**UNIVERSIDAD CATOLICA DE COSTA RICA**

**ESCUELA DE INGENIERIA EN SISTEMAS Y COMPUTACION**

**SISTEMAS OPERATIVOS 2**

**INVESTIGACION DE DESARROLLO DE APLIACIONES IOS**

**ANDRES JIMENEZ**

**EISNER LOPEZ ACEVEDO**

**SEDE SAN CARLOS**

**Junio de 2016**

Tabla de contenido

[Resumen Ejecutivo. 3](#_Toc466102284)

[Abstract 5](#_Toc466102285)

[CAPITULO I: INTRODUCCION. 8](#_Toc466102286)

[Planteamiento del problema. 8](#_Toc466102287)

[Objetivos 9](#_Toc466102288)

[Objetivos específicos. 9](#_Toc466102289)

[Objetivos Generales. 9](#_Toc466102290)

[Justificación del Tema 10](#_Toc466102291)

[Proyecciones. 11](#_Toc466102292)

[CAPITULO II: MARCO TEORICO 12](#_Toc466102293)

[Webs Móviles y Apps Móviles. 12](#_Toc466102294)

[Sistemas Operativos Móviles 13](#_Toc466102295)

[Android 13](#_Toc466102296)

[iOS 14](#_Toc466102297)

[Windows Phone 15](#_Toc466102298)

[BlackBerry 6 15](#_Toc466102299)

[Symbian 15](#_Toc466102300)

[Firefox O.S 16](#_Toc466102301)

[Ubuntu Touch 17](#_Toc466102302)

[Teléfono Inteligente. 17](#_Toc466102303)

[Xcode 18](#_Toc466102304)

[Objetive-C 19](#_Toc466102305)

[Swift 19](#_Toc466102306)

[Apache Flex 21](#_Toc466102307)

[XML 21](#_Toc466102308)

[Adobe ActionScript 21](#_Toc466102309)

[CAPITULO III: PROCEDIMIENTO METODOLOGICO. 23](#_Toc466102310)

[Enfoque de la investigación. 23](#_Toc466102311)

[Proceso desarrollo 23](#_Toc466102312)

[Etapa de Análisis. 23](#_Toc466102313)

[Definición de requisitos. 24](#_Toc466102314)

[Etapa de Diseño. 24](#_Toc466102315)

[CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES. 26](#_Toc466102316)

[Conclusiones. 26](#_Toc466102317)

[Recomendaciones. 26](#_Toc466102318)

[Referencias 27](#_Toc466102319)

# Resumen Ejecutivo.

Para crear aplicaciones en IOS, es necesario tener dos cosas invariablemente el sistema operativo, para Windows, se necesitan dos cosas, la primera es una Mac ya sea física, en la nube o virtualizada, y la segunda para crear una cuenta de desarrollador de Apple.

Siguiendo el ejemplo de Windows, puede realizar en diferentes IDE como son el Xamarin Visual Studio y Xcode Mac. Xamarin requiere una IP de un Mac, para programar. Para Xcode es donde cae, tener un Mac virtual o físico.

El último recurso es crear una aplicación que sea multiplataforma. Al igual que Adobe Flash Builder o Flash Developer, el primero de pago, el segundo código abierto.

Para desarrollar aplicaciones iOS utilizando las últimas tecnologías, necesita una computadora Mac (OS X 10.10 o posterior) que ejecute la última versión de Xcode. Xcode incluye todas las características que necesitas para diseñar, desarrollar y depurar una aplicación. Xcode también contiene el iOS SDK, que extiende Xcode para incluir las herramientas, compiladores y frameworks que necesitas específicamente para el desarrollo de iOS. Descarga la última versión de Xcode en tu Mac gratis desde la App Store.

No existe una forma oficial de ejecutar Xcode en una PC con Windows o Linux. Esto significa que, si desea desarrollar una aplicación de iPhone, pero no tiene una Mac, tendrá que invertir en una primera.

Para desarrollar aplicaciones iOS 10, necesitará Xcode 8 y el iOS 10 SDK, también disponible en Apple. Hay cambios significativos en el lenguaje Swift y en el SDK de esta versión de Xcode que querrás tener en cuenta, pero la buena noticia es que no habrá más cambios de "código" en el lenguaje Swift después de esta versión.

Familiarícese con Objective-C. Objective-C es el lenguaje de programación utilizado para crear funcionalidad dentro de las aplicaciones de iPhone. Maneja la manipulación de datos y objetos. Objetivo-C es un derivado de la familia C de lenguas, y es un lenguaje orientado a objetos. Si ya tiene una comprensión básica de C o Java, Objective-C debe ser, en su mayor parte, fácil de entender.

Una opción más nueva es crear su aplicación usando el lenguaje Swift, el seguimiento de Objective-C. Swift tiene una sintaxis mucho más amigable y una sensación más moderna.

