# Capítulo 5 - Aplicaciones Móviles

En este capítulo veremos que sistemas operativos se utilizan en plataformas móviles. Revisaremos como los dispositivos móviles toman mayor relevancia en el mercado de Internet(GLOSARIO), debido a que sus aplicaciones son de alta demanda por parte de los usuarios. Analizaremos los modos de construcción de aplicaciones diferenciando las nativas, las webs y las híbridas. A este respecto, HTML5 como tecnología de desarrollo emergente empuja permite la utilización de los conocimientos de aplicaciones web y por otro, las apps nativas ofrecen un mayor rendimiento. La brecha entre estas dos técnicas de desarrollo, deviene en las App Híbridas, tomando ventajas de cada modalidad. El advenimiento de tecnologías cómo Cordova, IntelXDK, Ionic y la popularización de HTML5, ha logrado que la comunidad de desarrolladores comienza a apostar a estos Frameworks(GLOSARIO) basados en tecnologías de Front-End (GLOSARIO) para el desarrollo de aplicaciones móviles. [[1]](#footnote-1)

5.1 Las Aplicaciones móviles

“Una aplicación móvil o App es una aplicación informática diseñada para ser ejecutada en teléfonos inteligentes, tabletas y otros dispositivos móviles y que permite al usuario efectuar una tarea concreta de cualquier tipo: profesional, de ocio, educativas, de acceso a servicios, etc; Facilitando las gestiones o actividades a desarrollar” (https://es.wikipedia.org/wiki/Aplicaci%C3%B3n\_m%C3%B3vil)

Al ser aplicaciones residentes en los dispositivos están escritas están escritas mayormente en Java (Android), Objective-C (iOS) y C# (Windows Phone). Su funcionamiento y recursos se encaminan a aportar una serie de ventajas tales como:

Ilustración - Aplicaciones móviles

* Un acceso más rápido y sencillo a la información necesaria sin necesidad de los datos de autenticación en cada acceso.
* Un almacenamiento de datos personales aislado para cada aplicación. En el caso de Android, basado en Linux, este concepto se lo denomina *Sandboxing*, limitando en gran medida el acceso al sistema de archivos e impide que los procesos puedan acceder a los recursos de otros procesos, como la memoria y la CPU.
* Una gran flexibilidad en cuanto a su utilización, dada la facilidad que presenta a la hora del manejo por parte del usuario, al ser necesario solo instalarla y teniendo como ventaja que el sistema operativo se encarga de mantenerla actualizada.
* Acceso funcionalidades específicas del dispositivo. Esto se debe a que las App nativas acceden directamente a recursos hardware (cámara, contactos, memoria, notificaciones push, etc.).
* Mejorar la conectividad y disponibilidad de servicios y productos entre usuarios, y usuario con proveedores de servicios).

### 5.2.1 Las Web Apps

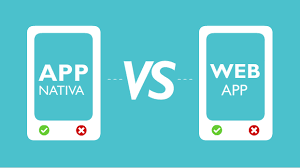
****Una Web App es una versión de una página web optimizada y adaptable a un grán número de dispositivos móviles independientemente del sistema operativo que utilice. Esta optimización es posible gracias al características provistas por lenguaje de marcado HTML5, combinado con hojas de estilo e cascada CSS3, que permiten proveer adaptabilidad, denominada en inglés “*Responsive Web Design*”. El diseño web responsivo es una filosofía de diseño y desarrollo donde el objetivo es adaptar la apariencia de las páginas webs al dispositivo(visualización) que se utiliza para visitarlas. Se caracteriza porque los layout y los contenidos multimediales son fluidos y se utiliza código media-queries de CSS3. [[2]](#footnote-2)

Ilustración - App nativa vs Web App

### 5.2.2 Ventajas de las Web-App:

* No ocupa espacio de memoria de almacenamiento en los dispositivos (no es una App).
* No requiere actualizaciones ya que al ser una página web siempre se accede a la última versión.
* No consume recursos dado que no instala servicios en segundo plano y además no consume espacio dado que no es necesario instalar la aplicación para su uso.
* En líneas generales la implementación de una Web App es más económica que el de una App.

