



Desarrollo y Programación de
Aplicaciones Avanzadas con Angular

Formación online



Módulo 2. Introducción a Angular

Módulo 2. Introducción a Angular

¿Qué es Angular?



¿Qué es Angular?

- Angular es un **framework** desarrollado y mantenido por Google, de código abierto
- Angular es la evolución de AngularJS, **re-escrito desde cero**
- Angular es un framework **escrito en TypeScript**
- Angular es **multi-plataforma**, aplicaciones web, híbridas, nativas y de escritorio
- Angular es **modular**, basado en **componentes**
- Angular actualiza su versión cada 6 meses, 2 veces al año
- Angular construye SPAs, **Single Page Applications**, donde el contenido cambia de forma dinámica sin necesidad de recargar el navegador constantemente

Angular es multiplataforma

- Angular se puede usar para crear aplicaciones web, pero *no sólo eso*
- Existen frameworks como Ionic que permiten usar Angular para crear aplicaciones híbridas (Aplicaciones móviles que ejecutan un webView/navegador en vez de código nativo)
- Otros frameworks como NativeScript permiten crear aplicaciones nativas en Android y iOS a partir del código Angular
- También podemos escribir otros tipos de aplicaciones como las Chrome Extensions, dado que el paquete final es JS plano.
- Podemos crear PWAs (Progressive Web Apps) e incluso publicarlas en las stores
- Incluso aplicaciones de escritorio gracias a Electron, podemos crear apps para Windows o MacOS

Angular VS AngularJS

- AngularJS fue anunciado en 2009, actualmente no lanzan nuevas versiones (1.7) y el soporte termina en Junio de 2021.
- AngularJS fue re-escrito desde cero con Angular, para crear un framework más robusto, aprovechando TS, nuevo motor de plantillas, basado en componentes, con DI en su núcleo...
- Angular fue lanzado en 2016, comúnmente conocido como Angular 2, aunque no es correcto
- Angular promete evolucionar poco a poco para no sufrir una re-escritura desde cero nunca más, sus versiones se mueven rápido pero de forma progresiva

Módulo 2. Introducción a Angular

Versiones en Angular

Angular es versionado según Semantic Versioning (Major.Minor.Issues)

Major release (Angular 2.0.0 -> 3.0.0): Contiene nuevas funcionalidades y puede convertir código en *deprecated** o necesitar refactors para poder actualizar (Breaking Changes). *Nueva versión cada 6 meses.*

Minor release (Angular 2.0.0 -> 2.1.0): Contiene pequeñas nuevas funcionalidades, pero conserva la compatibilidad (No Breaking Changes). *1-3 versiones por cada Major release.*

Patch release (Angular 2.0.0 -> 2.0.1): No contiene ningún riesgo, sólo contiene arreglo de errores (bug fixing) y cambios mínimos. *Nueva versión cada 1-2 semanas.*

*Cuando algo pasa a deprecated, seguirá funcionando durante 2 major releases más, para asegurar una buena y eficiente transición mientras actualizamos.

Esta estrategia de versiones tiene como objetivo que nunca más pase lo que pasó con AngularJS, y nos asegura que Angular siempre está en continua mejora y progreso, pero con ventajas para el soporte a largo tiempo.

Angular y TypeScript

- TypeScript es un superset de JavaScript, nos permite tener fuerte tipado, características novedosas de JS compatibles con navegadores más antiguos y mucho más...
- TypeScript es desarrollado y mantenido por Microsoft, de código abierto
- Una de las razones por la que Angular decidió crear otro framework distinto
- Exprime el uso de TypeScript en su core
- TypeScript necesita ser compilado (transpilado técnicamente), Angular usa el CLI que usa webpack para el proceso de transpilación y empaquetado de la app

Módulo 2. Introducción a Angular

Frameworks VS librerías

FRAMEWORKS (Angular, VueJS...)

- ✓ All-in-box (incluye todo lo necesario para hacer apps: routing, http request, state management, forms, testing...)
- ✓ Indican una guía de estilo, patrones y normas recomendables a seguir para ser eficientes
- ✗ Muchas abstracciones y poco margen de customización, difícil ir contra corriente
- ✗ Aprender el framework puede ser extenso y complejo en cuanto a curva de aprendizaje

LIBRERÍAS (ReactJS, jQuery...)