Cree una cuenta de desarrollo. Para distribuir tu aplicación en la App Store o darla a otros para que la pruebe, debes registrarte para una cuenta de desarrollador de Apple. La cuenta cuesta $ 99 por año y requerirá que ingrese información sobre impuestos y cuentas bancarias. Puede crear su cuenta en el sitio web iOS Dev Center.

Puede acceder a esto desde el iOS Dev Center. Si tiene algún paso pendiente para completar la instalación, aparecerá en la parte superior de la página. Asegúrese de que toda su información bancaria e impositiva se ha introducido correctamente.

No hay nada que hacer ahora, excepto esperar al proceso de revisión. Apple le notificará en unos pocos días o semanas por correo electrónico si su estado de aplicación cambia a "En revisión". Cuando entra en revisión, el proceso es muy rápido. Si falla el aluvión inicial de pruebas, obtendrá un correo electrónico de rechazo diciéndole por qué, y Apple ofrecerá sugerencias para solucionarlo. Si su aplicación pasa la inspección, Apple le enviará un correo electrónico informándole que su aplicación está lista para la venta. Su aplicación aparecerá ahora en la tienda de aplicaciones de iTunes.

# Abstract

To create applications in IOS, it is necessary to have two things invariably the operating system, for example for windows, you need two things, the first is a Mac either in physical, cloud or virtualized, and the second to create an Apple developer account.

Following the example of Windows, you can perform in different IDE's as are the Xamarin Visual Studio and Xcode Mac. Xamarin requires a IP of an Mac, to program. For Xcode it is where it falls, having a virtual or physical Mac.

The last resort is to create an application that is multiplatform. As is Adobe Flash Builder or Flash Developer, the first payment, the second open source.

To develop iOS apps using the latest technologies, you need a Mac computer (OS X 10.10 or later) running the latest version of Xcode. Xcode includes all the features you need to design, develop, and debug an app. Xcode also contains the iOS SDK, which extends Xcode to include the tools, compilers, and frameworks you need specifically for iOS development. Download the latest version of Xcode on your Mac free from the App Store.

There is no official way to run Xcode on a Windows or Linux PC. This means that if you want to develop an iPhone app but don't have a Mac, you'll need to invest in one first.

To develop iOS 10 apps, you'll need Xcode 8 and the iOS 10 SDK, also available from Apple. There are significant changes to the Swift language and the SDK in this version of Xcode that you'll want to be aware of, but the good news is that there are to be no more "code breaking" changes to the Swift language after this version.

Familiarize yourself with Objective-C. Objective-C is the programming language used to create functionality within iPhone apps. It handles data and object manipulation. Objective-C is a derivative of the C family of languages, and is an object-oriented language. If you already have a basic understanding of C or Java, Objective-C should be, for the most part, easy to grasp.

A newer option is to create your app using the Swift language, the follow-on to Objective-C. Swift has a much friendlier syntax and a more modern feel.

While it's possible to build a basic app without knowing Objective-C, you cannot perform any sort of advanced function without coding it yourself. Without Objective-C, all you can do is move back and forth between screens.

Some of the more popular online Objective-C and Swift communities include the Apple Developer Forums, the iPhoneSDK Google Group, and StackOverflow.

Create a development account. To distribute your app on the App Store or give it to others to test, you will need to sign up for an Apple Developer account. The account costs $99 per year and will require you to enter in tax and bank account information. You can create your account at the iOS Dev Center website.

Once you've signed up for a Developer account, you will have access to Apple's development resources. These resources include a variety of sample projects which can give you a tremendous amount of insight into how app development works. Find an example that is related to the kind of app you want to create and mess around with it in Xcode.

Select Device and Release from the drop-down menu. Press the "Build" icon. In the Finder, navigate to your project's build folder and look for the "Release-iphoneos" folder. Inside there will be an app. Zip it up into an archive.

For new apps to pass Apple certification, they will need to be optimized for iOS 8 and the Retina display.

You can access this from the iOS Dev Center. If you have any unfinished steps to complete for setup, they will be listed at the top of the page. Make sure that all of your banking and tax information has been entered correctly.

There is nothing to do now except wait for the review process. Apple will notify you in a few days or weeks via email if your application state changes to "In review". When it goes into review, the process is very quick. If it fails the initial barrage of tests, you will get a rejection email telling you why, and Apple will offer suggestions for fixing it. If your app passes inspection, Apple will send you an email informing you that your app is ready for sale. Your app will now appear on the iTunes app store.

# CAPITULO I: INTRODUCCION.

# Planteamiento del problema.

Hoy en día vemos como los celulares inteligentes (Smartphone) se posicionan en el mercado causando euforia entre los usuarios y un crecimiento notorio en las ventas. Al entrar al mercado han desplazado a los celulares analógicos dejándolos en el olvido y supliendo una gran necesidad entre los usuarios como el manejo de las redes sociales, internet, correos electrónicos entre otros.

