



UNIVERSIDAD TÉCNICA DE BABAHOYO

FACULTAD DE ADMINISTRACIÓN, FINANZAS E INFORMÁTICA.

PROCESO DE TITULACIÓN

ABRIL 2024 – AGOSTOS 2024

PREVIO A LA OBTENCIÓN DEL TÍTULO DE:

INGENIERO EN SISTEMAS DE INFORMACIÓN

TEMA:

ANÁLISIS COMPARATIVO DE FRAMEWORKS PARA EL DESARROLLO DE

APLICACIONES MOVIL

ESTUDIANTE:

LEONELA JULISSA MUÑOZ BARRIGA

TUTOR:

JOFFRE VICENTE LEON ACURIO

BABAHOYO- ECUADOR 2024

INDICE

1 Contenido

I. RESUMEN.....	3
II. Abstract.....	4
III. PLANTEAMIENTO DEL PROBLEMA	5
IV. JUSTIFICACION	6
V. OBJETIVOS	7
VI. LINEA DE INVESTIGACION	8
VII. MARCO CONCEPTUAL.....	9
VIII. MARCO METODOLOGICO.....	23
IX. RESULTADOS.....	26
X. DISCUSIÓN DE RESULTADO	29
XI. CONCLUSIONES	31
XII. RECOMENDACIONES	33
XIII. ANEXO.....	35
XIV. Bibliografía.....	38

I. RESUMEN

En este estudio de caso, se llevó a cabo un análisis comparativo de los Frameworks utilizados para el desarrollo de aplicaciones móviles, lo que llevó a la identificación de los Frameworks más utilizados que brindan a los programadores las características más importantes que les permiten realizar tareas específicas, a saber, costo, flexibilidad. Inversión de tiempo y energía. En el curso de la investigación, trate de determinar las preguntas y los objetos de la investigación, luego determine los objetivos y conceptos básicos del método de investigación y utilice la investigación bibliográfica, incluida información real, informes y artículos. Además, también se llevó a cabo un estudio comparativo, que incluyó el desarrollo de características para cada Frameworks seleccionado en forma escrita, es decir. Es decir, su confiabilidad, usabilidad o efectividad es proporcionar a los desarrolladores información que les indique qué tipo de Frameworks es recomendado, válido o más adecuado para el tipo de aplicación móvil que planean crear.

Palabras **Clave:**

- **Frameworks:** React Native, Flutter, Xamarin
- **Aplicaciones móviles**
- **Rendimiento**
- **Facilidad de uso**
- **Costos**
- **Experiencia del usuario**

II. Abstract

In this case study, a comparative analysis of the Frameworks used for mobile application development was conducted, leading to the identification of the most commonly used Frameworks that provide programmers with the key features enabling them to perform specific tasks, namely cost, flexibility, and investment of time and energy. During the research, the questions and objects of the investigation were determined, followed by the establishment of the objectives and basic concepts of the research method. Bibliographic research was employed, including real information, reports, and articles. Additionally, a comparative study was conducted, which involved developing features for each selected framework in written form. Specifically, the study examined their reliability, usability, or effectiveness to provide developers with information indicating which type of framework is recommended, valid, or most suitable for the type of mobile application they plan to create.

Keywords:

Frameworks: React Native, Flutter, Xamarin

Mobile applications

Performance

Ease of use

Costs

User experience

III. PLANTEAMIENTO DEL PROBLEMA

El incremento de aplicativos móviles ha ido evolucionando y teniendo gran aumento durante los últimos tiempos a medida que crece la demanda de soluciones móviles en todas las industrias. Actualmente, los programadores de software presentan grandes desafíos al momento de seleccionar el entorno ajustado para la creación de sus proyectos. Por lo tanto, las tomas de decisiones tendrán un gran impacto en el rendimiento, el costo, la eficiencia y la capacidad mantenibilidad de la aplicativo. Los Frameworks más usados son React Native, Flutter y Xamarin, cada uno de estos con sus ventajas y desventajas.

La inexistencia de información comparativa especificada y detallada imposibilita a los programadores que puedan tomar grandes decisiones acerca de qué frameworks utilizar.

Actualmente, la mayoría de la parte de las decisiones que se toman es apoyarse en experiencias individuales o en información de proyecto dividido, lo que podrían elegir un programa que no es aconsejable para poder mejorar las necesidades del proyecto. Esto puede resultar un mayor costo, largo tiempos de desarrollo, dificultades en el rendimiento y problemas en el mantenimiento de las aplicaciones a largo plazo.

La mayor parte de las pruebas que se presentan en la elaboración de los aplicativos móviles es seleccionar un espacio considerable para los frameworks como React Native, Flutter y Xamarin los mismos que se responsabilizan de minimizar el tiempo y los costos al dar acceso a dar un nuevo uso del código en todos sus proyectos.

Aunque, cada uno de los frameworks poseen sus propias ventajas y desventajas en cuanto al rendimiento es necesario posibilitar la curva de aprendizaje. No obstante, los beneficios potenciales y algunos programadores de software aún tienen problemas para seleccionar la adecuada plataforma que se ajusten a sus necesidades

IV. JUSTIFICACION

Actualmente, la sociedad se encuentra en constantes problemas debido a muchos problemas. De Actualizaciones tecnológicas que suceden año tras año. Este tipo de medios avanzados permite Crea relaciones increíbles que comienzan con una conversación compartida entre personas. Vecinos o incluso contactos al otro lado del mundo. En resumen, nosotros como ciudadanos estamos muy agradecidos por la tecnología tal como es. Es el mejor tipo de trabajo que puedes encontrar, y no solo eso. Apoyar a las grandes empresas en lugar de brindarles las posibles soluciones que necesitan. Permitir que las micro y pequeñas empresas lleguen a sus mercados objetivo de una manera más rentable. Clientes, esto se consigue a través del marketing, que suele utilizarse en las redes sociales. Parte de las tecnologías mencionadas anteriormente son aplicaciones móviles, con Uso diario de particulares para actividad económica, mantenimiento. Manténgase informado e incluso acceda a servicios. Estas aplicaciones deben ser reemplazadas Permitiéndoles satisfacer con mayor precisión los requisitos de cada cliente tal como son. Son muy diferentes y dependen del individuo. Por lo tanto es necesario El mejor mecanismo para desarrollar aplicaciones móviles que abarquen una gama tan amplia La necesidad de que sea aplicaciones móviles se puede solucionar con grandes ventajas Este frameworks está abierto a programadores. No hay duda de que la implementación de este mecanismo puede traer grandes beneficios. La empresa que lo utiliza porque además proporciona un costo razonable o ocurre Agilidad y velocidad en el desarrollo de aplicaciones que permiten escalar Los errores durante el proceso se pueden corregir y mantener. y garantiza una colaboración adecuada con otros desarrolladores.

