****

**CEF 440: INTERNET AND MOBILE PROGRAMMING**

**GROUP MEMBERS:**

|  |  |
| --- | --- |
| **NAME** | **MATRICULE NUMBER** |
| FORMASIT CHIJOH FOKUNAG | FE20A041 |
| GAMUAH PADJINOU RYANE JOY | FE20A044 |
| MOSONGO ADINA SAKWE NANGERI | FE20A066 |
| NDALEGH NOELA LUM MBAH | FE20A070 |
| RACHEAL NKONGHO TAFONGOH | FE20A100 |

**COURSE INSTRUCTOR**

DR NKEMENI VALERY

**Date**: March, 31 2023

1. **WHAT ARE THE MAJOR TYPES OF MOBILE APPLICATIONS AND THEIR DIFFERENCES? (SPECIFY THE CRITERIA FOR COMPARISON).**

Mobile applications can be classified under two main categories:

• Based on the technology used.

• Based on the industry they work in.

1. BASED ON THE TECHNOLOGY USED. Three types of applications can be identified based on the technology used. They include:

* NATIVE APPLICATIONS
* WEB APPLICATIONS
* HYBRID APPLICATIONS

|  |  |  |  |
| --- | --- | --- | --- |
| FEATURES | WEB | NAITIVE | HYBRID |
| Access | Can be accessed through the web browser of a computer or any other device with internet connectivity. | Must be installed on a device and can be used offline. | May be installed on a device and or can be accessed via the web. |
| Functionality | Allow users to access the interactions managed by the web browser. Cannot access the functionality of the device. | Allows users to interact with the internal components of their devices as well as the operating system of their device. | Allows users to interact with the devices’ internal components. |
| User experience | Lacks coherent user experience due to their dependence on the browser. | Better user experience as the app can fully use the available resources of the device like Bluetooth, phonebook contacts, camera roll, NFC, send notifications, and even the device UI | Better user experience as the app can fully used the available resources of the device. |
| Performance | They are slower, and less reactive, but permit better control of performance. Updates are done automatically. | They are more performant in that they are more responsive, interactive, can be optimized, faster, more reliable and more efficient with the device’s resources. Nonetheless they require regular updates for optimal performance. | Faster but may consume more battery life. |
| App development | Less costly, simple and faster to build. They are also easier to maintain | They are more costly, and require more expertise to build its functionalities in the native language used for development. | Permits apps to be built on one code base. Core application is written using web technologies an encapsulated within a native application using Apache Cordova and Ionic’s Capacitor which uses plugging to access the device’s features |
| Availability | Lower availability due to the need for internet connection. | More available to customers especially those used offline. | More available to customers especially those used offline. |
| Internal working | The client code in the browser communicates with the code base of the distant server. | The client code is written in a language specific to the device or platform where it is installed. | The client code and the browser code are encapsulated in a shell or a local container. |
| Deployment | Less costly and fast deployment. | Deployment is costly and slower. | Less costly and fast deployment. |
| Programming language used | HTML5, JavaScript, CSS, Ruby. | Java, Python, C++, kotlin, React, Swift, Objective-C | Objective-c, HTML5, Ionic, Swift. |
| Examples | Digital Renter, Google drive, Google docs, Gmail | Facebook, telegram, safari, Google Maps, Spotify, Whatsapp, LinkedIn | Instagram, Netflix, Amazon, Go-student, Google Classroom, twitter |

**NATIVE APPLICATIONS**

* **Advantages**
* Faster, has better performance
* Uses the native device UI
* Can access device features
* **Disadvantages**
* High cost to maintain
* Takes up more device space
* Updates must be downloaded

**WEB APPLICATIONS**

* **Advantages**
* Operates on all devices
* Easier to maintain
* Do not take up space on the device
* **Disadvantages**
* Dependent on the browser
* Needs internet connection
* May not always integrate with device hardware

**HYBRID APPLICATIONS**

* **Advantages**
* Faster and cheaper to build
* Loads quickly
* Less code to maintain since it uses a single code base
* **Disadvantages**
* Lacks power of the native apps as such cannot access the features of the device
* Slower since it has to download each element
* Certain features might not be usable on devices
* Progressive web apps are extensions of web apps that can be saved on devices for quicker access. They use web browser APIs and the functionalities of native apps. Example: Twitter
* Cross platform apps are apps compatible with multiple platforms. Example: Firefox, chrome…

1. **BASED ON THE INDUSTRY THEY WORK IN**

* **Lifestyle apps** (Supports the individual facets that define your lifestyle). Example: Spotify
* **Social media apps** .Example: Face book, Instagram
* **Utility apps** (They are generally pre-installed on devices).Example: Calculator, flashlight
* **Gaming apps.** Examples: Angry birds, subway surfer.
* **Productivity** apps (Helps users accomplish a task quickly and efficiently).Example: Docs, Sheets, and Wallets.
* **News and Information**. Examples: Google news and weather.

