

¿Qué son las aplicaciones híbridas?

Inicios

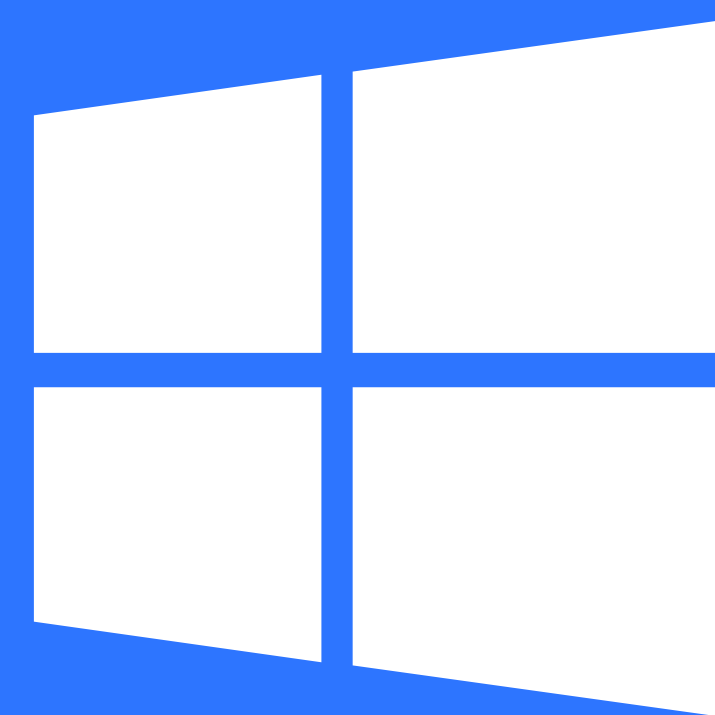
Software Development Kit **SDK**



Swift
Objective-C



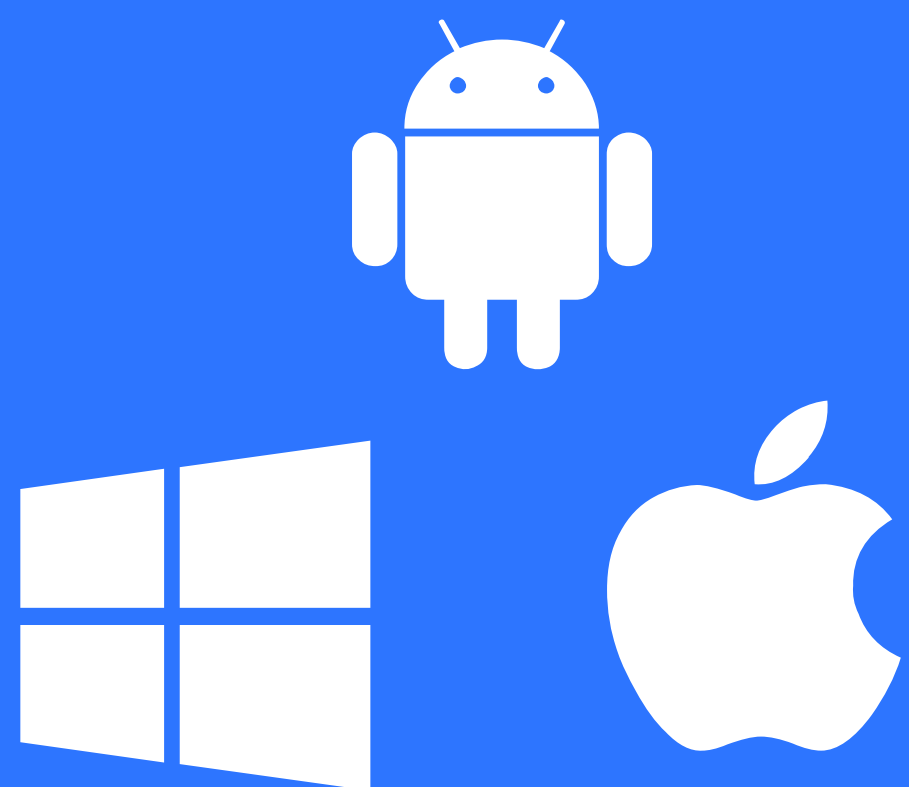
Java



C#

Problemas

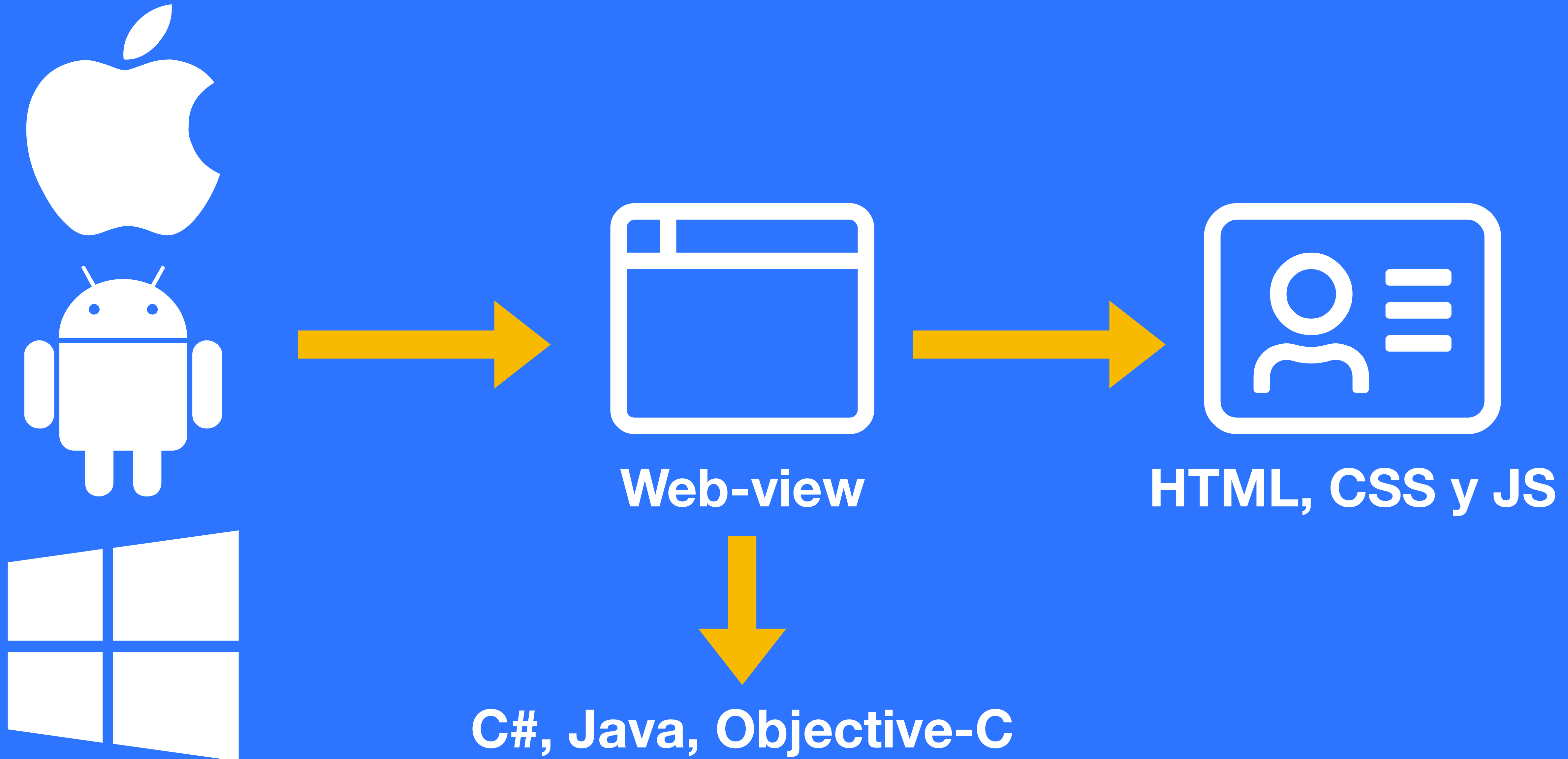
- **Tiempos de desarrollo para las 3 plataformas**
- **Manejar múltiples bases de código**
- **Contratar desarrolladores especializados en código nativo**



Para el 2022, el 70% de las interacciones en la web
serán en dispositivos móviles

-Gartner

Aplicaciones Híbridas



Aplicaciones Híbridas

¿Por qué seleccionarlás?