V. OBJETIVOS

OBJETIVO GENERAL

Analizar comparativamente Frameworks para el desarrollo de aplicaciones móviles con el fin de identificar sus ventajas y desventajas relativas, utilizando una metodología de evaluación basada en criterios de rendimiento, facilidad de uso, costos, y experiencia del usuario.

OBJETIVO ESPECIFICO

Analizar detalladamente las características de los principales Frameworks móviles, como React Native, Flutter, Xamarin.

Examinar la facilidad de uso y la curva de aprendizaje de cada Framework a través de encuestas y entrevistas con desarrolladores experimentados y novatos.

Comparar los costos asociados al desarrollo y mantenimiento de aplicaciones móviles utilizando diferentes Frameworks, analizando tanto los costos directos (licencias, herramientas) como los indirectos (tiempo de desarrollo, recursos humanos)

VI. LINEA DE INVESTIGACION

Este trabajo de estudio de caso está orientado y alineado con la línea de investigación de los sistemas de información y comunicación, el emprendimiento y la innovación. Estas áreas están estrechamente relacionadas con sublíneas de investigación como las tecnologías de redes y la computación y hardware inteligentes. El objetivo de este estudio de caso se presenta trabajo sobre el análisis comparativo de Frameworks para el desarrollo de aplicaciones móviles se enmarca dentro de esta línea de investigación. Este tipo de estudios busca facilitar a los programadores y empresas puedan utilizar herramientas prácticas y fundamentadas que les puedan ayudar a seleccionar la plataforma más adecuada para sus ideas de proyectos, facilitando recursos y asegurando la calidad de las aplicaciones móviles. La experiencia en sistemas de información y comunicación, emprendimiento e innovación, así como en tecnologías de redes y computación inteligente, esto puede ser fundamental para impulsar el desarrollo de soluciones móviles de poder avanzar para que se pueda cumplir las necesidades diferentes en el mercado y de los usuarios finales.

VII. MARCO CONCEPTUAL

Hoy en día, los programadores de aplicaciones móviles han ido evolucionando y cambiado significativamente, impulsado por el aumento de los dispositivos móviles y la carencia de aplicaciones más complejas y funcionales. Por lo tanto, marco conceptual tiene como objetivo comparar diferentes plataformas usadas al momento de programar aplicaciones móviles, basándose en el rendimiento de las aplicaciones, la facilidad de uso, ventajas, desventajas, la eficiencia y la adaptabilidad para desarrollar aplicaciones móviles.

Frameworks para el Desarrollo de Aplicaciones Móviles

Definición de Frameworks

De acuerdo con (Lucena, 2023) un Frameworks es un conjunto de reglas que están diseñadas para facilitar y agilizar la elaboración de software. Las mismas propone una estructura esencial para que los programadores puedan usar como punto de partida, economizando tiempo y esfuerzo. Adicionalmente, los Frameworks facilitan soluciones a problemas usuales al momento de diseñar software, lo cual permite a los programadores concentrarse en las características específicas de cada aplicación que se realice en lugar de tener que batallar con dificultades técnicas.

Importancia de los Frameworks

Señala (Martín Párraga, 2022) que los Frameworks son primordiales para obtener un desarrollo competente y efectivo de aplicaciones móviles. Facilitan procesos complejos lo que permite la valorización de código y la aseguración de la firmeza en cada diseño y la funcionalidad de las aplicaciones. Además, se seleccionan e implementan debidamente, cada Frameworks se pueden utilizar extremadamente para los

programadores. La importancia de utilizar Frameworks radica en las numerosas ventajas que ofrecen, que generalmente superan las desventajas, dependiendo del tipo de proyecto en desarrollo

Ventajas

- Ofrecen estructuras prefabricadas. De forma predeterminada, los Frameworks contienen construcciones de programación prediseñadas. Esto hace que el proceso de creación sea más fácil y rápido.
- Cada Frameworks tiene una comunidad de soporte. Gracias a los equipos que suelen estar detrás de cada estructura, la herramienta está en constante evolución.
- La comunidad desarrolla constantemente mejoras y crea extensiones que hacen que el Frameworks sea más fácil de usar. Además, esta comunidad es útil cuando necesita aclarar sus dudas mientras aprende la funcionalidad del Frameworks.
- Facilitan el trabajo en equipo. Los Frameworks son importantes en el trabajo de desarrollo colaborativo porque todos los miembros del equipo conocen la estructura de carpetas, por lo que todos saben a qué carpeta ir para realizar tareas específicas.
- Los Frameworks ofrecen código que ya ha sido probado y verificado. Esto significa que podemos confiar en que funcionará bien, ya que ha pasado por pruebas rigurosas

Desventajas

- Para adquirir un Frameworks de front-end, los programadores se deben dedicar tiempo a entender su sintaxis y funcionalidad. El front-end suelen tener una estructura determinada, lo que nos puede limitar la flexibilidad y la particularización del diseño. En ocasiones, los programadores pueden encontrarse restringidos por los estándares puestos por la infraestructura.

- En función de la variedad de proyectos la selección de la plataforma, el uso de un ámbito visible puede ir disminuyendo el rendimiento.
- Al agregar capas adicionales de abstracción y funcionalidad, la plataforma puede afectar el rendimiento de las aplicaciones. Sin embargo, la mayoría de las plataformas están optimizadas para minimizar esta carga y proporcionar herramientas para mejorar el rendimiento. (Rodríguez, 2024)

Flutter

Flutter es un conjunto de herramientas de Google para diseñar interfaces de usuario que te permite desarrollar aplicaciones para móviles, web y escritorio a partir de un solo código. Según (Cristancho, 2022)Flutter es un Frameworks de software gratuito y de código abierto, lanzado por Google en mayo de 2017. Su principal ventaja es que permite crear aplicaciones para iOS y Android usando una única base de código y un solo lenguaje de programación, lo que resulta en una gran eficiencia y valor.