Dentro de las causas se encuentra la aparición de las redes sociales, el manejo de correo electrónico, las señales inalámbricas como el Wifi, y los tipos diferentes de redes. Además de la internet.

Los diferentes sistemas operativos como Android de google, Windows Phone y iOS entre otros; donde los usuarios buscan interactuar teniendo una conectividad a Internet y estando al margen de la tecnología.

# Objetivos

## Objetivos específicos.

Investigar sobre las aplicaciones móviles en iOS.

## Objetivos Generales.

Identificar los métodos sobre la creación de las aplicaciones móviles en iOS.

Mencionar los usos de las aplicaciones móviles iOS.

Desarrollar una aplicación utilizando los métodos para la creación de las aplicaciones móviles en iOS.

# Justificación del Tema

El motivo principal para el desarrollo de aplicaciones móviles es el auge del concepto movilidad que existe en la actualidad y la previsible tendencia a futuro de este tipo de tecnologías.

La elección de desarrollar para iOS se basa en que dispongo de la creatividad y en mis conocimientos del lenguaje de programación Java.

Desconozco totalmente el sistema operativo IOS y el lenguaje de programación Objetive-C.

Previamente he realizado algún desarrollo a nivel formativo para Android con Android Studio por lo que a nivel formativo resulta más interesante aprender a desarrollar nuevas tecnologías como lo son iOS para mí.

Cada vez es mayor el acceso a Internet desde los dispositivos móviles. Dicha tendencia está afectando a todos los sectores.

# Proyecciones.

Con este trabajo se busca, identificar los métodos sobre la creación de las aplicaciones móviles en iOS. Mencionar los usos y por ultimo desarrollar una aplicación utilizando los métodos para la creación de las aplicaciones móviles en iOS.

Es necesario como estudiantes de Ingeniería en Sistemas, la investigación constante, y las posibles limitaciones que esta nos presente, como lo son las limitaciones de coste para poder desarrollar en esta plataforma. Ya que para poder desarrollar una aplicación y poder probarla se debe incurrir en gastos.

Es necesario poseer la curiosidad y las ganas de innovar en cada uno de los campos de la carrera.

Con este trabajo, se pretende desarrolla una aplicación funcional, en forma básica creada con Flash Builder de Adobe, de forma multiplataforma.

# CAPITULO II: MARCO TEORICO

# Webs Móviles y Apps Móviles.

El W3C define la Web Móvil como la web mediante la cual los usuarios pueden acceder desde cualquier lugar con independencia del tipo de dispositivo que se utilice para ello. Aquí se introduce el concepto de ubicuidad y multiplataforma, es decir no importa ni el lugar ni el tipo de dispositivo. Al igual que en su día se extendió el concepto de Web 2.0, donde se le daba importancia a la interacción del usuario, con la difusión de las redes sociales, los blogs etc., ahora con la tecnología móvil se está hablando por parte de los expertos de la Web Móvil 2.0 (Jaokar & Fish, 2006). Se puede pensar que la Web móvil 2.0 es lo mismo que la Web 2.0 incluyendo los conceptos de movilidad y/o ubicuidad. Dichos conceptos permiten, además de entender el uso de los aparatos móviles para la búsqueda de información, interactuar con las redes sociales y poder agregar contenidos para compartir información.

Las ‘apps’ son aplicaciones especialmente diseñadas para los ‘smartphones’ preparadas para la usabilidad móvil, con pantallas más pequeñas, con una interfaz más sencilla y amigable para estos dispositivos y con utilidades para el uso táctil. Muchas de estas aplicaciones vienen pre-instaladas en los terminales móviles o se pueden descargar mediante las tiendas de los principales fabricantes de teléfonos (PlayStore para terminales Android y Appstore para terminarles IPhone). Estas aplicaciones pueden ser de pago o gratuitas.

## Sistemas Operativos Móviles

Conjunto de programas que gestiona los recursos de hardware y provee servicios en un dispositivo móvil.

Un sistema operativo móvil o SO móvil es un sistema operativo que controla un dispositivo móvil al igual que los PCs que utilizan Windows o Linux, los dispositivos móviles tienen sus sistemas operativos como Android, IOS entre otros. Los sistemas operativos móviles son mucho más simples y están más orientados a la conectividad inalámbrica, los formatos multimedia para móviles y las diferentes maneras de introducir información en ellos.

Algunos de los sistemas operativos utilizados en los dispositivos móviles están basados en el modelo de capas.

