

Tecnologías Emergentes

---

# Tema 02. Servicios móviles

# Índice

## Esquema

### Ideas clave

- 2.1. ¿Cómo estudiar este tema?
- 2.2. Tipos de aplicaciones
- 2.3. Decisión sobre el tipo de aplicación móvil
- 2.4. Diseño de la interacción móvil

### A fondo

- Progressive Web Apps: todo lo que se tiene que saber
- Tipos de aplicaciones móviles: WebApp nativas-híbridas
- The Future of App Development
- Bibliografía

### Test

Tipos de aplicaciones móviles		
App nativas	Mobile Web	Aps híbridas
Máximo rendimiento y acceso a todo el hardware	Costes reducidos	Mobile Web en Contenedor nativo
Se pueden descargar de un App Store	Acceso universal y actualizaciones inmediatas	Se pueden descargar de un App Store
Coste muy elevado	No se pueden descargar de un App Store	Rendimiento penalizado

## 2.1. ¿Cómo estudiar este tema?

Para estudiar este tema, lee las ideas clave disponibles a continuación:

- ▶ **Alternativas para el desarrollo de aplicaciones móviles**, donde se abordan dos extremos del espectro, por un lado, el desarrollo de apps nativas, y por otro lado el desarrollo de aplicaciones Web adaptadas al canal móvil (Mobile Webs).
- ▶ **Aplicaciones híbridas**, como una mezcla que intenta obtener las ventajas de cada una de las anteriores, sin los inconvenientes de ambas.
- ▶ **Decisión sobre el tipo de desarrollo**: es importante por muchos factores elegir adecuadamente el modelo de aplicación móvil que debemos crear, dadas unas necesidades concretas.
- ▶ Se verán los muy numerosos elementos que se han de tener en cuenta a la hora de intentar **crear una satisfactoria experiencia de uso de la app** por parte de los usuarios.

## 2.2. Tipos de aplicaciones

En el momento en que se considera oportuno el diseño de una aplicación móvil surgen distintas alternativas a nivel de desarrollo que **van a condicionar fuertemente el diseño de la Experiencia de Usuario** de dicha app.

En primera instancia se ha de considerar para qué plataforma se debe desarrollar la aplicación, o qué decisión se ha de tomar cuando se contempla el desarrollo para varias plataformas. En la actualidad, al hablar de plataformas aquí nos referimos a los dos grandes sistemas operativos que gobiernan los dispositivos móviles: **Android (de Google) e iOS (de Apple)** son las plataformas que lideran en la actualidad el mercado. Esto no quiere decir que sean las únicas, puesto que existen algunas plataformas con presencia muy residual como Windows Phone y BlackBerry OS, y otras emergentes que irán ganando presencia como **Tizen**.

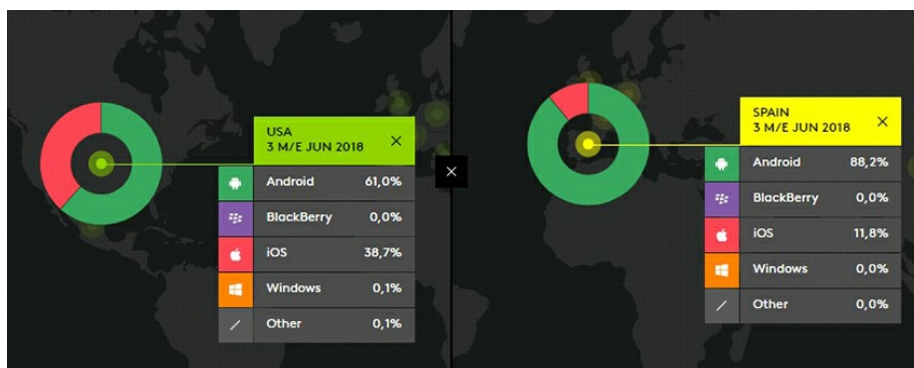


Figura 1. Comparativa de cuotas de mercado de las distintas plataformas móviles entre EE. UU. y España, en junio de 2018. Fuente: <http://www.kantarworldpanel.com/global/smartphone-os-market-share/>

En la actualidad, paralelamente a la plataforma Android estándar (también conocida como «*Stock*» Android), han aparecido dos variantes, denominadas Android One y Android Go.

- El **Stock Android** es la versión clásica, que ya va por su versión 9 («Android Pie»), y

admite personalizaciones fuertes por parte de los fabricantes, lo que dificulta las actualizaciones.

- ▶ **Android One** es una variante de Android que no admite personalizaciones de fabricante, salvo en aspectos muy puntuales como apps relacionadas con la cámara. El objetivo es que vaya fluido en dispositivos de gama media.
- ▶ **AndroidGo**, el último en incorporarse a la familia está específicamente diseñado para equipos muy modestos, con máximo 1 GB de RAM. El objetivo es extender una buena experiencia de uso a dispositivos muy limitados, y aumentar cuota de mercado.

Teniendo en cuenta la información que se ofrece en la figura 1, una de las primeras cuestiones a dilucidar es para qué plataforma o plataformas se ha de crear la app móvil.

Esta decisión es fundamental, ya que abre distintas alternativas para el desarrollo, que afectarán al resultado final percibido por el usuario. Pero, sobre todo, se trata de una decisión puramente empresarial, en función de objetivos, costes y plazos. En la actualidad, las principales alternativas son: aplicaciones de tipo Web móvil (o Mobile Web), aplicaciones nativas y finalmente aplicaciones híbridas.

Entre los factores clave que intervienen en esta decisión se encuentran:

- ▶ Presupuesto disponible.
- ▶ Plazos o limitaciones temporales para el desarrollo.
- ▶ Público al que se destina la app.
- ▶ Funcionalidad (lo que se puede hacer depende del modelo elegido).

