

DOI: https://doi.org/10.37811/cl_rcm.v6i4.2959

Desarrollo híbrido con flutter

Mg. Quisaguano Collaguazo Luis Rene

Luis.quisaguano1@utc.edu.ec

Mg. Pallasco Venegas Mirian Susana

segundo.corrales@utc.edu.ec

<https://orcid.org/0000-0002-2351-7278>

Andaluz Guerrero Alex Agustín

alex.andaluz1645@utc.edu.ec

<https://orcid.org/0000-0003-1688-9652>

AMg. Martines Freire Maira Natalia

maira.martinez@utc.edu.ec

<https://orcid.org/0000-0003-1345-0898>

Mg. Corrales Beltrán Segundo Humberto

mirian.pallasco@utc.edu.ec

<https://orcid.org/0000-0002-3964-8723>

Facultad de Ciencias de la Ingeniería y Aplicadas

Carrera de Sistemas de Información

Universidad Técnica de Cotopaxi

Latacunga-Cotopaxi-Ecuador

RESUMEN

El presente artículo contiene información relevante y resumida relacionada con el “Desarrollo Híbrido con Flutter” el mismo basado en una investigación la cual permite recolectar de distintas fuentes bibliográficas, los principales aspectos para la aplicación y ejecución de un proyecto informático basado en Flutter, dentro del documento se muestra el análisis de varios textos, con la finalidad de presentar al lector un panorama estructurado sobre los procesos a realizar para un correcto desarrollo e implementación de aplicaciones híbridas, esta investigación parte de la necesidad de dar a conocer este nuevo Framework de desarrollo el cual permita a nuevos programadores diseñar y codificar aplicativos móviles que permitan complementar el gran entorno evolutivo de las aplicaciones móviles en la actualidad.

Palabras claves: entorno; codificación; framework; implementación; híbridas.

Correspondencia: Luis.quisaguano1@utc.edu.ec

Artículo recibido: 15 julio 2022. Aceptado para publicación: 20 agosto 2022.

Conflictos de Interés: Ninguna que declarar.

Todo el contenido de **Ciencia Latina Revista Científica Multidisciplinar**, publicados en este sitio están disponibles bajo

Licencia [Creative Commons](https://creativecommons.org/licenses/by/4.0/) 

Como citar: Quisaguano Collaguazo, M. L. R., Pallasco Venegas, M. S., Andaluz Guerrero, A. A., Martínez Freire, M. N., & Corrales Beltrán, M. S. H. (2022). DESARROLLO HÍBRIDO CON FLUTTER. *Ciencia Latina Revista Científica Multidisciplinar*, 6(4), 4594-4609.

https://doi.org/10.37811/cl_rcm.v6i4.2959

HYBRID DEVELOPMENT WITH FLUTTER

ABSTRACT

This article contains relevant and summarized information related to the "Hybrid Development with Flutter" the same based on an investigation which allows collecting from different bibliographic sources, the main aspects for the application and execution of a computer project based on Flutter, within the The document shows the analysis of several texts, in order to present the reader with a structured overview of the processes to be carried out for a correct development and implementation of hybrid applications, this research is based on the need to publicize this new Development Framework. which allows new programmers to design and code mobile applications that complement the great evolutionary environment of mobile applications today.

Keywords: environment; coding; framework; implementation; hybrid.

INTRODUCCIÓN

Un buen desarrollador de software trabaja con disciplina y constancia desde el primer día.

En la actualidad el avance en el uso de los dispositivos móviles ha incrementado en gran medida que cada vez se exige a los programadores un más tiempo al momento de desarrollar aplicaciones con el fin de abarcar el mayor porcentaje del mercado, es por ello que se emplean mejores y más avanzadas técnicas siendo el caso del uso de frameworks que permitan el desarrollo de aplicaciones híbridas, siendo de esta manera que se ayuda a los desarrolladores a reducir el tiempo de codificación obteniendo mejores sistemas informáticos que permitan el uso del aplicativo en distintos sistemas operativos.