Ventajas

- Las primordiales ventajas de Flutter es su suficiencia para permitir el incremento de aplicaciones móviles. De acuerdo a una sola base de código, los programadores pueden elaborar aplicaciones que se ejecuten sin ningún problema en dispositivos Android
- Los mismo descartan la necesidad de diseñar aplicaciones independientes desde cero, lo que permite disminuir considerablemente el tiempo y los costos de las aplicaciones móviles.

- Flutter nos da acceso a hacer uso de componentes gráficos que acaten la estructura de diseño de de los sistemas operativos existentes.
- Otra ventaja es el límite de tiempo carga ya que es uno de los elementos de los que debemos prestar el mayor interés posible al momento de desarrollar una app.
- Flutter nos permite diseñar interfaces de usuario esquemáticamente atrayente. (Doonamis, 2023)

Desventajas

- Los aplicativos realizados con Flutter proceden a poseer tamaños de archivos mucho más amplios que aplicaciones nativas por la falta de la inclusión de bibliotecas y herramientas de widgets de Flutter. (Zhang, 2021)
- Flutter permite desarrollar aplicaciones web y de escritorio, estas capacidades menos elegantes y poderosas que sus dispositivos móviles. Es probable que algunas taras y comportamientos no estén totalmente compatibles u optimizados. (Patel, 2020)
- Además, Flutter da acceso a la incorporación de funciones nativas, las disponibilidades de unos complementos podrían en ocasiones ser limitada. En la mayoría de los casos, hay la posibilidad de que el desarrollador necesite escribir código personalizado adicional, lo que nos puede complicar el proceso de desarrollo.
- Aunque Flutter es conocido por su buen rendimiento, las aplicaciones son muy complicadas.

XAMARIN

De acuerdo con (Grosso, 2021) Xamarin es una herramienta estimable y utilizada ampliamente por los programadores de aplicativos móviles. Xamarin da acceso a escribir código en C#, q lo que luego se convierte para funcionar en iOS, Android. Estas características son muy valiosas porque permiten eliminar la necesidad de aprender diferentes lenguajes como Objective-C y Java.

Ventajas y desventajas de Xamarin



Ventajas: De acuerdo con (proscont.com, 2024),Xamarin da acceso a diseñar aplicaciones nativas para sin ninguna necesidad de reescribir los códigos para cada plataforma. Una de sus primordiales ventajas es la utilización del código. En diferencia a las demás herramientas, que solicitan que los módulos se puedan convertir en distintos lenguajes (como Java para Android y Objective-C).

Desventajas: No obstante, Xamarin posee algunas restricciones, como el acceso limitado a las librerías de código abierto. En tanto que el desarrollo nativo tiene por costumbre aprovechar numerosas tecnologías de código abierto sin embargo Xamarin logra hacer frente a las restricciones que se presenten.

React Native

De acuerdo con (Deloitte Spain, 2019), un Frameworks en JavaScript para diseñar aplicativos nativos para iOS y Android, se basan en librerías de JavaScript React con a creación de componentes visuales, mejorando el fin de los mismos para que en lugar de ser ejecutados en navegador puedan correr directamente sobre las plataformas móviles nativas, en este contexto iOS y Android. Dicho de otra manera, en lugar de elaborar un aplicativo web híbrida, lo que se logra al final como los resultados es una aplicativo real nativa, indistintamente de lo que se podría desarrollar con los códigos en Objective-C o Java.

VENTAJAS

- React Native, ofrece un buen rendimiento nativo al ofrecer elementos nativos como Imagen, Vista y Texto.
- Los códigos que se realizan en React Native son compatibles con numerosas plataformas. Por ello, es eficaz para diseñar aplicativos para Android e iOS y esta, como una de las React Native ventajas que se deberían tomar en cuenta.
- Permite entregar con más rapidez los proyectos de aplicaciones.

DESVENTAJAS

- No ofrecen un buen rendimiento inmediato a las aplicaciones nativas, en casos de que los aplicativos sean muy complejos o que soliciten gráficos intensivos, es posible que no alcancen el mismo grado de rendimiento que los aplicativos creados de modo nativa.
- Constituyen algunas funciones nativas esto puede depender de bibliotecas de

terceros. Estas bibliotecas no siempre pueden estar actualizadas o completamente compatibles con todas las versiones de las plataformas móviles, lo que puede ocasionar problemas.

- Aunque JavaScript y React son bastante conocidos por los programadores, aunque algunos desarrolladores que no tienen experiencia con ellos o con lo que es el desarrollo móvil pueden hallar que hay una curva de aprendizaje muy empinada al comenzar con React Native.

Sistemas Operativos Móviles

Según (Libre, 2022) Los sistemas operativos móviles son aquellos que procesan información de modos iguales a las computadoras. La mayor parte de ellos puede tener la habilidad de conectarse a Internet y poder tener una memoria limitada. Al dar uso a estos dispositivos, se puede realizar muchas cosas, como ver películas, comunicarte con amigos, gestionar tu agenda y jugar, entre otras actividades más que se pueden realizar.

La elaboración de aplicativos móviles se ha convertido en un sin número de críticos para las empresas y programadores individuales en vista al aumento de uso de los dispositivos.

Todo fabricante de dispositivos móviles elige cuidadosamente el sistema operativo para sus dispositivos móviles con su fin de predominar en el mercado y dar la mejor experiencia posible a sus usuarios.

A continuación, se destacan algunas de las marcas más vendidas y los sistemas operativos que utilizan:

Android: Utilizado por HTC, LG, Motorola, Samsung y BQ, entre otros. Android, propiedad de Google, ha lanzado varias versiones como Apple Pie, Banana Bread, Jelly Bean y

KitKat.

iOS: Sistema operativo exclusivo de Apple, utilizado en iPhone y iPad.

Symbian: Propiedad de Nokia y utilizado en sus teléfonos. Otras compañías necesitan permiso de Nokia para utilizar este sistema operativo.