### Android

El sistema operativo Android es sin duda el líder del mercado móvil en sistemas operativos, está basado en Linux, diseñado originalmente para cámaras fotográficas profesionales, luego fue vendido a Google y modificado para ser utilizado en dispositivos móviles como los teléfonos inteligentes y luego en tablets como es el caso del Galaxy Tab de Samsung , actualmente se encuentra en desarrollo para usarse en netbooks y PCs, el desarrollador de este S.O. es Google, fue anunciado en el 2007 y liberado en el 2008; además de la creación de la Open Handset Alliance, compuesto por 78 compañías de hardware, software y telecomunicaciones dedicadas al desarrollo de estándares abiertos para celulares, esto le ha ayudado mucho a Google a masificar el S.O, hasta el punto de ser usado por empresas como HTC, LG, Samsung, Motorola entre otros.

Android Inc., es la empresa que creó el sistema operativo móvil, se fundó en 2003 y fue comprada por Google en el 2005 y 2007 fue lanzado al mercado. Su nombre se debe a su inventor, Andy Rubin. Originalmente era un sistema pensado para las cámaras digitales.

Android está basado en Linux, disponiendo de un Kernel en este sistema y utilizando una máquina virtual sobre este Kernel que es la responsable de convertir el código escrito en Java de las aplicaciones a código capaz de comprender el Kernel.

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

Una de las grandes cualidades o características de este sistema operativo es su carácter abierto. Android se distribuye bajo dos tipos de licencias, una que abarca todo el código del Kernel y que es GNU GPLv2 ( implica que su código se debe poner al alcance de todos y que todos podremos hacer con este código lo que nos parezca oportuno, modificarlo, ampliarlo, recortarlo, pero siempre estaremos en la obligación de volver a licenciarlo con las misma licencia) Google también por supuesto tiene otra licencia para el resto de componentes del sistema que se licencia bajo APACHE v2 (implica que este código se pueda distribuir para ser modificado y usado a antojo del que lo utilice, pero a diferencia del primer caso, las modificaciones y el código resultante no es obligatorio el licenciarlo bajo las mismas condiciones en las que se encontraba). (Javier, 2013)

### iOS

iOS es el sistema operativo que da vida a dispositivos como el iPhone, el iPad, el iPod Touch o el Apple TV. Su simplicidad y optimización son sus pilares para que millones de usuarios se decanten por iOS en lugar de escoger otras plataformas que necesitan un hardware más potente para mover con fluidez el sistema operativo. Cada año, Apple lanza una gran actualización de iOS que suele traer características exclusivas para los dispositivos más punteros que estén a la venta en ese momento.

Anteriormente denominado iPhone OS creado por Apple originalmente para el iPhone, siendo después usado en el iPod Touch e iPad. Es un derivado de Mac OS X, se lanzó en el año 2007, aumento el interés con el iPod Touch e iPad que son dispositivos con las capacidades multimedia del iPhone, pero sin la capacidad de hacer llamadas telefónicas, en si su principal revolución es una combinación casi perfecta entre hardware y software, el manejo de la pantalla multi-táctil que no podía ser superada por la competencia hasta el lanzamiento del celular Galaxy S I y II por parte de Samsung.

### Windows Phone

Windows Phone (abreviado WP) es un sistema operativo móvil desarrollado por Microsoft, como sucesor de Windows Mobile. Con Windows Phone; Microsoft ofrece una nueva interfaz de usuario que integra varios de sus servicios propios como OneDrive, Skype y Xbox Live en el sistema operativo. Compite directamente contra Android de Google e iOS de Apple. Debido a la evidente fragmentación de sus sistemas operativos, Microsoft anunció en enero de 2015 que dará de baja a Windows Phone, para enfocarse en un único sistema más versátil denominado Windows 10 Mobile, disponible para todo tipo de plataformas (teléfonos inteligentes, tabletas y computadoras). Está diseñado para ser similar a las versiones de escritorio de Windows estéticamente y existe una gran oferta de software de terceros disponible para Windows Phone, la cual se puede adquirir a través de la tienda en línea de Windows Store para móviles. (Ros, 2016)

### BlackBerry 6

BlackBerry es un sistema desarrollado por Research In Motion el cual fue presentado en el WES 2010 junto con un video promocional donde se muestra algunas novedades. RIM apuesta que su BlackBerry 6 estará enfocado en el mercado corporativo y no-corporativo. La mejor experiencia de este sistema se encontrará en los equipos touchscreen (Pantalla Táctil), aunque RIM aseguro que en los equipos que cuenten con un TouchPad o TrackPad podrán ejecutarlo ya que ejerce casi la misma función. Así mismo todavía RIM no ha aclarado cuales son los equipos que se podrán actualizar a esta versión, aunque hay muchos rumores al respecto. RIM en el desarrollo de este OS se enfocó en la parte multimedia hacia el usuario, sin dejar a un lado la parte profesional, también se muestra la integración de las redes sociales y la mensajería instantánea en este. Sin duda RIM quiere dar al usuario una nueva experiencia en su equipo BlackBerry que nadie conocía.