Siendo así que se propone el uso del framework más actual “Flutter” mismo que es desarrollado por la empresa Google para la codificación de nuevas aplicaciones para dispositivos Android creando ventajas al momento de la creación de este, reduciendo los costos antes, durante y después.

Se conoce que los frameworks más utilizados hoy en día son para desarrollo web sin embargo durante los últimos años, en especial desde el 2010 han venido evolucionando, siendo más usados con el paso del tiempo, entre ellos están React, el cual fue desarrollado por Facebook y Angular de la mano de Google. Así como este existe otros más recientes como lo es Flutter de quien trataremos dentro de este artículo, el mismo fue creado por Google y cuenta con una gran diferencia ya que este permite compilar casi en un 100% todo código en aplicaciones nativas, gracias a que fue creado desde cero.

Un framework es también conocido como un cuerpo base, con parámetros que el desarrollador debe seguir, como la configuración de archivos, elementos creados por el programador y herramientas internas que puedan ser reutilizadas después dependiendo de la necesidad existente.

Según Enríquez & Rodríguez (2014) , “ Una de las principales dificultades al desarrollar aplicaciones para móviles es la gran cantidad de dispositivos y sistemas operativos existentes, teniendo que construir una versión diferente para cada caso en un lenguaje y una herramienta diferente ” Dado a la gran evolución tecnológica de los últimos años con respecto a los dispositivos móviles, esto ha ocasionado que se requieran de manera urgente herramientas que permitan el desarrollo de aplicaciones para acoplarse a estas nuevas tecnologías.

METODOLOGÍA

Para la realización de este documento se partió desde un enfoque cualitativo mediante una previa investigación, siendo que esta metodología permite analizar las cualidades más importantes del tema a tratar, con el fin de desarrollar una propuesta que permita al lector obtener una comprensión e interpretación más amplia de los conceptos que rodean un proceso de desarrollo basado en Aplicaciones Híbridas con Flutter.

Al igual se utilizó una estrategia documental, la cual permitió realizar un sondeo bibliográfico de carácter intelectual en relación con el tema, mediante la revisión literaria de revistas, tesis y páginas de internet, analizando el contenido de cada uno para orientar de mejor manera la estructura del documento y plasmando así un contenido práctico y con sentido literario.

Según (Mora, 2018) “la investigación bibliográfica consiste en la búsqueda, recopilación, organización, valoración, crítica e información de datos bibliográficos”. Esto implica que una investigación bibliográfica es parte principal dentro de la investigación del tema, ya que permite, elaborar ideas respecto al objeto de estudio, ya sea en un ámbito histórico, o metodológico.

RESULTADOS

¿Qué es Flutter?

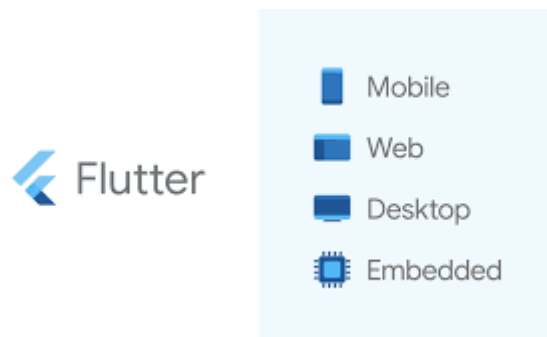


Ilustración 1 FLUTTER

Flutter es una tecnología híbrida diferente a lo que hay hoy en día, desarrollada por Google y basada en el lenguaje Dart, permite crear aplicaciones nativas para Android y iOS en tiempo récord con una interfaz muy amigable basada en Material Design, con la que puedes hacer cambios en el código y ver el resultado en el emulador en tiempo real, mientras la app sigue en marcha.

