



POLITÉCNICO COLOMBIANO
JAIME ISAZA CADAVID

Distintos de tipos de

FRAMEWORKS

Realizado por
Sebastian Zapata Castaño

Dirigido por
Hector Manuel Vanegas Solis

Universidad
Politecnico Colombiano Jaime Isaza Cadavid

Febrero, 2022

1. Angular

Angular (comúnmente llamado Angular 2+ o Angular 2) es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

La biblioteca lee el HTML que contiene atributos de las etiquetas personalizadas adicionales, entonces obedece a las directivas de los atributos personalizados, y une las piezas de entrada o salida de la página a un modelo representado por las variables estándar de JavaScript.

Angular se basa en clases tipo “Componentes”, cuyas propiedades son las usadas para hacer el binding de los datos. En dichas clases tenemos propiedades (variables) y métodos (funciones a llamar).



1.1. Arquitectura de angular

Una de las grandes diferencias entre un framework y una biblioteca, es que el framework es un montón de funcionalidades “genéricas” preparadas para que noso-

tros hagamos funcionalidad específica. En cambio, una biblioteca es una única función genérica. Basados en esta definición podemos decir que un framework consiste de varias bibliotecas escritas para manejarse todas juntas. Bajo este pensamiento, podemos decir que Angular preparó todo para que nuestra aplicación solo utilice los módulos (o bibliotecas) que vamos a necesitar en nuestra WebApp.

Con este pensamiento, cuando iniciamos a construir nuestra aplicación con Angular, solo vamos a tener el módulo principal llamado “core”, con él vamos a poder ejecutar nuestra aplicación y escribir cada uno de nuestros componentes. Si nuestra aplicación necesita generar rutas tenemos que agregar el módulo de ruteo, que Angular ya nos provee, y si necesitáramos agregar formularios, Angular también tiene un módulo increíble para eso. A estas alturas podrás estarte preguntando cuántos módulos/bibliotecas tiene Angular internamente, y la verdad es que son muchas pero la mayoría son independientes entre sí por lo que nuestra webapp solo necesita importar los módulos necesarios para poder trabajar.

Para construir aplicaciones Angular creamos:

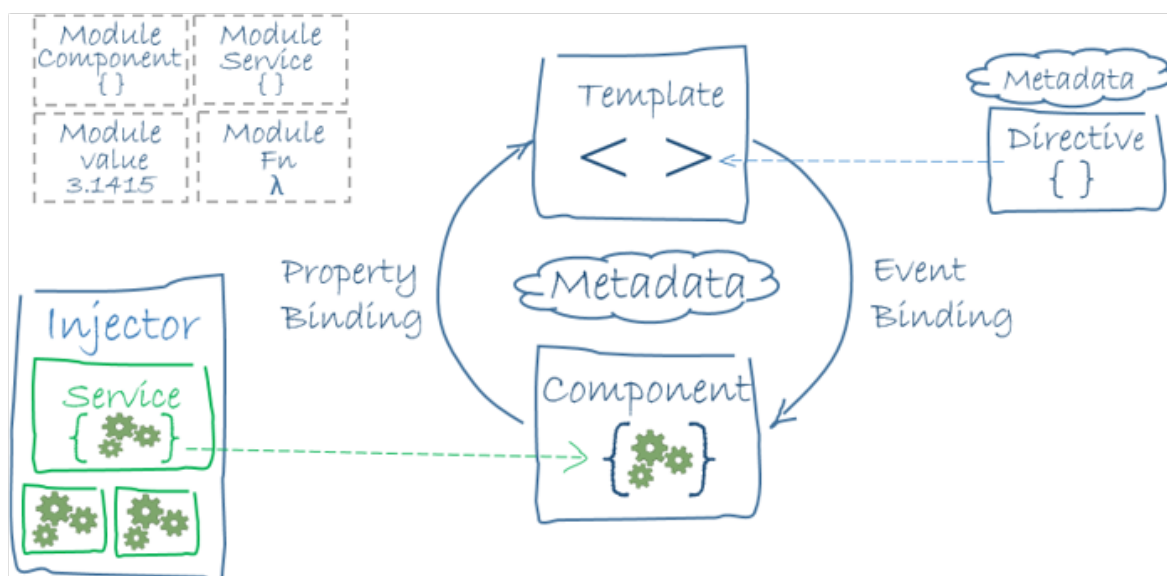
templates HTML que contienen etiquetas especiales de Angular,

componentes de clase que gestionan dichos templates,

servicios que encapsulan lógica de la aplicación,

y módulos que organizan los componentes y servicios.

