Platform Comparisons

**Web App**

Pros

* Updates itself, you do not have to download an update
* Deployment (web browser)
* Cost effective (development/maintenance)
* Good for reaching audiences
* Compatible across devices

Cons

* Limited number of native features
* Often slower
* Connection dependent

**Native App**

Pros

* No internet connection needed
* Can give user a rich experience
* Ideal for high-performance apps

Cons

* Multi User apps still require internet connection
* Maintainability for multiple apps is high

**Hybrid App**

Pros

* One codebase for multiple OS’s
* Can be placed in the app store
* Can leverage web development knowledge
* Reduced development cost
* Easy updates for web portion

Cons

* May be limited with platform components
* UX may be different from what user is used to
* Bad performance for high resource apps
* Middleware dependent
* Bug prone doe to updates
* May require updates to middleware

**Web applications:**

* Pivotal tracker
* Trello

**Native apps:**

* Asana

**Hybrid**

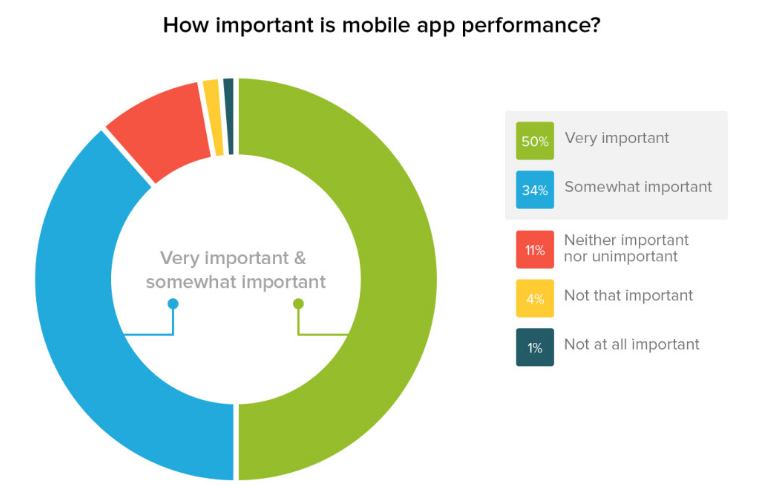
* Boot Camp

**Mobile App Frameworks**

* Ionic (AngularJS) - Well known, HTML5.
* React Native (React) - Good performance.
* Framework 7 - Close to native iOS apps.
* Onsen UI - Completes with Ionic.
* Intel XDK - Allows you to build apps for any platform on any platform

**Things to Consider**

* Do we want to use native features in the Mobile App? For example, do we have features now or in the future that would require native hardware functionality.
* Turnaround time for releasing a native app considering our timeframe during this semester. With proper resources and budget, then native apps are the way to go.
* For the best user experience, we need to consider that native is the way to go. Not that a hybrid approach would be bad.
* Poor mobile app experience is likely to discourage users from using an app again.
* *Do you want to astound and entice your users by building an entirely native application that integrates into the platform of their choice (Android or iOS)? Or are you more interested in taking a Minimum Viable Product approach and quickly developing a hybrid application which can be released across platforms? Though potentially easier to build and maintain, this second strategy is likely to result in a less than ideal user experience with sub-par performance.*
* The vast majority of users preferred mobile apps over mobile websites. But slightly more than half of the users have experience a problem with a mobile app, i.e. crashes, slow launch time, or simply would not launch.



https://info.dynatrace.com/rs/compuware/images/Mobile\_App\_Survey\_Report.pdf