### Symbian

Fue producto de la alianza de varias empresas de telefonía móvil, entre las que se encuentran Nokia como la más importante, Sony Ericsson, Samsung, Siemens, BenQ, Fujitsu, Lenovo, LG, Motorola, esta alianza le permitió en un momento dado ser unos de los pioneros y más usados.

El objetivo de Symbian fue crear un sistema operativo para terminales móviles que pudiera competir con el de Palm o el Smartphone de Microsoft. Técnicamente, el sistema operativo Symbian es una colección compacta de código ejecutable y varios archivos, la mayoría de ellos son bibliotecas vinculadas dinámicamente (DLL por sus siglas en inglés) y otros datos requeridos, incluyendo archivos de configuración, de imágenes y de tipografía, entre otros recursos residentes. Symbian se almacena, generalmente, en un circuito flash dentro del dispositivo móvil. Gracias a este tipo de tecnología, se puede conservar información aun si el sistema no posee carga eléctrica en la batería, además de que le es factible reprogramarse, sin necesidad de separarla de los demás circuitos. Las aplicaciones compatibles con Symbian se desarrollan a partir de lenguajes de programación orientados a objetos como C++, Java (con sus variantes como PJava, J2ME, etc.), Visual Basic para dispositivos móviles, entre otros, incluyendo algunos lenguajes disponibles en versión libre.

### Firefox O.S

Firefox OS es un sistema operativo móvil, basado en HTML5 con núcleo Linux, para smartphones y tabletas. Es desarrollado por Mozilla Corporation bajo el apoyo de otras empresas como Telefónica y una gran comunidad de voluntarios de todo el mundo. Este sistema operativo está enfocado especialmente en los dispositivos móviles incluidos los de gama baja. Está diseñado para permitir a las aplicaciones HTML5 comunicarse directamente con el hardware del dispositivo usando JavaScript y Open Web APIs. Ha sido mostrado en smartphones y Raspberry Pi, compatibles con Android.

En febrero de 2013 Mozilla anunció planes para el lanzamiento mundial de Firefox OS. Mozilla ha comunicado en rueda de prensa antes del inicio del Mobile World Congress en Barcelona, que la primera ola de dispositivos con Firefox OS estará disponible Brasil, Colombia, Hungría, México, Montenegro, Polonia, Serbia, España y Venezuela. Firefox también ha anunciado que LG Electronics, ZTE, Huawei y TCL Corporation se han comprometido a la fabricación de dispositivos con Firefox OS.

### Ubuntu Touch

Ubuntu Touch es un sistema operativo móvil basado en Linux. Es desarrollado por Canonical Ltd. Presentado el 2 de enero de 2013 al público mediante un anuncio en la web de Ubuntu, culmina el proceso de Canonical de desarrollar una interfaz que pueda utilizarse en ordenadores de sobremesa, portátiles, netbooks, tablets y teléfonos inteligentes.

Ubuntu Touch se caracteriza por ser un sistema diseñado para plataformas móviles. Una de sus características más destacadas es:

Pantalla de inicio sin sistema de bloqueo/desbloqueo (que funciona con un nuevo sistema de gestos, y que se aprovecha para mostrar notificaciones).

## Teléfono Inteligente.

El teléfono inteligente (smartphone en inglés) es un tipo de teléfono móvil construido sobre una plataforma informática móvil, con mayor capacidad de almacenar datos y realizar actividades, semejante a la de una minicomputadora, y con una mayor conectividad que un teléfono móvil convencional. El término inteligente, que se utiliza con fines comerciales, hace referencia a la capacidad de usarse como un computador de bolsillo, y llega incluso a reemplazar a una computadora personal en algunos casos.

Generalmente, los teléfonos con pantallas táctiles son los llamados teléfonos inteligentes, pero el soporte completo al correo electrónico parece ser una característica indispensable encontrada en todos los modelos existentes y anunciados desde 2007. Casi todos los teléfonos inteligentes también permiten al usuario instalar programas adicionales, habitualmente incluso desde terceros, hecho que dota a estos teléfonos de muchísimas aplicaciones en diferentes terrenos; sin embargo, algunos teléfonos son calificados como inteligentes aun cuando no tienen esa característica.

Entre otros rasgos comunes está la función multitarea, el acceso a Internet vía Wifi o redes 4G, 3G o 2G, función multimedia (cámara y reproductor de videos/mp3), a los programas de agenda, administración de contactos, acelerómetros, GPS y algunos programas de navegación, así como ocasionalmente la habilidad de y leer documentos de negocios en variedad de formatos como PDF y Microsoft Office.

## Xcode

Xcode es el entorno de desarrollo integrado (IDE, en sus siglas en inglés) de Apple Inc. y se suministra gratuitamente junto con Mac OS X. Xcode trabaja conjuntamente con Interface Builder, una herencia de NeXT, una herramienta gráfica para la creación de interfaces de usuario.

