



ASSIGNMENT 01

Name: Fatima Jabeen

Course: MAD Theory

Reg.no: SP20-BCS-002

Submitted To: Mr. M. Kamran

Dated: 30/09/2022

1. Comparison between the Native and Cross Platform Mobile application development:

Native Mobile App Development	Cross Platform Mobile App Development
<p>The process of developing a mobile app solely for a single native platform or type of device, typically Android or iOS, is known as native app development.</p> <p>An Android app and an iOS app would never work together. This is because distinct platforms call for distinct programming languages and development technologies. To build an iOS app, developers might use AppCode, Xcode, or Atom, but to make an Android app, they would need to use Android Studio. Programming languages like Swift and Objective-C are needed for iOS apps, whereas Java and Kotlin are used for Android apps. Simply put, this means that if you want your app to work on a variety of platforms and devices, you'll need different versions made by your developers.</p> <p>Native app development is a very popular option for many individuals and businesses, even though creating multiple apps may appear to be a disadvantage. This is due to a lot of good reasons.</p>	<p>Apps that are made to work on multiple platforms are called cross-platform apps. Because cross-platform apps use the same code for both Android and iOS, your developer will only need to create one version for the app to function on all platforms.</p> <p>For cross-platform app development, common programming languages include JavaScript, C#, and Dart. Technologies used by developers include React Native, Xamarin, and Flutter. A comprehensive comparison of Flutter and React Native can be found on our blog.</p>

2. Different scenarios where each native and cross-platform mobile application development is preferred.

Native Mobile App Development

Benefits of Native App Development's Wide Range of Functionality:

Every API and tool offered by the platform you're working on will be available to you. Technically, the programmers can work with the new app in any way they want.

1. Better Customer Service:

Because it provides better performance and speed, native apps are typically ranked higher on the platform's app store and are easier to publish.

2. Scalability Boosted:

Due to the availability of a wide variety of tools and the adaptability with which resources can be managed, native applications are also typically more scalable. High performance is achieved by the code's direct interaction with the underlying resources. Additionally, native apps typically provide a superior user experience that is synonymous with the platform.

Cons of Native App Development's Wide Range of Functionality:

1. Costly:

When you need to launch native apps for both iOS and Android, it can be expensive to build them. This means that two teams working on different platforms will need to be deployed.

2. Consuming Time:

Because the work done for one platform cannot be duplicated for another, developing native apps takes a long time. The alternative version requires the work of a separate team.

Cross-Platform Mobile App Development

Benefits of Lower-Cost Cross-Platform	Cons of Lower-Cost Cross-Platform Application Development:
<p>1. Application Development: To develop a cross-platform application, you will only need one developer team as opposed to two. As a result, you will save money on development costs.</p> <p>2. Accelerated Development: An app that works on multiple platforms can be made in a single development cycle.</p> <p>3. Single Coding System: A single code base is created because the app is developed using a single cross-platform development tool.</p>	<p>1. Slower Software: The cross-platform app is slower than its native counterpart due to the need for an additional abstraction layer and rendering process.</p> <p>2. Functionality is limited: It may be difficult for developers to access smartphone features like the microphone, camera, and geolocation in ways that a native app could.</p> <p>3. UX is limited: Native UX components are unavailable to cross-platform applications. As a result, it is unable to provide the same user experience as the platform.</p>

3. List of frameworks/Tech Stack for cross-platform mobile application development.

Ionic. Based on AngularJS, Ionic is one of the most impressive and well-liked cross-platform app frameworks.

- » React Native.
- » Flutter.
- » Xamarin.
- » Native Script.
- » Node
- » Appcelerator Titanium
- » PhoneGap.