1. **THOROUGH REVIEW OF ALL THE PROGRAMMING LANGUAGES USED FOR MOBILE PROGRAMMING**

**JavaScript**

It is a high-level, interpreted programming language that can support object-oriented and functional programming. It is a paradigm language run by a browser and is also used to develop web pages. Its frameworks are well-versed for mobile app development that can be used in any Android, iOS, and Windows platform. Its features include;

* Scripting: Executes the client-side script in the browser.
* Event handler: Provides options for event handling.
* Light weight. It is not a compiled language, source code never changes to byte code before run-time. Low-end devices can also run JS because of its light weight feature.
* Names, variables, keywords and functions are case sensitive. - Supports functional programming (writing functions…)
* It is platform independent-runs on multiple OS-and has improved processing time and responsiveness.

[**Scala**](https://www.scala-lang.org/)

If JavaScript is one of the most known, Scala is one of the newest programming languages available today. It is a modern, general-purpose programming language that has a reliable static type system. Its strongest asset, among others, is its multi-paradigm language.

Scala emerged as one of the most durable alternatives that can take in place of Java. It is designed to address the underlying struggles encountered by Java. It is interoperable with Java as it runs on Java Virtual Machine (JVM), making it one of Scala’s notable features.

**Java**

 JAVA is one of the most preferred languages by many mobile app developers. It is even the most searched programming language on different search engines. Java is an official Android development tool that can run in two different ways. It can be run either in a browser window or in a virtual machine.

**Java** is a general-purpose programming language that is [class-based](https://en.wikipedia.org/wiki/Class-based_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), and designed to have as few implementation dependencies as possible.

Though Java is not likely a top choice when you will be developing an iOS app, yet it can still provide well when working cross-platform apps. Some advantages includes;

* Its syntax is simple.
* Object-oriented language.
* Can run on multiple platforms/OS (Platform independent).
* It is secure. It is used in networked and distributed environments.
* It is robust and fault-tolerant.
* They are portable; do not need to be recompiled to run on another platform.

### [****Kotlin****](https://kotlinlang.org/)

Developers consider Kotlin as an advanced version of Java; it is an interoperable and versatile programming language that can easily overrule the shortcomings of Java.

Its language is clear, concise, and a statistically typed programming language. Working similarly like Java, it has several advanced features making it a better option.

What makes this option unique for your programming needs is that it supports almost all IDES. It is open-source and has extensive functions.

### [****Python****](https://www.python.org/)

Python is another widely used programming language used for any purpose. Developers used it to develop cross-platform applications in web, desktop, and mobile. It can also produce any Android and desktop applications from scratch.

Many use Python because of its ease of learning and reading. It is a language that is simple, readable, a bit complex but not too complicated.

Many popular applications like [Instagram](https://www.instagram.com/), [Dropbox](https://www.dropbox.com/?landing=dbv2), [YouTube](https://www.youtube.com/), [Reddit](https://www.reddit.com/), [Spotify](https://www.spotify.com/rs/), [Quora](https://www.quora.com/), and more were created in Python. Its features includes;

* Easy to code
* Easy to read –
* Free and open source
* Robust standard library
* Interpreted language
* Portable: The same code can be used on different machines.
* Object –oriented and procedure-oriented
* Extensible: Can be written in other languages.
* High-level language
* Have various areas of application: Game programming, AI, machine learning, data analytics just to name a few.

### [****PHP****](https://www.php.net/)

Most developers recommend PHP for mobile apps that require database access. It is an open-source programming language that is easy to learn, making it perfect for beginners with exceptionally smooth integration. It is highly favored for content-heavy apps. PHP loads quickly, even with a slow internet connection, and has built-in security features.

Hypertext Preprocessor or PHP was developed mainly for websites but, these days, it also serves in all-around development. It can now be used to build any Android and iOS applications. It is used as well for command-line scripting, server-side scripting, and coding applications.

