



**Técnico en**  
**< DESARROLLO DE SOFTWARE >**

***Introducción al Desarrollo de  
Aplicaciones Móviles***

(CC BY-NC-ND 4.0)  
International

Attribution-NonCommercial-NoDerivatives 4.0



## **Atribución**

Usted debe reconocer el crédito de una obra de manera adecuada, proporcionar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que tiene el apoyo del licenciante o lo recibe por el uso que hace.



## **No Comercial**

Usted no puede hacer uso del material con fines comerciales.



## **Sin obra derivada**

Si usted mezcla, transforma o crea un nuevo material a partir de esta obra, no puede distribuir el material modificado.

No hay restricciones adicionales - Usted no puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otros hacer cualquier uso permitido por la licencia.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



# ***Introducción al Desarrollo de Aplicaciones Móviles***

## ***Unidad I***

### **1. Historia de los dispositivos móviles**

Los dispositivos móviles en la actualidad están presentes prácticamente en cualquier rincón a donde vayamos y a pesar de que no hace mucho resultaba complejo construir aplicaciones para ellos, hoy en día es relativamente sencillo desarrollar programas, basta tomar algo de la abundante literatura existente, elegir un lenguaje de programación y descargar el software necesario que permita la instalación de un entorno de desarrollo apropiado para dicha actividad. A continuación relato de manera breve la historia de la programación de estos dispositivos, culminando con el panorama actual de esta interesante actividad.

Aunque los inicios de la programación para computadoras se remonta varias décadas atrás, iniciando con la programación física (cambio directo de cables para lograr la programación, es decir una especie de cambio de variables), pasando por el lenguaje máquina, ensamblador hasta llegar a lenguajes de alto nivel, la programación para móviles no ha tenido que pasar por tantos caminos escabrosos, pero aun así, por diversas etapas que han permitido su evolución. Si bien, antes de los dispositivos inteligentes o smartphones existieron los teléfonos de funciones o feature phones, el

enfoque se realizará sobre los primeros, ya que estos son lo que realmente permiten la instalación y configuración personalizada de aplicaciones.

## Los precursores: Las calculadoras programables



*Calculadora programable HP 65*

Por la década de los 70 y 80 compañías como Casio, Texas Instruments y Hewlett-Packard crearon calculadoras programables. Tal es el caso de la HP-65, la cual empaquetaba 9 registros de almacenamiento y tenía espacio para

100 instrucciones por teclado.

También incluyó una lectora/escritora de tarjeta magnética, los programas usaron RPN (Notación Polaca Inversa) para conservar la memoria. La documentación para los programas en la calculadora era muy completa, incluyendo algoritmos para centenares de aplicaciones, soluciones de ecuaciones diferenciales, estimación de precios, estadística, etc.

Pueden considerarse como precursores de los dispositivos móviles, ya que incluían conexión a impresora, capacidades gráficas, módulos de memoria expandibles de 1 kb e incorporación de Lenguaje Máquina, Basic y posteriormente Pascal y C.

## Newton: El abuelo de los PDA

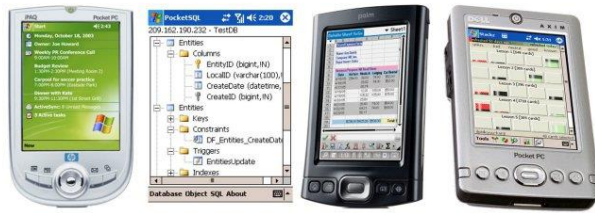


Por el año de 1993, Apple comercializo un dispositivo llamado Newton, que para muchos representa el primer PDA.

Este aparato contaba con interesantes funciones entre ellas el reconocimiento de escritura y la sincronización con una computadora de escritorio, aplicaciones para notas, contactos y fechas, calculadora, conversión de unidades, zona horaria, mapas, un modem, envió de faxes, pantalla táctil, llamadas telefónicas, navegador y gestión de correo electrónico. Usaba un sistema operativo llamado Newton OS y funcionaba con pilas AAA.

Es posible de hablar de desarrollo de aplicaciones para este dispositivo, no como las tiendas que conocemos ahora, pero si creadas por terceros para funciones muy específicas y de distribución limitada. Sin embargo, dada las avanzadas funciones para la época, el público no estaba preparado para ello, si sumamos el poco desarrollo y el impreciso reconocimiento de escritura fueron factores claves para que fuera retirado del mercado con más pena que gloria. Actividad que le toco precisamente al hijo prodigo: Steve Jobs.

