

REPUBLIC OF CAMEROON

Peace-Work-Fatherland

MINISTER OF HIGHER EDUCATION

FACULTY OF ENGINEERING AND TECHNOLOGY

REPUBLIQUE DU CAMEROUN

Paix-Travail-Patrie

MINISRE DE L’ENSEIGNEMENT SUPERIEURE

FACULTE D’ENGENIERE ET DE TECHNOLOGIE

**Instructor: DR NKEMENI VALERY**

**DEPARMENT OF COMPUTER ENGINEERING**

**COURSE** **TITLE**: Internet and Mobile Programming

PROGRAMMING

**GROUP MEMBERS**

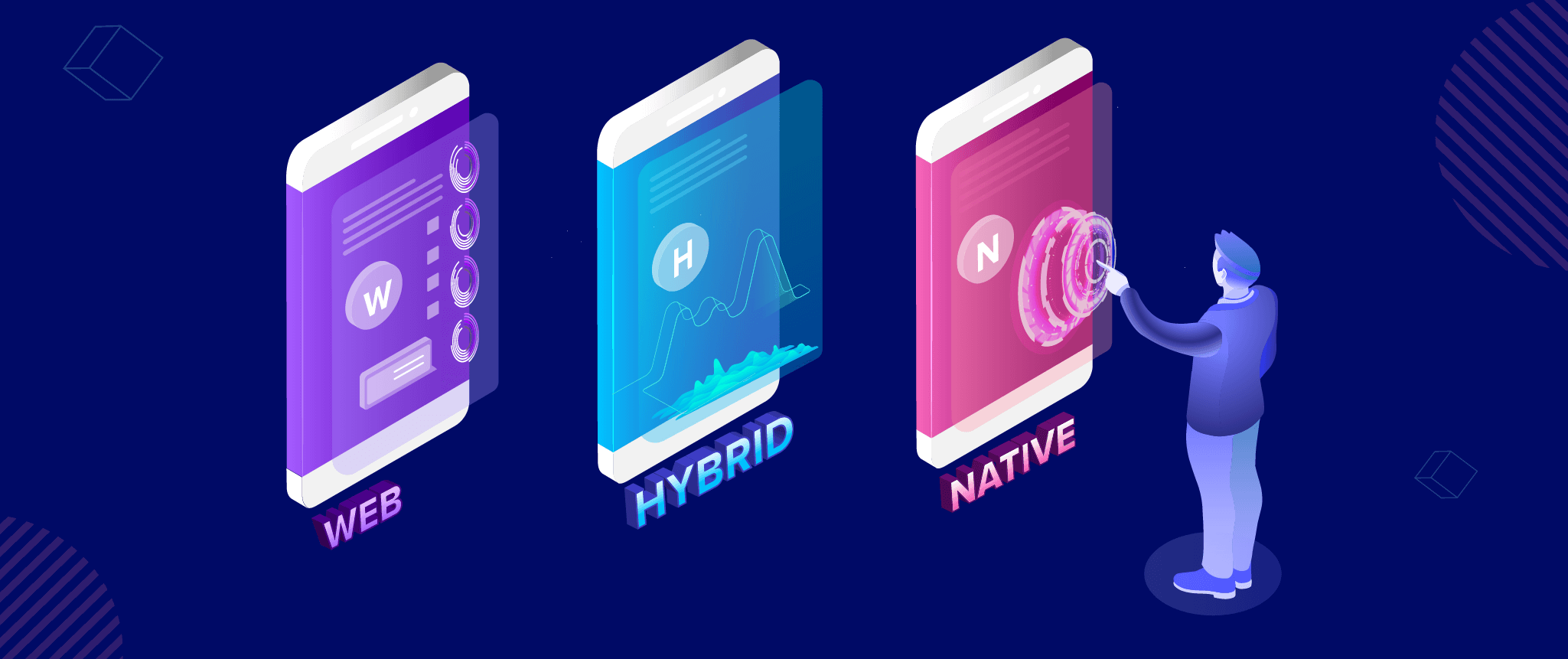
|  |  |
| --- | --- |
| **Names Matricul** | |
| ETOH KIERA EKUTY | FE20A035 |
| GWANA JUSLINE AKAH | FE20A047 |
| LASSI FOKOU BERICE JILDO | FE20A054 |
| YUVEN BRIAN | FE20A125 |
| AJONG NTI MAGCELOUS | FE20A006 |
|  |  |

