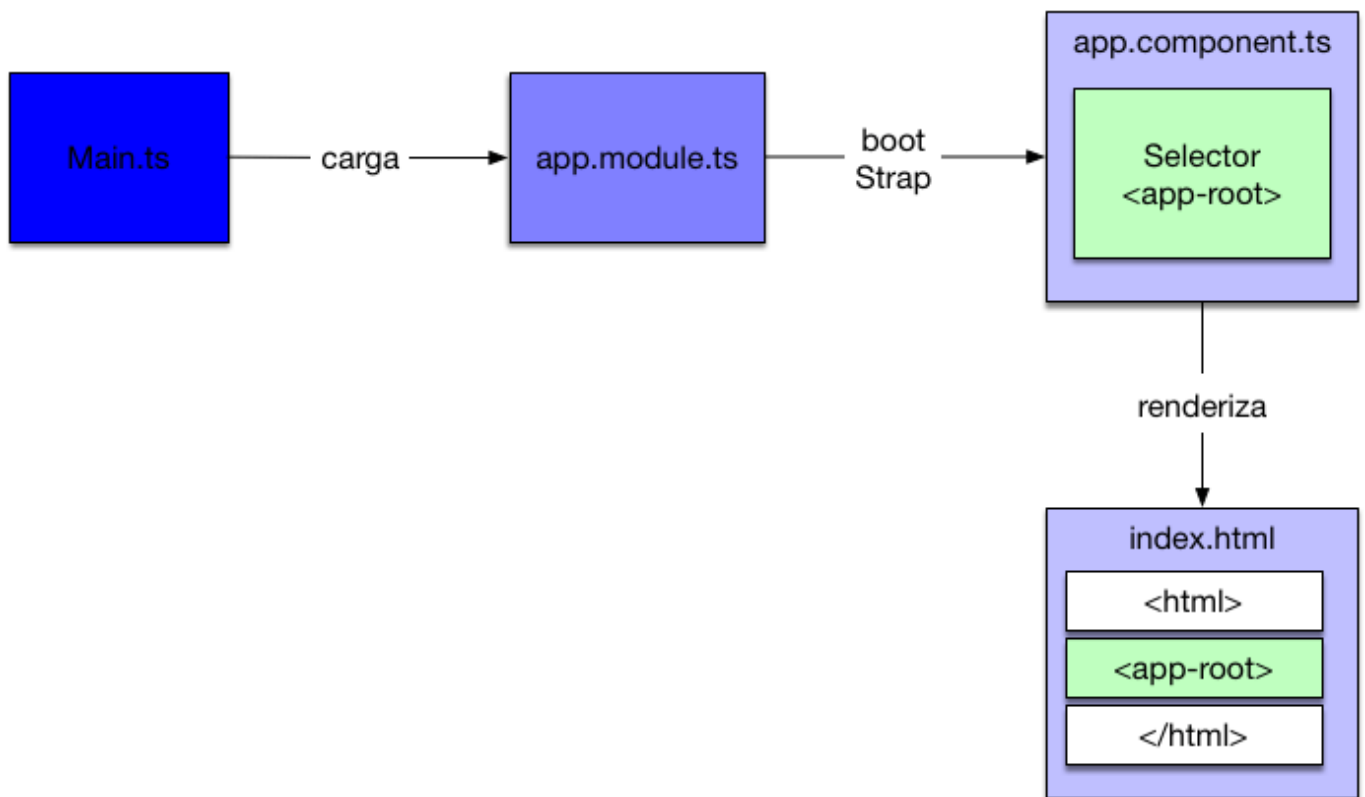
Podemos ver como solicita que se cargue AppModule. El fichero de app.module.ts se encuentra en la carpeta src. Vamos a mostrar su contenido:

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';

import { AppComponent } from './app.component';

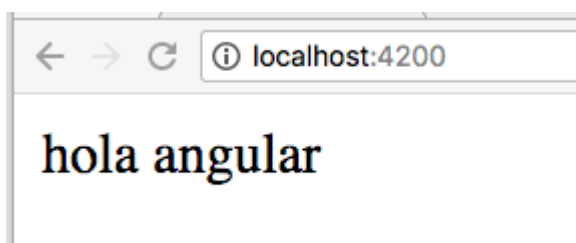
@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Como podemos ver es este fichero el que a través de la propiedad de bootstrap nos indica que componente es el de arranque en este caso AppComponent. Así pues es lo que sucede en una aplicación Angular es que en un primer momento se carga el fichero main.ts que define cual es el modulo de arranque. Angular accede al módulo de Arranque y busca cual es el componente inicial que arrancar y lo arranca. Este componente dispone de un decorador “selector”:app-root que indica al framework que etiqueta le identifica. Esta es la etiqueta que se encuentra en el fichero index.html.



Angular 5 hello world

Ya únicamente nos queda arrancar nuestra aplicación de Angular 5 Hello world . Para ello vamos a modificar el contenido del fichero **app.component.html** y dejar unicamente el mensaje de “hola angular”. Desde la linea de comandos ejecutamos **ng serve**. Angular arrancará el servidor y veremos nuestro primer componente cargado.



Otros artículos relacionados:

1. [Usando Rx Observables en JavaScript](#)
2. [JavaScript Deferred Objects y peticiones asíncronas](#)
3. [jQuery Promise y AJAX](#)

Artículos externos

Los 10 frameworks de JavaScript en 2018