

¿Qué es Angular?

Angular es una plataforma de desarrollo construida sobre TypeScript.

Como plataforma incluye

- Un marco basado en componentes para crear aplicaciones web escalables
- Una colección de bibliotecas bien integradas que cubren una amplia variedad de funciones como:
 - El enrutamiento
 - La administración de formularios
 - La comunicación cliente-servidor
- Un conjunto de herramientas para desarrolladores que le ayudarán a desarrollar, crear, probar y actualizar su código.

¿Para qué sirve Angular?

Angular sirve para desarrollar aplicaciones web, dinámicas y de una sola página (SPA) utilizando HTML, CSS y TypeScript. Es un framework front-end completo mantenido por Google, pensado principalmente para aplicaciones empresariales grandes que necesiten estar bien organizadas y escalables.

Historia de Angular

1. **Angular JS 1.x:**

AngularJS generalmente conocido como “Angular.js” o Angular 1.x fue creado por Misko Hevery y el equipo de Google en 2010. Era un framework de JavaScript para desarrollar aplicaciones web dinámicas, y su principal objetivo era facilitar el desarrollo de aplicaciones de una sola página (SPA). AngularJS introdujo conceptos como el enlace bidireccional de datos (two-way data binding) y las directivas.

2. **Angular 2:**

En 2016, Google decidió reescribir AngularJS desde cero para superar sus limitaciones. El nuevo Angular (sin el “JS” en el nombre) fue lanzado como una plataforma completamente nueva y mucho más moderna. La versión 2, que es conocida simplemente como Angular, fue escrita en TypeScript, un superconjunto de JavaScript que ofrece características como tipos estáticos y clases, lo que mejora el rendimiento y la escalabilidad.

3. **Evolución continua:**

Después del lanzamiento de Angular 2, el framework ha seguido evolucionando con nuevas versiones cada seis meses, manteniendo una actualización regular. Cada nueva versión de Angular mejoró el rendimiento, la facilidad de uso y agregó nuevas características. Algunas mejoras clave a lo largo de los años:

- a. **Angular 4 (2017):** Mejora en el rendimiento y reducción del tamaño de la aplicación.

- b. **Angular 5 (2017)**: Integración de herramientas como Angular Material y mejoras en la internacionalización.
- c. **Angular 6 (2018)**: Introducción de CLI mejorado, lazy loading y Angular Elements.
- d. **Angular 7 (2018)**: Mejoras en el rendimiento, la experiencia de desarrollo y el soporte de Angular CLI.
- e. **Angular 8 (2019)**: Soporte para Ivy Renderer (mejora en el rendimiento de la aplicación) y Angular Elements para la creación de componentes personalizados.
- f. **Angular 9 (2020)**: Introducción oficial de Ivy, el nuevo motor de renderizado de Angular, que mejora el rendimiento y la reducción de tamaño.
- g. **Angular 10 (2020)**: Mejoras en la calidad del código y actualizaciones de dependencias.
- h. **Angular 11 (2020)**: Mejoras en el desarrollo de aplicaciones, con una actualización de CLI y herramientas para hacer más fácil la creación de proyectos.
- i. **Angular 12 (2021)**: Introducción de Ivy en más áreas y mejoras en la integración con herramientas como TypeScript 4.2.

Estructura de trabajo en Angular

Los elementos clave en la arquitectura Angular son:

1. **Módulos**: Cada aplicación en Angular tiene un módulo raíz, convencionalmente nombrado AppModule, que **proporciona el mecanismo de arranque que inicia la aplicación**. Una aplicación generalmente contiene muchos módulos funcionales.
2. **Componentes**: Cada aplicación de Angular tiene al menos un componente, el componente raíz que conecta una jerarquía de componentes con el modelo de objetos del documento de la página (DOM). **Cada componente define una clase que contiene datos y lógica de la aplicación, y está asociado con una plantilla HTML que define una vista que se mostrará en un entorno de destino.**
3. **Servicios e inyección de dependencia**: Para los datos o la lógica que no están asociados con una vista específica y que deseas compartir entre componentes, crea una clase servicio.
4. **Enrutamiento**: Define las rutas de navegación de la aplicación.
5. **Plantillas, directivas y enlace de datos**: **Una plantilla es básicamente el HTML que usa un componente**, pero no es un HTML común: puede contener cosas especiales de Angular que cambian el contenido dinámicamente. Las directivas son instrucciones que Angular usa para cambiar cómo se ve o se comporta un elemento. **Las directivas son instrucciones especiales que Angular usa en el HTML para decirle cómo comportarse. El enlace de datos es la forma en que Angular conecta los datos del componente con lo que ves en la pantalla.**

Diferencias AngularJS y Angular

El relativo gran número de versiones de Angular puede crear confusión con las versiones; pero para aclarar esas dudas debemos volver al pasado, específicamente al año 2010. En este año fue lanzado Angular en su versión 1.0, conocido como AngularJS y dentro de poco tiempo se convirtió en uno de los proyectos de código abierto en Javascript más usado por compañías, tanto para desarrolladores frontend como entusiastas del desarrollo web.

