Types Of Mobile Apps – Part 1

In this lesson, I'll talk about the two different types of mobile apps that are the native apps and the hybrid apps.

We'll cover the following

Just for clarity, when I say mobile apps, I mean the apps that we download from the app stores like the Google Play Store & install on our mobile.

There are two types of mobile apps – *Native* & *Hybrid*. In this lesson, we'll find out what they are & what are the technologies, popular in the developer circles, that are required to build those apps. In the subsequent lessons, we'll also discuss things like:

- Why is it so important for developers to pick the right type of app to implement their use case?
- Why do we need different types of mobile apps? What are the pain points these app types are trying to solve?
- Which app type, *hybrid* or *native*, will suit best for my use case? We'll discuss this with some real-life examples.

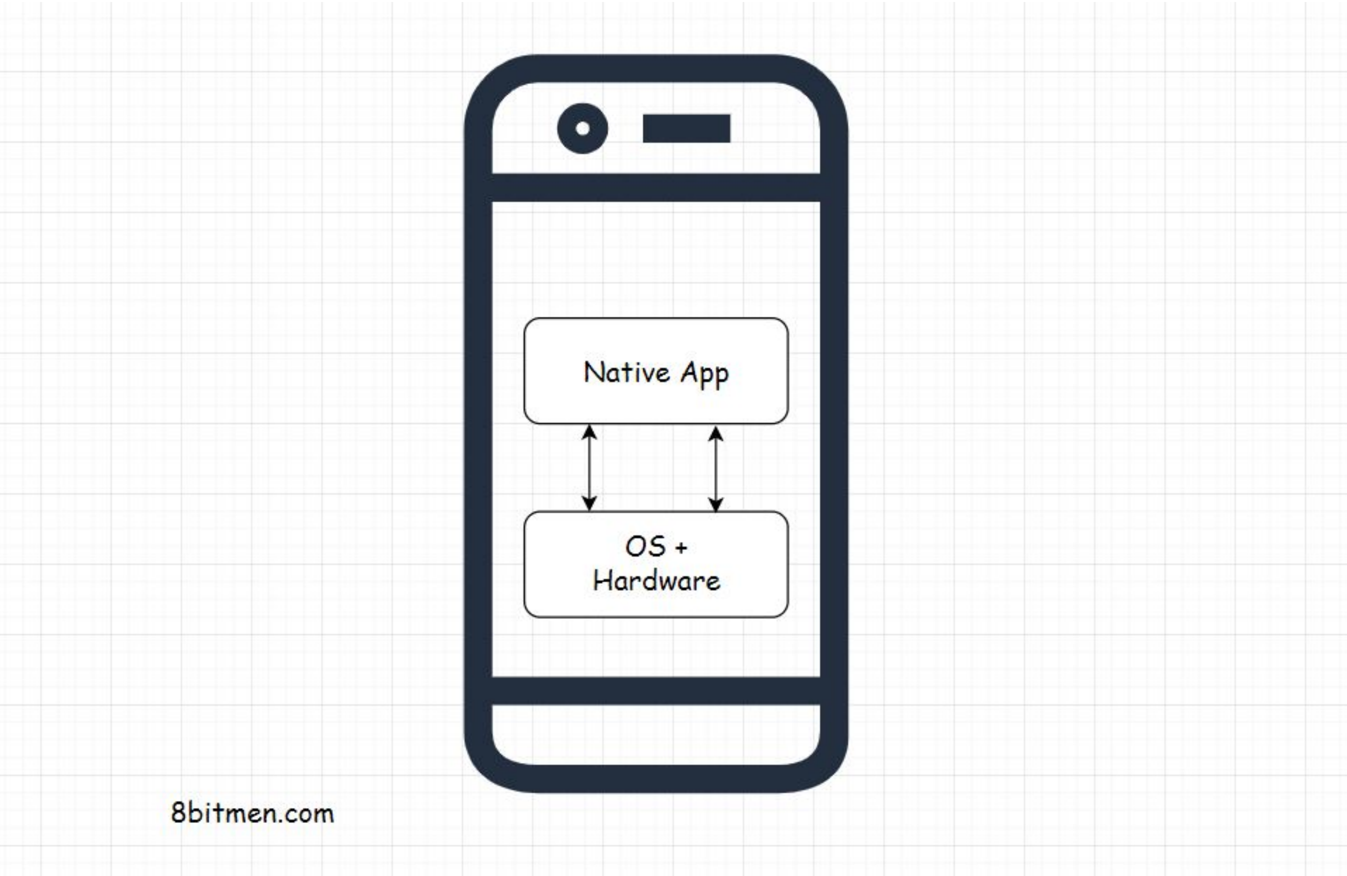
So, without further ado. Let's get on with it.

