

Comparison: Flutter vs React Native vs Ionic vs Native Development

Aspect	Flutter	React Native	Ionic	Native Development
Type	Cross-platform framework	Cross-platform framework	Hybrid framework	Platform-specific approach
Developed By	Google	Meta (Facebook)	Ionic Team	Google (Android), Apple (iOS)
Programming Language	Dart	JavaScript / TypeScript	HTML, CSS, JavaScript	Java/Kotlin (Android), Swift/Obj-C (iOS)
Codebase	Single codebase	Single codebase	Single web-based codebase	Separate codebase for each platform
Rendering Approach	Own rendering engine (no WebView)	Uses native components via bridge	Runs inside WebView	Direct OS rendering
Performance	Near-native / very high	Near-native (bridge overhead)	Low to medium	Highest
UI/UX Quality	Highly consistent and smooth	Native look & feel	Web-like UI	Best and fully native
Hardware Access	Wide access via plugins	Wide access via native modules	Limited access	Full access
Development Speed	Fast	Fast	Very fast	Slow
Learning Curve	Moderate (new language Dart)	Easy for JS developers	Easy for web developers	Steep (platform-specific)
Maintenance	Easy (single codebase)	Easy (single codebase)	Easy but limited	Difficult (multiple codebases)
Security	High	High	Moderate	Very high
Offline Support	Strong	Strong	Limited	Strong
Best Use Cases	Startups, scalable apps, UI-heavy apps	Social apps, dynamic apps	Simple content-based apps	Banking, gaming, system-level apps
Industry Usage	Growing rapidly	Widely adopted	Limited to simple apps	Enterprise & critical apps