### 5.2.3 Desventajas de las Web-Apps

* No permite la promoción y distribución a través de los *markets* o tienda de aplicaciones (Google Play, Nokia Store, App Store, Windows Phone Apps)
* Requiere de una conexión entre el cliente y el servidor (por ejemplo, por internet o una WLAN).
* Menor usabilidad, al ofrecer un acceso muy limitado a los elementos y capacidades hardware del dispositivo.
* Carece de un icono de lanzamiento específico.
* Necesitan de un espacio web.
* No funcionan en segundo plano (multitarea)

Ilustración – WebApps – Diseño multipropósito

## **5.3 Sistemas operativos para dispositivos móviles**

Al igual que en una computadora, las aplicaciones que se han mencionado se ejecutan sobre un sistema operativo móvil (SO) móvil. Se compone de un conjunto de programas de bajo nivel que permite la abstracción de las peculiaridades del hardware específico del aparato y proveen servicios a las aplicaciones. Al igual que los dispositivos de computación tradicionales dónde se utilizan Windows, Linux o Mac OS, en el caso de los móviles los SO son Android, iOS o Windows Phone, entre otros.

A medida que los dispositivos móviles crecen en popularidad, sus SO adquieren mayor importancia. La cuota de mercado de sistemas operativos móviles en el primer trimestre de 2016 fue el siguiente sobre una base de 6600 millones de dispositivos:

* Android 84,1 %
* iOS 14,8 %
* Windows Phone 0,7 %
* BlackBerry OS 0,2 %
* Otros 0,2 %

Android tiene la mayor cuota, desde enero 2011, con más de la mitad del mercado, experimentó un creciente aumento y en solo dos años (2009 a comienzos de 2011) ha pasado a ser el SO móvil más utilizado.

Es por esto, que en principio se pensó desarrollar una App para operar el SAR para esta plataforma.

## **5.4 Android**

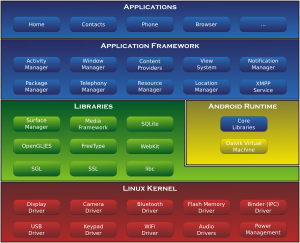
Se encuentra basado en Linux (GLOSARIO), diseñado originalmente para cámaras fotográficas profesionales, luego fue vendido a Google y modificado para ser utilizado en dispositivos móviles como los teléfonos inteligentes y posteriormente en tablets. Actualmente se encuentra en desarrollo para usarse en netbooks y PCs. Debido a la gran variedad de dispositivos que ejecutan Android, la *Open Handset Alliance*, compuesta por 84 compañías de hardware, software y telecomunicaciones, se dedicada al desarrollo de estándares abiertos para celulares, ayudado en gran medida a la masificación del SO de Google, hasta el punto de que estos estándares son usados por empresas como HTC, LG, Samsung, Motorola entre otros.

Ilustración - Arquitectura de Android



Las aplicaciones para Android se escriben y desarrollan en Java, aunque con unas APIs(GLOSARIO) propias, por lo que las aplicaciones escritas en Java para PC y demás plataformas ya existentes no son compatibles con este sistema.

Ilustración - Logo de Android

## **5.5 Aplicaciones móviles multiplataforma**

### 5.5.1 Diferencias entre aplicaciones y web móviles

Una aplicación móvil debe ser descargadas e instaladas para ser usada, mientras que una web puede accederse simplemente teniendo conexión a Internet y un navegador compatible. Pero estas últimas siempre pueden presentarse correctamente desde una pantalla generalmente más pequeña que la de un ordenador de escritorio.

Las “webs responsivas” (5.2.1 Las Web Apps) son como un subconjunto de las aplicaciones web y utilizan conceptos como el “diseño líquido” para que su contenido aproveche la forma del contenedor.

Previa a la existencia del CSS3, se carecía de tecnología para poder crear sitios “elásticos”, es decir, que su disposición se adapte a cualquier dimensión y relación de aspecto de pantalla, por lo tanto los desarrolladores de web estaban obligados a crear diferentes versiones de las páginas web. CSS3 provee mecanismos como las consultas de medio de presentación (*media queries (GLOSARIO))* para que las páginas pueden reaccionar ante distintas circunstancias como el cambio de ancho de la pantalla (como cuando ocurre una rotación. En conclusión, *Web responsive* se denomina a todas aquellas técnicas (no solamente redimensionado de pantalla) que permiten la adaptabilidad del contenido a los dispositivos terminales.

### 5.5.2 App Nativas

Una App nativa es aquella que se desarrolla de forma específica para un determinado sistema operativo, utilizando un *Software Development Kit*o SDK disponible a través del proveedor del dispositivo. Cada una de las plataformas, Android, iOS o Windows Phone, tienen un SDK diferente, por lo que si se desea que una App esté disponible en todas las plataformas se deberán de crear una para cada SO, impllicando la utilización no solo de múltiples APIs, sino también de distintos lenguajes según la plataforma:

* Las apps para iOS se desarrollan con lenguaje Objective-C o Swift.
* Las apps para Android se desarrollan con lenguaje Java o Kotlin
* Las apps en Windows Phone se desarrollan en C# o lenguajes *managed* que se ejecuten sobre el CLR de .Net.