Native App

Native apps are apps that are built exclusively for a particular operating system be it the *Android*, *iOS* or a *Windows-based* handheld device. These apps function only on the OS they are built for. For instance, an app that is built for Android OS will not work on Apple OS.

Native apps interact directly with the operating system and the device hardware as opposed to communicating with it via a wrapper, an adapter or a middle layer. For this reason, they have full access to the device hardware like camera, sensors & stuff.

These apps provide high performance, they have a consistent user interface & provide the look and feel of the native OS.



Native apps don't face any lag issues when rendering UI animations like the slider movement, hiding and display of UI elements etc. With these apps, the UI is pretty responsive, that means when the user clicks on the UI, he can immediately see the change as opposed to seeing the change after a bit of a lag.

Native apps are developed using the APIs and the SDKs provided by the native OS. Some of the examples of native apps are the android apps of LinkedIn, Tinder & Hike.

Technologies For Writing Native Apps

Every mobile OS supports a certain set of technologies for writing an app that would run on that particular OS. For instance, if you want to build an app that would run on Android OS, you can use *Java, Kotlin* or *C++*. This official Android developer documentation is a good place to start for the android application development.

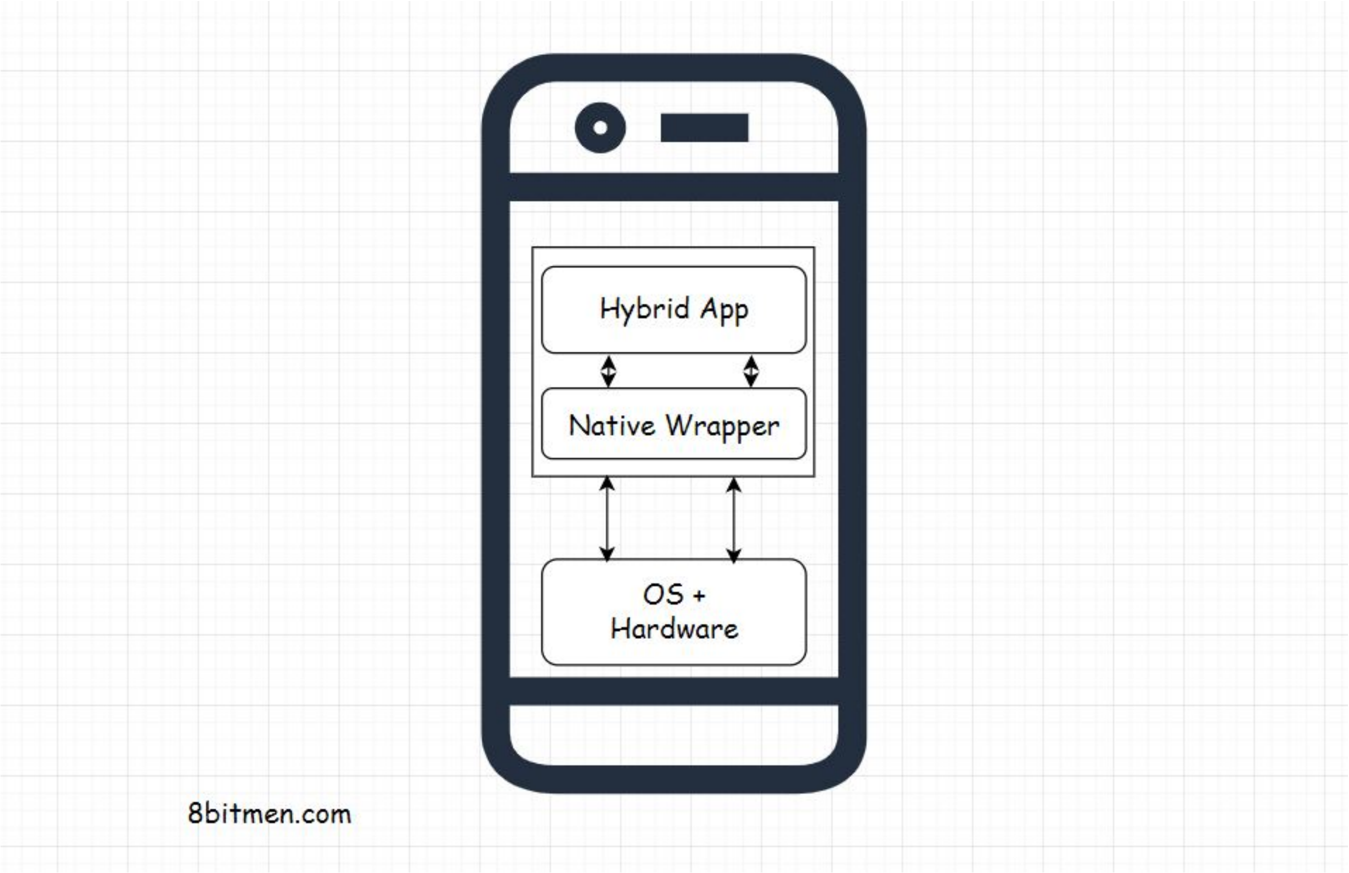
Likewise, for writing native apps for iOS you can use *Swift, Objective C* along with the *Cocoa Touch* framework. To write iOS apps, this apple developer documentation is a good place to start.

