Practical – 17

Aim : Introduction to React – Native.

What is React Native?

React Native (also known as RN) is a popular JavaScript-based mobile app framework that allows you to build natively-rendered mobile apps for iOS and Android. The framework lets you create an application for various platforms by using the same codebase.

React Native was first released by Facebook as an open-source project in 2015. In just a couple of years, it became one of the top solutions used for mobile development. React Native development is used to power some of the world’s leading mobile apps, including Instagram, Facebook, and Skype. We discuss these and other examples of React Native-powered apps further in this post.

How does React Native work?

Now that we’ve discussed cross-platform development, let’s take a look at the mechanics of React Native, and how it’s different from anything we’ve seen before.

Don’t worry if you’re not a technical person – we’re going to explain this in layman terms.
As mentioned earlier, React Native is written with a mixture of JavaScript and JXL, a special markup code resemblant of XML. The framework has the ability to communicate with both realms – JavaScript-based threads and existent, native app threads.

How does this communication work? React Native uses a so-called “bridge”. While JavaScript and Native threads are written in completely different languages, it’s the bridge feature that makes bidirectional communication possible.

What makes React Native unique?

The difference between React Native and other cross-platform development solutions (for example, Cordova and PhoneGap) is that React Native doesn’t render WebViews in its code. It runs on actual, native views and components. This is one of the reasons for React Native’s spectacular success.

Benefits of Using React Native for Mobile App Development :

1. Code Reusability

The biggest advantage of React Native is that developers don’t need to create separate codes for different platforms (Android and iOS). In fact, around 90%[3] of the code can be reused between the two platforms which help increase development speed and efficiency considerably. As a result, you get faster time-to-market and require lesser maintenance efforts.

2. Native Look and Feel

React Native components map 1:1 with native development components. It combines the building blocks from the native user interface with its own JavaScript which gives the app a native-like appearance. Moreover, as the building blocks are the same for Android and iOS, the look and feel of the app is also similar across the two platforms.

3. Live Reload

The live reload feature of React Native allows you to see and work with changes in real-time. You can make fixes in the code while the app is loading and it will be reflected in the app with an automatic reload. You can also reload a particular area of change to save time on the compilation.

4. UI Focused

React Native uses the React JavaScript library to build app interfaces that are fast and responsive. It has great rendering abilities and uses a component-based approach which makes it easy to create both simple and complex UI designs.

5. Cost-Efficiency

As the code is reusable in React Native, it helps you save development costs upto 40%[4]. You don’t have to hire two different Android and iOS dev teams to create the app. On top of it, there are many pre-built components in React Native that further fasten the development process.

6. Third-Party Plugins

Since building an app from scratch can be expensive, React Native offers a number of third-party plugins options including JavaScript-based and native modules. The third-party plugins eliminate the need for specific web view functions and help enhance the app’s functionality and performance.

7. Large Community Support

The development of React Native actually began as a response to the need of the developer community. As a result, it has a community-driven approach with over 50,000[5] active contributors. The engineers at Facebook are always developing and updating the platform and the availability of such enthusiasts makes it easier to get expert support.

Disadvantages of Using React Native for Mobile App Development :

1) **Performance is still lower than native –** React Native is not able to use all the blessings and potential of a specific platform. On the other hand, a native app can really maximize functionalities and deliver the ultimate user experience in the result.

   However, the re-architecture puts a lot of effort to make RN more performant, e.g. there is a new JS engine available – Hermes, and it makes apps running much faster on older Android devices.

2) **Not efficient for complex interfaces –** If you consider complicated designs or advanced interactions as a crucial part of your business advantage, you should definitely go for native development.

3) **Lack of some custom modules –** many custom modules are already available, yet, there may be a need for some specific components that you will be forced to build from scratch yourself.

4) **Updating issues –** it's hard to keep the app updated with the latest React Native version. Updating React Native versions is in most cases a complicated process.