### [****C#****](https://docs.microsoft.com/en-us/dotnet/csharp/)

A general-purpose programming language, [C#](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/), or pronounced as ‘C-sharp’ is fast, quick to deploy, and easy-to-use. It is a general-purpose programming language that is used in cross-platform, windows, and mobile app development. [C#](https://www.geeksforgeeks.org/csharp-programming-language/) is simple, strong typing, imperative, and one of the most preferred programming languages by most developers.

C# is made for Microsoft as it can produce durable Windows Phone applications. It is also supported by Unity3D, making it a great tool for building game applications. It is a versatile programming language that has been used to create applications in different sectors at the market.

### [****C++****](https://www.w3schools.com/cpp/cpp_intro.asp)

Many consider this programming language to be their top choice in mobile app development. C++ is used for Android app development as well as development of native apps. It is the best choice for those who are just starting to develop any mobile app as C++ has a built-in and massive pool of ready-to-use libraries for your app development.

[C++](https://www.geeksforgeeks.org/c-plus-plus/?ref=leftbar) is used in apps for industries like financial institutions, banking sectors, manufacturing businesses, and more. It has been in the market for quite a long time and has been loved by many developers for building mobile applications for iOS, Android, and Windows.

You can use C++ to create and develop various operating systems, game development, web browsers, medical applications, Enterprise tools, and more. It is a compiled language, simple, object-oriented, platform-dependent, popular, case-sensitive, powerful and fast, multithreaded language.

### [****Objective-C****](https://developer.apple.com/documentation/objectivec)

Objective-C is a primary programming language designed for and used by Apple app developers for its iOS and OS X operating systems. It is an object-oriented and general-purpose programming language that is easy to learn.

Though considered a bit old as many new programming languages are entering the market scene, Objective-C is still popular among developers.

[Objective-C](https://www.tutorialspoint.com/objective_c/index.htm) remained in the playing field even though there are newcomers. It is because many apps have already invested in the language in terms of security. Many still mostly rely on Objective-C even to this day.

### [****Swift****](https://swift.org/)

Swift is one of the latest open-source programming languages designed to work with iOS, OS X, and tvOS platforms. It requires little maintenance and avoids bugs of Objective-C.

Swift makes projects easier to scale. Its language is flexible and can add features to any app by adapting a programming pattern. Released mainly for iOS, iOS-supporting, and Linux, it has become the primary language developers lean on to develop iOS and OS X.

1. **MOBILE APP DEVELOPMENT FRAMEWORK?**

It is a software library that provides a fundamental structure to support the development of mobile applications for a specific environment or it is an application or tool that helps in mobile application development by providing templates and structure. EG Flutter, React Native, Xamarin, Native Script, JQuery Mobile.

There are three main types of app development frameworks namely;

|  |  |  |  |
| --- | --- | --- | --- |
| **Features** | **Cross-Platform ADF** | **Web ADF** | **Native ADF** |
| Run-time environment | Can be run on multiple platforms with a single codebase | They are run on browsers; they take data from websites. | They are designed to run on a single OS. |
| Flexibility | It is flexible in that one application can be developed and converted to several others to run on different platforms without changing the code. | Different frameworks are use for developing the front and back end. Moreover, they are flexible. | They are not flexible. |

## MOBILE APP DEVELOPMENT FRAMEWORKS COMPARISON

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **FEATURES** | **REACT NATIVE** | **FLUTTER** | **XAMARIN** | **IONIC** | **PhoneGap**  (previously known as APACHE CORDOVA) |
| Performance | High performance, allowing developers to use native modules for scripting code for complex operations. | High performance and high speed. | Lower code performance in various operations due to broad code sharing. | Reduced speed due to the fact that the tools used here are web technologies. | Very reduced speed and slow performance because it does not support plugins with hookes (a way for one piece of code to interact with the other) |
| Interface | Has the best JavaScript libraries for creating the UI | reactive (the user is immediately of the effect of each action) | Can be use to develop the UI | Focuses on the front-end UI/UX interactions of an app. Ease of use. | UI libraries improving the user interface |
| Language | JavaScript | Dart | C# ,  Objective-C | Typescript, JavaScript | JavaScript |
| Architecture | Flux | Flux | Mono / ART | Angular MVC | Javatpoint |
| Origin | Developed by Facebook | Developed by Google | Developed by Microsoft | Developed by Drifty | Developed by Adobe |
| Community support | 5 stars  Has the biggest community among all cross-platforms | 3 stars | 4stars | 3 stars | 3 stars |
| Code reusability | There is code reuse on different platforms | There is code reuse on different platforms | There is code reuse on different platforms | There is code reuse on different platforms | There is code reuse on different platforms |
| Type | Cross-platform | Cross-platform | Cross-platform | Allows for developers to create Progressive web Apps, cross-platform and hybrid mobile applications | Cross-platform |
| Cost | Single code base, hence decrease cost and saves time | Single code base, hence decrease cost and saves time | Single code base, hence decrease cost and saves time | Single code base, hence decrease cost and saves time | Single code base, hence decrease cost and saves time |