### Aplicaciones nativas

La idea más extendida entre los usuarios de lo que es una app móvil es

precisamente la que corresponde con una App Nativa: son aquellas que, una vez consultada una tienda de aplicaciones (sea Google Play para dispositivos Android, o Apple Store para iPhone o iPad), se descargan de la misma (tras un proceso de pago si no son gratuitas) y se instalan en el dispositivo. Lo único que se necesita para ejecutarlas es activar el icono correspondiente de la app.

Sin embargo, la característica realmente distintiva de las apps nativas es que se desarrollan de forma concreta para una plataforma o sistema operativo móvil. Esto significa que **un desarrollo para Android no es reutilizable para iOS**, lo que implica tener dos líneas de desarrollo diferentes.

Esto tiene un impacto importante en el coste global, pues no solo debemos contar con dos equipos de desarrollo, sino que el mantenimiento y actualizaciones suponen también una doble carga a posteriori. Evidentemente hablamos de la aplicación que se instala en el dispositivo. Si luego hay un servicio de *backend* (con bases de datos, aplicaciones de procesamiento o de almacenamiento en la nube, etc.) por supuesto que este podrá ser común.

Precisamente por estar desarrollada para un sistema operativo específico, muchas de las decisiones que afectan al proceso de diseño de una app ya vienen fuertemente condicionadas por la especificidad de las interfaces e interacciones de ese sistema (los fabricantes suelen incluso «sugerir encarecidamente» la utilización práctica de guías de estilo que ellos mismos publican). El problema es que incluso para un mismo sistema, puede haber cambios significativos entre versiones.

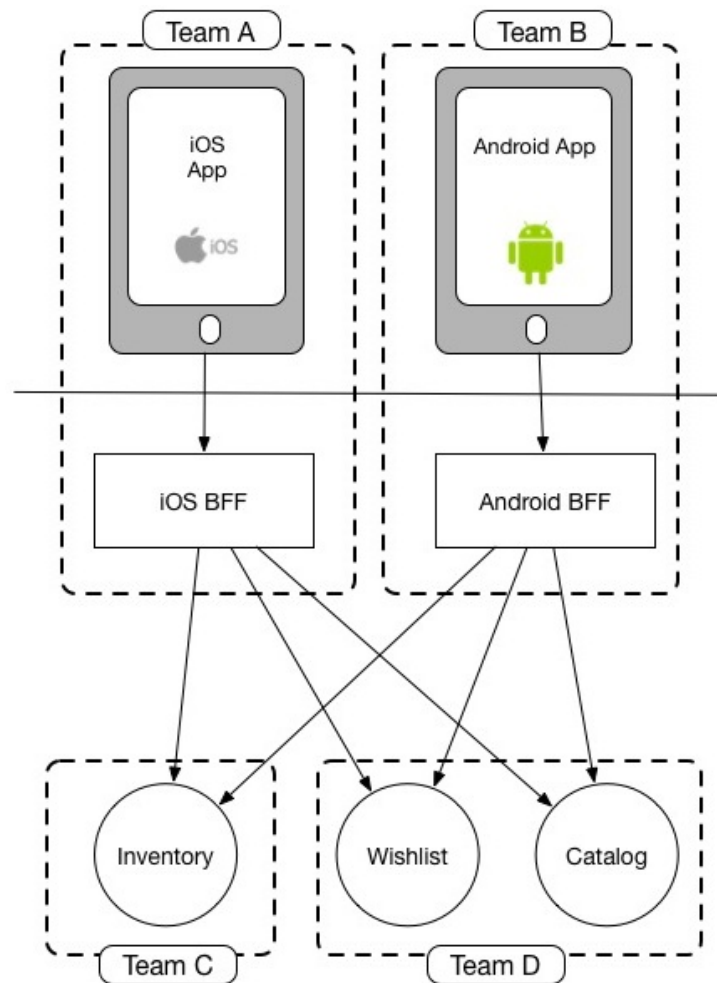


Figura 2. Desarrollo de una app por parte de dos equipos, A para iOS y B para Android, compartiendo un *backend* desarrollado por sendos equipos C y D. Fuente: <https://samnewman.io/patterns/architectural/bff/>

Otra característica distintiva de las apps nativas es que pueden sacar partido de todos los sensores del dispositivo, como puede ser el GPS, las cámaras frontal y trasera, los sensores de movimiento, etc.

Desde el punto de vista de la Experiencia de Usuario, precisamente uno de los requisitos no funcionales que se ven mejor satisfechos con este modelo de desarrollo de aplicaciones es el del rendimiento, que es el más alto posible.

Sin embargo, también existe unas importantes desventajas con este modelo de apps:



para cada plataforma se deberá seguir **un proceso específico de desarrollo**, y los costes globales de desarrollo se encarecerán más por el hecho **de tener que contar con especialistas específicos para cada plataforma**, con el agravante de que no suele ser sencillo encontrar equipos de desarrollo con la experiencia y garantía de calidad necesarios para que ambos desarrollos sean lo más equivalentes posible.