Existen grandes aplicaciones con Flutter hoy en día, como por ejemplo Alibaba, Hamilton Musical, Abbey Road Studios, Google ADS, Reflectly, entre otras. Estas empresas han utilizado Flutter para ofrecer aplicaciones nativas con un buen UX y UI asociado a cada ecosistema

Lo impresionante de Flutter es que no utiliza los widgets nativos de Android y iOS, sino que utiliza su propio código para crear widgets. Esto hace que todo el código relacionado con estos widgets se integre dentro de la app, y por ello las apps tienen un tamaño considerablemente más grande. Como resumen, Flutter permite añadir fácilmente nuevos widgets personalizados, además de incorporar un conjunto muy amplio de widgets para Android, iOS y Material Design. (Daniel, 2019)

Funcionalidades de Flutter

- **Calidad nativa:** Las aplicaciones nativas se desarrollan específicamente para un sistema operativo, Flutter utiliza todas las ventajas de las aplicaciones nativas para conseguir calidad en el resultado final.
- **Experiencia de usuario:** Flutter incluye Material Design de Google y Cupertino de Apple, con lo que la experiencia de usuario es óptima y los interfaces de usuario idénticos a los de las aplicaciones desarrolladas por las propias compañías.
- **Tiempo de carga:** Una de las principales causas de abandono de una aplicación es el tiempo que tarda en cargar, con Flutter se experimentan tiempos de carga por debajo de un segundo en cualquiera de los soportes iOS o Android.
- **Desarrollo ágil y rápido:** Gracias a la característica hot-reload, puedes programar y ver los cambios en tiempo real en tu dispositivo o en los simuladores. (Devs, 2019)

Ventajas y Desventajas del uso de Flutter

Las principales ventajas son:

- Mayor rendimiento posible del dispositivo.
- Independencia de la plataforma, flutter se encarga de compilar a bajo nivel.
- Independiente de la versión de iOS o Android.
- Abstracción a la hora de implementar características comunes en ambas plataformas.

Las Desventajas para destacar son:

- Mayor peso de la aplicación puesto que el framework va compilado en la misma.
- Gran dificultad para comunicarse con APIs nativas que no tengan plugin.
- Dificultad para integrarse con librerías de terceros nativas de cada plataforma. (Desing Soft, 2020)

Entorno de desarrollo

Actualmente flutter ofrece plugins para Android Studio y Visual Studio code. Estos plugins facilitan la escritura de código, la ejecución y depuración de este en ambas plataformas.

Una de las funcionalidades a destacar en tiempo de desarrollo es el Hot-reload que permite en la mayoría de los casos refrescar la vista en la que nos encontremos con los últimos cambios de código manteniendo el estado.

Puesto que hay mucho que contar y este framework puede convertirse en uno de los referentes en desarrollos híbridos, dado que también utiliza aspectos técnicos como la composición de las vistas (Widgets).

Flutter actualiza dicha interfaz de usuario a 60 FPS y usa la GPU para la mayor parte del trabajo. Si bien esto no afecta la forma en que se crea la aplicación, es la razón por la que la interfaz de usuario de Flutter es muy fluida. El código de la aplicación basado en Dart continuará ejecutándose a través de la CPU y en el subproceso de interfaz de usuario especializado

