



Hybrid Apps

Mike Hartington, GDE & Developer Advocate at Ionic
Gary Schultz, VP of Marketing & Sales at BrieBug



What is a Hybrid Application?

Hybrid applications are a blend of both native mobile (iOS / Android) and web technologies where the core of the application is written using web technologies (HTML, CSS, and JavaScript), which is then encapsulated within a mobile application.



Why Build or Evolve to a Hybrid App

- **Reuse Teams** - Time and money is saved by leveraging existing development team, avoiding time spent learning native iOS or Android platforms.
- **Faster Time to Market** - Advanced tooling such as Ionic enable rapid development of hybrid apps.
- **Consistency** - Developers can maintain a consistent feature set across native mobile and web platforms.



Common Best Practices

Hybrid Apps are not a one size fits all option.

- **Design for mobile, not the web** - UI/UX design and guidelines should follow that of a native mobile app rather than a website.
- **Play by platform rules** - Native platforms have Terms of Service that need to be followed.
- **Expect lower performance** - Be mindful that mobile devices have much lower processing speed and will not perform as fast as desktop computers.



Hybrid or Web Application?

Organizations often find that they can achieve their goals using a web app instead of a hybrid or native mobile application. This can also maximize profits by avoiding paying costly App Store/Google Play commissions.

Features that require a hybrid or native mobile app:

- **Bluetooth** - Apps that need manage or connect to bluetooth devices.
- **Camera** - Camera UX on native apps is far better than web apps.
- **Push Notifications** - Native mobile apps handle push notifications better than web apps.



Common Pitfalls

- **Going against the iOS/Android platforms** - Applications should closely adhere to design guidelines, no matter how “unique” they may be.
- **Building complex apps** - Simple is always better, work to simplify UI/UX and streamline feature - especially on mobile.
- **Rolling your own component library** - It is unnecessary to have code/support for older versions. There are mobile libraries that are being evolved and maintained by companies like Ionic.
- **Not Testing** - Build in tests to ensure your app is going to be easier to maintain and performant.



Getting Started

- **Decide if you need an app** - Determine whether or not your organization actually need native features in order to achieve your goals.
- **Pick the right tools** - Use frameworks and libraries that are built for hybrid apps. Companies like Ionic offer libraries of mobile-optimized UI components, gestures, and tools for building fast, highly interactive apps.

Take a look at documentation and get started at ionicframework.com

TRANSLATING TECHNOBABBLE

TRANSLATINGTECHNOBABBLE.COM