Dis

Discussion 6.1

Inquiry: Mobile Development

By Jonathan Alvarenga

CSC256

Discussion 6.1: Inquiry: Mobile Development

Using JavaScript in mobile development can provide some benefits but also some negatives. In mobile development Kotlin and Java are dominant for Android phones, while Swift is primarily for IOS. For cross-platform developments Dart and JavaScript are popular choices offering a single codebase for multiple platforms. I’ll explore the benefits and disadvantages of using JavaScript in mobile development (Blair, 2018).

JavaScript has an advantage of being a cross-platform codebase. It can be used with both IOS and Android software which are the 2 most popular phone software. This has the advantage of using both markets to produce content for (FireArt). You don’t have to only develop for one side. This can have a faster development cycle with its existing web development tools and rich ecosystem of libraries and tools to speed development of any projects. The code is also sharable across web applications. Meaning it can have a web version as well as a mobile version. It contains one of the largest communities with lots of shared documentation, support, and a wide array of open-source resources. It can be cost-effective by utilizing a single development team and codebase for multiple platforms, optimizing resource allocation. JavaScript can also provide a near-native performance providing smooth user experience. JavaScript create engaging and interactive user interfaces with dynamic content updates, animations, and responsive designs. JavaScript has a lot of advantages, but there are reasons why multiple codes exist to compensate for the disadvantages JavaScript can’t do.

JavaScript can be smooth in performance but can have bottleneck performance issues. JavaScript-based apps often rely on intermediary layers to interact with native functionalities and render UI. JavaScript has issues performing these functions and can lead to slower performance and less responsiveness compared to natively built applications, particularly for complex or data-intensive applications (R, 2018). JavaScript can also have limited access to native features. While frameworks strive to provide access to native APIs, some platform-specific features or the latest APIs might not be immediately available or may require custom native code integration, diminishing the cross-platform advantage. Sometimes the code can give suboptimal user experience. Apps built with JavaScript may struggle to deliver a polished look and feel like a truly native application. Making users have a worse experience in the long run. JavaScript structure depends on constant development, maintenance, updates, and security strength. Debugging can also be a challenge due to its multi-platform advantage. When one side has bugs you have to find different solutions than you would on the other side. JavaScript can also be very storage heavy. It can contain a significant amount of code and resources resulting in a large size compared to others.

Using JavaScript has advantages and disadvantages. It should be used in case-by-case projects. To some it can be simple and easier to use for development, but in certain projects it can hurt development. Only use it if applicable to the end product.

Bibliography

1. Fireart. (2022, January 5). Top Most Popular Programming Languages for Mobile App Development. Fireart Studio. https://fireart.studio/blog/top-most-popular-programming-languages-for-mobile-app-development/
2. Blair, I. (2018, August 10). 14 Programming Languages for Mobile App Development - BuildFire. BuildFire. https://buildfire.com/programming-languages-for-mobile-app-development/
3. R, G. (2018, December 19). Top Mobile Development Technologies Used for Mobile Apps | Fingent Blog. Fingent Technology. https://www.fingent.com/blog/top-technologies-used-to-develop-mobile-app/