

Desenvolvimento de Aplicações Móveis com a Plataforma React Native

48329 Ruben Farinha e 48738 Gonalo Verissimo

CLAV, Universidade de  vora, Portugal

<https://uevora.pt>

Engenharia Inform tica

2023/24

Abstract. As aplica es m veis fazem cada vez mais parte do nosso dia a dia e as plataformas evoluem constantemente para se adaptar  s necessidades dos utilizadores, o desenvolvimento de apps tem crescido exponencialmente nos  ltimos anos. Com este crescimento, surgem novas metodologias e ferramentas para otimizar o processo de cria o de apps. Frameworks como o React Native s o extremamente  teis para este prop sito, pois oferecem ferramentas que simplificam e agilizam o desenvolvimento.

Keywords: Aplica o M vel · React Native · Framework.

1 Introdu o

O desenvolvimento de aplica es m veis   um mercado em constante crescimento, com a necessidade de criar apps para diferentes plataformas (Android e iOS), surgiram as **frameworks multiplataforma**. Neste short paper vamos abordar como a plataforma **React Native** facilita este processo de desenvolvimento de aplica es m veis eficientes e escal veis, acabando por discutir tamb m vantagens e desvantagens da sua utiliza o, compara o a outras alternativas e exemplos desenvolvidos utilizando esta plataforma. React Native   uma poderosa ferramenta que **revolucionou** a maneira como os desenvolvedores criam experi ncias de aplica es m veis. Criado e mantido pelo Facebook, React Native foi lanado pela primeira vez em 2015 e desde ent o tem sido amplamente adotado pela comunidade. Foi concebido pela necessidade de simplificar e acelerar o processo de desenvolvimento de aplica es m veis, mantendo a capacidade de criar experi ncias de utilizador ricas e de alta qualidade. Antes da sua introdu o, os desenvolvedores enfrentavam o desafio de criar aplica es nativas para cada plataforma (iOS e Android), o que consumia tempo e recursos consider veis.

2 Funcionamento

O React Native   uma estrutura de **desenvolvimento de aplica es m veis multiplataforma** que utiliza JavaScript como linguagem de programa o principal. Ao contr rio de outras frameworks, o React Native compila o c digo JavaScript em c digo nativo da plataforma (iOS ou Android) atrav s de um processo chamado "**compila o ahead-of-time**" (AOT).

O funcionamento do React Native pode ser resumido da seguinte forma:

C digo JavaScript: Os desenvolvedores escrevem o c digo da aplica o em JavaScript, ou em uma linguagem que   traduzida para JavaScript, como TypeScript.

Bridge de Comunica o: O React Native utiliza um "bridge" de comunica o que permite a intera o entre o c digo JavaScript e os componentes nativos do dispositivo. Este bridge   respons vel por traduzir as chamadas de fun o e os eventos entre o c digo JavaScript e o c digo nativo.

Componentes Nativos Reutiliz veis:   fornecida uma biblioteca de componentes nativos reutiliz veis, escritos em Objective-C para iOS e Java para Android. Estes componentes encapsulam a funcionalidade nativa, como bot es, listas e barras de navega o, e s o renderizados diretamente nos elementos da interface do utilizador da plataforma.

Renderiza o Din mica: O React Native utiliza um mecanismo de renderiza o din mica que permite atualizar a interface do utilizador de forma eficiente em resposta a eventos e mudan as de estado. Isso   alcan ado atrav s de uma reconcilia o virtual do DOM (Document Object Model) em JavaScript e da atualiza o correspondente dos componentes nativos na interface do utilizador.

Empacotamento, Compila o e Otimiza o: O c digo JavaScript e os recursos da aplica o s o empacotados num arquivo de bundle que   carregado e executado no dispositivo. Durante o processo de compila o, o c digo JavaScript   pr -compilado e otimizado para melhorar o desempenho e a efici ncia da aplica o.

Em resumo, o React Native permite que os desenvolvedores criem **aplica es m veis multiplataforma** utilizando JavaScript e componentes nativos reutiliz veis.   usado um **bridge de comunica o** para interagir com os recursos nativos do dispositivo e um mecanismo de **renderiza o din mica** para atualizar a interface do utilizador de forma eficiente. Ao empregar uma abordagem de compila o ahead-of-time, o React Native oferece um desempenho e uma experi ncia de utilizador compar veis  s aplica es desenvolvidas nativamente, ao mesmo tempo que permite a partilha de c digo entre plataformas.

3 Vantagens e Desvantagens face à competição

O React Native emerge como a escolha preferida entre as frameworks de desenvolvimento de aplicações móveis, conforme um estudo realizado pelo Politécnico de Coimbra. Este estudo, conduzido através de diversos testes com programadores de diferentes níveis de conhecimento, revelou várias vantagens distintas desta plataforma.

Um dos pontos-chave destacados foi a **intuitividade** e **facilidade** do uso do React Native. A sua abordagem baseada em componentes simplifica o desenvolvimento, tornando-o acessível para programadores menos experientes. Além disso, a estrutura modular da framework, centrada em componentes reutilizáveis, contribui para uma arquitetura de código limpa e de fácil manutenção. Outra vantagem significativa é a **manipulação eficiente de listas**, facilitada pela estrutura do React Native que permite a manipulação de arrays. Isso simplifica a exibição e a gestão de dados, tornando o desenvolvimento de interfaces dinâmicas mais ágil e intuitivo.

O React Native também se destaca pelo **sólido suporte para networking**, permitindo a integração de solicitações HTTP de forma direta e eficaz. Isso simplifica a comunicação com servidores e serviços externos, tornando o desenvolvimento de aplicações conectadas à internet mais fluído e eficiente.

No entanto, apesar das suas vantagens, o React Native não está isento de desafios. Os desenvolvedores podem enfrentar dificuldades ao controlar o código para garantir uma experiência consistente em diferentes plataformas móveis. A personalização da aparência e do comportamento da aplicação para atender às diretrizes específicas de cada plataforma também pode ser um obstáculo, especialmente para iniciantes.

Além disso, a **dependência de bibliotecas de terceiros** pode aumentar a complexidade do projeto e introduzir potenciais **problemas de compatibilidade**. Embora o React Native seja acessível para programadores com conhecimento prévio de JavaScript, pode haver uma curva de aprendizagem inicial para entender os conceitos específicos e as melhores práticas de desenvolvimento associadas a esta framework.

References

1. Fernando Fortunato de Lima. Avaliao de frameworks para o desenvolvimento de aplicaes h bridas. *Trabalho de Curso* (2019)
2. Hugo Brito, Anabela Gomes, Alvaro Santos, and Jorge Bernardino. Javascript in mobile applications: React native vs ionic vs nativescript vs native development. In *2018 13th Iberian conference on information systems and technologies (CISTI)*, (pages 1–6. IEEE, 2018)
3. Documentao oficial React Native, <https://reactnative.dev/docs/getting-started>, last accessed 2023/03/23
4. Native Modules Intro, <https://reactnative.dev/docs/native-modules-intro>, last accessed 2023/03/23
5. Top 10 Reasons to use Platform Native APIs, <https://blogs.remobjects.com/2013/03/14/top-10-reasons-to-use-platform-native-apis/>, last accessed 2023/03/23