Windows Phone: Fue Desarrollado por Microsoft, primordialmente utilizado en los que son Smartphone de gama alta de Nokia.

BlackBerry OS: Desarrollado por RIM (Research In Motion), utilizado en los dispositivos BlackBerry, que han desarrollado su propio sistema operativo móvil.

Firefox OS: Desarrollado por la Fundación Mozilla, creadora del navegador Firefox. Este sistema operativo busca establecerse en el mercado, con empresas como Geeksphone y Alcatel adoptándolo. (aretecnologia, 2022)

Clasificación de Frameworks de Desarrollo Móvil

Los Frameworks de desarrollo de aplicaciones móviles se pueden clasificar en tres categorías principales:

Frameworks Nativos: Esto Incluyen herramientas oficiales proporcionadas por los creadores del sistema operativo, estos pueden ser Swift/Objective-C para iOS y Kotlin/Java para lo que es Android.

Frameworks Híbridos: Esto también utilizan tecnologías web, que pueden ser como HTML, CSS y JavaScript para poder crear aplicaciones que funcionen en un contenedor nativo. Ejemplos incluyen Ionic y Apache Córdova.

Frameworks Multiplataforma: Permiten escribir código en un solo lenguaje y desplegarlo en múltiples plataformas. (Tecsol Group, 2020)

JAVASCRIPT

Según (Gauchat, 2021) JavaScript es un lenguaje de programación interpretado que se ha utilizado para diversas tareas, aunque anteriormente se veía principalmente como un complemento. La captación de JavaScript cambió con la implementación de nuevos implementos de análisis que aceleraron el procesamiento del código

Gracias a esta mejora en el rendimiento, JavaScript superó sus antiguas limitaciones y se consolidó como una excelente opción para el desarrollo web.

Según (Zubikarai, Sil, 2021) JavaScript muestra algunas ventajas:

- Ejecuta abiertamente en un navegador, lo que ocasiona retrasos al momento de llamar al servidor.
- Su sintaxis es suficientemente sencillo ya que se inspira en Java, lo que nos puede posibilitar su aprendizaje.
- Existen algunos recursos disponibles para adquirir conocimiento en JavaScript, y su uso sigue avanzando cada día .
- Nos permite utilizar cualquier tipo de página web, a diferencia de otros lenguajes de scripting.
- Al ejecutarse a lo que puede ser lado del cliente, JavaScript disminuye la carga en los servidores.
- Este Permite idear elementos interactivos, como puede ser la funcionalidad de arrastrar y soltar.
- Los desarrolladores pueden agregar extensiones y agregar funcionalidades adicionales.
- también se puede utilizar tanto en el front-end como en el backend, gracias a

Node.js.

Desventajas

Seguridad client-side: Al ejecutarse el código JavaScript en el lado del cliente, los errores y descuidos pueden ser explotados con fines maliciosos. Esto lleva a que algunas personas decidan desactivar JavaScript por completo.

Soporte de navegadores: A diferencia de los scripts del lado del servidor, que producen los mismos resultados, el código JavaScript puede ser interpretado de manera diferente por distintos navegadores. Sin embargo, en la actualidad estas diferencias son mínimas y no deberías tener problemas si verificas tu código en los principales navegadores.

widgets

como afirma (Ramírez, 2024) los widgets son pequeñas aplicaciones que se pueden sumar a lo que es a la pantalla principal de un dispositivo Android. Esto Sirven para mostrar información, puede ser el clima, o para realizar actividades directamente desde la pantalla de inicio, como inactivar el Wifi. Los widgets pueden ser aplicaciones independientes y puede formar parte de una aplicación más grande. Pueden venir preinstalados en tu dispositivo o ser parte de aplicativo que hayas instalado. También, hay aplicaciones diseñadas específicamente para el funcionamiento como widgets y personalizar tu pantalla de inicio, como Zooper Widget.

Comparativa de Frameworks

A continuación, se muestra una tabla con una comparativa de las principales características de cada uno de los Frameworks existentes.

Característica	Xamarin	Flutter	React Native
-----------------------	----------------	----------------	---------------------

Lenguaje de programación	C#	Dart	JavaScript
Desarrollado por	Microsoft	Google	Facebook
Rendimiento	Alto (cerca del nativo)	Alto (cerca del nativo)	Alto (cerca del nativo)
Acceso a APIs nativas	Sí, a través de Xamarin. Essentials y librerías nativas	Sí, a través de plugins y librerías nativas	Sí, a través de módulos nativos y librerías
Hot Reload	Sí	Sí	Sí
Base de código	Compartida (pero con dependencias específicas para iOS y Android)	Totalmente compartida	Compartida (pero con algunos módulos específicos para iOS y Android)
Componentes UI	Xamarin.Forms (UI multiplataforma) y acceso directo a controles nativos	Widgets personalizados propios	Componentes nativos mapeados a través de JavaScript
Soporte multiplataforma	iOS, Android, Windows, macOS	iOS, Android, Web, Windows, macOS	iOS, Android, Web (experimental)

Comunidad y soporte	Fuerte, especialmente en empresas que usan tecnologías Microsoft	Creciente, con fuerte respaldo de Google	Muy fuerte, con una gran comunidad de desarrolladores
Ecosistema	Integrado con Visual Studio y Azure	Integrado con herramientas de Google	Integrado con el ecosistema de Facebook y herramientas de desarrollo web
Curva de aprendizaje	Moderada (para desarrolladores con experiencia en .NET/C#)	Moderada (para desarrolladores nuevos en Dart)	Baja a moderada (para desarrolladores con experiencia en JavaScript)
Licencia	MIT	BSD	MIT
Popularidad	Menos popular que Flutter y React Native, pero con una base sólida de usuarios	Muy popular y creciendo rápidamente	Muy popular, especialmente en startups y proyectos móviles rápidos
Costos	Gratis (Visual Studio Community es gratuito, pero Visual	Gratis	Gratis

	Studio Enterprise es pago)		
Presupuesto	<p>- Herramientas de desarrollo: Gratis (Visual Studio Community) o pago (Visual Studio Enterprise, ~\$45-\$250 por mes)</p> <p>- Desarrolladores: \$70,000 - \$120,000 por año.</p> <p>- Costos adicionales: Servicios Azure, licencias adicionales</p>	<p>- Herramientas de desarrollo: Gratis</p> <p>- Desarrolladores: \$70,000 - \$120,000 por año.</p> <p>- Costos adicionales: Ninguno significativo</p>	<p>- Herramientas de desarrollo: Gratis</p> <p>- Desarrolladores: \$70,000 - \$120,000 por año.</p> <p>- Costos adicionales: Ninguno significativo</p>

Herramientas de desarrollo:

Xamarin: Puedes usar Visual Studio Community gratis, pero para equipos grandes o necesidades avanzadas, Visual Studio Enterprise tiene un costo que varía entre \$45 y \$250 por mes

por usuario.