Iniciamos la aplicación en el módulo “root”, y Angular asume el control, presentando el contenido de la aplicación en el navegador y respondiendo a las interacciones del usuario de acuerdo con las instrucciones que hemos proporcionado



1.2. Principales características de Angular

Velocidad y rendimiento

- Generación de código: Angular convierte tus plantillas en código altamente optimizado para las máquinas virtuales de JavaScript de hoy en día, ofreciéndote todas las ventajas del código escrito a mano con la productividad de un framework.
- Universal: Ejecuta la primera vista de tu aplicación en node.js, .NET, PHP, y otros servidores para renderizado de forma casi instantánea obteniendo solo HTML y CSS. También abre posibilidades para la optimización del SEO del sitio, incluyendo configuración.
- División del código: Las aplicaciones de Angular se cargan rápidamente gracias al nuevo enrutador de componentes. Este ofrece una división automática de códigos para que los usuarios solo carguen el código necesario para procesar la vista que solicitan.

Productividad

- Plantillas: Permite crear rápidamente vistas de interfaz de usuario con una sintaxis de plantilla simple y potente.
- Angular CLI: Las herramientas de línea de comandos permiten empezar a desarrollar rápidamente, añadir componentes y realizar test, así como previsualizar de forma instantánea la aplicación.
- IDEs: Obtén sugerencias de código inteligente, detección de errores y otros comentarios en la mayoría de los editores populares e IDEs.

Historia completa del desarrollo

- Testing: Utiliza Karma para realizar pruebas unitarias, y Protractor para realizar pruebas end-to-end de forma rápida, estable y eficaz.
- Animación: Permite crear animaciones complejas y de alto rendimiento con muy poco código a través de la intuitiva API de Angular.
- Accesibilidad: Posee características para crear aplicaciones accesibles con los componentes disponibles para ARIA.

2. Vue.js Framework

Vue.js (también llamado Vue o VueJS) es un framework progresivo Javascript para crear interfaces de usuario, o en otras palabras, se trata de una «capa» añadida a Javascript formada por herramientas, convenciones de trabajo y un lenguaje particular que nos permite crear aplicaciones de forma rápida, agradable, sencilla y muy práctica.



2.1. Arquitectura de VueJS

Vue.js cuenta con una arquitectura de adaptación gradual que se centra en la representación declarativa y la composición de componentes. La biblioteca central se centra sólo en la capa de vista. Las características avanzadas necesarias para aplicaciones complejas como el enrutamiento, la gestión de estados y las herramientas de construcción se ofrecen a través de librerías y paquetes de apoyo mantenidos oficialmente, con Nuxt.js como una de las soluciones más populares.

Vue.js permite extender el HTML con atributos HTML llamados directivas. Las directivas ofrecen funcionalidad a las aplicaciones HTML, y vienen como directivas incorporadas o definidas por el usuario.

2.2. Principales características de VueJS

- La curva de aprendizaje es, con diferencia, la más sencilla de los tres frameworks más populares: React, Vue y Angular.
- Se trata de un framework muy amigable y respetuoso con las tecnologías de frontend y los estándares. Utiliza HTML, CSS y Javascript y es compatible con

WebComponents (de hecho, sus componentes se basan en ellos). Si eres desarrollador con base fuerte de HTML/CSS, muy probablemente te guste más Vue que otras opciones.