**Major type of mobile application and their differences**

**Introduction**

What is a mobile application?  It is a type of application software designed to run on a mobile device. Anyone planning to build an app for their business will have to answer the question: which type of mobile app do I want?

**Types of Mobile Apps by Technology**

There are three basic types of mobile apps if we categorize them by the technology used to code them: 

* Native apps are created for one specific platform or operating system.
* Web apps are responsive versions of websites that can work on any mobile device or OS because they’re delivered using a mobile browser.
* Hybrid apps are combinations of both native and web apps, but wrapped within a native app, giving it the ability to have its own icon or be downloaded from an app store

1. **Native Apps**

Native apps are built specifically for a mobile device’s operating system (OS). they’re built for just one platform, you cannot mix them – say, use a Blackberry app on an Android phone or use an iOS app on a Windows phone.

**Technology Used**: Native apps are coded using: Java, Kotlin, Python, Swift, Objective-C, C++, and React.

**Pros:** Because of their singular focus, native apps have the advantage of being faster and more reliable in terms of performance.. Native apps utilize the native device UI, giving users a more [optimized customer experience](https://clevertap.com/blog/customer-experience-optimization/).

And because native apps connect with the device’s hardware directly, they have access to a broad choice of device features like Bluetooth, phonebook

**Cons:** However, the problem with native apps lies in the fact that if you start developing them, you have to duplicate efforts for each of the different platforms. The code you create for one platform cannot be reused on another. This increases costs.

And every time there’s an update to the app, the user has to download the new file and reinstall it. This means they take more space in the device’s storage.



1. **Web Apps**

Web apps behave similarly to native apps but are accessed via a web browser on your mobile device. They’re not standalone apps in the sense of having to download and install code into your device. They’re actually responsive websites that adapt its user interface to the device the user is on. In fact, when you come across the option to “install” a web app, it often simply bookmarks the website URL on your device.One kind of web app is the progressive web app (PWA), which is basically a native app running inside a browser.

**Technology Used**: Web apps are designed using HTML5, CSS, JavaScript and Ruby

**Pros:** Because it’s web-based, there is no need to customize to a platform or OS. This cuts down on development costs. Plus, there’s nothing to download. They won’t take up space on your device memory like a native app, making maintenance easier – just push the update live over the web. Users don’t need to download the update at the app store.

**Cons:** Web apps are entirely dependent on the browser used on the device. There will be functionalities available within one browser and not available on another, possibly giving users varying experiences.