1- Velocidad:

Codificar en una única base de código hace que puedas entregar aplicaciones a múltiples plataformas entre 2-3 veces más rápido que el nativo

2- Mejor soporte:

Una aplicación híbrida tiende a ser más fácil de dar soporte sin contar que no necesitas personal especializado en código nativo

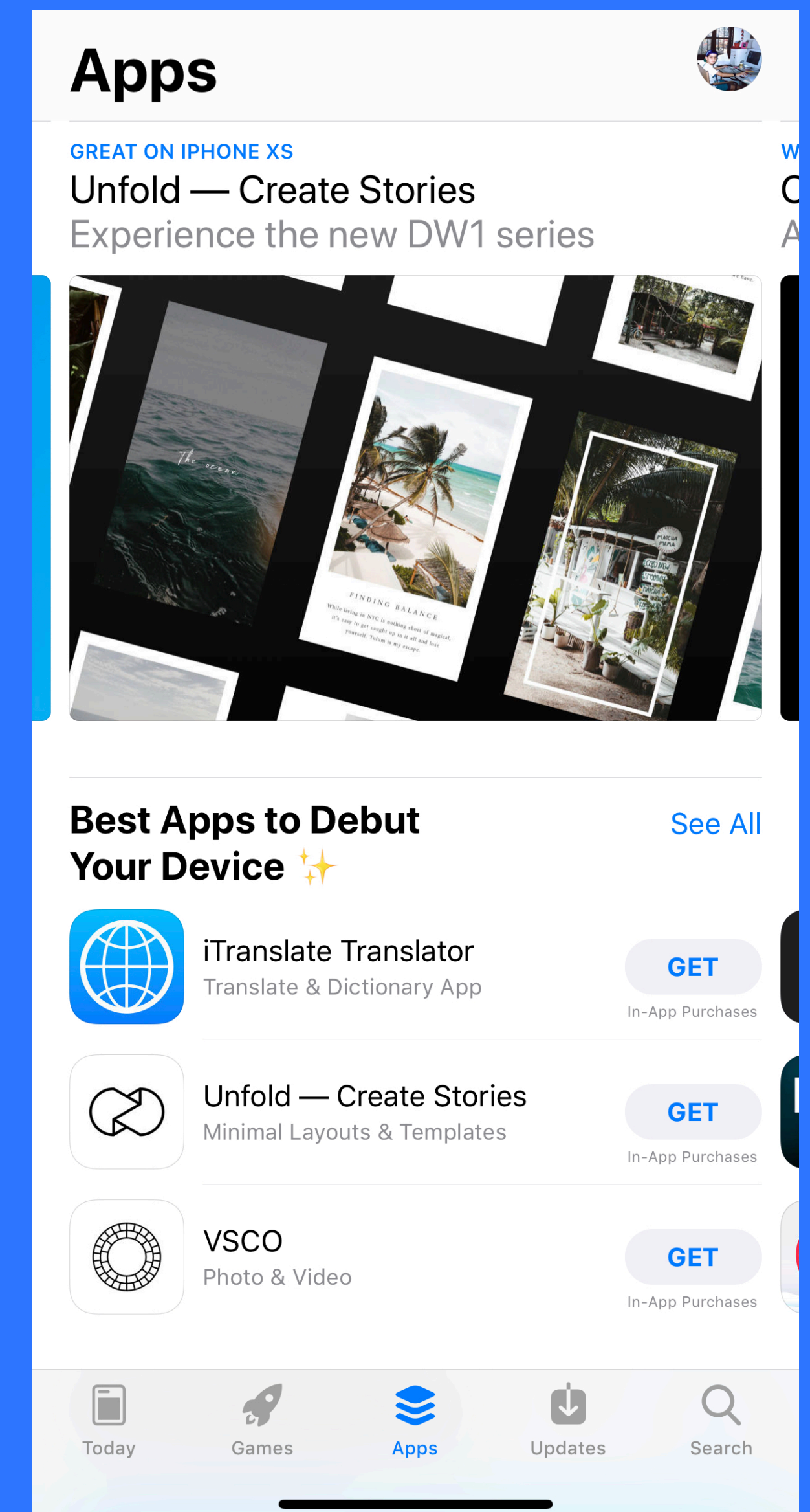
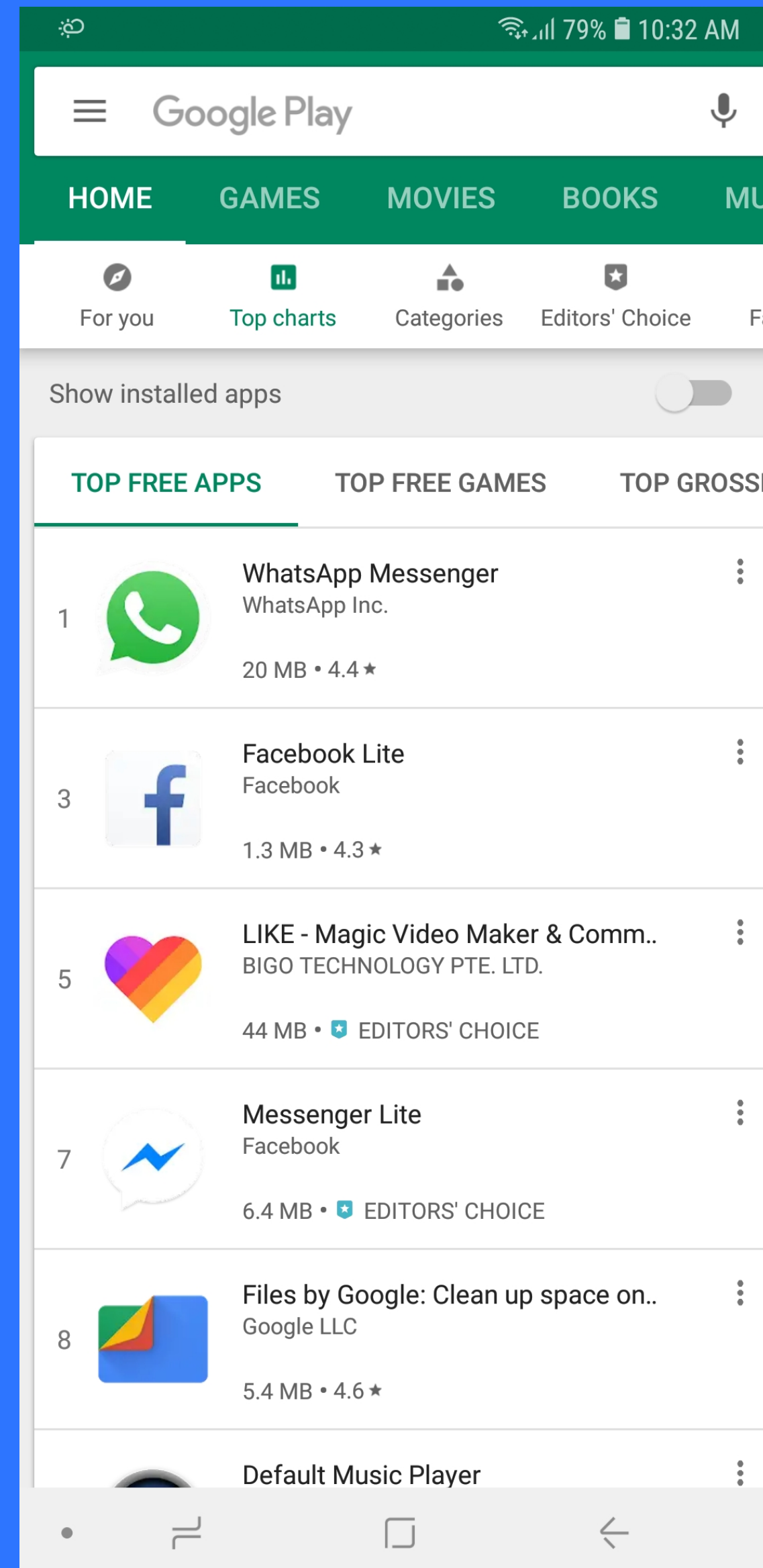
3- Un único canal:

Las aplicaciones híbridas pueden correr en donde sea que la web corra. En un desktop, móvil, una PWA, carros, etc.

¿Qué es una aplicación híbrida?

Las aplicaciones híbridas son esencialmente aplicaciones nativas.

Pueden acceder a los mismos recursos nativos y tienen performance acelerado por hardware como si fuera creada con su SDK nativo.



Híbrido vs Nativo

Importante

Es imprescindible tener en mente que la decisión de escoger híbrido o nativo debe de ser basada en los objetivos de tu organización.



Le apuesta al gran poder de la web y su potencial de ser multi-plataforma en un ambiente híbrido de desarrollo.

**Espero que lo que te explique a continuación,
te ayude a escoger la mejor aproximación
para tu siguiente proyecto.**

Usa el talento que tu ya tienes

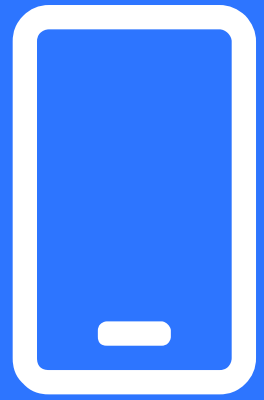
La comunidad de desarrolladores web es 30 veces más grande que la comunidad de desarrolladores de apps nativas 2017- Stack overflow survey

Muchos equipos de desarrollo YA tienen programadores que entienden HTML, CSS y JavaScript



Eso es más fácil que buscar, entrenar y contratar especialistas, sin contar que no centralizas el conocimiento en una única persona

¿Por qué híbrido?



La mejor interfaz de usuario entre plataformas

Así como las aplicaciones nativas, las aplicaciones híbridas hechas en ionic también son aceleradas por hardware.

Una consistente experiencia de usuario entre plataformas, dispositivos móviles y otros modos de interacción incluyendo navegadores web.

¿Por qué híbrido?

Construyendo para el futuro

Las aplicaciones del futuro deberán correr en un creciente número y diversidad de plataformas



Afortunadamente, la web es la forma más usada de correr aplicaciones en el mundo

¿Por qué híbrido?

Inconvenientes de las aplicaciones híbridas

1- Mayor consumo de recursos

- El uso del webview puede introducir un grado mayor de consumo en comparación de una app nativa.
- La abundancia de Apis y el crecimiento de poder de los dispositivos hacen que este factor sea de menos consideración cada año.
- Pero para juegos 3D o uso de 3D en general o bien aplicaciones de performance intensivo, lo híbrido no es la mejor opción.

2- Plugins de terceros

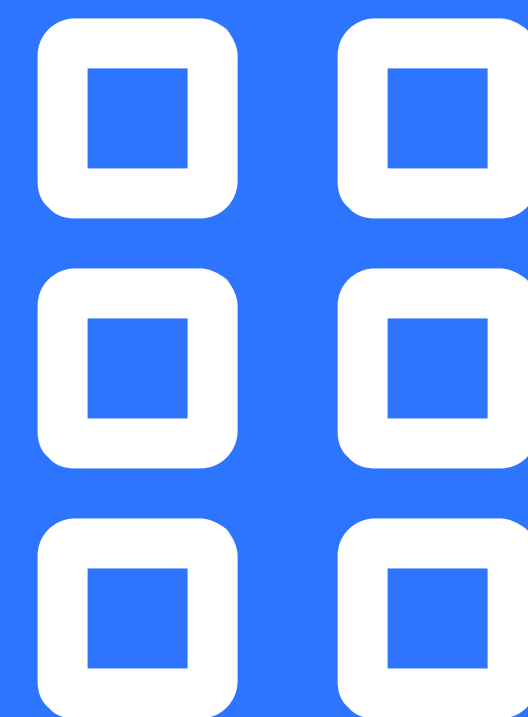
- Las aplicaciones híbridas pueden acceder a casi todas las características nativas del dispositivo, tales como la cámara o el giroscopio usando plugins.
- Plugins open-source como los de Cordova son muy populares, esto añade un nivel adicional de complejidad a nuestra aplicación que muchos lo consideran problemático.

3- Dependencias de frameworks

- Escoger una aproximación multi-plataforma significa que pondrás tu confianza en un framework para mantener las mejores prácticas, características y patrones de desarrollo para cada plataforma.
- ionic esta comprometido en mantenerse con las últimas versiones de Android e IOS



Performance



**Gran librería de
recursos nativos**



**Cero dependencias
de terceros**

¿Por qué Nativo?