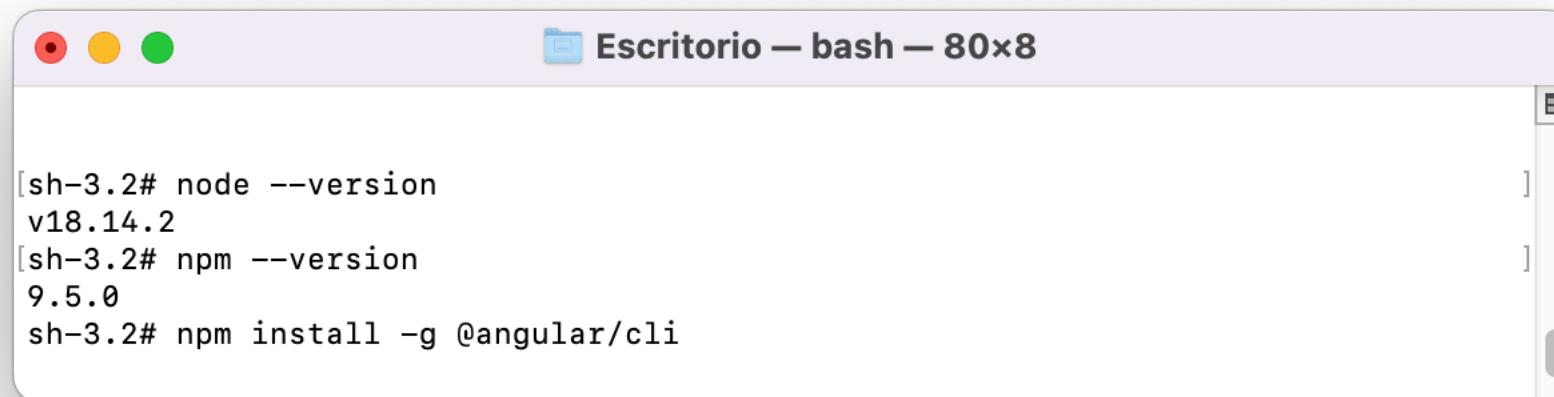
- ✗ Se limita a lo mínimo y más básico, si necesitas otros componentes, necesitas añadir librerías lo cual puede llegar a ser un infierno por las versiones
- ✗ Pueden haber muchas implementaciones muy distintas usando la mismas tecnologías,
- ✓ Muy configurable, puedes elegir que librerías complementan tu app y personalizar como quieras
- ✓ Curva de aprendizaje baja, suele estar acoplada al lenguaje y con pocas abstracciones

Módulo 2. Introducción a Angular

Angular CLI: Instalando Angular

Herramienta de Angular que nos permite, mediante comandos, automatizar ciertas tareas (crear proyectos Angular, añadir librerías mediante schematics, crear componentes, servicios, directivas...)

AngularCLI se puede encargar de compilar, arrancar, empaquetar, lanzar tests y crear librerías dentro de nuestro proyecto. Por debajo usa Webpack para muchas de estas tareas.