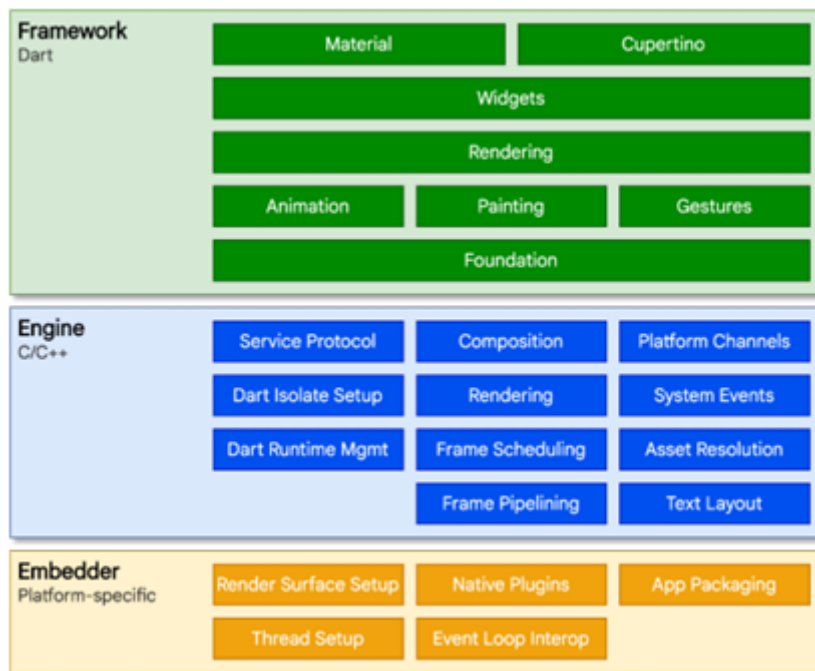


Ilustración 2 ARQUITECTURA DE FLUTTER

Instalación de Flutter

Herramientas de instalación de Flutter en Windows:

- Sistema operativo: Windows 7 requiere de las siguientes especificaciones y SP1 o siguientes (64 bits)
- Espacio de almacenamiento: 1.32 GB
- Herramientas: Windows Powershell 5.0 y Git, Primeramente, se descarga el Framework para Windows Flutter SDK del sitio oficial: versión 2.x. <https://flutter.dev/docs/getstarted/install/windows>.
- Instalación de Android Studio

Este es un archivo .zip que se extrae en la carpeta deseada.

Seguido de esto es posible ejecutar comandos de Flutter desde el cmd estando en la ubicación del SDK, o es posible configurar Flutter en acceder a sus comandos desde cualquier ubicación.

Para configurar Flutter, las variables de entorno para en las variables de entorno, es necesario editar las variables de entorno del sistema, agregar una nueva variable, y establecer la ruta de flutter como el valor de esta. \ bin Flutter tiene una serie de requerimientos para su correcto funcionamiento, estos requerimientos se pueden visualizar utilizando el comando flutter doctor terminal de Windows. (Desing Soft, 2020)

Librerías externas usadas por Flutter

Por defecto Flutter trae muchas librerías que el desarrollador puede usar, pero en algunos casos es necesario instalar algunas externas para lograr ciertas funcionalidades. A continuación, se describen este tipo de librerías. (Nicolas, 2019)

```
geolocator: ^5.1.4+1
geocoder: ^0.2.1
image_picker: ^0.6.0+9
firebase_core:
firebase_messaging:
flutter_datetime_picker: ^1.2.8
```

Ilustración 3 LIBRERIAS EXTERNAS DE FLUTTER

Comparativa de frameworks

A continuación, se muestra una tabla con una comparativa de las principales características de cada uno de los frameworks existentes. (Lizarraga, 2020)

	Flutter	React Native	Ionic
Lenguaje de programación	Dart	Javascript	Javascript
Reusabilidad	Alta	Media	Alta
Librerías	Media, en aumento	Alta	Alta
Popularidad	Alta	Alta	Media
Desempeño	Excelente	Bueno	Medio
Costo	Sin costo	Sin costo	Sin costo, paquete pro de paga
Tiempo de desarrollo	Medio	Alto	Bajo

Tabla 1: Comparativa de frameworks 2.6 Controlador de versiones

¿Qué es Dart?



Ilustración 4 DART

Dart es un lenguaje de programación open source, relativamente nuevo, que fue desarrollado por Google y que lanzó su primera versión en 2011. Este lenguaje se creó con el objetivo de permitir a los desarrolladores utilizar un lenguaje orientado a objetos y con análisis estático de tipo. (Moraga, 2019)

La programación Dart es una alternativa a reemplazar JavaScript y convertirse en el lenguaje prioritario para los navegadores actuales, aunque este lenguaje todavía está en proceso de mejoras y adaptaciones.

En mayo de 2017, Google lanzó la primera versión alfa pública del lenguaje Dart Flutter, un marco de desarrollo de interfaz de usuario multiplataforma, que utiliza Dart para crear aplicaciones móviles tanto para iOS como para Android. Con el tiempo, Flutter empezó a ser muy popular.

