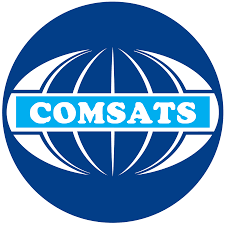
**Assignment No 01**

**MAD-Theory**

****

**Submitted By:**

**AIMAN LIAQAT**

**SP20-BSE-004**

**Submitted to:**

**Sir Mohammad Kamran**

**Date:02-oct-2022**

**Comsats University Islamabad, Attock Cantt**

Contents

[**Native And Cross Platform Mobile Development:** 2](#_Toc115471783)

[**Native Platform Mobile Development:** 2](#_Toc115471784)

[**Cross Platform Mobile Development:** 2](#_Toc115471785)

[ **Comparison Between Native And Cross Platform Mobile Development:** 2](#_Toc115471786)

[ **Scenarios where each native and cross-platform mobile app development is preferred.** 3](#_Toc115471787)

[ **Time to market** 3](#_Toc115471788)

[ **Security** 3](#_Toc115471789)

[ **Performance** 3](#_Toc115471790)

[ **Development costs** 3](#_Toc115471791)

[**Conclusion** 4](#_Toc115471792)

[ **List of frameworks/Tech Stack for cross-platform mobile Application development.** 4](#_Toc115471793)

# 

# **Native And Cross Platform Mobile Development:**

## **Native Platform Mobile Development:**

 The mobile application is designed exclusively for one platform. The application is built using programming languages ​​and tools that are specific to one platform. For example, you can develop a native Android app using Java or Kotlin and choose Swift and Objective-C for iOS apps

## **Cross Platform Mobile Development:**

cross-platform is a type of software that can run on multiple computer platforms i.e. Android, iOS, Windows, Blackberry, etc.

# **Comparison Between Native And Cross Platform Mobile Development:**

|  |  |  |
| --- | --- | --- |
| Criteria | Native | Cross\_platfrom |
| Architecture | Different Apps for different platforms | One app for multiple platforms |
| performance | Full native and higher | Native like and performance issues |
| Development time | Requires time as needed to write different code | Significantly used development time |
| cost | No Cost effects | Lower cost |
| User Experience | superior | Less reliable |
| Code reusability | Almost none | 60% used |
| Hardware accessibility | Complete hardware support | limited |

# **Scenarios where each native and cross-platform mobile app development is preferred.**

The decision to develop a native or cross-platform app is crucial. Various scenarios benefit native over cross-platform apps and vice-versa.

Let’s consider a few areas when determining which development method to use.

# **Time to market**

This is a widespread problem for startups and new product lines. You want to ship the product early to start getting quality feedback from users. Cross-platform application development would be best suited for this scenario, as they are quick to build and iterate. Mobile native development would be time-consuming and more difficult to maintain, ultimately slowing time to market.

## **Security**

You need to consider the reputation of the business and the impact that losing users’ trust would have. The risks can be very high for certain types of mobile applications, such as mobile banking apps. In these scenarios, native mobile development would be a better route. It provides many built-in security features, including file encryption and intelligent fraud detection using specific OS libraries. While cross-platform mobile development is ideal for getting the app to market faster, native apps provide superior security, stability, and scalability.

## **Performance**

Mobile apps can have high computing requirements – especially for games. For applications that require accelerated performance, native mobile development is often best suited. Using cross-platform frameworks in this scenario would require additional effort and native application expertise. Optimizing performance for a specific operating system allows you to ensure that your application runs as efficiently and effectively as possible.

## **Development costs**

Some companies have higher budgets for mobile app development than others. For lower budgets, opt for cross-platform apps as you only need a small team. In addition, cross-platform development allows you to keep costs under control by reusing code and projects.

## **Conclusion**

To create a successful, stable, and well-received mobile app, you need to determine which operating system – or systems – you want your app to be compatible with. Although both native and cross-platform development methods can be used for Android and iOS operating systems, the tools, methods, advantages, and disadvantages accompanying each method must be considered when developing a mobile app.

# **List of frameworks/Tech Stack for cross-platform mobile Application development.**

Some of the best frameworks for cross-platform mobile application development are:

* Ionic
* React Native
* Flutter
* Xamarin
* Native Script
* Node. js
* Appcelerator Titanium
* PhoneGap
* Sencha Touch