Xcode incluye la colección de compiladores del proyecto GNU (GCC), y puede compilar código C, C++, Swift, Objective-C, Objective-C++, Java y AppleScript mediante una amplia gama de modelos de programación, incluyendo, pero no limitado a Cocoa, Carbón y Java. Otras compañías han añadido soporte para GNU Pascal (microbizz, 2011), Free Pascal (Pascal Central, s.f.), Ada y Perl (Perl CamelBones, s.f.).

Entre las características más apreciadas de Xcode está la tecnología para distribuir el proceso de construcción a partir de código fuente entre varios ordenadores, utilizando Bonjour.

Xcode se introdujo el 24 de octubre de 2003 junto con la versión 10.3 de Mac OS X, siendo desarrollado a partir del anterior entorno de desarrollo, Project Builder, al que sustituyó. Project Builder, a su vez, también era una herencia de la compañía NeXT, fusionada con Apple en 1996.

La aparición de Xcode 2.1 en junio de 2005 fue significativa porque proporcionó a la comunidad de desarrolladores las herramientas para crear binarios universales que permiten al software creado para Mac OS X ser ejecutado tanto en la arquitectura PowerPC como en la nueva, basada en Intel (x86). Esta versión integró además las herramientas y marcos de trabajo WebObjects de Apple para construir aplicaciones y servicios web de Java, que anteriormente se vendían como un producto separado por un precio de 699$.

Con el lanzamiento de Mac OS X v10.5 también lo fue el Xcode 3.0, que tenía como principales novedades la inclusión de Objetive-C 2.0, un nuevo Interface Builder, la opción de re factorizar proyectos y hacer "snapshosts" del proyecto entre otras.

Xcode 4, lanzado a principios de 2011, incluía como novedades una nueva interfaz y la compatibilidad con Mac OS X 10.7 Lion. Con esta versión, Xcode dejó de ser compatible con Mac OS X 10.5 Leopard.

### Objetive-C

Objective-C es un lenguaje de programación orientado a objetos creado como un súper conjunto de C para que implementase un modelo de objetos parecido al de Smalltalk. Originalmente fue creado por Brad Cox y la corporación StepStone en 1980. En 1988 fue adoptado como lenguaje de programación de NEXTSTEP y en 1992 fue liberado bajo licencia GPL para el compilador GCC. Actualmente se usa como un lenguaje principal de programación para Mac OS X, iOS y GNU step, además de Swift. (Wikipedia, s.f.)

### Swift

Swift es un lenguaje de programación de propósito general, multiparadigma, compilado desarrollado por Apple Inc. para iOS, macOS, watchOS, tvOS y Linux. SWIFT está diseñado para trabajar con Apple Cocoa y Cocoa Touch Marcos y el gran cuerpo de código Objective-C (ObjC) para productos de Apple. Swift pretende ser más concisa y más resilientes al código erróneo ("más seguro") que Objective-C. Está construido con el framework de compilador LLVM incluido en Xcode 6 y versiones posteriores y en plataformas que no sean Linux, utiliza la biblioteca de tiempo de ejecución de Objective-C, que permite C, Objective C, C++ y Código Swift para ejecutar dentro de un programa.

Swift respalda los conceptos de base que Objective-C despacho notablemente dinámico, flexible, extenso enlace, programación extensible y características similares. Estas características también tienen rendimiento conocido y seguridad ventajas y desventajas, que Swift fue diseñado para la dirección. Para seguridad, Swift presentó un sistema que ayuda a errores de programación comunes de dirección como punteros nulos y azúcar sintáctica introducida para evitar que la pirámide de la fatalidad que puede resultar. Para problemas de rendimiento, Apple ha invertido un esfuerzo considerable en optimización agresiva que puede aplanar las llamadas a métodos y descriptores de acceso para eliminar esta sobrecarga. Más fundamentalmente, Swift ha añadido el concepto de extensibilidad del Protocolo, un sistema de extensibilidad que se puede aplicar a tipos, estructuras y clases. Apple promueve esto como un verdadero cambio en los paradigmas que término "Protocolo orientado a programación" de programación.

SWIFT fue introducido en Apple 2014 en todo el mundo los desarrolladores Conferencia (WWDC). sufrió una actualización a la versión 1.2 durante 2014 y una actualización más importante a Swift 2 en 2015 WWDC. Inicialmente un lenguaje propietario, versión 2.2 era software de código abierto y bajo licencia de Apache 2.0 en 03 de diciembre de 2015, para plataformas de Apple y Linux. IBM anuncio su sitio web de Swift Sandbox, que permite a los desarrolladores escribir código Swift en una salida de panel y pantalla en otro.