Características de Dart

Características del lenguaje Dart:

- Programación estructurada y flexible. Los ingenieros de Google diseñaron Dart de manera que pueda ser utilizado en proyectos de una sola persona hasta proyectos más desarrollados o complejos.
- Lenguaje familiar y fácil de aprender. Es un lenguaje realmente sencillo y fácil de aprender. En su sitio web se pueden encontrar varios tutoriales, y también permite colaboraciones de otros desarrolladores.
- Permite la adaptación de nueva herramienta a cualquier navegador web. El lenguaje de programación Dart se puede ejecutar de dos maneras; en una máquina virtual (MV), o en un motor de Javascript utilizando un compilador para traducir el código. Esto le permite adaptarse a cualquier navegador.
- Lenguaje basado en clases e interfaces o POO. Gracias a sus basamentos en clases o en la programación orientada a objetos, se facilita la encapsulación y la reutilización del código. (Fernando, 2020)

Diferencias entre Dart y otros lenguajes de programación

La principal diferencia que presenta Google Dart frente a otros lenguajes de programación es que es más fácil de leer ya que la sintaxis de este se acerca al lenguaje humano, tiene más posibilidades y menos comandos.

Es posible utilizar caracteres en blanco, pestañas y saltos de línea a discreción, lo que permite a los programadores de estructurar bien el código, de manera que sea últimamente ignorado por el compilador. Sin embargo, hay unas cuantas excepciones, como las palabras clave y los nombres de variables y funciones, es decir, todas las definiciones establecidas dentro del lenguaje Dart. Entre ellas se encuentran, por ejemplo, "if", "else", "string" o "void".

Ventajas de Dart:

- Es de acceso gratuito para cualquier persona.
- Detrás de su programación se encuentra Google, lo que ofrece perspectivas a largo plazo para el desarrollo del lenguaje.
- Dart es fácil de aprender debido a que los desarrolladores han simplificado características complicadas de otros lenguajes.
- Funciona en todos los navegadores móviles y de escritorio actuales.

¿Dónde utilizar Dart?

Dart es un lenguaje de propósito general, y se lo puede utilizar en casi cualquier cosa:

- En aplicaciones web, utilizando la librería de arte: html y el transpilador para transformar el código en Dart en JavaScript, o utilizando frameworks como AngularDart.
- En servidores, utilizando las librerías de arte: http y arte: io. También hay varios frameworks que se pueden utilizar, como por ejemplo Aqueduct.
- En aplicaciones de consola.
- En aplicaciones móviles gracias a Flutter.

Arquitectura del framework

Está compuesto por varios módulos que permiten tener a disposición los componentes necesarios para desarrollar una aplicación con funcionalidades normales. Podemos distinguir principalmente: (Moraga, 2019)

- Motor gráfico 2D Skia, próximo motor gráfico de Chrome optimizado al máximo para ofrecer un gran rendimiento en cualquier hardware.
- Core del framework que gestiona los ciclos de vida y permite el desarrollo basado en estados.
- Componentes visuales. Divididos en dos grandes módulos
- Material contiene los componentes tradicionales del diseño de Google y proporciona una experiencia de uso similar a la que encontramos en Android.

- Cupertino los elementos imitando la experiencia que encontramos en iOS. En la última reléase ha sido potenciado ofreciendo una experiencia personalizada completa dentro de iOS.
- Componentes que nos facilitan el uso de temas y adaptación a diferentes plataformas-pantallas. Nos dividen las plataformas en:
- Activas: Móvil, Tablet... plataformas con una gran interacción por parte del usuario
- Pasivas: TV, Embebidos... plataformas con poca interacción.

¿Qué son las Aplicaciones Móviles?



Ilustración 5 Apps NATIVAS / HIBRIDAS

Una aplicación móvil diseñada para ser ejecutada en. teléfonos inteligentes otros, tabletas y dispositivos móviles. Este tipo de aplicaciones permiten al usuario efectuar un variado conjunto de tareas, facilitando las gestiones o actividades a desarrollar.

Por lo general, se encuentran disponibles a través de ciertas plataformas de distribución, o por intermedio de las compañías propietarias de los móviles tales como Android, iOS, BlackBerry OS, sistemas operativos Windows Phone. (Gsoft, 2020)

¿Qué son las Aplicaciones Nativas?