Las aplicaciones nativas, como se mencionó anteriormente, tienen acceso a las caracteristicas espeíficas de HW, además de la capacidad ser ejecutadas sin necesidad de conectividad a internet. Por otro lado estas Apps son promocionadas por medio de las tiendas de aplicaciones, que facilitan su descarga y ofrecen un mejor rendimiento que las alternativas de desarrollo.

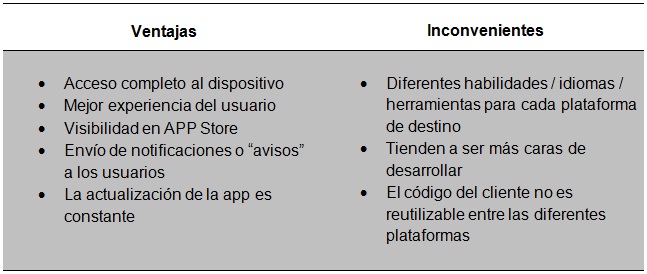


Ilustración - Cuadro comparativo nativas

### 5.5.3 Desarrollo de Web Apps

Una aplicación web o Web App se desarrollada con los lenguajes HTML, Javascript y CSS que revisten de gran popularidad en la actualidad.

En el contexto de aplicaciones móviles, su principal ventaja con respecto un desarrollo nativo es la posibilidad de programar independientemente del SO en el que se usará la aplicación. De esta forma se pueden ejecutar en diferentes dispositivos sin tener que crear varias aplicaciones.

Las aplicaciones web se ejecutan dentro del propio navegador web del dispositivo a través de una URL.

La diferencia mayor con una aplicación nativa es que carece del proceso de instalación, pero con la contraparte de no poder estar visibles en la tienda de aplicaciones y por ende, la promoción y comercialización debe realizarse de forma independiente. Carecen también de la capacidad de accederse desde el lanzador del dispositivo, pero en algunas plataformas, este inconveniente puede suplirse con la creación de un acceso directo o link.

Las Web Apps móviles son una opción atractiva si el objetivo es adaptar la web a formato móvil.

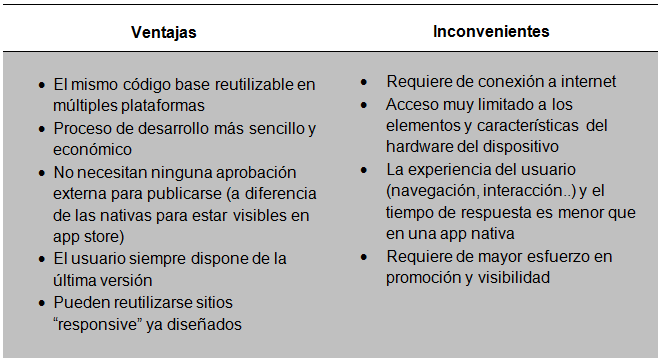


Ilustración - Cuadro comparativo - Aplicaciones Web

### 5.5.4 Aplicaciones Híbridas

Una aplicación híbrida es una combinación de las dos anteriores. Las apps híbridas se desarrollan con lenguajes propios de las Web Apps, es decir, HTML, Javascript y CSS, pero también dan la posibilidad de acceder a gran parte de las características del hardware del dispositivo. La principal ventaja es que es posible empaquetarla y distribuirla en la tienda de aplicaciones de cada SO.

Tanto PhoneGap como Apache Cordova, son los frameworks (GLOSARIO) más utilizados por los programadores para el desarrollo multiplataforma de aplicaciones híbridas.[[3]](#footnote-3)

### 5.5.5 Creación de una Aplicación híbrida

Consiste en diseñar la aplicación como si fuera una Web App para ser ejecutada en el navegador del cliente. La facilidad de desarrollo debe ser balanceada con una experiencia de usuario y apariencia en principio inferior a una aplicación nativa.

### 5.5.6 Aplicación híbrida: app interpretada

La aplicación interpretada significa que la aplicación es programada y luego cada terminal la traduce a su propio lenguaje de programación. Facilita el desarrollo de aplicaciones y reduce el esfuerzo considerablemente. Aunque el resultado no es idéntico a la nativa, la apariencia es bastante buena, y en muchas ocasiones puede ser la solución al problema del desarrollo de aplicaciones multiplataforma.