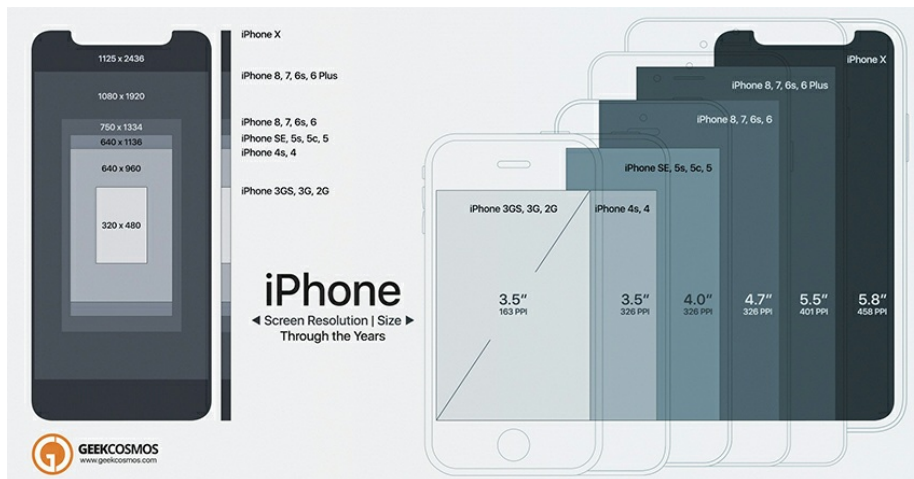


Figura 3. Evolución del tamaño físico de la familia de dispositivos iPhone de Apple. Fuente: [geekcosmos.com](http://geekcosmos.com)

## Aplicaciones Web para móviles (Mobile Web)

Con la estandarización de tamaños de pantalla de gran formato en dispositivos móviles, hoy día ya son normales pantallas como mínimo de 5 pulgadas o más, incluso en el caso de los dispositivos de Apple (que fue muy reacia a ofrecer estos tamaños). Curiosamente, no hace mucho hubo una categoría específica para este tipo de dispositivos, denominados **Phablets**, por ser puente entre móvil y tablet.

Esto significa que casi cualquier web convencional se puede visualizar adecuadamente en un dispositivo móvil, si acaso posiblemente sea preciso usar en zoom para ver algunos contenidos. Sin embargo, **cuando se adapta específicamente la Web para su visualización en dispositivos móviles, la experiencia de usuario se notablemente superior**

A efectos de codificación, en todas las aplicaciones Web se suelen utilizar tres tipos de elementos:

- ▶ **HTML**, como lenguaje de contenidos.
- ▶ **Hojas de estilo CSS** (Cascading Style Sheets), que permiten modelar y definir la representación de los contenidos.
- ▶ **JavaScript**, como lenguaje para el control de las interacciones del usuario con los distintos elementos de la página.

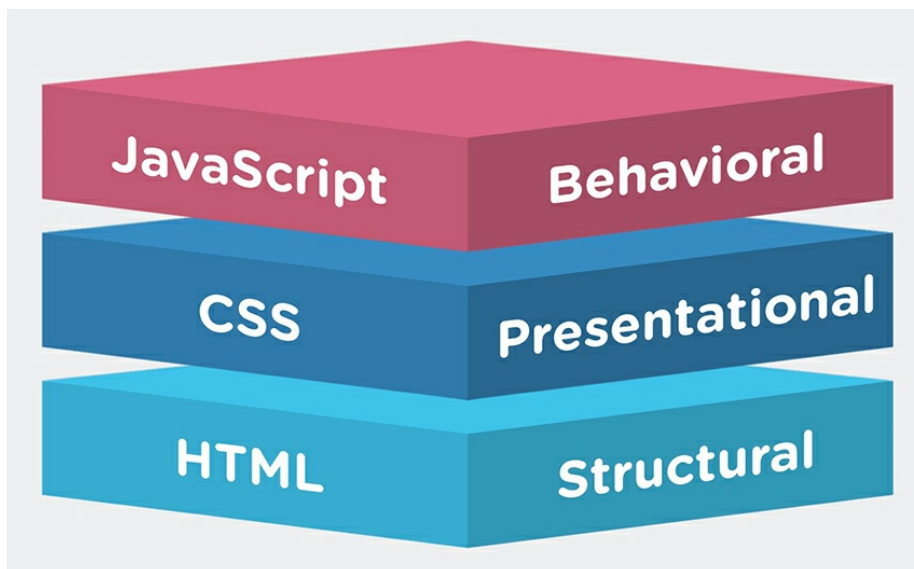


Figura 4. Separación conceptual de los lenguajes usados para Mobile Web. Fuente: [www.meiwen.com.cn](http://www.meiwen.com.cn)

El estándar HTML ha ido evolucionando con el tiempo, y de ser muy limitado en muchas cuestiones, ha pasado recientemente a dar un paso de gigante en algo que afecta al desarrollo de aplicaciones Web para móviles: **la capacidad de acceder a los sensores del dispositivo**, fundamentalmente al GPS y otros servicios de geolocalización, y a la cámara (normalmente la trasera). Además, permiten almacenar información persistente en el dispositivo, usando una pequeña base de datos.

Una de las grandes ventajas de este modelo de desarrollo es que, en esencia no se diferencia gran cosa del modelo de desarrollo de aplicaciones web convencionales, no orientadas a su utilización en dispositivos móviles. Esto significa que se puede aprovechar en gran medida el conocimiento y la experiencia de los equipos de desarrolladores Web, y, sobre todo, **reutilizar el conocimiento de diseñadores desarrolladores del ámbito Web**. Esto, sin duda contribuye a hacer de este modelo de desarrollo una opción notablemente superior en términos coste-efectivos al de desarrollo de aplicaciones nativas. Por otra parte, la aplicación así desarrollada se convierte de alguna forma, por la única necesidad de utilizar un navegador móvil estándar, en una aplicación multiplataforma.

Ante la duda de si en el ámbito móvil se reproducen los problemas de incompatibilidades entre navegadores que en ocasiones han supuesto grandes quebraderos de cabeza, hay que indicar que en este caso la situación está bastante más controlada, puesto que el motor de representación visual usado tanto por los navegadores por defecto de las plataformas Android e iOS es idéntico (WebKit), por lo que queda asegurada la representación consistente de los contenidos para un elevadísimo porcentaje de dispositivos móviles existentes en el mercado.