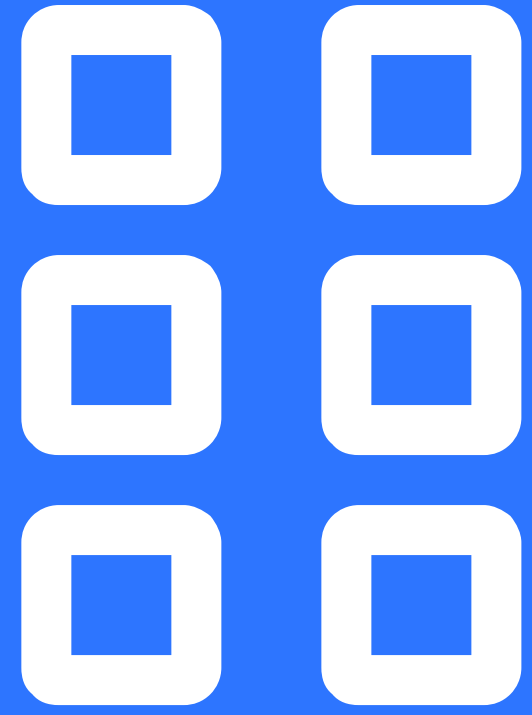


Performance

El código nativo sigue siendo más rápido que JavaScript y el HTML.

Esto es importante cuando el desarrollador busca crear aplicaciones de alta demanda gráfica como juegos o animaciones intensivas.

¿Por qué Nativo?



Gran librería de recursos nativos

Usar el SDK nativo le permite al desarrollador usar las últimas características diseñadas para esas plataformas.

¿Por qué Nativo?



Cero dependencias de terceros

Al crear aplicaciones con el SDK nativo, los desarrolladores no están atados a recursos de terceros y no hay dependencia de la comunidad open-source como Cordova, para mantenerse al día con las últimas características.

¿Por qué Nativo?

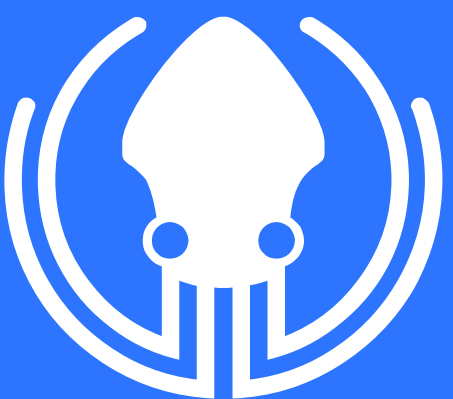
Inconvenientes del desarrollo de aplicaciones nativas

1- Ciclos de desarrollo más largos

Especialmente cuando de crear aplicaciones en múltiples plataformas se trata.

IOS tiene su forma de crear una lista, Android también, Windows Phone también, la web también, etc...

También reduce la agilidad de lanzar actualizaciones de nuestras aplicaciones.



2- Altos costos de desarrollo

Si no lo haz escuchado anteriormente, desarrollar aplicaciones nativas es más caro y usualmente toma más trabajo hacerlas.

Principalmente por el tiempo que demora crear algo para cada plataforma, que conlleva a contratar y mantener el talento de un desarrollador o varios que están especializados en esta tarea.



3- El talento nativo es difícil de encontrar

Encontrar y contratar personal para IOS y Android es difícil, consume tiempo y por lo general es caro.

Muchas empresas prefieren contratar un desarrollador que pueda crear apps para iOS y Android sin importar si son nativas o híbridas.



Atributo	Nativo	Híbrido multi-plataforma
Habilidad necesaria	Objective-C, IOS SDK, Java, Android SDK	HTML, CSS y JS + algún framework
Método de distribución	App Stores	App Stores, Desktop Browser, Desktop App (Como electron), Mobile Browser, PWA
Velocidad de desarrollo	Lenta (multiplataforma)	Rápida
Costo de mantenimiento	Medio / Alto	Bajo / Medio
Performance Gráfico	Alto	Moderado / bajo
Performance de la App	Alto	Depende del caso de uso
Acceso a funcionalidad nativa	Librería completa	Librería casi completa mediante plugins
Consistencia entre diversas plataformas y dispositivos	Requiere apps separadas	Si, con el mismo código

¿Qué es ionic?



ionic

Es un framework open source

Que también tiene un conjunto de herramientas que facilitan el desarrollo y pruebas.

Puedes desarrollar para múltiples plataformas con el mismo código.

ionic adapta su diseño dependiendo de la plataforma en la que este corriendo.



Aprenderemos ionic utilizando Angular

Angular es quien realiza la lógica de validación entre pantallas y manipulación del HTML

Usaremos el sistema de rutas de Angular para navegar entre pantallas



Veamos un poco de
 ionic