Ilustración 6 Apps NATIVAS

Las aplicaciones, entre otros. nativas son aquellas aplicaciones que están desarrolladas para un equipo o plataforma determinada. Es decir, funciona en el equipo sin necesidad de ningún programa externo ya que se ha desarrollado en el lenguaje de programación específico de cada equipo.

El término de App Nativa está habitualmente asociado a los dispositivos móviles y por tanto hay Apps Nativas para cada sistema operativo como iOS o Android. Este tipo de

aplicaciones se programan con el código nativo de la plataforma en cuestión utilizando las herramientas del kit de desarrollo proporcionado por el equipo responsable del sistema operativo.

La instalación de estas aplicaciones se realiza directamente en el dispositivo, usualmente a través de una tienda de aplicaciones como lo es la Play Store de Google. La principal ventaja de las Apps Nativas es que se adapta dispositivo, hecho que le da un buen rendimiento, problemas todas las funcionalidades del y puede dispositivo en su totalidad al utilizar sin mayores (Rendimiento gráfico, GPS, cámara, acelerómetro). (Antonio, 2020)

¿Qué son las Aplicaciones Híbridas?



Ilustración 7Apps Híbridas

Las aplicaciones híbridas aprovechan al máximo la versatilidad de un desarrollo web y tienen la capacidad de adaptación al dispositivo como una app nativa. Actualmente, existen múltiples herramientas/frameworks que permiten el desarrollo de aplicaciones híbridas, dentro de los cuales se encuentra React Native que utiliza JavaScript y Flutter que utiliza Dart.

Este tipo de aplicaciones han ido recibiendo una mayor adopción por parte de los desarrolladores, hecho que ha conllevado en una mejora en el proceso de desarrollo y desempeño de esta.

A diferencia de las aplicaciones nativas, este tipo de aplicaciones no se realiza en el lenguaje nativo del sistema, sino que lo que suele suceder es que se convierte el código fuente en código nativo o se emula la vista, de tal manera que funciona correctamente sin perjudicar el desempeño de esta. (Antonio, 2020)

¿Qué es Android?



Ilustración 8 ESTRUCTURA DE ANDROID

Android es un popular sistema operativo para teléfonos móviles basado en Linux desarrollado por Google y presentado por primera vez en el año 2007 operativo Android se. El sistema encuentra diseñado para dispositivos con pantalla táctil, como teléfonos inteligentes, tabletas, relojes inteligentes, televisores, entre otros.

Android es un proyecto de código abierto ampliamente adoptado. Google desarrolla activamente la plataforma de Android, pero ofrece una parte de ella de forma gratuita a los fabricantes de hardware y operadores de telefonía que desean utilizar Android en sus dispositivos.

El sistema permite el desarrollo de aplicaciones móviles con los lenguajes Java y Kotlin, así como también ofrece una serie de herramientas para facilitar el desarrollo en su kit de desarrollo de software (SDK) tales como debuggers, librerías, emuladores y documentación.

Así mismo, el sistema operativo facilita la interacción con el hardware del dispositivo móvil para poder acceder a los componentes de este como lo sería la cámara, llamadas telefónicas o manejo del GPS. (Antonio, 2020)

¿Qué es IOS?