Figura 5. Distintos tipos de apps con su estructura, y su relación con la Experiencia de Usuario, de menor (izquierda) a mayor (derecha). Fuente: <http://www.ibm.com>

## Aplicaciones híbridas

Existe un tercer modelo de desarrollo de aplicaciones móviles, el de las **Apps híbridas**, en el que se trata de conjugar por un lado, los puntos fuertes de una app nativa (en esencia, acceso a todos los elementos hardware del dispositivo, y la posibilidad de descarga e instalación —y monetización— gracias al uso de una tienda de aplicaciones), y los puntos fuertes de una app Mobile web (fundamentalmente la reutilización de la experiencia de desarrollo en entornos Web y el hecho de tener una única línea de desarrollo).

Desde el punto de vista del proceso de diseño, tanto de la aplicación en sí como de la Experiencia de Usuario que esta proporciona, este modelo es especialmente interesante porque **se puede aprovechar el conocimiento de expertos en desarrollo Web y en diseño de la Experiencia de Usuario para la Web**, normalmente más asequibles que los especializados en plataformas nativas.

Quizás, el único inconveniente serio que existe hoy sobre las apps híbridas, y que en la actualidad impacta en la Experiencia de Usuario es que las apps que se generan con este mecanismo **suelen ser de gran tamaño, y su velocidad dista de ser la de una app nativa pura**. Quizás con el tiempo esto sea un factor cada vez menos a tener en cuenta, conforme evolucionen las tecnologías de desarrollo híbrido de apps, y la propia potencia de los dispositivos móviles.

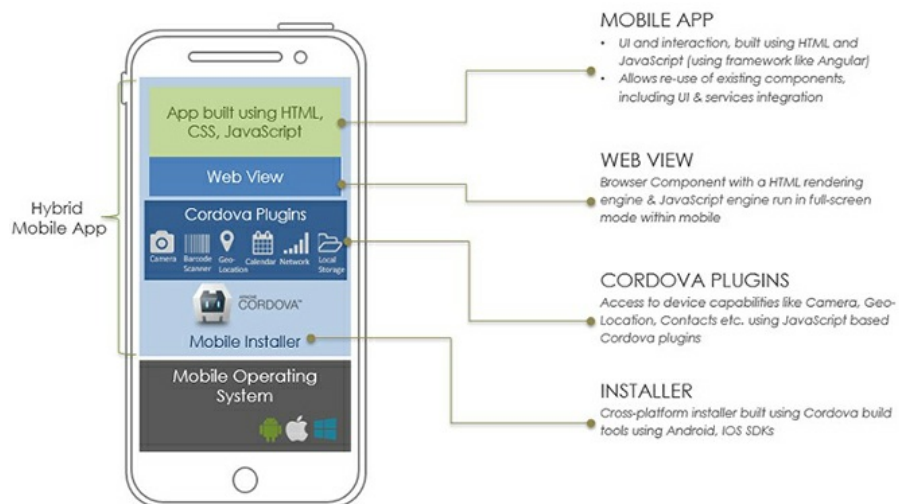


Figura 6. Arquitectura de ejecución de una app híbrida, usando Apache Cordova como contenedor nativo.

Fuente: <https://www.wavemaker.com/>

## 2.3. Decisión sobre el tipo de aplicación móvil

Tomando como referencia únicamente la información anteriormente presentada, sería posible establecer un primer cuadro comparativo sobre los distintos tipos de aplicaciones móviles, que se puede tomar como cuadro-guía a la hora de decidir el modelo a usar para una nueva aplicación móvil:

	Comparativa de características de tipos de aplicaciones móviles		
	Mobile WEB	NATIVA	HIBRIDA
Interfaz de usuario	Basada en Web	Específica de la plataforma (Android, iOS...)	Basada en Web
Comportamiento multiplataforma, multidispositivo	Si	No	Si
Posibilidad de reutilización de código	Si	No	Si
Acceso a los sensores del dispositivo móvil	Parcial	Completo	Depende del contenedor usado
Experiencia de Usuario (UX)	Se puede reutilizar conocimiento de UX Web	Específica de cada plataforma (necesidad de profesionales muy especializados)	Se puede reutilizar conocimiento de UX Web
Tienda de aplicaciones	No	Si (bajo validación en el caso de iOS)	Bajo validación
Coste desarrollo	Bajo o Medio	Alto	Medio
Interfaz de usuario	Basada en Web	Específica de la plataforma (Android, iOS...)	Basada en Web

Tabla 1. Comparativa de características de tipos de aplicaciones móviles.

Sin embargo, cuando una empresa necesita tomar la decisión, sobre todo en aquellos casos en los que el presupuesto es limitado, y se debe centrar al máximo el tiro para que los resultados sean óptimos, habría que hacer **un análisis más**

**pormenorizado**, que tenga en cuenta otras variables.

En la actualidad hay gran cantidad de artículos técnicos serios en los que se comparan las distintas tecnologías antes mencionadas. Sin embargo, cuando se trata de contemplar la última tecnología (en este sector concreto el ritmo de evolución tecnológica es altísimo), el número de artículos que pueden ser relevantes se reduce dramáticamente.

Más aún, cuando se trata de examinar las posibilidades tecnológicas para un desarrollo multiplataforma, que permita abarcar la mayor cantidad posible de plataformas móviles usando una única base de código común para posibilitar un mayor retorno de la inversión (ROI) para las empresas, el cuerpo de conocimiento ofrecido en las distintas bases de datos científicas (entre las consultadas se encuentran Thomson Reuters, SpringerLink, ACM Digital Library, IEEE Xplore) es prácticamente testimonial, muy probablemente debido a los tiempos habituales de revisión y publicación de la producción científica, normalmente superiores a los de la evolución de determinadas tecnologías móviles.