Flutter: Las herramientas de desarrollo son gratuitas, incluyendo Android Studio y Visual Studio Code.

React Native: Las herramientas de desarrollo son gratuitas, incluyendo Android Studio y Visual Studio Code.

Desarrolladores:

Los salarios para desarrolladores móviles con experiencia en cualquiera de estas tecnologías son similares, oscilando entre \$70,000 y \$120,000 por año, dependiendo de la ubicación y el nivel de experiencia.

Costos adicionales:

Xamarin: Puede haber costos adicionales relacionados con el uso de servicios de Azure y licencias adicionales.

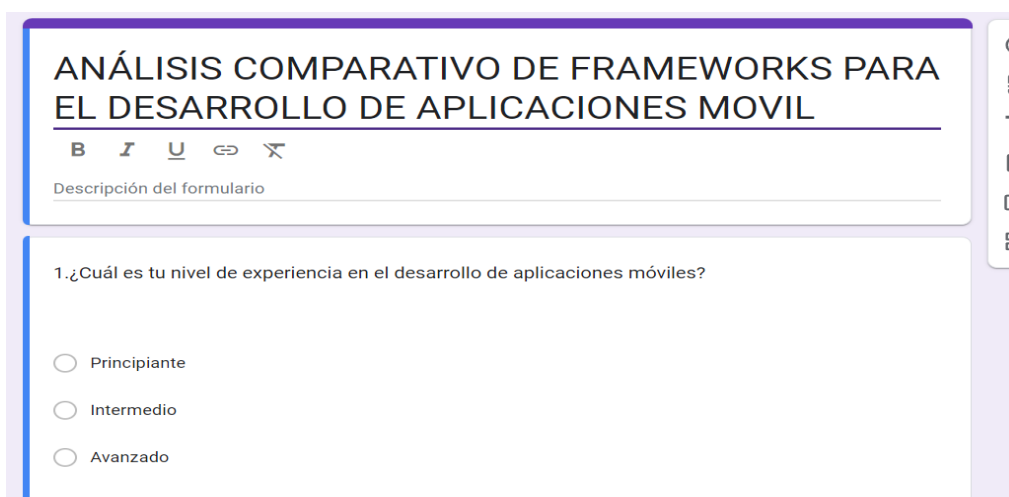
Flutter y React Native: No suelen tener costos adicionales significativos.

VIII. MARCO METODOLOGICO

Para realizar una evaluación comparativa integral y objetiva, se desarrolló la siguiente metodología, que incluye criterios detallados y procedimientos de recopilación de datos a través de una encuesta de desarrollo que se les hizo a los estudiantes de 8vo semestre vespertino.

Materiales

En cuanto al material utilizado para el proyecto de investigación se consideró como instrumento la Encuesta.



The image shows a screenshot of a survey form. The title is "ANÁLISIS COMPARATIVO DE FRAMEWORKS PARA EL DESARROLLO DE APLICACIONES MOVIL". Below the title is a description field labeled "Descripción del formulario". The first question is "1. ¿Cuál es tu nivel de experiencia en el desarrollo de aplicaciones móviles?". There are three radio button options: "Principiante", "Intermedio", and "Avanzado".

Métodos

Modalidad de la Investigación

El proyecto de investigación se basará en las siguientes modalidades:

Investigación bibliográfica: se revisó la teoría en diversas análisis comparativo, restringiéndose la búsqueda a fuentes como artículos científicos, libros y tesis a través de palabras claves como:

- **Frameworks:** React Native, Flutter, Xamarin

- **Aplicaciones móviles**
- **Rendimiento**
- **Facilidad de uso**
- **Costos**
- **Experiencia del usuario**

Investigación descriptiva: Esta investigación descriptiva se realizó con el fin de analizar el rendimiento, lenguaje de programación, componente, soporte, costo y sus características de cada aplicativo.

Investigación de campo: Se procedió a la recolección de datos mediante el envío de encuestas electrónicas a los estudiantes del octavo semestre, modalidad vespertina, de la Universidad Técnica de Babahoyo. Este proceso tuvo como objetivo obtener información relevante para el análisis comparativo de los diferentes Frameworks móviles utilizados por los estudiantes durante su proceso académico.

Población y Muestra

Para llevar a cabo este estudio, se determinó que la población estaría constituida por todos los estudiantes de octavo semestre de la carrera de Ingeniería en Sistemas de la Universidad Técnica de Babahoyo, en modalidad vespertina. Estos estudiantes fueron seleccionados debido a su acceso frecuente a la tecnología y su experiencia en el desarrollo de aplicaciones móviles, lo que los hace idóneos para proporcionar información relevante sobre el uso de frameworks móviles.

Muestra: Se optó por un muestreo por conveniencia, dada la accesibilidad y proximidad de los datos. Se envió una encuesta electrónica a la totalidad de la población objetivo, conformada por 30 estudiantes, de los cuales se obtuvo respuesta por parte de 23. Esta muestra, aunque no probabilística, se considera adecuada para los fines del estudio al proporcionar un panorama general sobre la percepción y uso de Frameworks móviles entre los futuros ingenieros.

Descripción	Detalle
Método de muestreo	Muestreo por conveniencia
Razón para el muestreo	Se seleccionó debido a la accesibilidad y proximidad de los datos disponibles.
Población objetivo	Totalidad de los estudiantes de 8vo semestre, modalidad vespertina, de la Universidad Técnica de Babahoyo.
Tamaño de la población	30 estudiantes
Tamaño de la muestra	23 estudiantes respondieron la encuesta.
Tipo de muestra	No probabilística
Justificación	Aunque no es una muestra probabilística, se considera adecuada para proporcionar un panorama general sobre la percepción y uso de frameworks móviles entre los futuros ingenieros.