Just like this, every respective mobile OS supports a different set of technologies to enable developers to build apps for its platform.

Hybrid App

Hybrid apps as the name implies are a hybrid between the native and the web-based technologies. Just like native apps, they can be installed from the app stores on the mobile and can access the hardware & communicate with the OS of the device.

Hybrid apps are primarily built using open web-based technologies such as *Html5, CSS, JavaScript*. They run in a native-container and communicate with the native OS via a wrapper or a middle layer. This middle layer enables the open web technologies to talk to the native OS. Now, because of this additional middle layer which native apps don’t have, hybrid apps are a bit slower than the native apps when it comes to performance & the rendering of the UI.



There are several popular frameworks available to write hybrid apps such as *React-Native, Ionic, Cordova* etc. Let's discuss this up next.

Technologies For Writing Hybrid Apps

Below are a few popular technologies available to us for developing hybrid mobile apps.

React Native

React Native is an open-source mobile application development framework, written in *JavaScript*, developed by Facebook. With it, we can develop applications for multiple platforms like Android, iOS, Windows etc.

Before releasing the framework, Facebook was already using it for its ad manager, analytics & the group app. React-Native is a pretty popular framework for writing hybrid apps. In 2018, it had the highest number of contributors for any repository on GitHub.

Some of the companies using React-Native for their mobile apps are Bloomberg, Walmart, Uber Eats, Discord.

Apache Cordova - PhoneGap

Apache Cordova is an open-source hybrid mobile application development framework released by Adobe. The framework enables the developers to build mobile apps for Android, Windows, iOS, using *Html, JavaScript, CSS*.

There are several ecosystems and frameworks built on top of Cordova like *Ionic Framework, PhoneGap* etc. PhoneGap is Adobe's commercial version of Cordova. Besides the open Cordova framework, Adobe provides several tools, an entire ecosystem, to facilitate mobile development via PhoneGap.

Here is a list of apps developed using PhoneGap.

Ionic Framework

Ionic is an open-source SDK for writing hybrid mobile apps built on top of *Apache Cordova* and *Angular JS*. Here are some of the companies developing their apps using the Ionic Framework.

Flutter

Flutter is an open-source hybrid mobile application SDK by Google. It can be leveraged to develop applications for platforms like Android, iOS, Windows, Mac, Linux, Google Fuchsia & the web. Some of the apps developed using Flutter are:

Square, Google Assistant.

For a full list of apps developed using Flutter here you go.

This is a good Wikipedia resource that lists out the various mobile app development tools, SDKs & platforms for writing mobile apps.

So, these are some of the popular technologies used by the industry to write hybrid apps. Let's continue this discussion on hybrid and native apps in the next lesson.