**Name Umair Khan**

**Reg SP20-BCS-011**

**Date 2/10/2022**

**Mad Theory Assignment # 1**

1. **Comparison of Native and Cross Platform Mobile App development:**

What does it mean to develop native mobile apps? The term "Native mobile app development" refers to the process of creating an application specifically for Android or iOS. Which Operating System you are creating for will determine this. Java and Kotlin are the programming languages we use to create native Android apps, and Swift or Objective-C are the languages we use to create native iOS apps.

**Pros and Cons of Native Mobile App development.**

**Pros:**

* Better performance: You design and enhance native apps for a certain platform. You also compile them using the APIs and primary programming language of the platform. They become much quicker, more effective, and responsive to user’s actions.
* Security: Data security for your users can be improved with native apps. They have access to built-in security mechanisms that are specific to the platform.
* Quality UX: Native apps have a smoother experience as compared to user input and output. The look and feel are consistent because they inherit the device’s OS interfaces.
* Minimal Bugs: Native app developers gain access to updated SDKs as soon as they are released. These updates always ship with improvements and new features.

**Cons:**

* Cost: Deploying native mobile applications is frequently expensive. For the various platforms, you must have distinct development teams. To construct a native Android app, for instance, you require developers skilled in Java or Kotlin.
* Development Time: Separate teams working on similar apps for separate platforms take time. Maintenance and enhancements also require a lot of effort.
* Lack of code reusability: Create and keep code in separate projects to target different mobile operating systems.

What is cross-platform mobile app development? Cross-platform development means that we can create mobile application for both android and IOS operating system from one codebase. Cross-platform software development aims to target many operating platforms with a single project. React Native, flutter and Xamarin are cross-platform examples.

**Pros and Cons of Cross-platform Mobile App development**

**Pros:**

* Low cost: Only one team of developers is required for cross-platform mobile development. They must be informed about the selected framework.
* Code reusability: Cross-platform frameworks let you employ a single codebase, which increases code reuse. This guarantees consistency in the applied business logic.
* Rapid development: You can reach the market more quickly by increasing productivity and reusing code. Cross-platform frameworks have also improved the way that apps are tested as they are being developed.
* Maintenance: Updating a cross-platform is simpler due to easier maintenance. You are working with a single codebase, which is why. Using the same code across all builds for the various platforms ensures consistency.

**Cons:**

* BIG digital footprint: Cross-platform apps typically have a bigger digital footprint. The alternatives for mobile optimization must be taken into consideration.
* Integration challenges: Not all platform-specific features are integrated by cross-platform frameworks. Some hardware integrations, such as the use of GPUs, may need knowledge of native app development.
* Lower performance: Custom runtimes are frequently bundled with cross-platform frameworks to execute programme. It is in charge of interacting with services that are specialized to a given platform. This increases the amount of computation, which lowers performance.
* Platform updates: New SDK versions frequently come with new updates and functionality. For access to these more recent capabilities, you must wait for a separate upgrade with cross-platform frameworks.

1. **Scenarios where each Native and Cross-platform is preferred:**

The choice of whether to create a native or cross-platform app is essential. There are many situations when native apps are preferable to cross-platform ones, and vice versa. Making decisions will be easier if you carefully evaluate your audience, budget, development schedule, performance and security requirements, and other factors. Few factors which you should consider before developing.

1. **Security**

You must take into account the company's reputation as well as the effects of losing the trust of your customers. For some mobile application types, like a financial app, the dangers can be very significant. Native mobile development would be a superior option in these situations. It offers a wide range of built-in security measures, such as intelligent fraud detection using certain OS libraries and file encryption. Native apps offer higher security, stability, and scalability, even while cross-platform mobile development is perfect for getting the app to market more quickly.

1. **Performance**

Particularly for games, mobile apps might have demanding computational requirements. The best option for apps that require enhanced performance is frequently native mobile development. In this case, using cross-platform frameworks would demand additional labor and knowledge of native apps. You can make sure your programmed functions as effectively and efficiently as possible by optimizing performance for a particular operating system.

1. **Development Costs**

Some businesses can spend more money than others on creating mobile apps. Cross-platform apps are better for cheaper budgets because they require a smaller team to develop. By sharing your code and projects, cross-platform development also enables you to control costs.

1. **List of frameworks for cross-platform:**

The most popular frameworks for cross-platform are:

* Flutter
* React native
* Kotlin
* Ionic
* Xamarin
* Native Script