IX. RESULTADOS

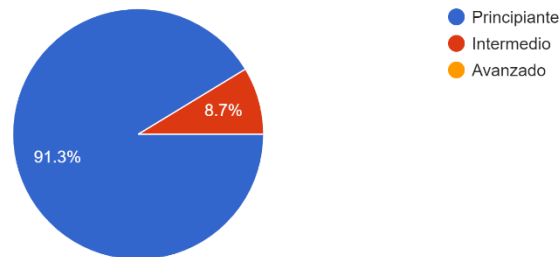
Resultados Cuantitativos

Encuestas a Desarrolladores:

Se obtuvieron **23** respuestas de desarrolladores con diversos niveles de experiencia.

El análisis de las encuestas reveló las siguientes preferencias:

¿Cuál es tu nivel de experiencia en el desarrollo de aplicaciones móviles?

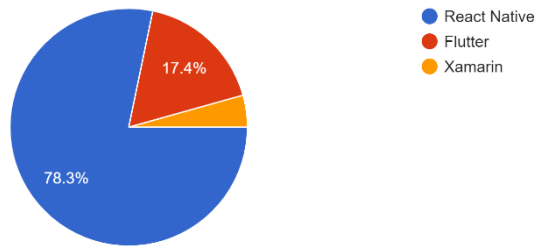


(Muñoz, 2024)

Análisis e Interpretación:

En su gran mayoría de los encuestados se consideran principiantes en el desarrollo de aplicaciones móviles. Esto puede influir en sus percepciones y opiniones sobre los frameworks que utilizan, ya que su experiencia es limitada.

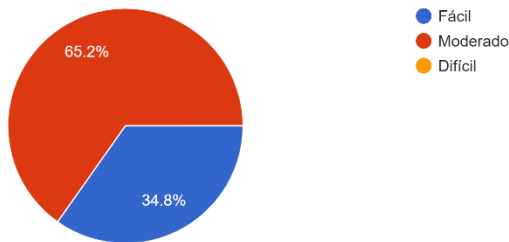
2 ¿Con cuál de los siguientes frameworks móviles has trabajado?



Análisis e Interpretación:

La mayoría de los encuestados han trabajado con React Native, mientras que solo uno ha trabajado con Flutter. Esto podría indicar una preferencia o mayor familiaridad con React Native entre los principiantes.

3. ¿Cuál consideras que es la facilidad de uso del frameworks que utilizas principalmente?

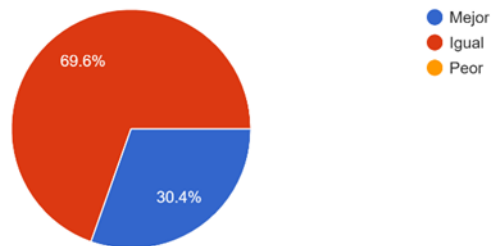


Análisis e Interpretación:

Todos los encuestados calificaron la facilidad de uso del frameworks que utilizan como "Moderado". Esto sugiere que, aunque son principiantes, encuentran los frameworks relativamente manejables, pero no necesariamente fáciles.

4. ¿Cuál es tu percepción sobre el rendimiento del frameworks que usas

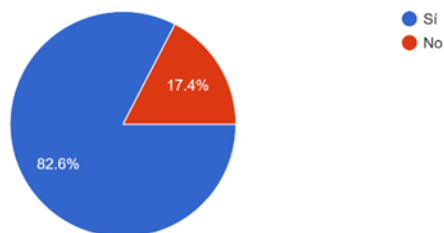
principalmente en comparación con otros?



Análisis e Interpretación:

La percepción del rendimiento está dividida entre "Mejor" e "Igual". Esto indica que algunos encuestados consideran que el Frameworks que utilizan supera a otros en rendimiento, mientras que otros lo ven comparable.

5. ¿Recomendarías el Frameworks que utilizas a otros desarrolladores?



Análisis e Interpretación:

Todos los encuestados recomendarían el Frameworks que utilizan. Esto refleja una satisfacción general con su elección, a pesar de su nivel de experiencia

X. DISCUSIÓN DE RESULTADO

Los resultados obtenidos de la encuesta y el análisis de datos brindan una imagen completa de las percepciones y experiencias de los desarrolladores con diferentes plataformas móviles. A continuación, se muestra una discusión detallada de estos resultados: La mayoría de los encuestados se consideraban nuevos en el desarrollo de aplicaciones móviles. Este dominio puede influir significativamente en su percepción y evaluación de los Frameworks. Los principiantes tienden a buscar herramientas que sean más accesibles y requieran menos curva de aprendizaje, lo que puede explicar su preferencia por Frameworks con documentación sólida y soporte de comunidades como React Native. La clara preferencia por React Native entre los encuestados indica una mayor conciencia del manejo y el conocimiento del Frameworks. Esto es posible gracias a un amplio documento, una comunidad activa y el apoyo de las principales empresas de tecnología. Flutter, aunque conocido por su productividad y capacidades de diseño, parece ser menos conocido en las personas principiantes, tal vez debido a su curva de aprendizaje más pronunciada o su menor claridad. Todos los encuestados calificaron la facilidad de uso de los Frameworks que utilizaron como "promedio". Esto indica que los Frameworks seleccionados, aunque no se consideran extremadamente complejos, no se consideran completamente accesibles para los principiantes. Este hallazgo resalta la necesidad de mejorar la usabilidad y accesibilidad de los Frameworks para desarrolladores novatos.

Es alentador ver que todos los encuestados recomendarían el Frameworks React Native que usan a otros desarrolladores, lo que es un buen indicador de satisfacción general. Los desarrolladores, independientemente de su nivel de experiencia, siguen encontrando valor en las herramientas que utilizan y sienten que estas plataformas cumplen con sus expectativas y

necesidades básicas.