1. **HOW TO COLLECT AND ANALYZE THE REQUIREMENTS FOR THE DEVELOPMENT OF A MOBILE APPLICATION:**

**A) Requirement Gathering and Analysis Techniques:**

I) **Interview**

Interviews of stakeholders and users are critical to creating the great software. Without understanding the goals and expectations of the users and stakeholders, we are very unlikely to satisfy them. We also have to recognize the perspective of each interviewee, so that, we can properly weigh and address their inputs. Listening is the skill that helps a great analyst to get more value from an interview than an average analyst.

II) **Prototyping**

Prototyping is a relatively modern technique for gathering requirements. In this approach, you gather preliminary requirements that you use to build an initial version of the solution - a prototype. You show this to the client, who then gives you additional requirements. You change the application and cycle around with the client again. This repetitive process continues until the product meets the critical mass of business needs or for an agreed number of iterations.

III) **Survey/Questionnaire**

When collecting information from many people – too many to interview with budget and time constraints – a survey or questionnaire can be used. The survey can force users to select from choices, rate something (“Agree Strongly,agree…”), or have open ended questions allowing free-form responses. Survey design is hard – questions can bias the respondents.

IV) **Brainstorming**

Brainstorming is used in requirement gathering to get as many ideas as possible from group of people. Generally used to identify possible solutions to problems, and clarify details of opportunities.

V) **Document Analysis**

Reviewing the documentation of an existing system can help when creating AS–IS process document, as well as driving gap analysis for scoping of migration projects. In an ideal world, we would even be reviewing the requirements that drove creation of the existing system – a starting point for documenting current requirements. Nuggets of information are often buried in existing documents that help us ask questions as part of validating requirement completeness.

VI) **Focus Group**

A focus group is a gathering of people who are representative of the users or customers of a product to get feedback. The feedback can be gathered about needs/opportunities/ problems to identify requirements, or can be gathered to validate and refine already elicited requirements. This form of market research is distinct from brainstorming in that it is a managed process with specific participants.

VII) **Reverse Engineering**

When a migration project does not have access to sufficient documentation of the existing system, reverse engineering will identify what the system does. It will not identify what the system should do, and will not identify when the system does the wrong thing.

1. **HOW TO ESTIMATE THE COST OF MOBILE APPLICATION DEVELOPMENT:**

The final cost of developing an app depends on seven key factors:

* **Outsourcing vs. in-house app development**

**Working with an in-house development team** has one main advantage – direct communication and complete control over the team.

**Differences**

|  |  |
| --- | --- |
| **In-House App Development** | **Outsource App Development** |
| Close proximity as there is direct communication and face-to-face meetings | There isn’t close proximity as the development process is migrated to an external company |
| Lack of talent on the local market | Unlimited talent pool |
| Need for huge initial investments | No need for high initial investments |
| Deep involvement and high level of control | Lack of personal control |
| Non-flexibility | Flexibility in the development process |
| Higher rates | Lower rates |

Although it’s generally considered that outsourcing is much cheaper than hiring in-house team members, the cost depends on a number of other factors.

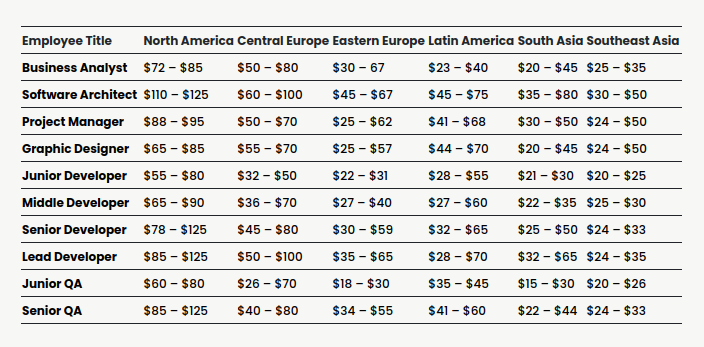
* **Location and structure of a development team**

If it is decided to outsource the app development, there will be need to choose the country for it; giving the following two types of outsourcing methods:

**Nearshore outsourcing -** working with a company from your country or a nearby country. The time difference is no more than 2-4 hours.