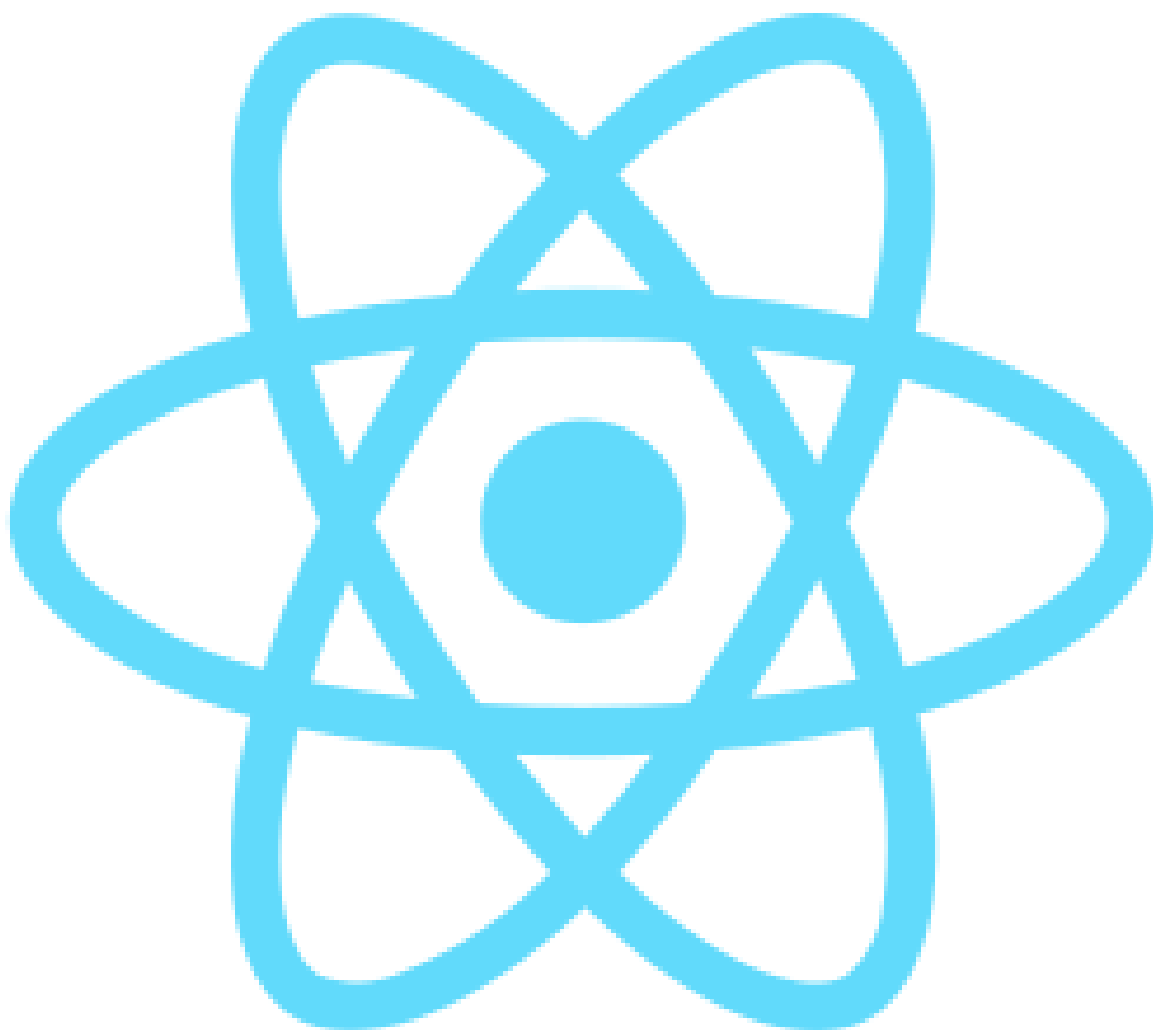
- Se trata de un framework progresivo. Esto significa que es ideal para migrar y adaptar proyectos existentes hechos en otras tecnologías y pasarlos poco a poco a Vue. Algo muy común cuando trabajas en proyectos legacy (la mayoría de los casos).
- A los desarrolladores que provienen de lenguajes o frameworks exclusivamente de backend están acostumbrados a ciertos patrones de programación que no son exactamente iguales en frontend. Es aconsejable aprender ciertas bases de frontend general. Una buena base de Javascript también es muy recomendable.
- Vue le da mayor protagonismo al enfoque tradicional «centrado en HTML» así como a los sistemas de plantillas. Si te gustan, Vue probablemente te resulte muy atractivo. Por otro lado, el enfoque de React se suele centrar más en programación pura en Javascript, utilizando HTML y CSS sólo como complementos que se añaden a Javascript.