Consecuencias y recomendaciones Dado que entre los encuestados predominaban los recién llegados, es necesario desarrollar más recursos educativos y de formación centrados en facilitar el proceso de aprendizaje. Esto puede incluir tutoriales, cursos en línea y documentación más detallada y accesible. Comunidad: Hacer crecer y fortalecer su comunidad de usuarios puede ser beneficioso porque proporciona un importante sistema de soporte para los desarrolladores, especialmente aquellos que recién comienzan. Fomentar más foros, blogs y recursos de soporte puede mejorar la experiencia del usuario. Dada la controversia en torno al rendimiento, es importante que los desarrolladores de Frameworks sigan optimizando sus herramientas para garantizar un rendimiento superior, especialmente para proyectos complejos y aplicaciones exigentes

Para equilibrar la adopción de diferentes Frameworks, sería útil aumentar la visibilidad y promoción de opciones como Flutter entre los nuevos desarrolladores. Esto se puede lograr mediante campañas educativas y comparativas que destaquen las ventajas y características únicas de cada herramienta.

XI. CONCLUSIONES

Este estudio comparó los Frameworks de desarrollo de aplicaciones móviles más populares: React Native, Flutter y Xamarin. El objetivo es determinar sus fortalezas y debilidades en términos de rendimiento, facilidad de uso, costo y experiencia de usuario. Los principales hallazgos se resumen a continuación:

El principal hallazgo

Reaccionar preferencia nativa:

La mayoría de los desarrolladores encuestados utilizan React Native. Esto muestra que las personas, especialmente los principiantes, están más familiarizadas y prefieren este Frameworks debido a su extensa documentación, comunidad activa y soporte de las grandes empresas de tecnología. fácil de usar:

Los desarrolladores calificaron estos Frameworks como "moderados" en términos de facilidad de uso. Esto demuestra que, aunque no son extremadamente complejos, los principiantes no los comprenden completamente, lo que destaca la necesidad de mejorar la usabilidad y accesibilidad para los nuevos desarrolladores. Percepción del desempeño:

Las opiniones difieren sobre cómo funcionan estos Frameworks. Algunos desarrolladores creen que su Frameworks principal tiene un mejor rendimiento, mientras que otros creen que está a la par con otros Frameworks. Esto indica una diferencia en la experiencia del usuario en términos de eficiencia.

Satisfacción general:

Todos los desarrolladores recomiendan los Frameworks que utilizan a otros. Refleja su satisfacción general con su elección, independientemente del nivel de experiencia. Los desarrolladores encuentran valiosas estas herramientas y creen que satisfacen sus expectativas y necesidades básicas.

Por último, la encuesta proporciona un entendimiento claro de las preferencias y experiencias de los desarrolladores con diferentes sistemas operativos. Las respuesta muestran que React Native es el más conocido y el más popular debido a su sólida documentación y comunidad, aunque todavía pueden haber algunas áreas que se pueden mejorar en términos de usabilidad y la productividad para poder satisfacer mejor las necesidades de los programadores, en particular los principiantes.

XII. RECOMENDACIONES

Responder a las preferencias locales:

Impulsar las adopciones de otros Frameworks como son Flutter y Xamarin que requiere implementar programas de educación y capacitación que resalten los beneficios únicos de cada Frameworks. Esto se pueden incluir seminarios web y materiales de capacitación diseñados para demostrar cómo funcionan estos sistemas en diversos entornos de desarrollo. Trabajar con la comunidad técnica y las empresas para crear estudios de casos y proyectos de referencia también puede aumentar la visibilidad y la familiaridad de estos sistemas para los principiantes.

fácil de usar:

Para mejorar la usabilidad de todos los Frameworks, las empresas deberían invertir en mejorar la documentación, guías paso a paso y capacitación detallada. Para responder a las preferencias locales, también sería beneficioso crear interfaces de usuario y herramientas de desarrollo más fáciles de entender. Los principiantes también se benefician de una base de conocimientos centralizada y accesible con soluciones a problemas comunes. Consideraciones sobre el desempeño:

Los desarrolladores de Frameworks deben concentrarse en la optimización continua de sus herramientas para abordar las diferentes percepciones de rendimiento. Esto pueden incluir revisiones del desempeño regulares y poder mejoras basadas en resultados. Las superiores herramientas de análisis y monitoreo del rendimiento pueden ayudar a los desarrolladores a optimizar las aplicaciones de manera más eficiente. Aclarar las confusiones y poder proporcionar una visión más objetiva también se pueden ayudar a compartir puntos de referencia que muestran el rendimiento en distintos escenarios.

Satisfacción general:

Si bien en general es satisfactorio, siempre se puede mejorar. Para mantener y mejorar este

nivel de satisfacción, se anima a los usuarios a proporcionar comentarios continuos y responder rápidamente a sus sugerencias y quejas. Implementar un programa de lealtad y reconocimiento para los desarrolladores que contribuyen activamente a la comunidad es una excelente manera de mantener a la comunidad comprometida y satisfecha. Además, la inversión continua en soporte técnico y recursos de capacitación ayudará a garantizar que los desarrolladores siempre se sientan respaldados. En general, estas recomendaciones apuntan no solo a mejorar la experiencia de los desarrolladores actuales, sino también a atraer nuevos usuarios a Frameworks menos conocidos, optimizando así el panorama general de desarrollo de aplicaciones móviles.

XIII. ANEXO

ANÁLISIS COMPARATIVO DE FRAMEWORKS PARA EL DESARROLLO DE APLICACIONES MOVIL

B *I* U ↻ ✕

Descripción del formulario

1.¿Cuál es tu nivel de experiencia en el desarrollo de aplicaciones móviles?

- Principiante
- Intermedio
- Avanzado

2.¿Con cuál de los siguientes frameworks móviles has trabajado?

- React Native
- Flutter
- Xamarin

⋮

3.¿Cuál consideras que es la facilidad de uso del framework que utilizas principalmente?

- Fácil
- Moderado
- Difícil



4. ¿Cuál es tu percepción sobre el rendimiento del framework que usas principalmente en comparación con otros?

- Mejor
- Igual
- Peor

5. ¿Recomendarías el framework que utilizas a otros desarrolladores?