## Palm y las PocketPC: El paso de la estafeta

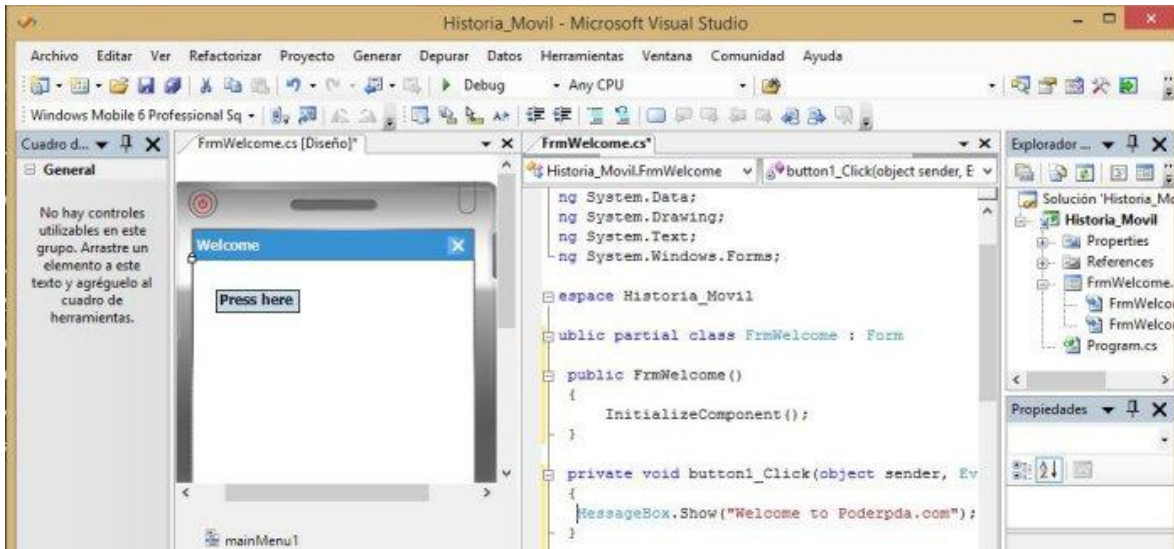


Palm comenzó como una empresa dedicada al desarrollo de software en los 90, generando aplicaciones de sincronización para Casio y HP, software de reconocimiento de escritura

para la Newton entre otras; sin embargo poco después crea el sistema operativo Palm Os, generando sus propios dispositivos y licenciando el sistema a terceros.

Por espacio de más de una década constituyeron equipos de gran prestigio llegando a poseer en el 2001 el 60% del mercado americano. Para el caso de las Pocket Pc, Microsoft entró en escena un par de años después que Palm e introdujo el termino Computadora de Bolsillo, en cual tenía instalado como sistema operativo Windows CE, un conjunto de aplicaciones en ROM, pantalla táctil y un dispositivo apuntador o stylus. Para ese entonces la disposición de aplicaciones ya era más extendida, existiendo un catálogo con decenas de ellas, además, los dispositivos se diversifican en dos ramas: los PDA y los Smartphone, diferenciándose estos últimos por las funciones de telefonía como mensajes y llamadas. Precisamente fue con un iPaq de HP similar al mostrado en la imagen con el cual abordé el barco e inicié la aventura y pasión por la tecnología móvil, ese cacharro tecnológico aún está conmigo y me recuerda lo mucho que ha avanzado esta tecnología en tan pocos años.

## 2. IDE Mobile: El entorno de desarrollo integrado para dispositivos móviles



Cuando el uso de dispositivos móviles se hizo algo común, también la programación de aplicaciones para estos se tomó más en serio, así diversas compañías crear entornos de programación o complementos a IDEs existentes y emuladores para permitir la creación de aplicaciones, entre ellos:

- Pda Toolbox,
- Satellite Forms,
- Scotbuilder,
- Palm Os Developer Studio,
- HS Pascal,
- Mobile Visual Basic
- Embedded Visual Tools



Al mencionar las características de estos entornos destacan el entorno visual, el uso de controles como botones y etiquetas, los compiladores, emuladores y la generación de aplicaciones con código nativo para el equipo específico. En el uso de los lenguajes de programación se incluyen C, C++, Basic, Pascal y Java. Para la instalación de las aplicaciones también existían varias opciones tales como instalación mediante sincronización con una computadora, copiado hacia la memoria del dispositivo y recientemente mediante descargas de internet. Aunque el Internet ya existía en ese entonces aún no era de fácil acceso por lo que capacitarse en tales actividades requería adquirir libros, buscar desesperadamente manuales ocultos en los archivos de instalación o en caso adverso acudir a un experto con la esperanza que este compartiera su valioso conocimiento en el “difícil y oscuro mundo de la programación”.

