

# Análise de desempenho de aplicações mobile híbridas e nativas

Autor: Robson Novato Lobao

Orientador: Tiago Garcia de Senna Carneiro

27 de outubro de 2022

## Resumo

Os smartphones, tão comuns hoje em dia, que contam com diversas funcionalidades e ajudam hodiernamente seus usuários, não possuem uma solução de desenvolvimento única. Há dois sistemas operacionais que dominam o mercado atualmente, o Android manteve sua posição como o sistema operacional móvel líder em todo o mundo em agosto de 2022, controlando o mercado de sistemas operacionais móveis com cerca de 70% de participação, enquanto o iOS representou cerca de 28% do mercado de sistemas operacionais móveis (LARICCHIA, 2022). Os aplicativos podem ser desenvolvidos de forma dependente de Sistemas Operacionais, em que, o desenvolvimento é dedicado a plataforma que deseja se obter um aplicativo. Ou podemos utilizar uma única linguagem de programação para ambas plataformas, em que contamos com transferência de código para compilação interna em seus respectivos SO's, alguns frameworks híbridos que são utilizados hoje em dia é o ReactNative, Flutter e Xamarin. Entretanto, os ganhos de tempo para desenvolvimento único podem afetar o desempenho no âmbito de processamento, ponto abordado nesse trabalho, diferença de desempenho entre aplicações híbridas e nativas. Apesar das versões nativas demonstrarem algumas vantagens de performance, o framework obteve bons resultados de tempo de resposta, além de possibilitar acesso as funcionalidades nativas, ter boa documentação e alto reúso de código.

## 1 Introdução

De forma geral, as aplicações costumam ser desenvolvidas para sua plataforma nativa, a linguagem para o iOS é o Swift e para plataformas Android é o Kotlin necessitando manter muitos colaboradores em um único projeto e não permitindo alterações ágeis, logo o custo para desenvolver, ferramentas e pessoal pode tornar inviável o desenvolvimento do software (BOCARD, 2021).

Com relação as linguagens nativas, as duas em destaque é o Swift, como citado, linguagem de propósito geral desenvolvida pela Apple que conta com funções de integrar em todos dispositivos da marca e interagir com a antiga linguagem recomendada, o objective-C. Além disso, temos o Kotlin, linguagem da Google que é a oficial do Android e mais recomendada perante o Java, a antiga linguagem oficial, por ser mais conciso, seguro e reduzir erros e bugs no código (BOSE, 2018).

Diante dos problemas acarretados pelo desenvolvimento nativo surge o desenvolvimento híbrido, que através de uma única linguagem atende as necessidades básicas do usuário com relação a funcionalidade específicas do aparelho, como um GPS ou tirar fotos. Entretanto, as vezes, a custo de desempenho, recurso importante e muito levado em conta por usuários ao definir o software que utilizará, além do design (FANG et al., 2017).

Como exemplo de desenvolvimento híbrido, o Nubank, maior fintech da América Latina segundo Gagne (GAGNE, 2019), cujos produtos (conta digital, cartão de crédito, entre outros) são totalmente controlados por meio do seu aplicativo, recentemente iniciou a adoção do Flutter, framework de desenvolvimento multiplataforma da Google, integrando-o à já existente base de código em React Native, Kotlin e Swift de acordo com Freire e Andrade (FREIRE; ANDRADE, 2019).

Flutter foi lançado recentemente, em 2017, e pode ser uma boa alternativa aos frameworks e ferramentas multiplataforma existentes. Neste trabalho será comparado o desenvolvimento multiplataforma utilizando esse framework com o desenvolvimento nativo, após ter sido identificada a falta de estudos publicados na revisão bibliográfica apresentada no capítulo 3.

Gasparotto (GASPAROTTO, 2022) diz que essa abordagem tem importância por reduzir custos do projeto já que há reaproveitamento de código e otimização do tempo de desenvolvimento, e diminuição de colaboradores, todavia, a prova cabal de utilização de tecnologias é o mercado, e a Facebook, como grande empresa tech, experienciou cometer, segundo palavras de seu CEO Mark Zuckerberg, "O maior erro da história do Facebook nos últimos anos" ao apostar em aplicações híbridas frente as nativas, gerando imensa insatisfação nos usuários.

Por conseguinte, tendo em vista que a possibilidade de economia com utilização de tecnologias híbridas coloca o nativo à prova, o objetivo dessa pesquisa é mostrar a diferença de desempenho em ambas aplicações, tanto híbridas como nativas, respondendo a pergunta: tecnologias híbridas ou nativas, qual possui melhor desempenho médio?

Tendo isso em vista, poucos esforços foram mostrados para tentar analisar a ótica de desempenho das tecnologias, se atentando mais a funcionalidades e facilidade de aprendizado de ambas formas de desenvolvimento, no artigo de Olusola Olajide Ajayi (AJAYI, 2018) foram utilizados alguns benchmarks de performance, entretanto, possivelmente enviesados e sem análise de consumo de bateria, condição indispensável ao se desenvolver para smartphones, tópico abordado nessa pesquisa.

