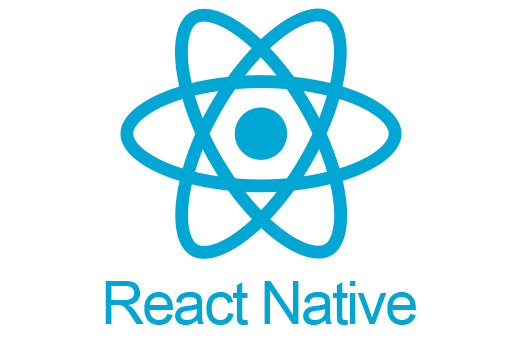
**FRAMEWORKS DE DESARROLLO WEB E HÍBRIDO EN APLICACIONES MÓVILES**

**React Native** combina las mejores partes del desarrollo nativo con React, la mejor biblioteca de JavaScript de su clase para crear interfaces de usuario.

Puede usar React Native hoy en sus proyectos existentes de Android y iOS o puede crear una aplicación completamente nueva desde cero.

****

Facebook lanzó React Native en 2015 y lo ha mantenido desde entonces. En 2018, React Native tuvo el segundo mayor número de contribuyentes para cualquier repositorio en GitHub.

**Ventajas**

* **Vea sus cambios tan pronto como guarde.**Con el poder de JavaScript, React Native te permite iterar a la velocidad del rayo. Ya no tendrá que esperar a que finalicen las compilaciones nativas. Guardar, ver, repetir.
* Los componentes de React envuelven el código nativo existente e interactúan con las API nativas a través del paradigma de IU declarativo de React y JavaScript. Esto permite el desarrollo de aplicaciones nativas para equipos de desarrolladores completamente nuevos y puede permitir que los equipos nativos existentes trabajen mucho más rápidos.
* **Desarrollo rentable**: React Native ofrece a los desarrolladores una vía económica para crear aplicaciones multiplataforma con React Native. En lugar de crear dos aplicaciones diferentes para Android y iOS, el desarrollador puede implementar el mismo código para ambas plataformas.
* **Comunidad de desarrolladores activa**: El sitio web Statista muestra que React Native es el marco de desarrollo de aplicaciones multiplataforma más popular hoy en día, con una cuota de mercado del 42% y 90k en Github.

**Desventajas**

* **Desafíos de compatibilidad y depuración**: A pesar de todas las excelentes características, ¡React Native todavía está en la versión Beta! Es por eso que todavía tiene algunos problemas aparentes, como la complejidad de depurar aplicaciones, entre otras limitaciones, incluidos los problemas de compatibilidad.
* **La gestión de la memoria no es sobresaliente**: React Native crea aplicaciones con excelentes características. Sin embargo, React Native podría no ser la mejor plataforma para crear aplicaciones que administren los recursos de hardware de manera eficiente.
* **Aún necesita desarrolladores nativos**: Aunque la experiencia en JavaScript puede llevarlo lejos en el desarrollo de React Native, aún necesita el servicio de desarrolladores nativos para implementar ciertas funciones de la aplicación.

**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 primer lanzamiento fue el 15 de septiembre de 2016.

Sus principales características son:



* **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.

**Ventajas:**

* **Implementación de la arquitectura MVC**: La arquitectura Modelo-Vista-Controlador, no solo otorga valor al marco al crear una aplicación del lado del cliente, sino que también sienta las bases para otras características como el enlace de datos y los alcances.

### ****Arquitectura de diseño mejorada:****Algunas de las grandes aplicaciones web contienen muchos componentes. Angular simplifica la forma de administrar estos componentes incluso si un nuevo programador se une al proyecto después de que el proceso de desarrollo ya ha comenzado.

### ****TypeScript:** mejores herramientas, código más limpio y mayor escalabilidad** Angular está escrito usando TypeScript, que es un superconjunto para JavaScript. Cumple plenamente con ****JavaScript**** y también ayuda a detectar y eliminar errores comunes durante la codificación.

### ****Desventajas:****

### Complejidad: Angular es un marco complejo y puede ser difícil de aprender para los desarrolladores que son nuevos en los marcos de JavaScript.

### Rendimiento: las aplicaciones angulares pueden ser lentas, especialmente en comparación con otros marcos como React o Vue.

### Documentación deficiente: la documentación de Angular no es tan completa

### Curva de aprendizaje pronunciada: Angular tiene una curva de aprendizaje pronunciada y requiere que los desarrolladores tengan una buena comprensión de TypeScript, HTML y CSS.

**Flutter** es un SDK de código fuente abierto de desarrollo de aplicaciones móviles creado por Google. Suele usarse para desarrollar interfaces de usuario para aplicaciones en Android, iOS y Web, así como método primario para crear aplicaciones para Google Fuchsia.

### 

### Su lanzamiento inicial se llevó a cabo el 11 de mayo de 2017 y en este ultimo año ha tenido un crecimiento significativo en cuanto a su popularidad, fue programado en Dart y C++, admite plataformas como Android, iOS, Google Fuchsia, Web platform, Linux, macOS y Windows.

### Ventajas:

* **Recarga caliente:** como comentábamos antes, al hacer algún cambio en el código se podrán ver los efectos reflejados inmediatamente, sin tener que compilar la aplicación de nuevo y sin perder el contexto en el que estábamos.
* **Renderizado de vistas muy rápido y constante:** Flutter se ha planteado objetivos de renderizado muy altos para ciertos dispositivos, lo que supera con creces a cualquiera otra solución de desarrollo móvil híbrido.
* **Desarrollo multiplataforma**: no es necesario construir por separado para las dos plataformas: Android y IOS. Flutter ya genera un código base que sirve para ambas plataformas.
* **Acceso a las funciones nativas**: algunas funciones específicas de la plataforma, como la cámara y la geolocalización, requieren acceso a funciones nativas. Estas funciones deben implementarse mediante lenguajes nativos, y Flutter da la sensación de desarrollarse en la plataforma nativa. Flutter permite reutilizar código existente de Java, Swift y Objective-C para acceder a las funciones nativas y SDK en iOS y Android.

**Desventajas:**

* **Dart necesario:** para poder usar Flutter es necesario aprender el lenguaje de programación Dart.
* **Framework muy joven:** y aún no tiene una gran comunidad detrás, por lo que se deberán afrontar los problemas que nos encontremos con menos ayuda que en otros frameworks.
* **Está enfocado solo a móvil**: por el momento solo hay una versión oficial de Flutter y solo está enfocada para móvil. Así si nuestra aplicación va a tener un sitio web tendremos que desarrollarlo paralelamente a la versión de móvil.
* **Librerías limitadas**: las bibliotecas a las que pueden acceder los desarrolladores de aplicaciones móviles están muy limitadas en Flutter. No siempre proporcionan todas las funcionalidades que necesita el desarrollador. Dichas funcionalidades deben ser desarrolladas por los desarrolladores de aplicaciones por sí mismas.