3. React Framework

React (también llamada React.js o ReactJS) es una biblioteca Javascript de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página. Es mantenido por Facebook y la comunidad de software libre. En el proyecto hay más de mil desarrolladores libres.

React intenta ayudar a los desarrolladores a construir aplicaciones que usan datos que cambian todo el tiempo. Su objetivo es ser sencillo, declarativo y fácil de combinar. React sólo maneja la interfaz de usuario en una aplicación; React es la Vista en un contexto en el que se use el patrón MVC (Modelo-Vista-Controlador) o MVVM (Modelo-vista-modelo de vista). También puede ser utilizado con las extensiones de React-based que se encargan de las partes no-UI (que no forman parte de la interfaz de usuario) de una aplicación web.

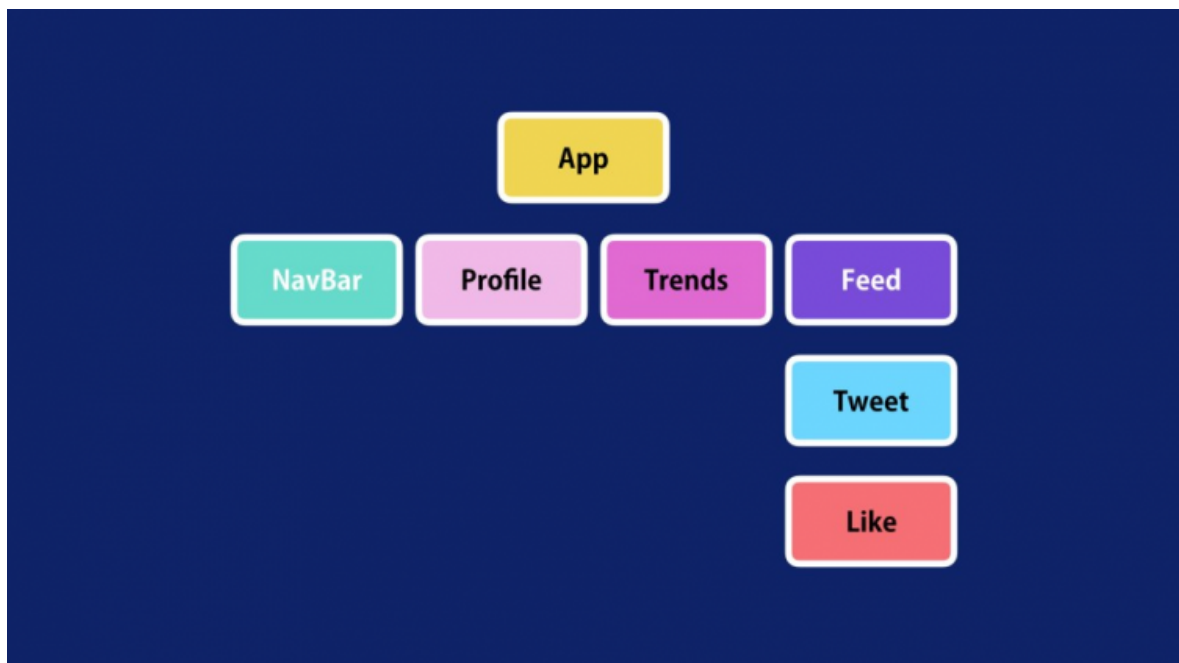
Según el servicio de análisis JavaScript (en inglés "JavaScript analytics service"), Libscore, React actualmente está siendo utilizado en las páginas principales de Imgur, Bleacher Informe, Feedly, Airbnb, SeatGeek, HelloSign, entre otras.



3.1. Arquitectura de React

El elemento más importante de React es el componente, que es, en esencia, una pieza de la interfaz de usuario. Como norma general, al diseñar una aplicación con React, lo que estamos haciendo es crear componentes independientes y reusables para, poco a poco, crear interfaces de usuario más complejas.