En cualquier caso, con la información relevante extraída se ha realizado un **análisis DAFO tanto de las apps nativas como de las aplicaciones web**, señalando además aquellos criterios más relevantes para empresas que deban afrontar el desarrollo de una aplicación móvil. En total se han examinado 21 fuentes de información relevante, y se indica la frecuencia de aparición de cada criterio en las fuentes, en las tablas 2 y 3.

Costes de desarrollo, despliegue y mantenimiento.

Como se puede comprobar tras un examen de las dos tablas correspondientes a los análisis DAFO, en especial examinando los criterios marcados como relevantes en términos de costes o inversión necesaria, **el modelo de desarrollo basado en aplicaciones Web el claro ganador en este aspecto.**

En efecto, las aplicaciones nativas sufren una elevada fragmentación debido a la proliferación de plataformas (Android, iOS, y en mucha menor medida, Windows Phone), cada una de ellas con una base de código propia.

El coste de desarrollar para estas plataformas es significativamente más elevado que el necesario para desarrollar una aplicación web, por la gran cantidad de factores que juegan un papel importante en el proceso de desarrollo global: gestión de varias líneas de producción activas, distintas especializaciones del mantenimiento correctivo y perfectivo, un perfil más elevado y mayor nivel de especialización de los desarrolladores involucrados, etc.).

En cambio, el desarrollo de las aplicaciones web móviles, basadas en tecnologías como HTML5 y jQuery Mobile implican menores costes puesto que solo requieren única base de código (una única línea de producción) y se pueden desplegar en la práctica totalidad de dispositivos móviles que tengan un navegador y acceso a Internet

Se trata, por tanto, de la solución más coste-efectiva a día de hoy para las empresas cuando ambos modelos permiten satisfacer las necesidades tecnológicas que suponen las funcionalidades de la aplicación a desarrollar.



Aplicaciones Web	
<b>Fortalezas</b> <ul style="list-style-type: none"> <li>▶ Modelo de despliegue muy simple, no requiere instalación (basta con proporcionar una URL al usuario) (89,2 %)</li> <li>▶ Facilidad de desarrollo, y una mayor base de desarrolladores (menores costes) (81,1 %)</li> <li>▶ Estabilidad: Solo se usa un mismo conjunto de tecnologías (Web) y son estándar (78,4 %)</li> <li>▶ Reusabilidad en las distintas plataformas (70,3 %)</li> <li>▶ Las actualizaciones se hacen en el servidor, por lo que son inmediatas y totalmente transparentes al usuario. No hay retrasos en la publicación, y menor impacto económico de resolución de errores (59,5 %)</li> <li>▶ Todos los usuarios tienen siempre la misma versión de la aplicación. Menores costes de mantenimiento correctivo y perfectivo (45,9 %)</li> </ul>	<b>Debilidades</b> <ul style="list-style-type: none"> <li>▶ No tienen acceso a todos los recursos hardware del dispositivo, lo que puede limitar el tipo de aplicaciones (86,5 %)</li> <li>▶ La monetización –en caso necesario– ha de hacerse mediante suscripciones o vía publicidad (73 %)</li> <li>▶ Apariencia (aún) distinta a la de apps nativas (67,6 %)</li> </ul>
<b>Oportunidades</b> <ul style="list-style-type: none"> <li>▶ Requiere menor inversión (94,6 %)</li> <li>▶ Ciclo de desarrollo software para varias plataformas más corto (75,7 %)</li> <li>▶ Mejor ROI (67,6 %)</li> </ul>	<b>Amenazas</b> <ul style="list-style-type: none"> <li>▶ Puede que nunca lleguen a equipararse a apps nativas (54,1 %)</li> <li>▶ La competencia puede desarrollar apps nativas (18,9 %)</li> </ul>

Tabla 2. Análisis DAFO del modelo de desarrollo móvil basado en Aplicaciones Web.

Aplicaciones nativas	
<b>Fortalezas</b> <ul style="list-style-type: none"> <li>▶ Muy buena integración con el sistema operativo de la plataforma (86,5 %)</li> <li>▶ Mejor rendimiento (78,4 %)</li> <li>▶ Experiencia de usuario consistente dentro de la plataforma (35,1 %)</li> <li>▶ Interés propio del fabricante por evolucionar la plataforma para que siga siendo competitiva y por ende lo sea su tienda de apps (fuente de ingresos) (10,8 %)</li> </ul>	<b>Debilidades</b> <ul style="list-style-type: none"> <li>▶ Estrictamente vinculadas a una plataforma, cada una requiere su propio proceso de desarrollo software. (97,3 %)</li> <li>▶ Necesaria mucha mayor cualificación (coste de personal para desarrollo) (78,4 %)</li> <li>▶ Mayor inversión de desarrollo por ser necesarias varias líneas de producción (una por plataforma) y gestionar varias bases de código (59,5 %)</li> <li>▶ Monetización necesariamente compartida con el propietario de la plataforma (43,2 %)</li> <li>▶ Fragmentación de versiones entre usuarios que decidan no actualizar la app. (10,8 %)</li> </ul>
<b>Oportunidades</b> <ul style="list-style-type: none"> <li>▶ Se puede sacar el máximo partido a las capacidades del dispositivo (83,8 %)</li> <li>▶ Soporte sencillo de la tienda de aplicaciones para la distribución de la app (70,3 %)</li> <li>▶ El proceso de aprobación de una app para su publicación en algunas tiendas de apps da cierta confianza a los usuarios sobre su calidad y seguridad (37,8 %)</li> </ul>	<b>Amenazas</b> <ul style="list-style-type: none"> <li>▶ Dependencia absoluta de los criterios de publicación en la tienda de aplicaciones del fabricante (91,9 %)</li> <li>▶ Dependencia total del fabricante de la plataforma (cambio de versiones, futuras incompatibilidades, nuevas exigencias de hardware mínimo) (89,2 %)</li> <li>▶ Puede popularizarse rápidamente una nueva plataforma (32,4 %)</li> </ul>

Tabla 3. Análisis DAFO del modelo de desarrollo móvil basado en aplicaciones nativas.

