

NativeScript Framework

48681 Rodrigo Alves e 48973 Diogo Mestre

Universidade de Évora, Colégio Luis António Verney, Portugal
<https://www.uevora.pt>

Resumo As aplicações móveis estão cada vez mais presentes nas nossas vidas. Desta forma, estas plataformas têm vindo a ser melhoradas para satisfazer as necessidades de cada pessoa. O desenvolvimento de aplicações por sua vez aumentou nos últimos tempos, e consequentemente as metodologias e técnicas tem vindo a ser aperfeiçoadas, ou seja, a utilização de frameworks como o *NativeScript* são muito úteis para a criação de aplicativos, pois possui ferramentas que tornam este processo de criação muito mais simples e prático. Neste documento, vamos descrever esta framework, as vantagens e as desvantagens da mesma no desenvolvimento de apps, o seu modo de funcionamento e uma comparação com outras frameworks concorrentes.

Keywords: NativeScript · Native API · Código Nativo.

1 Introdução

A criação de aplicações móveis pode ser por vezes um processo complicado, deste modo é necessário a utilização de frameworks que permitam facilitar o processo de desenvolvimento de programas a partir de ferramentas/métodos acessíveis ao utilizador, como resultado é possível desenvolver algo complexo de forma simples.

Desta forma, têm vindo a aparecer diferentes soluções que possibilitam a produção de aplicações móveis. O *NativeScript* é o exemplo de uma framework de código aberto, que foi disponibilizado sobre a licença de Apache 2 e suportado pela Telerik desde 2014, com o objetivo de construir aplicações móveis nativas com base em múltiplos sistemas operativos usando *JavaScript*, *TypeScript*, *Angular*, *Vue.js* ou *React*. Até ao momento foram lançadas 8 versões desta framework, sendo que a última inclui suporte para o chip M1 da Apple e novas funcionalidades, tais como suporte melhorado de acessibilidade e um novo plugin oficial para integração com Stripe.

2 Vantagens e Desvantagens

Como todas as frameworks de desenvolvimento de aplicativos móveis, o *NativeScript* possui vantagens e desvantagens. Algumas vantagens são: permite aos programadores desenvolver aplicações tanto para iOS como para Android, usando uma única base de código; permite aceder a diversas APIs de dispositivos nativos, de forma a enriquecer as aplicações e torná-las mais eficientes; uma vez que

utiliza tecnologias Web, torna-a mais fácil de manusear, pois os developers utilizam uma linguagem mais familiar; permite ainda ter uma interface nativa, ou seja, como os developers têm a possibilidade de criar as interfaces com base em XML e CSS, esta irá apresentar a aparência e o desempenho de uma aplicação nativa.

Apesar das suas vantagens, existem também algumas desvantagens na utilização do *NativeScript* para o desenvolvimento de aplicações móveis. Esta framework tem uma comunidade mais pequena em comparação com outras estruturas de desenvolvimento móvel, o que significa que há menos bibliotecas e plugins de terceiros disponíveis para os programadores utilizarem. Embora o *NativeScript* se baseie em tecnologias de desenvolvimento web, requer que os programadores aprendam novos conceitos e APIs específicos para o desenvolvimento móvel, o que pode ser um desafio para os principiantes. O *NativeScript* baseia-se fortemente no *JavaScript*, o que pode levar a problemas de desempenho em certos cenários, especialmente quando se lida com cálculos pesados ou grandes conjuntos de dados. Como as aplicações *NativeScript* incluem um tempo de execução JavaScript, o tamanho da aplicação resultante pode ser maior do que outras estruturas de desenvolvimento de aplicações nativas. Embora a biblioteca de widgets *NativeScript* UI seja personalizável, pode não oferecer o mesmo nível de flexibilidade de design que outras estruturas de desenvolvimento móvel, tais como *React Native* ou *Flutter*.

3 Modo de funcionamento/uso

O *NativeScript* converte o código em *JavaScript* para código nativo através de um processo de compilação chamado just-in-time (JIT). Durante este processo esta framework converte o código usando APIs nativas, que após a execução das mesmas geram o código nativo que é executado diretamente no dispositivo.

O código nativo é um código executável que é compilado diretamente para a arquitetura de hardware de uma plataforma específica, como iOS ou Android. Estas plataformas são escritas em linguagens de programação de baixo nível que estão definidas, como por exemplo o *Objective-C*, *Swift* ou *Java* para iOS e Kotlin ou *Java* para Android. A partir deste tipo de código existe um aproveitamento máximo dos recursos de hardware disponíveis em cada dispositivo, este também é otimizado para oferecer o melhor desempenho possível, o que é especialmente importante em dispositivos móveis, onde a velocidade e a eficiência são fatores essenciais. A execução é feita diretamente pelo processador do dispositivo, sem a necessidade de interpretar o código cada vez que é executado. Os criadores de aplicações móveis utilizam o código nativo para desenvolver as suas apps, pois estes são executados mais rapidamente, consomem menos recursos do sistema e oferecem uma melhor experiência ao utilizador. No entanto, escrever em código nativo pode ser mais complexo e demorado do que escrever em linguagens de programação de alto nível, como *JavaScript* ou *TypeScript*, e é aqui que entra a *API native*.