En ese sentido, React es similar a Angular, ya que toda aplicación en React tiene al menos un componente; al que normalmente nos referimos como componente «raíz», que contiene otros componentes “hijos”, y estos a su vez otros, etc. De esta manera, se puede afirmar que la vista es un árbol de componentes, como se muestra en el siguiente diagrama de ejemplo:



3.2. Características de React

- Composición de componentes.
- Desarrollo Declarativo Vs Imperativo.
- Flujo de datos unidireccional.
- Performance gracias al DOM Virtual.
- Isomorfismo
- Elementos y JSX.

- Componentes con y sin estado.
- Ciclo de vida de los componentes.
- Ideal para aplicaciones de alta demanda.
- Permite el desarrollo de aplicaciones móviles.

¿Cual es la principal diferencia entre estos tres Frameworks?

Angular frece de todo, desde enrutamiento hasta plantillas. No tienes que mudarte a cualquier otra herramienta para desarrollar sus aplicaciones. Esto puede ser bueno o malo de acuerdo a tu perspectiva. Tiene opiniones sobre cómo estructurar una aplicación. No necesita agregar librerías de routing. Es decir se puede comenzar a desarrollo con todo lo que se proporciona en el paquete Angular sin necesidad de agregar nada adicional.

A diferencia de Angular, no proporciona mucho en la biblioteca oficial de React. Esta te brinda la libertad de elegir las herramientas de programación que prefiera. En otras palabras, no es tan obstinado como Angular. Las soluciones de terceros como el enrutamiento de reacción se pueden integrar con este Framework de JavaScript. Aparte de esto, también puede usar MobX y Redux para apoyar las tareas de administración. Si contás con un equipo de Desarrollo con mucha experiencia, entonces disfrutarán mucho de esta flexibilidad. Además, obtendrás una increíble aplicación dinámica desarrollada para su negocio.

El Framework Vue es tan básico como el Framework de react. Pero el ecosistema Vue en su sitio web oficial tiene mucho que ofrecer a sus desarrolladores. Algunos de los complementos son Vue Router para enrutamiento y para la administración de estados ofrecen Vuex. También tiene un render vue del lado del servidor para iniciar el desarrollo del lado del servidor de una aplicación. Por lo tanto, no es tan obstinado como Angular, pero no es tan flexible como React.

ANGULAR VS REACT VS VUE : LEARNING CURVE



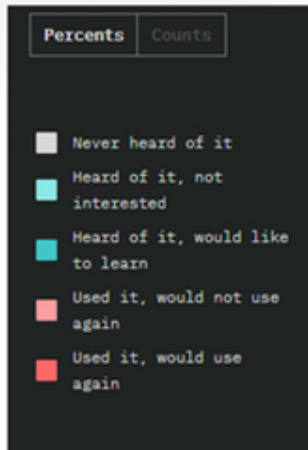
REACT



VUE



ANGULAR



Referencias

- Angular: Mucho más que un framework. (n.d.). SG Buzz. Retrieved February 21, 2022, from <https://sg.com.mx/revista/56/angular>
- ¿Qué es ReactJS? Conceptos básicos. (2020, March 26). IfgeekthenNTTdata. <https://ifgeekthen.nttdata.com/es/que-es-reactjs-conceptos-basicos>
- ¿Qué es Vue? (n.d.). Lenguajejs.com. Retrieved February 21, 2022, from <https://lenguajejs.com/v-es-vue/>
- React. (n.d.). Reactjs.org. Retrieved February 21, 2022, from <https://es.reactjs.org>
- Wikipedia contributors. (n.d.-a). Angular (framework). Wikipedia, The Free Encyclopedia. [https://es.wikipedia.org/w/index.php?title=Angular\(framework\)&oldid=137371139](https://es.wikipedia.org/w/index.php?title=Angular(framework)&oldid=137371139)Wikipediacontributors.(n.d.-b).Vue.js.Wikipedia,TheFreeEncyclopedia.https://es.wikipedia.org/w/index.php?title=Vue.js&oldid=141651147
- (N.d.). Medium.Com. Retrieved February 21, 2022, from <https://medium.com/somoswigou/angular-vs-react-vs-vue-cuál-es-la-mejor-opción-941a207951c7>