### 3. Entornos de desarrollo

#### BlackBerry: El solitario empresarial



Los móviles BlackBerry se crearon unos años después del surgimiento de los dispositivos Palm, para ser exacto el primero de ellos, el BlackBerry 850 en 1999, el cual tan solo era un buscapersonas, es decir, no contaba con funciones de telefonía.

Para los subsecuentes dispositivos se fueron agregando características de telefonía y servicios que lo hicieron único y triunfador, enfocado a un mundo empresarial, con un teclado Qwerty, las soluciones robustas y seguras BIS y BES y, por supuesto el famoso pin que le permitieron hacerse un hueco en el prometedor mercado de la tecnología móvil. En el apartado de la programación, pronto soporto la instalación de aplicaciones y



entornos de desarrollo basados en los lenguajes c y java, optando constantemente por crear plugins especiales para IDEs conocidos como Visual Studio, Eclipse o Netbeans.

## Java: El surgimiento de un Gigante



En 1995 Sun Microsystems presento el lenguaje de programación Java de la mano de James Gosling. Desde su nacimiento java ponía cierto énfasis en especificaciones

esenciales para dispositivos de bajo consumo, por ello, con la llegada de Java ME (Micro Edition), se establecieron las bases para la creación de aplicaciones enfocadas a dispositivos con memoria, pantalla y en general recursos limitados. Pronto la mayoría de los fabricantes sin importar el sistema operativo ofrecieron soporte java, que básicamente consistía en implementar una máquina virtual sobre la cual se pudiese ejecutar programas creados en este lenguaje, ya fuese Symbian, Web Os, Maemo, Windows Mobile, Palm OS, BlackBerry o cualquier otro sistema operativo. Al día de hoy Java es uno de los lenguajes más utilizados a nivel mundial por lo que invertir en su aprendizaje es altamente recomendado.

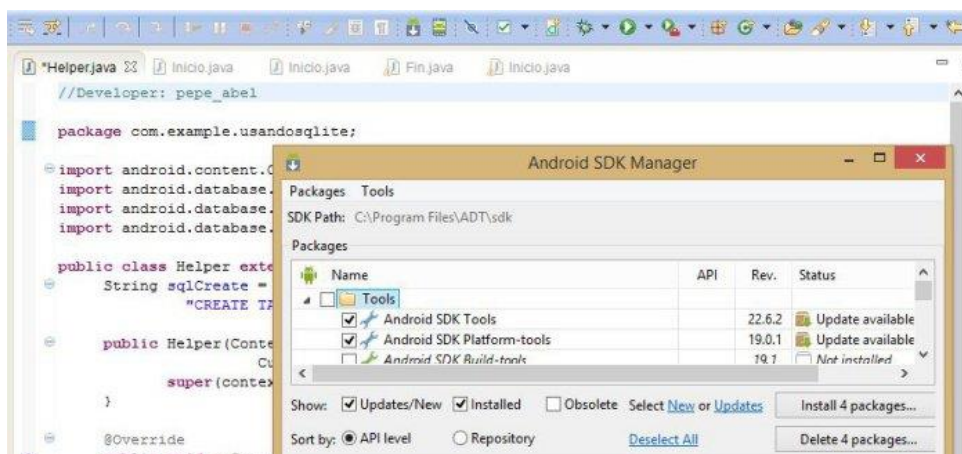
## Objetivo C: El declive de algunos y el salto evolutivo