## 2.4. Diseño de la interacción móvil

Una de las claves del éxito de cualquier aplicación móvil, tome la forma arquitectural que tome, es **la satisfacción del usuario ante la experiencia de uso de la aplicación**. La experiencia de usuario móvil es muy particular, pues engloba las percepciones y sensaciones del usuario antes, durante y tras su interacción con nuestro producto móvil, bien sea a través de un navegador o mediante una app, usando un dispositivo móvil que puede estar en una larga línea de gamas que va desde terminales de bajo coste hasta terminales de última generación y tabletas de ultra alta definición y potencia de cálculo más que sobrada

La creación de experiencias de usuario móvil satisfactorias y que deleiten a los usuarios obliga a repensar buena parte de lo que hasta ahora se consideraba válido en diseños para equipos de escritorio y Web:

- ▶ Por un lado, existen complicaciones derivadas de las características de los dispositivos: tamaños de pantallas muy variables, restricciones en la conectividad y en la forma de uso.
- ▶ Por otro lado, la gran dificultad que supone el contexto de uso, siempre cambiante y complicado de identificar.



Figura 7. Una pequeña muestra de la gran diversidad de dispositivos y factores de forma de los mismos.

Fuente: <https://solutionsreview.com/>

El buen diseño de aplicaciones para el móvil tiene distintos componentes clave que modelan la experiencia de usuario móvil, como se puede ver en la siguiente figura:



Figura 8. Elementos que intervienen en la satisfacción percibida por el usuario sobre una aplicación móvil.

Fuente: <http://www.slideshare.net/lycerejo/elements-of-the-mobile-user-experience>

La relevancia de cada uno de estos elementos clave dependerá del tipo de dispositivo (móvil o tablet), así como de la interfaz de usuario (tipo app o tipo Mobile Web).

Veamos en la tabla siguiente cada uno de estos elementos, y recomendaciones al respecto:

## RECOMENDACIONES PARA CADA ELEMENTO DE UX MÓVIL

### Funcionalidad

- ▶ **Priorizar las características más relevantes** para un usuario en movilidad, por ejemplo, usar la localización para detectar la entrada en aeropuerto e informar solo sobre nuestro vuelo, o usar la localización para informar solo sobre tiendas más cercanas y facilitar link al número de teléfono para llamar directamente.
- ▶ Facilitar herramientas de interacción con el mundo físico, **como lectores de códigos de barras, de códigos QRs** en aplicaciones para comparar precios u obtener más información.
- ▶ Ofrecer determinadas características en todos los canales. Por ejemplo, los ajustes de cuenta de usuario han de estar accesibles tanto en móvil como en tablet como en Web normal.

### Arquitectura de Información

- ▶ Presentar los **elementos** de contenido y enlaces más **importantes** en el contexto móvil en **la primera página o pantalla**.
- ▶ **El diseño de navegación debería ser más vertical** y no en horizontal como en sitios web convencionales.
- ▶ **Minimizar el número de toques** para llegar a las funciones más importantes (realizar un análisis detallado para ello, para cada caso de uso y reorganizarlo todo con esta premisa).
- ▶ Intentar **evitar que el usuario introduzca información**, es preferible poner listas de opciones de forma inteligente (poner al principio las opciones más probables, por ejemplo, para indicar ciudad, o elegirla con servicios de localización, con una simple confirmación).
- ▶ Utilizar **mecanismos** (etiquetas, indicadores, textos) muy **descriptivos** para que el usuario vea simplificada la navegación y sepa bien en todo momento en qué parte se encuentra.

## Contenido

- ▶ **Simplificar** mucho el contenido presentado en la interfaz móvil con respecto al de la Web convencional.
- ▶ Si se usa **contenido multimedia**, dar siempre el **control al usuario**, no haciendo que dicho contenido se empiece a reproducir automáticamente, y siendo respetuoso con el consumo de datos.
- ▶ **Optimizar las imágenes** usadas, en algunos casos supone reducir su tamaño, pero en otras ofrecer definición suficiente para dispositivos con pantallas con alta densidad de puntos (como la de los últimos iPads Pro o Galaxy Tab S, y similares).
- ▶ **Evitar contenidos que no se puedan ver adecuadamente** en dispositivos móviles sin instalar complementos (que a veces no están disponibles), como es el caso de Flash.

## Diseño

- ▶ Es **muy importante** entender que el **diseño móvil** no consiste en reducir o adaptar el diseño para Web convencional, sino **repensar y rediseñar las aplicaciones** para este nuevo entorno en movilidad.
- ▶ Se ha de **diseñar para proporcionar información de forma rápida**, que se puede captar con un solo vistazo o un rápido examen a la pantalla.
- ▶ Intentar mantener la **coherencia visual de nuestro servicio a través de todos los canales**: web y móvil, tanto para teléfonos como tabletas: usar los mismos códigos de colores y todo aquello que define la personalidad o identidad del sitio.
- ▶ **Considerar siempre las orientaciones vertical y horizontal como diferenciadas**, y no cambiar lo que el usuario esté haciendo si se produce un cambio de orientación física del dispositivo.