Los siguientes puntos distinguen Angular de Angular JS:

- En los años 2009 y 2014 se vio el desarrollo de Angular JS y Angular, respectivamente.
- Mientras que AngularJS se basa en JavaScript, Angular está construido sobre TypeScript.
- Mientras AngularJS emplea un conjunto de directivas, Angular utiliza directivas estándar.
- AngularJS utiliza el framework MVC, mientras que Angular se adhiere a una arquitectura basada en componentes.
- Mientras que AngularJS hace uso de ámbitos y controladores, Angular sigue la jerarquía de componentes.
- A diferencia de AngularJS, que sólo admite la vinculación de datos unidireccional, Angular permite la vinculación de datos unidireccional y bidireccional.
- A diferencia de AngularJS, que carece de una interfaz de línea de comandos, Angular viene con una.
- AngularJS carece del mismo soporte móvil que Angular, que es proporcionado por Ionic y NativeScript.

Ventajas de Angular

1. **Arquitectura basada en componentes:** Facilita la reutilización y el mantenimiento del código.
2. **TypeScript:** Angular está escrito en TypeScript, lo que permite un desarrollo más estructurado, con tipado estático y herramientas avanzadas como autocompletado y detección de errores.
3. **Desarrollo escalable:** Ideal para aplicaciones grandes y de empresa gracias a su estructura modular.
4. **Soporte para pruebas (Testing):** Angular está diseñado con testing en mente y ofrece herramientas como Karma y Jasmine para pruebas unitarias y de integración.
5. **Soporte oficial y comunidad activa:** Desarrollado por Google y con una gran comunidad, lo que garantiza actualizaciones, mejoras y soluciones constantes.
6. **Framework integral y estructurado:** Angular tiene todo lo necesario para desarrollar aplicaciones complejas y de gran escala: inyección de dependencias, módulos, routing, formularios, validaciones, pruebas, HTTP, animaciones, internacionalización, etc. Angular brilla cuando necesitas una solución robusta, todo-en-uno y bien estructurada para proyectos grandes.

Desventajas de Angular

1. **Curva de aprendizaje empinada:** Su complejidad puede ser abrumadora para los que van empezando, especialmente por su uso de TypeScript, decoradores, directivas y servicios.
2. **Peso del framework:** Angular es más pesado comparado con alternativas como React o Vue, lo cual puede afectar el rendimiento inicial en apps pequeñas.
3. **Rendimiento en apps muy dinámicas:** Aunque Angular es rápido, en aplicaciones con muchos cambios en tiempo real puede resultar menos eficiente que frameworks más livianos.
4. **Migraciones entre versiones mayores:** Aunque Angular ha mejorado en esto, actualizar entre versiones mayores puede ser laborioso y requiere planificación.
5. **Mayor consumo de recursos:** Puede requerir más memoria y procesamiento, especialmente en dispositivos móviles o antiguos.
6. **SEO limitado por defecto:** El HTML inicial está vacío (<app-root>), lo que afecta negativamente el posicionamiento en buscadores. No es ideal para sitios que dependen fuertemente del SEO (como blogs, e-commerce o landing pages públicas).

¿Qué se necesita para correr un proyecto angular?

Requisitos previos:

- **Node.js y npm:** Node.js es necesario para ejecutar Angular y sus herramientas. Al instalar Node.js, también se instala npm (el gestor de paquetes).
Link de descarga: <https://nodejs.org/es>

Instalar Angular CLI:

Primero instalemos Angular-Cli, ejecutamos en la terminal:

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

El -g significa que se instalará globalmente en el sistema.

Crear el proyecto:

Ahora podemos crear un proyecto angular utilizando el comando “ng new”, en este caso, nombraremos el proyecto “my-proyecto-angular”.

```
ng new my-proyecto-angular
```

Ejecutar la aplicación:

```
ng serve --open
```

Usando la opción -- open (o simplemente -o) nos abrirá automáticamente la aplicación en el navegador en “http://localhost:4200”

Referencias: (Al último las pasamos a APA)

Angular. (n.d.).

https://v17-angular-io.translate.google.com/guide/what-is-angular?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc&_x_tr_hist=true

campusMVP. (2019, November 18). Las 10 principales diferencias entre AngularJS y Angular. campusMVP.es.

<https://www.campusmvp.es/recursos/post/las-10-principales-diferencias-entre-angularjs-y-angular.aspx>

Singh, R. P., & Singh, R. (2024, October 17). Angular vs AngularJS: conozca la diferencia. Richestsoft. <https://richestsoft.com/es/blog/angular-vs-angularjs-know-the-difference/>

Radkovskyy, B., & Muzyka, R. (2024, September 2). Why to use Angular: An overview of angular pros and cons. Leobit.

<https://leobit.com/blog/why-to-use-angular-in-an-overview-of-angular-pros-and-cons/>

Angular. (n.d.-b). <https://docs.angular.lat/guide/architecture>

Nandhan, S. (2023, March 28). Angular History - Saravanan Nandhan - Medium. Medium. <https://medium.com/@sarancruzer/angular-history-293515a16524>

Bastidas, W. (2020, July 23). Angular : Inicio rápido con angular-cli y angular material. Medium.

<https://medium.com/williambastidasblog/angular-5-inicio-r%C3%A1pido-con-angular-cli-y-angular-material-e52e4ab0df3a>