- Sí
- No

23 respuestas

 [Vínculo a Hojas de cálculo](#) 

Se aceptan respuestas

Resumen

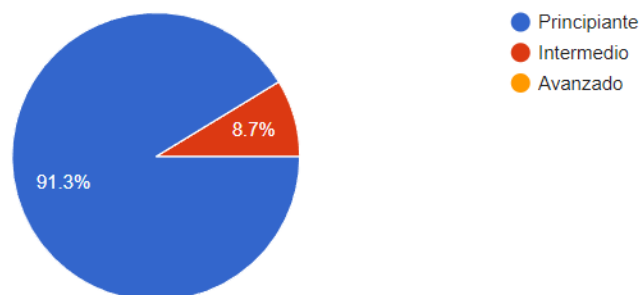
Pregunta

Individual

1. ¿Cuál es tu nivel de experiencia en el desarrollo de aplicaciones móviles?

 Copiar

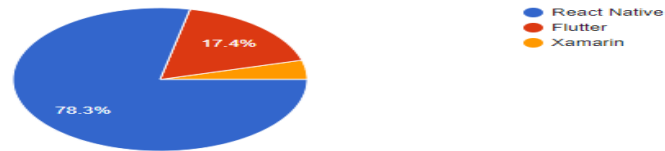
23 respuestas



2. ¿Con cuál de los siguientes frameworks móviles has trabajado?

[Copiar](#)

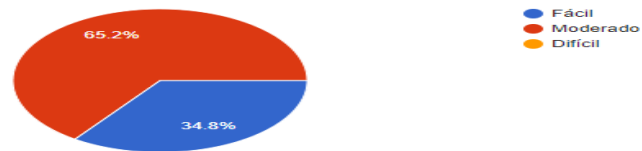
23 respuestas



3. ¿Cuál consideras que es la facilidad de uso del framework que utilizas principalmente?

[Copiar](#)

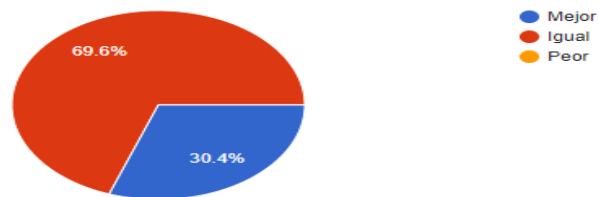
23 respuestas



4. ¿Cuál es tu percepción sobre el rendimiento del framework que usas principalmente en comparación con otros?

[Copiar](#)

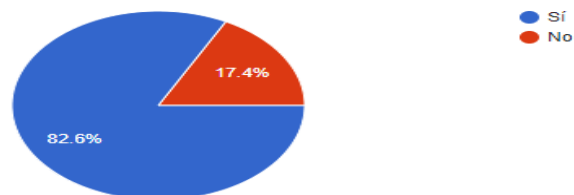
23 respuestas



5. ¿Recomendarías el framework que utilizas a otros desarrolladores?

[Copiar](#)

23 respuestas



XIV. Bibliografía

Doonamis. (2023, noviembre 23). *Flutter, descubre sus ventajas y desventajas*.

Doonamis. <https://www.doonamis.com/descubre-ventajas-desventajas-flutter/>

Grosso, N. (2021, octubre 8). *Xamarin: qué es y cómo funciona*. okdiario.com.

<https://okdiario.com/tecnologia/que-xamarin-2022974>

Libre, G. A. (s/f). *Sistemas operativos para dispositivos móviles*. Edu.py. Recuperado el

1 de agosto de 2024, de https://aprendizaje.mec.edu.py/dw-recursos/system/content/12c54d4/tecnologia/curso/informatica_basica/sistemas_operativos/5.do.html

Lidekrans, G. T. M. (s/f). *React Native vs. Flutter: A performance comparison between cross-platform mobile application development frameworks*. Diva-portal.org.

Recuperado el 1 de agosto de 2024, de <https://www.diva-portal.org/smash/get/diva2:1768521/FULLTEXT01.pdf>

Los mejores frameworks para desarrollo de aplicaciones móviles híbridas. (2020, julio

27). Tecsol Group. <https://tecsolgroup.com.ar/los-mejores-frameworks-para-desarrollo-de-aplicaciones-moviles-hibridas/>

Lucena, P. (2023, mayo 6). *¿Qué es el framework?* Maestrías y MBA.

<https://www.cesuma.mx/blog/que-es-el-framework.html>

Martín-Párraga, L., Llorente-Cejudo, C., & Cabero-Almenara, J. (2022). Análisis de las competencias digitales docentes desde los marcos e instrumentos de

evaluación. *International journal of educational research and innovation*, 2022(18), 62–79. <https://doi.org/10.46661/ijeri.7444>

Ramírez, I. (2024, enero 2). *Cómo añadir widgets a la pantalla de inicio en Android*. Xatakandroid.com; Xataka Android. <https://www.xatakandroid.com/tutoriales/como-anadir-widgets-a-la-pantalla-de-inicio-en-android>

Rodríguez, F. (2024, marzo 15). *Ventajas y Desventajas de Utilizar Frameworks Frontend en el Desarrollo Web*. *cursosadvanize.com*. <https://cursosadvanize.com/frameworks-frontend/>

Sistemas Operativos Mviles. (s/f). Recuperado el 1 de agosto de 2024, de <https://www.areatecnologia.com/informatica/sistemas-operativos-mviles.html>

Ventajas y desventajas de Xamarin. (2024, marzo 28). *proscont.com*. <https://www.proscont.com/ventajas-y-desventajas-del-xamarin/>

Zubikarai, S. (2021, marzo 15). *Ventajas y Desventajas de JavaScript*. *freecodecamp.org*. <https://www.freecodecamp.org/espanol/news/ventajas-y-desventajas-de-javascript/>

(S/f-a). Doria.fi. Recuperado el 1 de agosto de 2024, de https://www.doria.fi/bitstream/handle/10024/180002/hjort_elin.pdf?sequence=2&isAllowed=y

(S/f-b). Deloitte.com. Recuperado el 1 de agosto de 2024, de

<https://www2.deloitte.com/es/es/pages/technology/articles/que-es-react-native.html>

(S/f-c). Edu.sv. Recuperado el 1 de agosto de 2024, de

http://biblio.uls.edu.sv/doc_num.php?explnum_id=2