**Which PHP Framework to use:**

One way to show CakePHP on all mobile devices is to build native apps. This forces users to stop and download your app. In order to be downloaded, you also need to be one of the few apps that users have on their screens. Without having the app on your homescreen, you cannot access the “real” mobile experience..

Another way is to have a mobile website, a responsive website, for mobile devices. They are slower to load than native apps, limited in accessing smartphone APIs and do not work offline.

A way to get the best of both worlds is to build a Progressive Web App. This also avoids having to write code specific to Android and to IoS.

Extra features with PWAs:

* Push Notifications
* Offline Work Mode
* Option to have Launch Icon
* Launch in Full Screen

You already know CakePHP. There seems to be a limit to showing CakePHP on mobile devices, certainly to show it on all mobile devices. With React, Vue or Angular you can extend your CakePHP environment to all devices. Good idea to build one application for all devices on PWA.

Need to choose a PWA Library.. Let’s go with Vue.js.

* Already know Vue.js : choice to do it through CDN..
  + Write event-driven application.. Easy to quickly update parts of page. Live pages when change in data. (example live results) (integrate Vue.js CDN on pages that need live results).
  + PWA advantages..
  + Vue StoreFront.. (E-Commerce)

**To Do:** Can use Vue.js for live pages. Still need to use Vue.js PWA capabilities.

**Advantages PWA to Responsive Website:**

Extra features with PWAs:

* Push Notifications
* Offline Work Mode
* Option to have Launch Icon : add Shortcut to Home Screen (A2HS)
* Launch in Full Screen

Use CakePHP as API Back-End..

<https://stackoverflow.com/questions/60335298/cakephp-3-to-progressive-web-app-back-end/60341051#60341051>