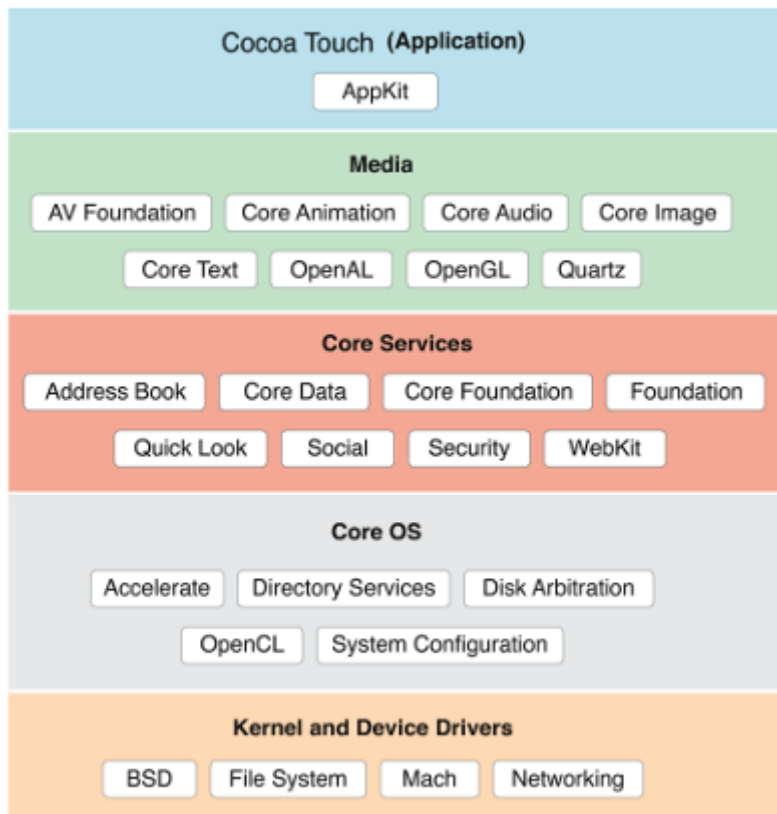


Ilustración 9 ESTRUCTURA DE IOS

IOS es un sistema operativo móvil de Apple lanzado oficialmente el 29 de junio de 2007 derivado de macOS, que a su vez se deriva de Darwin BSD, siendo por lo tanto un sistema operativo tipo Unix.

IOS cuenta con cuatro capas de abstracción: la capa del núcleo del sistema operativo, la capa de "Servicios Principales", la capa de "Medios" y la capa de "Cocoa Touch".

A diferencia de Android, este sistema operativo se encuentra únicamente en los equipos desarrollados por la empresa como lo son sus iPhone, iPods, iPads o Apple Watch. Así mismo, el kit de desarrollo de este sistema operativo únicamente es accesible desde los equipos del fabricante, hecho que potencialmente aumenta los costos de desarrollo al requerir de hardware específico. (Antonio, 2020)

DISCUSIÓN

En visión del tema analizado se observa que existen varios conceptos y puntos de vista relacionados con la correcta implementación y funcionamiento de los Frameworks actuales, siendo que la mayor parte de conceptos coinciden en que Flutter al ser relativamente más nuevo permite a las empresas llevar un mejor control costo-beneficio al momento de desarrollar una aplicación híbrida, así como también consideran que los programadores podrían reducir en gran cantidad el tiempo de codificación por cada aplicación que de pretenda crear.

Este documento de investigación permite observar el ciclo de vida de un proyecto y sus determinadas fases, como son el análisis, diseño, codificación y pruebas pertinentes, con el fin de obtener un software híbrido de calidad para aplicaciones móviles utilizando una API REST.

CONCLUSIÓN

En mi opinión personal la investigación de este tema me a parecido muy interesante permitiéndome conocer más conceptos sobre el funcionamiento de un framework así como también de cómo trabaja la parte del front-end y el back-end, así como también como se lleva el desarrollo de aplicaciones móviles compiladas de forma híbrida así como también los componentes necesarios para el diseño de la interfaz gráfica que utilizara el usuario tanto de IOS como de Android

Teniendo en cuenta lo anterior puedo decir que toda la investigación resulto muy fructífera al momento de conocer más sobre el Framework Flutter, el cual considero que en cuestión de aprendizaje es el más fácil y el que mejor desempeño brinda, dado que al trabajar con widgets cuenta con un amplio catálogo, de igual manera cuenta con una librería externa muy amplia gracias a la cual no es necesario recurrir a librerías hechas por terceros.