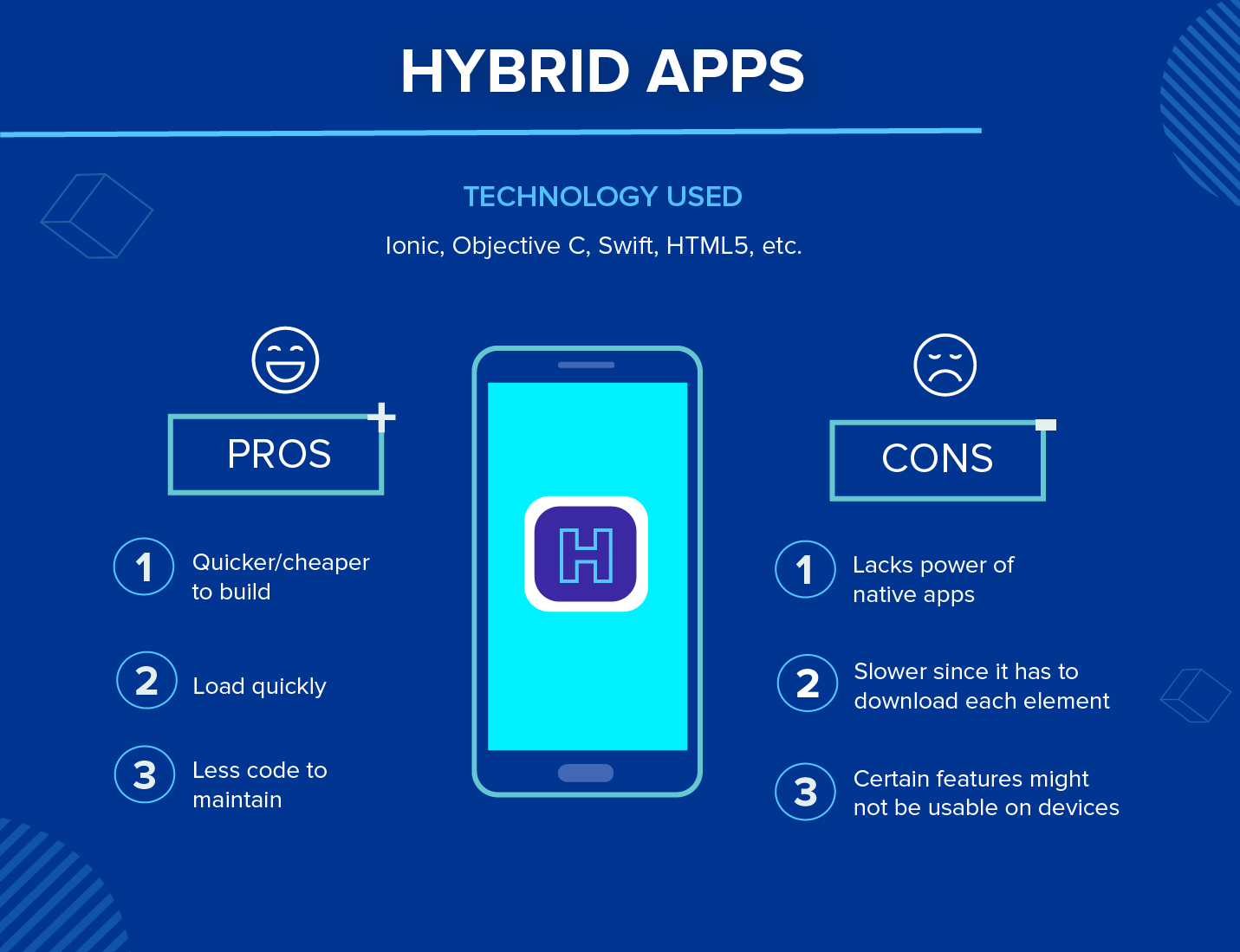
1. **Hybrid Apps**

These are web apps that look and feel like native apps. They might have a home screen app icon, responsive design, fast performance.

**Technology Used**: Hybrid apps use a mixture of web technologies and native APIs. They’re developed using: Objective C, Swift, HTML5.

**Pros**: Building a hybrid app is much quicker and more economical than a native app. They load rapidly, are ideal for usage in countries with slower internet connections, and give users a consistent user experience.

**Cons**: Hybrid apps might lack in power and speed, which are hallmarks of native apps.



Native app is best in terms of performance(fast) while Web and Hybrid are best in terms of resources(least).

**Mobile app programming Language**

1. **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.

1. **Java**

JAVA is one of the most preferred languages by many mobile app developers. 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, object-oriented, and designed to have as few implementation dependencies as possible.

1. **Python**

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 popular applications like Instagram, Dropbox, YouTube, Reddit, Spotify, Quora, and more were created in Python.