Uma *API native* consiste num conjunto de interfaces de programação de aplicativos que são fornecidas pelos sistemas operacionais dos dispositivos, permitindo aos criadores usarem recursos dos mesmos, tais como: câmara, localização, sensores e muitos outros. Estas APIs possibilitam que as apps se integrem perfeitamente com o sistema operacional, o que proporciona uma experiência mais fluida e robusta da mesma, assim concede aos criadores da aplicação um uso máximo dos recursos de hardware e software disponíveis no dispositivo, garantindo um desempenho rápido e eficiente. O *NativeScript* fornece assim uma camada de abstração que conecta o código JavaScript aos recursos nativos do dispositivo, o que leva a que os developers não precisem aprender uma nova linguagem de programação. O processo de conversão do código é efetuado de forma automática e não requer a escrita de nenhum código nativo diretamente.

Para desenvolver aplicações usando o *NativeScript* primeiro é necessário instalar a framework e configurar, no nosso caso, o Android Studio, de seguida podemos começar a desenvolver a aplicação usando *JavaScript*, *TypeScript* ou *Angular*. O *NativeScript* também fornece um conjunto de ferramentas executáveis na linha de comando para certas tarefas relacionadas com a aplicação.

4 *NativeScript vs React Native vs Ionic*

Após uma análise comparativa entre o *NativeScript*, o React Native e o Ionic realizada através de vários testes a certos programadores com diversos graus de conhecimento na matéria (desenvolvimento de vistas, comunicação entre vistas e lógica de negócio, listagem de dados, networking e diferenciação entre plataformas), feita pelo politécnico de Coimbra, verificou-se que o React Native foi a framework com mais vantagens para desenvolvimento de aplicações móveis. Neste estudo todos os programadores possuíam as bases de *JavaScript*, sendo que o *NativeScript* ficou um pouco abaixo das expectativas, uma vez que a maioria dos programadores apesar de entenderem o necessário para resolver os problemas, perdiam-se no fluxo de desenvolvimento da aplicação e acabavam por cometer alguns erros, o que levou a considerar que esta framework apresenta uma ligeira dificuldade na construção das vistas, que consiste numa mistura de XML com CSS.

Assim neste estudo verificou-se que o *React Native* é bastante intuitivo e fácil de usar, não só por as listas serem baseadas em arrays, mas também pelo facto do desenvolvimento da framework ser totalmente baseado em componentes. O *Ionic* apesar de possuir CSS puro e ainda por ter um desenvolvimento quase igual ao AngularJS para web, os developers apresentaram dificuldades em controlar o código consoante a plataforma móvel. O *NativeScript* teve um feedback positivo relativamente aos requests http serem feitos tal e qual como o *JavaScript ES5*, mas houve uma grande maioria de opiniões negativas relativas à criação de vistas em XML com recurso a Grids, que acabou por se tornar um tipo de implementação muito pobre de tão confusa que se apresentava, tendo dificuldade em controlar a lógica entre uma vista e um controlador e em como chamar funções entre vistas e controladores.

Em suma, o *NativeScript* não é uma escolha popular entre os programadores, apesar de parecer ser uma framework mais intuitiva e fácil de entender, uma vez que utiliza uma linguagem mais familiar.

Referências

1. Hugo Brito, Anabela Gomes, Álvaro 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.
2. Fernando Fortunato de Lima. Avaliação de frameworks para o desenvolvimento de aplicações híbridas. *Trabalho de Curso*, 2019.
3. Hugo Brito, Álvaro Santos, Jorge Bernardino, and Anabela Gomes. Learning analysis of mobile javascript frameworks. In *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6. IEEE, 2019.
4. Developers NativeScript. Nativescript. <https://docs.nativescript.org/>. Accessed on 2023-04-12.
5. Jim. Top 10 reasons to use platform native apis. <https://blogs.remobjects.com/2013/03/14/top-10-reasons-to-use-platform-native-apis/>. Accessed on 2023-04-12.
6. Wikipedia. Windows native api. https://en.wikipedia.org/wiki/Windows_Native_API. Accessed on 2023-04-12.
7. Margaret Rouse. Native code. <https://www.techopedia.com/definition/3846/native-code>. Accessed on 2023-04-12.
8. Malcolm Tatum. What is native code? <https://www.easytechjunkie.com/what-is-native-code.htm>. Accessed on 2023-04-12.
9. Imaginedone. Aplicativo nativo: entenda o que é e conheça as principais vantagens para empresas. <https://imaginedone.com.br/blog/web-e-mobile/aplicativo-nativo/>. Accessed on 2023-04-12.
10. Arline. Código nativo. <https://definirtec.com/codigo-nativo>. Accessed on 2023-04-12.
11. stackoverflow. O que seria algo "nativo"? <https://pt.stackoverflow.com/questions/458760/o-que-seria-algo-nativo>. Accessed on 2023-04-12.