Una segunda implementación libre de Swift que cacao, común idioma infraestructura (. net del Microsoft) y el Java y Android la plataforma de los objetivos existe como parte del compilador elementos de Rem Objects Software puesto que el lenguaje es código abierto, existen perspectivas de que ser portado a la web algunos marcos de web ya se han desarrollado, como el de IBM Kitura, perfecto y Vapor. Un grupo de trabajo oficial "APIs de servidor" también se ha iniciado por Apple, con miembros de la comunidad de desarrolladores Swift desempeña un papel central. (Developer Apple, 2014)

## Apache Flex

FlexJS ™ es un nuevo marco de desarrollo de aplicaciones que cruzar compila MXML y ActionScript en HTML y JavaScript. Se ejecuta en la web, de escritorio y móvil (app) en formato SWF y se ejecuta en entornos basados ​​en navegador como archivos / JS / CSS HTML sin necesidad de que el plugin de Flash Player. Reúne las ventajas de Flex al mundo JavaScript. (Apache Flex, 05)

### XML

XML significa Extensible Markup Language. XML fue diseñado para almacenar y transportar datos. XML fue diseñado para ser a la vez humano-y-legible por máquina.

MXML es un lenguaje descriptivo desarrollado inicialmente por Macromedia hasta el 2005 para la plataforma FLEX de Adobe. Lenguaje que describe interfaces de usuario, crea modelos de datos y tiene acceso a los recursos del servidor, del tipo RIA (Rich Internet Application).

### Adobe ActionScript

(fecha de lanzamiento en 1997) es el lenguaje de programación de la plataforma Adobe Flash. Originalmente desarrollado como una forma para que los desarrolladores programen de forma más interactiva.

ActionScript es un lenguaje de programación orientado a objetos desarrollado originalmente por Macromedia Inc. (ya fusionado en Adobe Systems). Es una derivación de HyperTalk, el lenguaje de script para HyperCard. Ahora es un dialecto de ECMAScript (que significa que es un súper conjunto de la sintaxis y la semántica del lenguaje más ampliamente conocidos como JavaScript), a pesar de que inicialmente surgió como un hermano, ambos siendo influenciado por HyperTalk.

ActionScript se utiliza principalmente para el desarrollo de sitios web y software destinados a la plataforma de Adobe Flash Player, que se utiliza en las páginas Web en forma de archivos SWF incrustados.

ActionScript 3 también se utiliza con el sistema de Adobe AIR para el desarrollo de aplicaciones de escritorio y móviles. El lenguaje mismo es de código abierto en el que su especificación se ofrece de forma gratuita y un compilador de código abierto (como parte de Adobe Flex) y la máquina virtual de código abierto (Mozilla Tamarin) están disponibles.

ActionScript también se utiliza con Scaleform GFX para el desarrollo de interfaces de usuario de videojuegos en 3D y HUD.

# CAPITULO III: PROCEDIMIENTO METODOLOGICO.

## Enfoque de la investigación.

De acuerdo a los objetivos el proyecto corresponde a una investigación de tipo descriptiva, planteando un marco metodológico y una arquitectura tecnológica para el desarrollo del prototipo software móvil. Siendo necesario la recolección de la información y seleccionar el área de las ciencias en las que se aplicaría el marco metodológico y documentar la información proporcionada es fundamental para implementar el prototipo de software propuesto.

El desarrollo de aplicaciones para dispositivos móviles, desde el punto de vista de la Ingeniería del Software, se considera que no debe cambiar sustancialmente de las etapas o fases que se tienen en cuenta cuando se desarrolla aplicaciones para computadoras, es claro que hay que tener en cuenta una variedad de aspectos tecnológicos que caracterizan a estos dispositivos y que influyen gradualmente en las etapas del diseño e implementación, como son la conectividad (aplicaciones en línea, fuera línea o parcialmente conectadas), energía, almacenamiento, administración de recursos y periféricos, seguridad, adaptación al tamaño de la pantalla, entrada de datos, entre otros, según Yorio (s.f, 10)

## Proceso desarrollo

Para el proceso de desarrollo se recomienda seguir un proceso iterativo e incremental, el cual es un enfoque donde el ciclo de vida está compuesto por una secuencia de iteraciones. Donde cada iteración es un pequeño proyecto, compuesto por actividades tales como análisis, diseño, implementación y pruebas (tanto en diversos dispositivos como emuladores), dando como resultado un producto completo, señala Pressman (2005, 67).

## Etapa de Análisis.

Se basa en la descripción de los requisitos de los datos, las funciones y el comportamiento requerido. El analista, deberá recolectar la información necesaria, utilizando técnicas de recolección de datos como entrevistas, encuestas, observación, entre otras. En esta etapa se logra claridad sobre lo que desea el usuario y la forma en la cual se le va a presentar la solución que está buscando, obteniendo los requisitos de un producto de software. Las actividades que se realizan en esta etapa son:

### Definición de requisitos.

requisitos funcionales de la aplicación para representarlos como casos de uso. Da lugar a un diagrama de casos de uso. Los casos de uso capturan los requisitos del sistema.