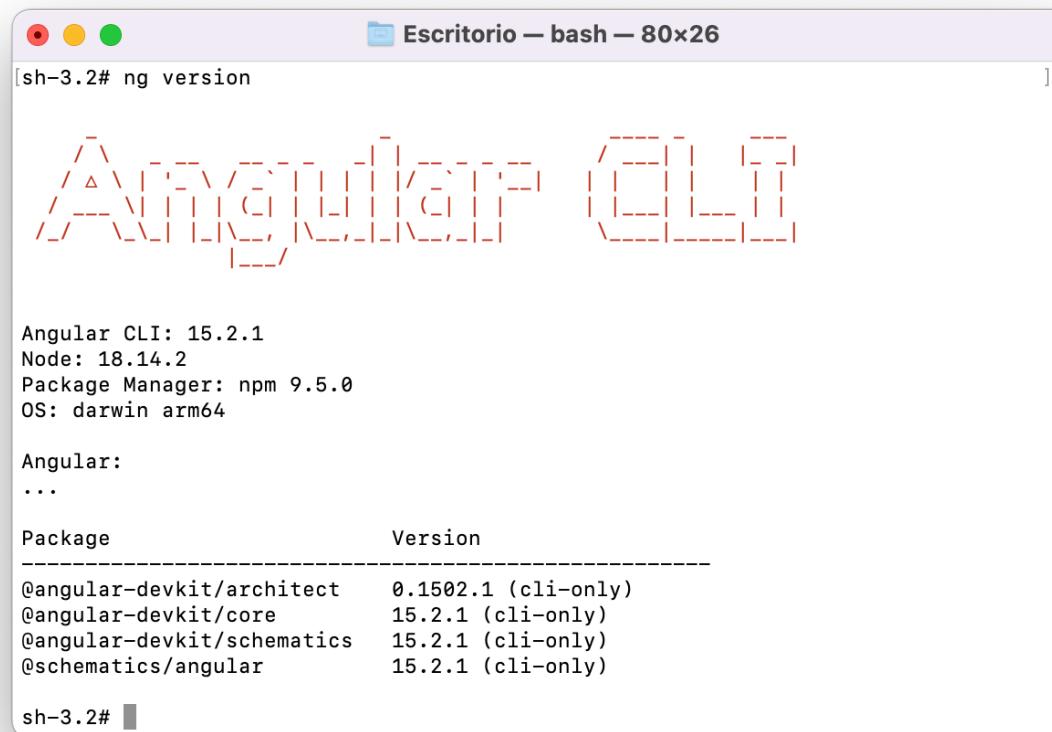


```
[sh-3.2# node --version
v18.14.2
[sh-3.2# npm --version
9.5.0
sh-3.2# npm install -g @angular/cli
```

<https://angular.io/guide/setup-local>

Módulo 2. Introducción a Angular

Angular CLI: versión



A screenshot of a macOS terminal window titled "Escritorio — bash — 80x26". The window shows the output of the command "ng version". The output includes the Angular CLI logo, system information (Angular CLI: 15.2.1, Node: 18.14.2, Package Manager: npm 9.5.0, OS: darwin arm64), Angular version information, and a table of installed packages and their versions.

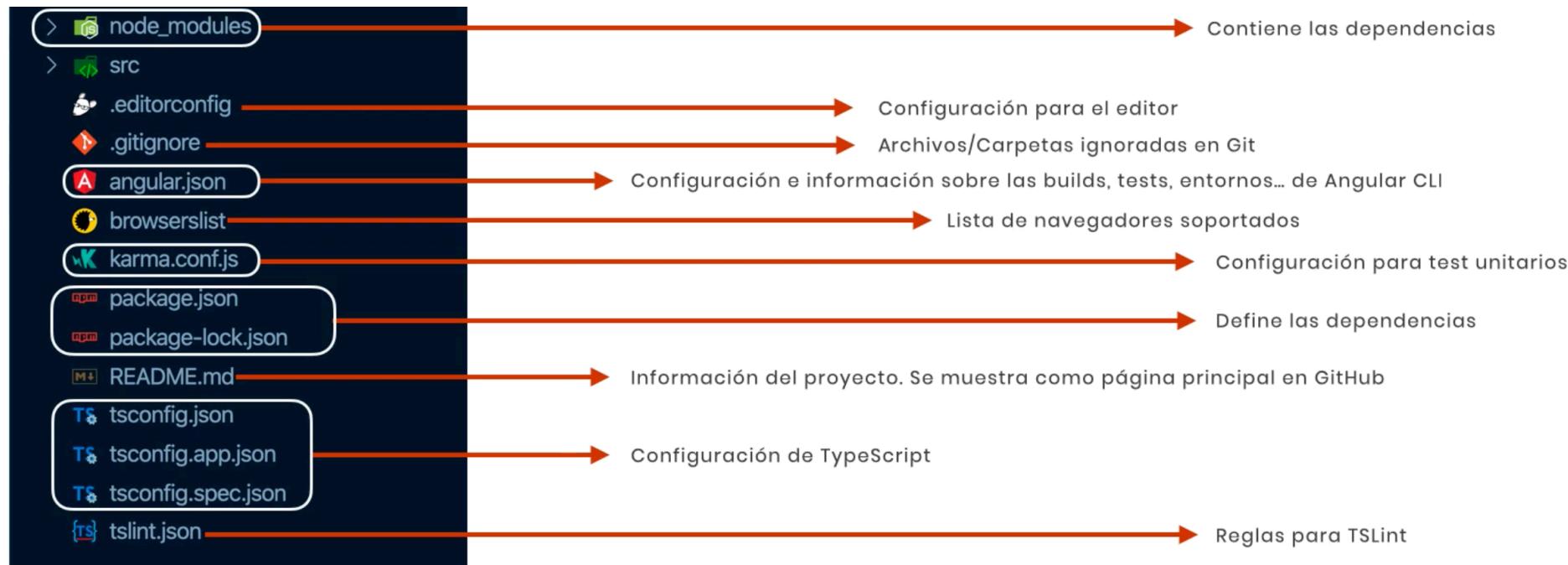
```
Angular CLI: 15.2.1
Node: 18.14.2
Package Manager: npm 9.5.0
OS: darwin arm64

Angular:
...
Package          Version
-----
@angular-devkit/architect    0.1502.1 (cli-only)
@angular-devkit/core         15.2.1 (cli-only)
@angular-devkit/schematics   15.2.1 (cli-only)
@schematics/angular          15.2.1 (cli-only)

sh-3.2#
```