1. **PHP**

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.

1. **C++**

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. You can use C++ to create and develop various operating systems, game development, web browsers, medical applications, Enterprise tools, and more.

1. **C#**

A general-purpose programming language, C#, or ‘C-sharp’ is fast, quick to deploy, and easy-to-use. A general-purpose programming language is used in cross-platform, windows, and mobile app development.

**Mobile app development frame work**

1. **Native script**

is an open-source framework to create native mobile applications empowered with Angular, Typescript, JavaScript, CSS, and Vue.js.

1. **React Native:**

 is the best JavaScript library to build native applications for all devices and platforms. With React Native, you can develop rich applications for both iOS and Android

1. **Xamarin:**

 is one of the top mobile app development frameworks and it is .Net based. This native platform is introduced by Microsoft. Being a cross-platform and open source app building platform, [Xamarin](https://www.clariontech.com/blog/why-xamarin-is-your-best-pick-for-building-cost-effective-mobile-apps) offers a development ecosystem with backend, API, components

1. **Flutter:**

is a UI toolkit by Google help in building native applications for the web, mobile, and desktop. This UI toolkit is featured with fully customized widgets, which support to create native applications in a short period

1. **JQwery Mobile:**

 is a user interface system based on HTML5. This web is touch-optimized and used to build highly responsive applications for smartphones, tablets as well as desktops

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Frame work** | **User**  **experience** | **Community**  **support** |  | **cost** | Programming language |
| Native script | Yes | Apple Ios,android |  | $249 | Java script |
| JQuery mobile | yes | Deprecated since oct 2021 |  | &20monthly | Javascript |
| Flutter | yes | Mobile apps |  | $15000 | Dart(C/C++) |
| React native | yes | Meta platforms |  | $15000/3months | Java, swift |
| Xamarin | Open source |  |  | free | C# |
|  |  |  |  |  |  |

**Analyzing the requirement of a mobile app**

The process of mobile app development can somewhat vary on an individual basis. However, the following six steps are the ones that you’ll likely have to go through:

* [Strategy Development](https://www.velvetech.com/blog/mobile-app-development-process/#strategy-development)
* [Analysis and Planning](https://www.velvetech.com/blog/mobile-app-development-process/#analysis)
* [UI/UX Design](https://www.velvetech.com/blog/mobile-app-development-process/#design)
* [App Development](https://www.velvetech.com/blog/mobile-app-development-process/#app-development)
* [Application Testing](https://www.velvetech.com/blog/mobile-app-development-process/#testing)
* [Deployment](https://www.velvetech.com/blog/mobile-app-development-process/#deployment)

1. **Strategy development**:

The very first step of the app development process is defining the strategy. Think carefully about your future application, its goals, capabilities, and business model.You state your objective(What problem will my application solve?Who are its target users?),Check if there are competitors(are there any existing apps that serve the same purpose),Select platform your app will be run on,Choose monetization method (In-app advertising,In-app purchases,Subscriptions,Affiliate marketing,Paid apps).

1. **Analysis and Planning**:

**Functional and Non functional Requirements**

We specific the actions that you want your mobile app to perform.Example, If you’re building an app for the [finance industry](https://www.velvetech.com/financial-software-development/), perhaps you’ll need it to create new accounts, manage credit cards, make payments, generate account statements, and so on.The tasks that your software has to perform will go into your functional requirements document. We also need to think about the non-functional requirements of the software. These relate to the quality of how your system performs and it’s user experience.

**Technology Stack:**

Here we identifying the skills needed for your mobile project.

If you’re building an iOS app, you’ll need experienced iOS developers. If you want to create a cross-platform app, then you need someone skilled in technologies like Titanium, Xamarin, and others fit for multi-platform development.

1. **UI/UX Design**

In order for users to get the best experience, a lot of attention has to be paid to an app’s design.So, your UI/UX design has to be user-friendly, engaging, and offer a seamless experience. the UI/UX design process is all about developing prototypes.