REFERENCIAS BIBLIOGRÁFICAS

Acervolima. (15). *GeeksforGeeks*. Obtenido de FLUTTER - APLICACIÓN DE GENERACIÓN DE PDF SIMPLE: <https://www.geeksforgeeks.org/flutter-simple-pdf-generating-app/>

Alferez, J. (2018). *Universidad Politecnica de Madrid*. Obtenido de estudio comparativo deframeworksmultiplataforma para desarrollo de aplicaciones móviles: <https://1library.co/document/z31oe0my-estudio-comparativo-frameworks-multiplataforma-desarrollo-aplicaciones-moviles.html>

Android. (2020). *Developers*. Obtenido de Arquitectura de la plataforma android: <https://developer.android.com/guide/platform>

Antonio, L. (09 de 12 de 2020). *Universidad Politecnica Sinaloa*. Obtenido de <http://repositorio.upsin.edu.mx/formatos/A031LIZARRAGAOSUNAKEVINANTONI06608.pdf>

Daniel, E. (2019). *Platzi*. Obtenido de <https://platzi.com/blog/flutter-multiplataforma/#:~:text=Flutter%20es%20una%20tecnolog%C3%ADa%20h%C3%ADbrida,amigable%20basada%20en%20Material%20Design>.

Desing Soft. (19 de 02 de 2020). *Desing Soft*. Obtenido de <https://designsoftcr.com/desarrollo-movil-hibrido-con-flutter/>

- Devs, Q. (05 de 07 de 2019). *Quality Devs*. Obtenido de Qué es Flutter y por qué utilizarlo en la creación de tus apps:
<https://www.qualitydevs.com/2019/07/05/que-es-flutter/>
- Fernando, H. (2020). *Lenguajes de programacion.net*. Obtenido de
<https://lenguajesdeprogramacion.net/dart/>
- Flutter. (18 de 11 de 2020). *Flutter*. Obtenido de Flutter architectural overview:
<https://flutter.dev/docs/resources/architectural-overview>
- Gabit. (2020). *Gabit*. Obtenido de Que es los: <http://www.gabit.org/gabit/?q=es/que-es-ios>
- Google. (2019). *Android*. Obtenido de Que es Android: <https://www.android.com/what-is-android/>
- Gsoft. (12 de 11 de 2020). *Gsoft*. Obtenido de
<https://www.gsoft.es/articulos/quenesesito-web-apps-app-nativa-o-app-hibrida/>
- Jaimes, E. (2019). *Scribd*. Obtenido de Flutter_tutorial en Español:
<https://es.scribd.com/document/564024913/Flutter-tutorial-en-Espanol>
- Lizarraga, K. (09 de 12 de 2020). *Universidad Politecnica de Sinaloa*. Obtenido de “Implementación de Flutter para el desarrollo de aplicaciones móviles nativas en iOS y Android”:
<http://repositorio.upsin.edu.mx/formatos/A031LIZARRAGAOSUNAKEVINANTONI06608.pdf>
- Menacho, A. (2020). *Universidad de Sevilla*. Obtenido de Aplicación móvil con Flutter y FlutterFire: <https://biblus.us.es/bibing/proyectos/abreproy/93024/fichero/TFG-3024+MENACHO+RODR%C3%8DGUEZ%2C+%C3%81LVARO.pdf>
- Mora.L. (2018). *Introduccion a la metodologia de investigacion*. Puyo: informe de investigacion.
- Moraga, J. L. (2019). *Deloitte*. Obtenido de
<https://www2.deloitte.com/es/es/pages/technology/articles/apps-mas-visuales-y-rapidas-con-flutter.html>
- Nicolas, S. (01 de 2019). *Uniandes*. Obtenido de
<https://repositorio.uniandes.edu.co/bitstream/handle/1992/44885/u831158.pdf?sequence=1>

Point, T. (2019). *Tutorial Point*. Obtenido de Flutter:

https://www.tutorialspoint.com/flutter/flutter_tutorial.pdf

Quinn, N. (2015). *How2Geek*. Obtenido de Cómo crear archivos estándar PDF/A en

Flutter: <https://how2geeks.com/es/como-crear-archivos-estandar-pdf-a-en-flutter>

Victor, V. (2019). *Universidad de Almeria*. Obtenido de Desarrollo de aplicaciones móviles multiplataforma con flutter:

http://repositorio.ual.es/bitstream/handle/10835/8010/TFG_VAZQUEZ%20RODRIGUEZ%2C%20VICTOR.pdf?sequence=1&isAllowed=y