### Etapa de Diseño.

El Diseño proporciona detalles acerca de la estructura de los datos, las arquitecturas, las interfaces y los componentes, con suficientes detalles como para permitir su interpretación y realización física. Las actividades que se realizan en esta etapa son:

- Diseño conceptual de datos. Su objetivo es construir un modelo conceptual del dominio de la aplicación considerando los requisitos

- Diseño arquitectónico. Básicamente, explica Pressman (2005, 277), en el diseño arquitectónico se busca la elaboración de la arquitectura de la aplicación, donde se muestra la estructura general del sistema y las formas en que la estructura proporciona una integridad conceptual para un sistema. Se recomienda utilizar patrones de diseño en la solución, los cuales han sido creados para problemas conocidos.

- Diseño interfaz y navegación. Su objetivo es que las aplicaciones sean más atractivas y la interacción con el usuario amigable y fácil de usar.

- Etapa de Implementación. Esta etapa dependerá de la elección del lenguaje, plataforma y herramientas de desarrollo (emuladores, entornos integrados, etc.). El objetivo es la implementación de acuerdo al diseño y la tecnología seleccionada.

- Etapa de Pruebas. Las pruebas se pueden hacer sobre emuladores y sobre los diferentes dispositivos. Dado que se perdería mucho tiempo al tener que instalar el programa en un dispositivo móvil, cuando es local, cada vez que se requiera probarlo, es importante contar con emuladores de los dispositivos y de los sistemas operativo que se van a implementar.

El presente proyecto permitió la investigación del desarrollo de aplicaciones para el iOS por medio de las herramientas necesarias para su desarrollo además de técnicas y herramientas necesarias para el desarrollo de aplicaciones para dispositivos móviles. Se presentaron los resultados mediante el desarrollo de prototipo de una lista en forma de agenda para la materia de Sistemas Operativos 2. Se logró además fomentar la investigación en cuanto a estas plataformas. Se pudo constatar las ventajas del uso del lenguaje ActionScript para el desarrollo y diseño de aplicaciones móviles, y del framework open base movil. En trabajos futuros, además de continuar con el desarrollo de la aplicación.

# CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES.

## Conclusiones.

* El proyecto de investigación permite obtener el estudio del contexto de desarrollo de aplicaciones móviles en forma general, dejando las bases para el desarrollo de proyectos de este tipo y el diseño de un prototipo utilizando el IDE de Flash Builder, y los conocimientos de XML para el desarrollo de aplicación de investigación.
* Se implementó otro IDE aparte del Xcode porque para su desarrollo se necesitan una Mac física o virtual para la cual se necesitan recursos mayores a los que posee mi equipo personal.
* Además de existir Xamarin como sustito para Microsoft del desarrollo, no se implementó por ese medio, ya que para su ejecución se necesita una IP virtual de una Mac física para su desarrollo.
* El Flash Builder es una IDE de pago por lo que se encuentra en forma de Demo y está limitada. Por esa razón no se pudo emplear el uso de Base de Datos en el Sistema. Además de no contar con un grupo de desarrolladores para ese lenguaje.

## Recomendaciones.

* Se recomienda el uso de aplicaciones Multiplataforma para el desarrollo de aplicaciones móviles. Como lo es el Flash Developer.
* Al ser aplicaciones multiplataforma simplifica la creación de aplicaciones para los distintos dispositivos que existen actualmente en el mercado, sin necesidad de tener que desarrollar en otros lenguajes y centrarse en solo uno.

# Referencias

Apache Flex. (2016 de 11 de 05). *Apache Flex*. Obtenido de Apache Flex: http://flex.apache.org/

Developer Apple. (9 de 09 de 2014). *Developer Apple*. Obtenido de Developer Apple: https://developer.apple.com/swift/blog/?id=14

J. M. (01 de 10 de 2013). *El Pais*. Obtenido de El Pais: http://tecnologia.elpais.com/tecnologia/2013/10/01/actualidad/1380620296\_338056.html

Jaokar, A., & Fish, A. (2006). *Mobile Web 2.0: The Innovator's Guide to Developing and Marketing Next Generation Wireless/Mobile Applications.* Oxford: Futuretext Publications.

microbizz. (1 de 10 de 2011). *microbizz*. Obtenido de microbizz: http://www.microbizz.nl/gpcxcode.html

Pascal Central. (s.f.). *Pascal-Central*. Obtenido de Pascal-Central: http://pascal-central.com/fp-xcode/

Perl CamelBones. (s.f.). *camelbones*. Obtenido de camelbones: http://camelbones.sourceforge.net/index.html

Ros, I. (23 de 05 de 2016). *muy computer*. Obtenido de muy computer: http://www.muycomputer.com/2016/05/23/windows-phone-hundido-mercado/

Wikipedia. (s.f.). *Wikipedia*. Obtenido de Wikipedia: https://es.wikipedia.org/wiki/Objective-C