## Entrada de Usuario

- ▶ **Limitar la entrada a campos** absolutamente esenciales, en especial en formularios de registro. Siempre han de ser menos que los solicitados en Web estándar.
- ▶ **Mostrar valores por defecto siempre que sea posible** (según la última selección realizada por el usuario o la más frecuentemente elegida por el usuario en su historial).
- ▶ Intentar utilizar la **geolocalización** siempre que sea posible. No se ha de olvidar que las páginas Web tienen acceso ya a esta información.
- ▶ Siempre que se pida un **dato numérico mostrar el teclado numérico**.
- ▶ **Intentar guardar la información de sesión iniciada en el dispositivo**, y tratar de evitar nuevos inicios de sesión, ya que los dispositivos móviles tienden a ser dispositivos altamente personales, que no se comparten.

## Contexto móvil

- ▶ El **contexto de uso de la aplicación debe ser primordial**, y esto significa que se han de tener en cuenta cuestiones como la luminosidad ambiente (usar temas con más o menos brillo), el uso de geolocalización para ofrecer solo información relevante, y tener en cuenta que, si el usuario está en movimiento, la interfaz se ha de simplificar.
- ▶ En general, se han de **usar todas las capacidades hardware del dispositivo para adecuar lo que se presenta** al usuario según el contexto de uso.
- ▶ Siempre se han de tener en cuenta las **preferencias y ajustes** del usuario.

## Usabilidad

- ▶ Dejar muy claro qué **cosas se pueden seleccionar, tocar o deslizar**.
- ▶ En **interfaces táctiles**, hacer que los elementos seleccionables o sobre los que se pueda dar un toque tengan el tamaño suficiente y estén bien separados para evitar errores de selección (en especial separar bien los elementos «destructivos», de cancelar operación y similares).
- ▶ Seguir razonablemente bien las guías oficiales de diseño de UI y UX que existen para las distintas plataformas (específicas de Apple y de Google para Android).
- ▶ **Intentar adaptar las condiciones de visibilidad** y contraste al momento del día (por ejemplo, las apps de navegación suelen cambiar a modo noche, usando colores más tenues para no deslumbrar)

## Confiabilidad

- ▶ Se debe hacer todo lo posible para que el usuario se sienta **cómodo y confiado** al usar la app.
- ▶ A la hora de instalar la app, se deben los **permisos realmente necesarios**. No hay nada más inquietante que pedir permiso de acceso a los contactos para una aplicación de tipo linterna o de información meteorológica.
- ▶ Se ha de **evitar cualquier posible preocupación sobre la privacidad y la seguridad**, con medidas técnicas y también explicando claramente al usuario las prácticas de negocio seguidas en la app.
- ▶ No se han de usar largos textos de Términos y Condiciones, que parezcan diseñados para que no se lean. **Simplicidad** es la palabra aquí, como casi siempre en el buen diseño de la interacción

## Realimentación (feedback)

- ▶ Se debe atraer la atención del usuario solo cuando realmente sea necesario.
- ▶ Hay que procurar minimizar el número de alertas. Y cuando se producen con textos concisos y muy claros.

## Ayudar y asistencia

- ▶ Se han de ofrecer opciones de asistencia al usuario en el uso de la app.
- ▶ Es recomendable usar un asistente en la primera ejecución de la app.
- ▶ Se deberían ofrecer múltiples canales de consulta (página web específica, Twitter, botón de llamada a soporte, etc.).

## Componente social

- ▶ Se han de potenciar los contenidos y funcionalidades que proporcionen un sentido de participación social, que permita al usuario la interacción y la compartición en redes sociales.
- ▶ Es buena idea hacer que la app permita generar contenidos con la «marca» propia del usuario.

## Marketing

- ▶ Se deben explorar distintas vías para que el usuario encuentre la app o el sitio web para móviles que se ha desarrollado: utilizando códigos QR, etiquetas NFC, posicionamiento SEO, promoción multicanal.
- ▶ Si se solicitan valoraciones sobre la app (con estrellas o algún mecanismo similar), siempre ha de hacerse de forma no intrusiva. Como mínimo incluir la opción «no volver a recordarme».

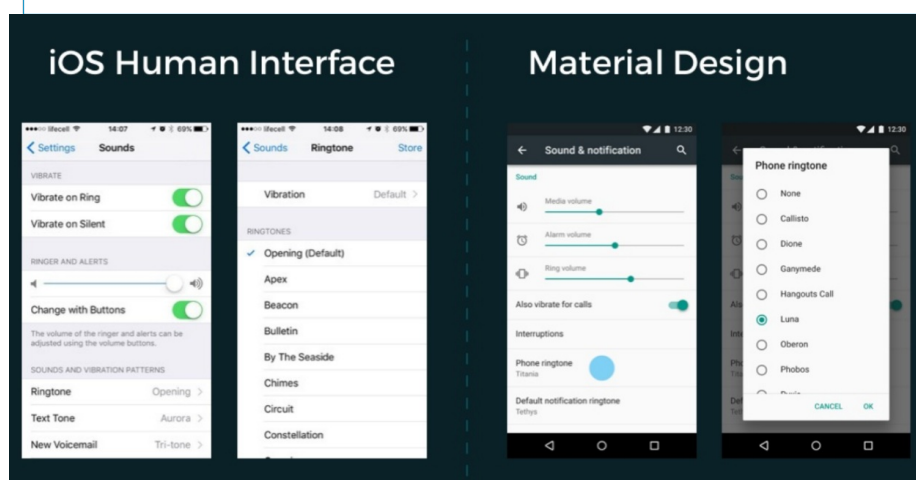


Figura 9. Diferencias entre las guías de estilo de Apple (*Human Interface*) y de Android (*Material Design*), que condicionan la usabilidad de las apps en cada plataforma. Fuente: mac9to5.com





### Progressive Web Apps: todo lo que se tiene que saber

Santoni, M. (16 de marzo 2018). Las Progressive Web Apps: Todo lo que se tiene que saber. *GoodBarber*. [Mensaje en un blog]. <https://es.goodbarber.com/blog/las-progressive-web-apps-todo-lo-que-se-tiene-que-saber-a604/>

Artículo completo en el que se explica un nuevo modelo de aplicaciones web, las PWA o aplicaciones web progresivas, por supuesto aplicables al canal móvil.

