**Minería de repositorios: Benchmark Kotlin Multiplatform, React Native, Flutter**

Santiago Hernández Facio Lince

Ingeniería de Sistemas y Computación

Universidad de los Andes

Bogotá, Colombia

s.hernandezf@uniandes.edu.co

Resumen:

Frente al caótico y extenso mundo de posibilidades que abarca en desarrollo de aplicaciones móviles, este escrito espera ofrecer una guía a los diferentes desarrolladores móviles con respecto a 3 de los lenguajes de programación más populares para el desarrollo de aplicaciones híbridas. Se espera comparar y clasificar estos 3 lenguajes: Kotlin Native, React Native, Flutter, en distintas métricas que puedan ser útiles para definir las fortalezas y debilidades de cada lenguaje, buscando ayudar en la elección del lenguaje de programación a futuros proyectos.

Introducción:

Frente a la creciente expansión de variedad de fabricantes de dispositivos y/o sistemas operativos para dispositivos móviles, los diversos grupos interesados en el desarrollo móvil se enfrentan a un mar de tecnologías para el desarrollo de sus aplicaciones móviles, empezando por elegir desarrollo nativo, web o híbrido para posteriormente elegir un lenguaje de programación que cumpla de la mejor forma con los requerimientos funcionales y atributos de calidad buscados por el desarrollador.

No es un secreto que existe una gran cantidad de lenguajes de programación en el mercado y que al momento de elegir uno para desarrollar un nuevo proyecto o incluso para aprenderlo e iniciarse en el mundo de la programación, puede llegar a ser una elección un poco abrumadora. Afortunadamente, los lenguajes de programación pueden llegar a ser clasificados en base a sus fortalezas, algunos son buenos para el desarrollo de videojuegos, otros, los que nos interesan, son enfocados específicamente para el desarrollo de aplicaciones móviles híbridas.

Ahora bien, aunque es de gran ayuda reducir la elección a solo los lenguajes enfocados al desarrollo móvil híbrido, sigue habiendo mucha oferta a considerar. Debido a esto, distintos programadores de todo el mundo, a lo largo de los años han empezado a generar pruebas para comparar lenguajes de programación desde distintas métricas: velocidad, usabilidad, mantenimiento, respuesta, etc. Posteriormente, se comparan los resultados entre lenguajes y se definen fortalezas y debilidades. Todo este análisis es reportado en un informe llamado Benchmark y publicado buscando ayudar a la comunidad.

Sin embargo, siendo el desarrollo de aplicaciones móviles híbridas un sector tecnológico relativamente nuevo, no existen muchas referencias ni Benchmarks que comparen los lenguajes de programación que están hoy en día en el mercado, pese a que este sector tecnológico es de los más demandados en la actualidad.

Solución:

Buscando mitigar esta problemática y pensando en contribuir a la comunidad de desarrolladores de aplicaciones móviles, esta tesis esperar construir un Benchmark confiable y completo que compare 3 Frameworks de aplicaciones híbridas más populares: Kotlin Multiplatform, React Native, Flutter.

Este Benchmark espera comparar los lenguajes desde distintos ángulos, definiendo métricas para comparar los lenguajes de programación base de los frameworks:

* Kotlin: Kotlin Multiplatform.
* JavaScript: React Native.
* Dart: Flutter.

Adicionalmente a esto, se definirán de igual manera métricas para comparar los frameworks directamente, así como la comparación de la aplicación final realizada con cada framework.

Objetivos:

* Usar métricas de calidad básicas y generales.
* Usar métricas de calidad enfocadas en el desarrollo móvil.
* Realizar una comparación objetiva de los lenguajes (definir un proyecto común a desarrollar).
* Proporcionar todo el ambiente necesario para que otros desarrolladores puedan realizar o contribuir sus propias comparaciones.
* Realizar pruebas controladas.
* Realizar pruebas en ambientes de producción “reales”

Resultados esperados

Se espera contar con un Benchmark que refleje de forma clara y concisa los distintos “experimentos” realizados y los resultados obtenidos gracias a estos. Se espera comparar de forma objetiva los lenguajes y poder definir fortalezas y debilidades de cada lenguaje. Adicionalmente se busca dejar un repositorio con todo el código y con todo lo necesario para que, si otro desarrollador lo desea, pueda recrear los experimentos y añadir contribuciones al repositorio, generando cada vez un benchmarking más extenso y confiable, esperando que en el futuro sirva de guía para los desarrolladores móviles.

# Referencias

Corporación Universitaria Comfacauca. (2 de 9 de 2016). *https://www.unicomfacauca.edu.co/*. Obtenido de PROPUESTA PROYECTO DE GRADO : https://www.unicomfacauca.edu.co/wp-content/uploads/EDO-01-PROPUESTA-PROYECTO-GRADO.pdf

Team Asana. (8 de 11 de 2022). *6 pasos para redactar una propuesta de proyecto convincente*. Obtenido de asana: https://asana.com/es/resources/project-proposal