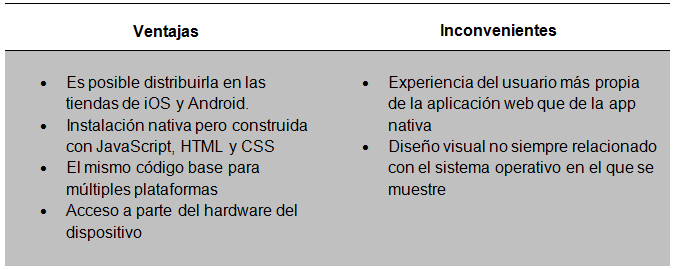


Ilustración - Comparativa aplicaciones híbridas

## **5.6 Entornos y herramientas para el desarrollo**

En el contexto de esta tesis, al momento de seleccionar el tipo de aplicación móvil, dentro de los tipos antes mencionados, se analizaron los entornos de desarrollo y herramientas para el desarrollador.

Ilustración - Herramientas para desarrollo de apps

### 5.6.1 Android Studio

Android Studio es el IDE oficial para el desarrollo de aplicaciones para Android, basado en la tecnología IntelliJ IDEA. Además del editor de códigos y las herramientas para desarrolladores de IntelliJ, Android Studio ofrece la posibilidad de instalarse en los distintos sistemas operativos como Linux, MS Windows, MacOs. Además, incluye características[[4]](#footnote-4) como:

* Integración de ProGuard (reducción de código, eliminación de atributos, clases, métodos sin utilizar)
* Firma de aplicaciones
* Renderizado en tiempo real
* Consola de desarrollador: consejos de optimización, ayuda para la traducción, estadísticas de uso
* Editor de diseño, con posibilidad de arrastrar y soltar elementos
* Herramientas Lint (detección de problemas de rendimiento, usabilidad, compatibilidad de versiones, etc)
* Soporte para crear diseños en Android Wear (Sistema operativo para dispositivos corporales de Android)
* Google Cloud Platform
* Dispositivos virtuales para simular las aplicaciones

Provee también emuladores para diferentes plataformas de hardware, destinados a la prueba de Apps.

### 5.6.2 App Inventor

Es un entorno de desarrollo de software creado por Google Labs para la elaboración de aplicaciones destinadas al sistema operativo Android. El usuario puede, de forma visual y a partir de un conjunto de herramientas básicas, ir enlazando una serie de bloques para crear la aplicación. El sistema es gratuito y se puede descargar fácilmente desde la web[[5]](#endnote-2) la aplicación generada. Las aplicaciones creadas con App Inventor están limitadas por su simplicidad, aunque permiten cubrir un gran número de necesidades básicas en un dispositivo móvil.

Con Google MIT App Inventor, se espera un incremento importante en el número de aplicaciones para Android debido a dos grandes factores: la simplicidad de uso, que facilitará la aparición de un gran número de nuevas aplicaciones; y Google Play, el centro de distribución de aplicaciones para Android donde cualquier usuario puede distribuir sus creaciones libremente.

Otra gran cualidad es la posibilidad de insertarlo en la educación dado su programación por medio de bloques gráficos, que resultan ser muy intuitivos en aquellas personas que se introducen.

Unas primeras apps para el SAR fueron realizadas bajo esta plataforma, de dicha experiencia se pudo concluir que tiene una baja curva de aprendizaje.

### 5.6.3 Tecnologías del lado del cliente - Open Web Stack (HTML, CSS y JS)

A continuación, se mencionan tecnologías utilizadas tanto para aplicaciones web como, Web Apps y aplicaciones híbridas.

### 5.6.3.1 HTML

 Es un [lenguaje de marcado](https://es.wikipedia.org/wiki/Lenguaje_de_marcado) para la elaboración de [páginas web](https://es.wikipedia.org/wiki/P%C3%A1gina_web). Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros.

### 5.6.3.2 CSS

Como se mencionó anteriormente (Sección 5.5.1), CSS (Hoja de estilos en cascada) es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado.

### 5.6.3.3 JS

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, ​ basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (*client-side*), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, ​ aunque existe una forma de JavaScript del lado del servidor.

### 5.6.3.4 SASS

Es un metalenguaje de estilos en cascada. Es un lenguaje que preprocesado produce CSS, denominado SassScript. Existen dos formatos SCSS (Sintaxis con bloques) y SASS (Sintaxis con indentación). CSS3 consiste en una serie de selectores y pseudo-selectores que agrupan las reglas que son aplicadas.

SassScript proporciona los mecanismos necesarios para ofrecer variables, nesting (anidamientos), mixins, y herencia de los selectores.

Las variables Permite reutilizar valores que podemos manejar desde un solo sitio de forma sencilla y centralizada

Un mixin permite aprovechar un fragmento de código al que podemos llamar repetidamente, evitando repetición.