Corría el año 2006 y todo parecía normal: BlackBerry con buena salud, las Pocket PC en buen estado y Palm con salud suficiente, sin embargo en los laboratorios de Apple ya se estaba cocinando algo y era precisamente el iPhone. Este smartphone surgió en enero de 2007 y aunque fue presentado como una gran revolución en realidad carecía de elementos esenciales de otras plataformas como el copiar y pegar y el bluetooth liberado entre otras cosas; pero gracias a la mercadotecnia y a factores de diseño y estética logro tomar posición de manera rápida y fomentar la investigación y desarrollo que hacía falta en todo el ecosistema móvil. Para el desarrollo de aplicaciones esta plataforma ha utilizado el lenguaje Objective C, que básicamente está basado en C con algunas implementaciones tomadas de SmallTalk, si bien este lenguaje fue creado en los 80, hasta que Apple lo tomó que comenzó a ser popular; como entorno de desarrollo se ha utilizado Xcode, que incluye compiladores GCC que no se limita a Objective c, ya que permite C, C++, Java, Free Pascal, Ada y Lisp; además el componente Interface Builder se encarga de la creación de interfaces de usuario. Aunque en ese tiempo ya existían tiendas de aplicaciones para otras plataformas, la tienda para el iPhone marco la pauta

ya que todo estaba disponible desde una aplicación embebida en el sistema, con lo cual las demás plataformas siguieron esta tendencia. Desafortunadamente para los sistemas operativos existentes, no se vio venir el cambio y se durmieron en sus laureles, generando poca innovación y lanzando solo parches y actualizaciones menores, por ello, no pudieron seguir el paso y la mayoría de ellas sucumbió o está en ese doloroso proceso.

## Android y Windows Phone: Renovarse o morir



Google siempre ha sido ambicioso y con gran visión, por ello desde el año 2005 compró la compañía Android Inc. con el firme objetivo de

contar con un sistema operativo y el consecuente desarrollo de dispositivos. La llegada del iPhone representó un cambio de planes y aceleró el desarrollo, de forma que en el 2008 presentó su primer teléfono inteligente, el HTC Dream. Desde sus inicios la programación se ha realizado en el IDE Eclipse con un kit de desarrollo y plugins para cada versión específica de la API, aunque también es posible configurar un entorno con NetBeans o IntelliJ IDEA. Como lenguaje de programación se utiliza básicamente Java, Javascript, Basic y C, adicionalmente existe un entorno web llamado Android Inventor, que sin necesidad de código se pueden crear aplicaciones de mediana complejidad. Dado que el sistema es licenciado, en Android existe un conjunto de empresas asociadas

que desarrollan emuladores y herramientas específicas para el diseño, compilado y depuración, lo cual permite contar con una gama de opciones al momento de elegir las herramientas más adecuadas para programar. Estas características le han permitido crecer rápidamente al grado que al día de hoy, es la plataforma número uno a nivel mundial.

Por su parte Microsoft no pudo evitar la caída y desaparición de Windows Mobile, pero gracias a su musculo económico, diseñó Windows Phone y presentó una versión incipiente en el 2010. Aprovechando sus tecnologías consolidadas, hoy es posible crear aplicaciones con el IDE Visual Studio que en la actualidad contiene una gran cantidad de herramientas, emuladores y ayudas que hacen del desarrollo una tarea relativamente sencilla (necesario instalar el SDK de Windows Phone para habilitar las plantillas de creación). Los lenguajes de programación incluyen C++, C# y Visual Basic y el uso de emuladores requiere que el equipo de cómputo admita la virtualización vía hardware.