Prototypes let you simulate the user experience and the app’s entire workflow. That way, you get a good sense of how everything will work upon completion.

1. **App development**

Now that the design phase is complete, it’s time to move on to one of the most important stages of your mobile app development process. Typically, it’s made up of two main parts — backend and frontend.

**Backend**

The backend stage of app development involves creating databases and server-side objects that are responsible for your app performance.At this point, your project team will choose appropriate programming languages and begin coding the app. Additionally, database engines and a hosting environment will be selected.

**Frontend**

A mobile app’s frontend is what the end-user will interact with most. There are three main approaches to building it.

**Platform-specific:** These apps are created exclusively for each mobile platform. Meaning that the code isn’t reused between iOS and Android, but rather fully optimized for each one. It might be a more costly approach, but the speed and responsiveness can be worth it.

**Cross-platform:** Solutions that are built in this manner can run on different platforms with one single codebase. They are compatible with various operating systems and aren’t attached only to one. The main benefits of this approach are cost-effectiveness and code reusability.

**Hybrid:** These solutions contain elements of both — platform-specific apps and web applications. They allow developers to write the application code with standard web technologies and execute within a native shell. They are great if you want your app to run on multiple platforms but don’t have the time to build a cross-platform solution.

1. **Mobile app testing**

Once you’re done with app development, it’s time to begin testing. For the highest quality delivery, your app should undergo testing methods prior to deployment.

**Functionality**. This is when you check the features of your application and ensure that everything works well.

**Performance**. During performance testing, you’ll need to pay attention to your app’s responsiveness and how it handles an increase in concurrent users.

**Security**. When building enterprise solutions like mobile insurance apps and healthcare applications, which operate in highly regulated industries, data protection is of utmost importance.

1. **Deployment**

When you’re finally ready to deploy your app, you’ll have to go through different distribution models depending on the platform you were developing for.The mobile application deployment strategy to follow is quite simple. You’ll either need to submit your application to Apple’s App Store or Google’s Play Store or go for private distribution.Both stores require you to fill out several forms and submit your app for review. You should know, the App Store is considered to be much stricter with the applications it allows onto its marketplace. So, you might face difficulties if your iOS-based tool isn’t up to the highest standards.

**How to estimate the mobile app development cost**

The price of building an app depends highly on its functionality and the wages of those who are going to build it. Every feature takes a number of hours to program, and the more complex the app, the higher will be the price of building it and maintaining it after release. The cost estimation process is as follows:

**Supported platforms**

If you plan to develop an app specifically for one platform, native development (Kotin for Android and Swift for iOS) is the best choice in terms of performance, stability and price. However, if you want your app to be present in both Google Play and Apple Store, consider building the app with a cross-platform framework, such as Flutter, React Native, and Xamarin.

**Maintenance**

Every app requires updates to maintain complete compatibility with new devices and OSes, as well as fixing bugs (there are always some). Usually, these expenses amount to about 15% of the initial development cost per year. Also, since almost every app stores user data, you will have to rent a secure server or cloud storage. The annual bill on renting a server’s space starts at $1500 and depends on the amount of data and number of your users.

**Development team size**

How many people develop your app should be determined according to the project complexity. Generally, a common setup includes:

* 2 developers (iOS and Android or cross-platform).
* QA (Quality Assurance) Engineer reviews all specifications and technical documents, as well as creates and plans all testing activities.
* Backend developer oversees data storage, payment systems and logic of app operation.
* UX/UI Designer creates a user-friendly interface for the app.
* Project-manager manages the development process and communicates directly with the client.

**Team’s location**

The geographical location of the team plays an important role in the development costs. For example, rates of software developers in North America are usually over $100 per hour, while developers in Western Europe provide code of the same quality for half of that price. When looking for developers, consider hiring an offshore development company or freelancers, because it can dramatically change the final price of the app.