La sintaxis de .sass y .scss no puede ser interpretada directamente por los navegadores, por ende es necesario la compilación.

### 5.6.3.5 Angular JS

Es un framework MVC (Model, View, Controller) de JavaScript para el Desarrollo Web Front End que permite crear aplicaciones SPA Single-Page Applications (una única página).

### 5.6.4 Cordova

Apache Cordova es un entorno de desarrollo de aplicaciones móviles, originalmente creado por Nitobi y comprado por Adobe [[6]](#footnote-5). Más tarde fue liberado como Apache Cordova. Permite, construir aplicaciones para dispositivos móviles utilizando CSS3, HTML5, y Javascript. Las aplicaciones resultantes son híbridas, lo que significa que no son ni una aplicación móvil nativa o App (debido a que la representación gráfica se realiza con vistas Web) ni puramente basadas en web (Están empaquetadas como aplicaciones para su distribución y tienen acceso a las APIs nativas del dispositivo en lenguaje JavaScript).

### 5.6.5 Intel XDK

Es un kit de desarrollo creado por Intel para crear aplicaciones nativas para los teléfonos celulares y las tabletas que utilizan tecnologías web como HTML5, CSS y JavaScript. Las aplicaciones se compilan mediante un servicio on-line. Hace uso de la plataforma Cordova para crear aplicaciones cross-platform, enfocado en el segmento de Apps para IoT. Posee un emulador, pre-visualización de aplicaciones mediante el scan de un código QR, Drag and Drop y soporte de plantillas. Brinda soporte para Android, iOS, Windows Phone, entre otras plataformas.

### 5.6.6 Ionic

Es una Framework, Open Source y de distribución gratuita, para el desarrollo de aplicaciones híbridas, inicialmente pensado para móviles y *tablets*, basadas en HTML5, CSS y JS. Está construido con Sass y optimizado con AngularJS.

### 5.6.6 Meteor

Es una plataforma para crear aplicaciones *webs en tiempo real* construida sobre Node.js. Meteor se localiza entre la base de datos de la aplicación y su interfaz de usuario y se encarga que las dos partes estén sincronizadas.

Meteor puede compartir código JavaScript entre el cliente y en el servidor.

### 5.6.7 Meteor y Cordova

Existe una integración del framework Meteor con Cordova, que permite que una aplicación web creada con Meteor, sea ejecutarla en un dispositivo iOS o Android de forma híbrida. Un beneficio importante de empaquetar la aplicación web como una aplicación de Cordova es que todos sus recursos no deben ser descargados desde la web, asegurando una velocidad de carga mayor, beneficiando a los usuarios con conexiones lentas. Otra característica es la compatibilidad con *hot code push*, que le permite actualizar la aplicación en los dispositivos de los usuarios sin pasar por el proceso habitual de revisión de la tienda de aplicaciones. Cordova también permite el acceso a ciertas características nativas a través de una arquitectura de complementos. Los complementos permiten utilizar funciones que normalmente no están disponibles para aplicaciones web, como acceder a la cámara del dispositivo o al sistema de archivos local, interactuar con lectores de código de barras o NFC.

5.7 Resumen

Como vimos en este capítulo, las aplicaciones móviles son aplicaciones informáticas diseñadas para ser ejecutadas en teléfonos inteligentes, tabletas y otros dispositivos móviles y que permite al usuario efectuar una tarea con mayor versatilidad que con una computadora de escritorio.

Se detallaron ventajas y desventajas sobre cada tipo de aplicación y su forma de desarrollo; como por ejemplo, las Apps nativas no requieren de conectividad a internet en comparación a las Apps web. Las Apps hibridas poseen ventajas agregadas de las otras dos.

1. <https://www.developereconomics.com/graphs/de11>

   <https://www.xatakamovil.com/mercado/desarrollo-de-aplicaciones-moviles-i-asi-esta-el-mercado> [↑](#footnote-ref-1)
2. https://es.wikipedia.org/wiki/Dise%C3%B1o\_web\_adaptable [↑](#footnote-ref-2)
3. <https://es.wikipedia.org/wiki/PhoneGap>

   <https://es.wikipedia.org/wiki/Apache_Cordova> [↑](#footnote-ref-3)
4. https://es.wikipedia.org/wiki/Android\_Studio# [↑](#footnote-ref-4)
5. http://appinventor.mit.edu/explore/ai2/windows.html [↑](#endnote-ref-2)
6. <https://es.wikipedia.org/wiki/Apache_Cordova>

   http://cordova.apache.org/ [↑](#footnote-ref-5)