## 4. Tienda de Aplicaciones: La Fuente de ingresos

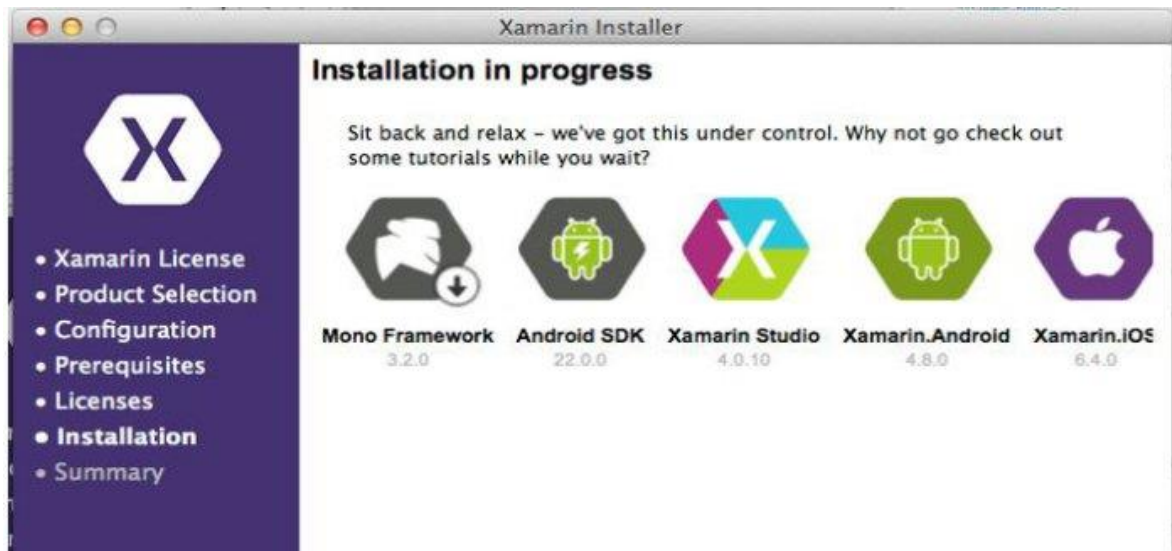


Las tiendas de aplicaciones modernas existen desde el año 2008. En la actualidad todo sistema estable y robusto cuenta con una tienda de aplicaciones que permiten al usuario funcionalidad adicional para suplir necesidades específicas, donde el desarrollador se queda con un o porcentaje de las ganancias obtenidas. Si bien al principio se tenía la idea de abundantes ganancias por la venta de aplicaciones hoy en día es difícil monetizar una aplicación basado solo en la venta y existen otras opciones como:

- Aplicaciones gratuitas completas con fecha de caducidad con opción a pago al terminar el periodo
- Aplicaciones con pago por suscripción
- Aplicaciones gratuitas completas con publicidad
- Aplicaciones gratuitas con funcionalidad reducida y venta de extras

Para hacer disponible una aplicación en alguna de las tiendas es preciso seguir los lineamientos que marca la misma, la cual generalmente tiene una serie de pasos que garantiza la estabilidad, seguridad y funcionalidad de la aplicación. Cabe señalar que existen tiendas extraoficiales que ofrecen condiciones un tanto distintas a las establecidas en la oficial, además es posible la instalación de las mismas de manera directa hacia el dispositivo lo cual puede comprometer el sistema si la fuente de la aplicación es desconocida y en algunos casos es preciso realizar rooteo con lo cual se pierde la garantía del equipo.

## 5. Desarrollo Multiplataforma: Desarrolla una vez, impleméntalo en diversas plataformas



El uso de un emulador siempre ha sido una forma barata de realizar pruebas y depuración cuando no se tiene físicamente el dispositivo, pero siempre ha representado una limitante por no simular al 100% las características del equipo, generación de errores y lentitud; pero, para bien de los desarrolladores hoy en día es posible conseguir un equipo de prestaciones medianas por un costo módico y así, agilizar las actividades inherentes. Por otra parte, en los últimos años han surgido plataformas de desarrollo que permiten la creación de aplicaciones multiplataforma con mínima configuración para su implementación en ambientes específicas:

- Xamarin
- Appcelerator Titanium
- Phonegap



Está claro que esta modalidad de desarrollo genera ventajas ya que solo se necesita crear el código una sola vez y posteriormente compilar e implementar en más de una plataforma con el consecuente ahorro de tiempo, desarrollo estándar, etc.; también es cierto que presenta algunos inconvenientes según el IDE seleccionado, tales como Complejidad del código fuente, dificultad de mantenimiento, coherencia, experiencia de usuario menos enriquecida y generalmente son entornos de paga. Al final la elección depende de las características particulares y enfoque de cada aplicación.

## Referencia:

URL: <http://www.poderpda.com/> Blog: Fecha: 05/09/2014 Autor: José Abel

---

### ***Descargo de responsabilidad***

La información contenida en este documento descargable en formato PDF o PPT es un reflejo del material virtual presentado en la versión online del curso. Por lo tanto, su contenido, gráficos, links de consulta, acotaciones y comentarios son responsabilidad exclusiva de su(s) respectivo(s) autor(es) por lo que su contenido no compromete al área de e-Learning del Departamento GES o al programa académico al que pertenece.

El área de e-Learning no asume ninguna responsabilidad por la actualidad, exactitud, obligaciones de derechos de autor, integridad o calidad de los contenidos proporcionados y se aclara que la utilización de este descargable se encuentra limitada de manera expresa para los propósitos educativos del curso.