Como pretendo inovar na área? Como dito anteriormente, a maioria das pesquisas destaca questões de desenvolvimento com habilidades específicas dos profissionais, ou seja, se a equipe tem facilidade ou não de aprender novas tecnologias, se seria melhor aproveitado se todos de uma equipe utilizassem apenas uma linguagem. Ademais, frisam questões de disponibilidade de funcionalidades como o faceID, pagamento digital com GoolePay/ApplePay e similaridade da User Interface entre os dispositivos de diferentes sistemas operacionais, destacado no trabalho de Seung-Ho Lim, um dos mais referenciados na área de desenvolvimento mobile. O foco do trabalho está em, empiricamente, provar a superioridade em desempenho, utilizando benchmarks e testes específicos de processamento.

Em conclusão, proponho a hipótese que tecnologias nativas vão ter, em todos os casos, desempenho superior ao híbrido, tanto em tempo de execução como em saúde da bateria, já que as próprias plataformas já desenvolvem ou utilizam as linguagens com otimização máxima para aquele contexto, e no caso da Apple, que preza por exclusividade, e não precisam lutar por fatia de mercado já que detém cerca de 78% de todas as vendas de smartphones premium (BEN, 2022), essa diferença é ainda mais visível. E que linguagens que são duplamente compiladas são quase que idealmente inferiores em desempenho.

## 2 Objetivos

O objetivo geral é comparar o método de desenvolvimento nativo com o desenvolvimento multiplataforma, implementando uma aplicação com versões nativas para iOS e Android.

Será feito utilizando os métodos nativos, implementação da mesma aplicação no Android utilizando o Kotlin e no iOS utilizando o Swift. Logo após será escolhido o framework Flutter, que utiliza a linguagem Dart de autoria do Google para representar o desenvolvimento híbrido, tendo em vista que é o maior expoente do mesmo e grandes empresas com Nubank utilizam ela para fazer o desenvolvimento dos seus produtos.

Todos os testes serão feitos em ambiente controlado, tanto em dispositivos físicos como simulações virtuais com emuladores, tentando ao máximo representar o cotidiano do usuário e mantendo condições semelhantes e utilizando a bateria nos dispositivos físicos como ponto importante.

Logo após, será feita a comparação das versões por critérios de consumo de CPU, consumo de memória, tamanho da aplicação, tempo de resposta, acesso a funcionalidades nativas e reuso de código.

## 2.1 Objetivos específicos

Com relação a objetivos específicos teremos: (1) identificar os métodos de desenvolvimento para plataformas móveis e seus desafios ou dificuldades, (2) definir e implementar uma aplicação para plataformas móveis com versões nativas para iOS e Android, e versão multiplataforma utilizando Flutter, (3) comparar as versões do aplicativo desenvolvidas pelas critérios de consumo de CPU, (4) consumo de memória, (5) tamanho da aplicação, (6) tempo de resposta, (7) acesso a funcionalidades nativas, (8) consumo de bateria e (9) reuso de código.

## Referências

AJAYI, Olusola Olajide. Performance Evaluation of Native and Hybrid Android Applications. en. **Communications on Applied Electronics**, v. 7, n. 16, p. 1, 2018.

BEN, Lovejoy. iPhone US market share hits all-time high, overtaking Android; dominates global premium sales. en. **9To5Mac**, v. 1, n. 1, p. 1–2, 2022.

BOCARD, Taysa. Quanto custa um aplicativo? pt. **Usemobile**, v. 1, n. 1, p. 4, 2021.

BOSE, Subham. A COMPARATIVE STUDY: JAVA VS KOTLIN PROGRAMMING IN ANDROID APPLICATION DEVELOPMENT. en. **International Journal of Advanced Research in Computer Science**, v. 9, n. 3, p. 4, 2018.

FANG, Jiaming et al. Design and performance attributes driving mobile travel application engagement. en. **International Journal of Information Management**, v. 37, n. 4, p. 269–283, 2017.

FREIRE, Alexandre; ANDRADE, Vinicius. Why we think Flutter will help us scale mobile development at Nubank. en. **Medium**, v. 1, n. 1, p. 1, 2019.

GAGNE, Yasmin. Banking in Brazil is hard. Here's how Nubank is changing that. en. **FastCompany**, v. 1, n. 4, p. 1, 2019.

GASPAROTTO, Henrique. Desenvolvimento multiplataforma com Xamarin. pt. **DevMedia**, v. 1, n. 1, p. 1, 2022.

LARICCHIA, Federica. Mobile operating systems' market share worldwide from January 2012 to August 2022. en. **Statista**, v. 1, n. 1, p. 4, 2022.