Módulo 2. Introducción a Angular

Estructura de proyecto: root



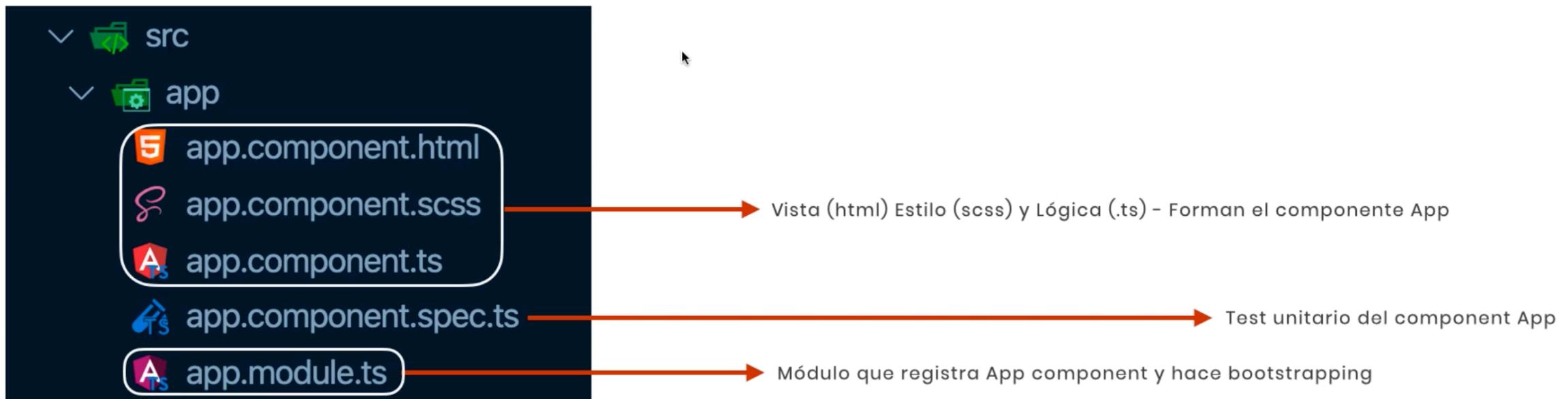
Módulo 2. Introducción a Angular

Estructura de proyecto: src



Módulo 2. Introducción a Angular

Estructura de proyecto: src/app



Módulo 2. Introducción a Angular

Hello World

- Ir al directorio en el que queremos crear el proyecto Angular
- **\$ ng new angular-hello-world**

EXTRA: ¿Cómo actualizar Angular?

- Desinstalar Angular CLI:
 - **\$ npm uninstall -g @angular/cli**
- Forzar el borrado de caché de npm:
 - **\$ npm cache clean --force**
- Verificar la versión de angular para comprobar que se ha desinstalado:
 - **\$ng --version**
- Instalar de nuevo Angular CLI
 - **npm install -g @angular/cli**