**Offshore outsourcing -** working with a company from anywhere in the world. The time difference surpasses 4 time zones.

Many companies decide on the latter option because of it’s financial advantages. Generally, developers in the West are more expensive than those in Asia or Eastern Europe. The following table shows how much developers charge based on their location.



* **Android vs. iOS**

Android and iOS are two of the leading mobile operating systems in the world. It is often a choice to create an app for either of the platforms.

The prices to build an app for iOS or Android don’t differ much, as the timeline is almost the same for both types of development. If you have a bigger budget, you can work on two applications simultaneously and launch them on the market.

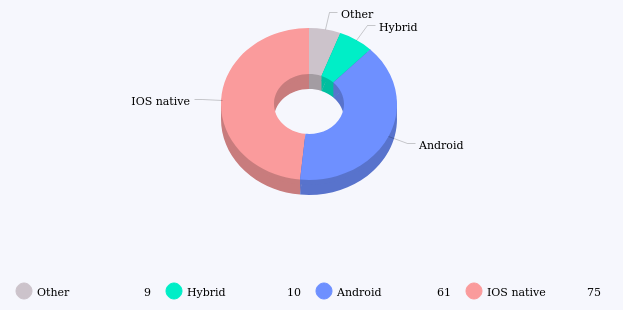
* **Native vs. Hybrid**

To be able to estimate the cost of an application, it is vital to know what type of application is to be created. It could either be a **native** or **hybrid app** depending on the requirements.

**Native app development** creates an application for a specific platform: iOS or Android. If you decide to target both groups, you will need to build two separate apps, automatically increasing the final app development cost.

**Hybrid app development** entails creating an app that will work on multiple operating systems. The main advantage of such apps is the lower cost to develop an app. At the same time, such apps have some disadvantages, which incites many companies to develop native apps. One of them is unstable performance, tricky user experience based on the device an app is being used, and technical limitations.

We can see the investment rate in hybrid and native apps below



* **Visual design complexity**

The first thing users see when they open your app is design. You either make a good first impression or lose a chance to attract them.

The cost of designing an app depends on the number of screens, visuals complexity, the need for custom elements, complex illustrations, and branding.

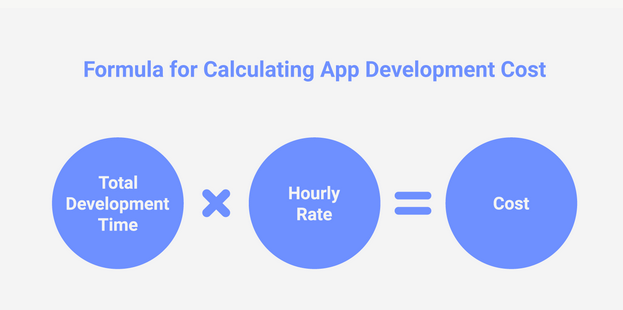
* **Features and functionality**

In order to know how much your application will cost, it is important to understand the features it would offer to its users. These are the most significant cost drivers, as some can take months to implement.

The average price to develop an app can shift in either direction based on the number and complexity of features you want to implement. Some of them, like push notifications, can eat up half of your app budget and reach hundreds of thousands of dollars in implementation.

The cost of building an application can vary based on the features that are going to be implemented including; ***hardware features, user login, geolocation, payments, data encryption, cross-device synchronization,*** and ***user engagement.*** Depending on which features are to be implemented, the number of hours of app development varies; from **10-70 hrs** for user login to **20-200 hrs** for cross-device synchronization, which then gravely influences the cost of development.

The formula for app development cost breakdown is simple: the number of hours multiplied by the hourly rate.



Usually, app development companies use two main charging approaches:

* **Fixed charge** – this approach is mostly used for small projects with a very strict timeline and predefined scope of work. The main advantage of this calculation technique is the clarity in app development cost. You don’t have to wonder how much does it cost to build an app, as you know the final price even before the development starts.
* **Time and material** – this pricing approach differs from the first one in terms of flexibility. It is used mainly for complex projects that might have unexpected turns in development. The price is usually based on the hourly rate and time spent on development.