#### **Características de las herramientas de *testing* de aplicaciones móviles**

BBVA Open4U. (2 de julio de 2018). Características de las herramientas de testeo de aplicaciones móviles. [Mensaje en un blog]. <https://bbvaopen4u.com/es/actualidad/caracteristicas-de-las-mejores-herramientas-de-testeo-de-aplicaciones-moviles>

Interesante documento desarrollado por BBVA en su iniciativa Open4U, en el que se explican las características de algunas de las herramientas más usadas para hacer pruebas o testing de aplicaciones móviles.

### Tipos de aplicaciones móviles: WebApp nativas-híbridas

Cristian Henao. (18 de febrero de 2017). 1. *Tipos de Aplicaciones Móviles: WebApp - Nativas - Híbridas* [Vídeo]. YouTube. <https://youtu.be/ykgYAaxL2dA>

Clara exposición realizada por Cristian Henao, sobre los distintos tipos de aplicaciones móviles.

### The Future of App Development

Phillips, A. (29 de agosto de 2018). The Future of App Development. [Mensaje en un blog]. <https://hackernoon.com/the-future-of-app-development-e89f72c3ea8b>

Interesante aproximación a cuestiones relacionadas con el futuro del desarrollo de aplicaciones móviles, incluyendo aspectos novedosos como Internet de las cosas (IoT) y la tecnología *Blockchain*.

## Bibliografía

Ríos, S. J. (2014). Sistemas de información para la mejora de la competitividad empresarial basados en tecnologías de localización de dispositivos móviles. (Tesis doctoral). Universidad de Granada.

1. En las apps nativas, en cuanto al diseño visual:
  - A. Es absolutamente idéntico al de las aplicaciones mobile Web.
  - B. Dependerá fuertemente de las cuestiones específicas tanto de las interfaces como de las interacciones de la plataforma en cuestión.
  - C. Es totalmente idéntico incluso dentro de las diferentes versiones de la plataforma móvil.
  - D. Ninguna de las anteriores es correcta.
  
2. Una aplicación móvil nativa:
  - A. Puede funcionar offline, sin necesidad de establecer una conexión móvil de datos.
  - B. Puede acceder a distintos elementos hardware como las cámaras, los receptores de geolocalización GPS y otros sensores.
  - C. Su rendimiento es el más alto de entre los distintos tipos de aplicaciones móviles.
  - D. Todas las anteriores son correctas.
  
3. Una aplicación de tipo mobile web:
  - A. Su alcance es superior que el de una app nativa.
  - B. Tiene acceso, si bien limitado, a algunos elementos hardware del móvil, como la cámara.
  - C. Su coste de desarrollo es el menor de entre los distintos tipos de aplicaciones móviles.
  - D. Todas las anteriores son correctas.

4. Siguiendo el criterio de ventas, las principales plataformas móviles a nivel mundial son (en orden decreciente):
- A. Android de Google e iOS de Apple.
  - B. iOS, Tizen y Android.
  - C. iOS, Android y Tizen.
  - D. Ninguna de las anteriores es correcta.
5. Sobre las aplicaciones Mobile Web:
- A. Se accede directamente a ellas con una URL (dirección Web).
  - B. Son más costosas a nivel de desarrollo que las apps nativas.
  - C. Se deben desarrollar específicamente, y con dos equipos independientes, para cada tipo de móvil (iOS, Android, etc.).
  - D. Todas las anteriores son correctas.
6. Sobre las apps Híbridas:
- A. Son más económicas de desarrollar que las aplicaciones Mobile Web.
  - B. Se trata de una web móvil que queda encapsulada dentro de una app nativa llamada contenedor, y que es específica de cada plataforma.
  - C. Son las que menos volumen en MB proporcionan en todos los casos, y siempre son más pequeñas que las apps nativas.
  - D. Todas las anteriores son correctas.

7. Una de las grandes dificultades a la hora de crear una app satisfactoria para el usuario es:

- A. La enorme diversidad y variedad de dispositivos móviles.
- B. La necesidad de simular un iPhone cuando se tiene un Android.
- C. Que el usuario ya solo entiende la interfaz de WhatsApp y las apps deben ser lo más similares a dicha app.
- D. Todas las anteriores son correctas.

8. La segunda de las grandes dificultades a la hora de crear una app satisfactoria para el usuario es:

- A. El contexto de uso.
- B. Que solo se pueden hacer aplicaciones de éxito en dispositivos de gama muy alta, con precios superiores a 900 €.
- C. La notable falta de concentración del usuario móvil medio, ya que se distrae enseguida con las notificaciones, normalmente de mensajería instantánea.
- D. Todas las anteriores son correctas.

9. La monetización de una App Nativa:

- A. Solo puede realizarla Apple con sus dispositivos iPhone.
- B. Se puede realizar con el uso de la correspondiente tienda de aplicaciones.
- C. Solo es posible si la aplicación admite pagos con Apple Pay o Samsung Pay.
- D. Ninguna de las anteriores es correcta.



10. La monetización de una aplicación Mobile Web:
- A. Solo se puede hacer con suscripciones, muros de pago o técnicas similares.
  - B. Se puede realizar con el uso de la correspondiente tienda de aplicaciones.
  - C. Solo se puede hacer con transferencia bancaria.
  - D. Ninguna de las anteriores es correcta.