Riabova Yelyzaveta, master's student
Vakaliuk Tetiana, Dr Sc., professor
Zhytomyr Polytechnic State University

APPROACHES TO DEVELOPING CROSS-PLATFORM MOBILE APPLICATIONS: A COMPARATIVE ANALYSIS OF FLUTTER, REACT NATIVE AND XAMARIN

Cross-platform mobile development is increasingly important due to rapid technological advancements and the need for efficient resource use. Traditional methods require separate app versions for each operating system, while cross-platform development uses a single codebase for Android, iOS, and more, reducing costs and expanding reach.

Popular frameworks include Flutter, React Native, and Xamarin, each with unique features, benefits, and limitations suited to different project needs. This study compares these frameworks, focusing on development ease, performance, and cross-platform support.

Flutter, by Google, uses Dart and native code compilation for high performance. Its custom widgets enable flexible, native-like UIs. It supports Android, iOS, web, and desktop apps but has a steeper learning curve than JavaScript or C#. React Native, by Facebook, uses JavaScript and React, with extensive libraries and community support. It integrates with native components, making it ideal for MVPs and simpler apps, though its performance may lag for complex projects. Xamarin, by Microsoft, uses C# and .NET for high performance. It supports enterprise needs with Universal Windows Platform (UWP) compatibility but has a steeper learning curve for non-C# developers. Each framework—Flutter, React Native, and Xamarin—suits different needs. Flutter excels in performance and UI flexibility, React Native in rapid development and community support, and Xamarin in .NET power and Microsoft integration.

When choosing a cross-platform framework for mobile app development, key criteria include performance, ease of learning, UI capabilities, platform support, and development time. Below is a comparison of Flutter, React Native, and Xamarin based on these factors.

Table 1.

Criterion	Flutter	React Native	Xamarin
Performance	High with native code.	Depends on JavaScript.	High; delays in large apps.
Ease of Learning	Requires Dart; good docs.	Easy for JS/React users.	Easy for C# users, harder for begginers.
User Interface	Custom widgets; flexible.	Uses native components.	Native but less flexible.
Platform Support	Android, iOS, Web, more.	Android, iOS, Windows.	Android, iOS, Windows.
Development Time	Fast with strong tools.	Fast with libraries.	Slower setup.
Native Integration	Strong native support.	Needs extra config.	Strong via Xamarin.Android.
Community	Active, growing.	Large, active.	Smaller, stable.
Best For	Custom designs, games.	MVPs, business apps.	Enterprise, Microsoft.

The analysis shows that Flutter and Xamarin offer high performance through native code compilation, making them ideal for resource-intensive apps. React Native, though versatile, lags due to its JavaScript-based architecture. Flutter excels in custom UIs with flexible widgets, while React Native and Xamarin use platform components for native-like designs. Flutter supports the most platforms, while Xamarin excels in the Microsoft ecosystem. React Native is easiest for JavaScript developers, while Flutter and Xamarin require knowledge of Dart and C#.

Framework selection depends on project goals, performance needs, speed, native feature integration, and target platforms. React Native is best for quick MVPs with its libraries and JavaScript ease. Flutter suits high-performance apps with complex graphics and multiple platform support, simplifying updates with a single codebase. Xamarin is ideal for enterprise projects requiring Microsoft integration, offering compatibility, stability, and security, perfect for financial or medical apps.

Looking ahead, Flutter will expand desktop and web support, React Native will improve performance, and Xamarin will deepen .NET integration. All three will continue evolving to meet market demands.

References

1. W. Wu, "React Native vs Flutter", Cross-platforms mobile application frameworks, 2018.
2. M. Uciński and M. Dzieńkowski, "A comparative analysis of the performance of Flutter and Xamarin development frameworks", Journal of Computer Sciences Institute, vol. 25.