# Major Types of Mobile Applications: Review and Comparison

An application is a software that lets you exchange information with customers and help them complete specific tasks. Different types of applications, or apps, are based on their development method and internal functionality. Web apps are delivered over an internet browser. Users don't need to install them on their devices. Native apps, on the other hand, are built for a specific platform or device type. The user must install the appropriate software version on their device of choice. Hybrid apps are native applications with a web browser embedded inside them

* 1. **Native Mobile Apps**

Native apps are software applications built specifically for a particular operating system (OS), like iOS or Android, and are designed to leverage the device's hardware and software for optimal performance and a seamless user experience

Native apps are developed using programming languages and tools specific to the target OS, such as Swift and Objective-C for iOS or Java and Kotlin for Android.

This means that if you want to have an app published on Google Play and App Store, you will have to program it twice, once for each of the languages mentioned above.

Native apps are typically distributed through app stores like the Apple App Store or Google Play Store.

Popular examples of native apps include Instagram, Google Maps, Uber, and Spotify.

* + 1. **Advantages of Native Applications**

**Performance and User Experience:**

Native apps are optimized for the specific platform, resulting in faster performance, smoother transitions, and a more intuitive user experience.

**Offline Functionality:**

Native apps can store data locally and access features even without an internet connection, offering a more reliable and seamless user experience.

**Seamless Integration:**

Native apps can seamlessly integrate with the operating system's features, such as notifications, camera, geolocation, and other device functionalities.

**Access to Platform-Specific Security Features:**

Native apps have access to platform-specific built-in security features, enhancing the security of user data.

**Widgets:**

A native app can benefit from widgets. That is, it can incorporate a thumbnail view of the app on the smartphone's homepage.

* + 1. **Disadvantages of Native Apps**

**Development Time and Cost**: Creating individual apps for iOS and Android can be both costly and time-consuming. This is due to the requirement of maintaining two separate codebases, one for each platform.

**Skill Requirements**: Building native apps requires platform-specific programming languages and frameworks. Developers need to master platform-specific languages (Swift/Objective-C for iOS, Java/Kotlin for Android). Acquiring proficiency in multiple languages poses a learning curve.

**Updates and Maintenance**: Maintaining two codebases means that updates and bug fixes must be implemented separately for iOS and Android. This can lead to longer development cycles and higher maintenance costs.

* 1. **PWAs (Progressive Web Apps)**

Progressive Web Apps (PWAs) are web applications built with web technologies that offer a native-app-like experience, including the ability to be installed, work offline, and receive push notifications, all while being accessible through a web browser.

They are called "**progressive**" because they gradually absorb functionalities from Native apps

They are cheaper and faster to develop than native apps.

Example of PWA Apps are**: Pinterest and Trivago, Tinder, Starbucks**

**Advantages Of PWAs**

**Installable:**

Users can install PWAs on their devices, making them easily accessible and giving them the appearance of a native app.

**SEO-Friendly:**

PWAs are built using web technologies, making them easily indexable by search engines.

**Cross-Platform Compatibility:**

PWAs can run on multiple platforms and devices (smartphones, tablets, desktops) from a single codebase, simplifying development and reducing costs.

**Lower Development Costs:**

PWAs are generally less expensive and faster to develop compared to native apps, as they leverage existing web technologies and a single codebase.

**No App Store Dependence:**

PWAs can be installed directly from a browser and don't require users to go through app stores, simplifying the installation process.

**Disadvantages Of PWAs**

### Compatibility with iOS

Since iOS 11.3, it’s been possible to run PWAs on Apple devices, but you can forget about compatibility with older devices. What’s more, Apple doesn’t allow PWAs to access many important features, including Touch ID, Face ID, ARKit, Bluetooth, serial, Beacons, altimeter sensor, and even battery information.

**Hardware Limitations:**

PWAs, being web applications, may not have full access to device features like NFC, Bluetooth, advanced camera controls, fingerprint scanners, or proximity sensors, as native apps do.

**Legacy Device Compatibility:**

PWAs may not be fully compatible with older devices or browsers that don't support the latest web standards.

**Performance:**

PWAs, built with web technologies like JavaScript, HTML, and CSS, might not perform as efficiently as native apps, especially for complex tasks or resource-intensive applications.

* 1. **Hybrid Apps**

Hybrid mobile apps combine elements of native and web apps, using web technologies (HTML, CSS, JavaScript) wrapped in a native app shell to enable cross-platform functionality with a single codebase.

**How they work:**

They are built using web technologies (HTML, CSS, JavaScript) and then packaged within a native app container, allowing them to access device features and run on multiple platforms (iOS and Android) with a single codebase.

**Advantages of Hybrid Apps**

### One codebase to rule them all:

Hybrid apps use one codebase and can work across multiple platforms.

### Quicker build:

A hybrid app is quicker to build because it doesn’t have any native requirements and uses the most basic coding languages; HTML, CSS, and JavaScript.

**Disadvantages of Hybrid Apps**

## ****Dependency on Third-Party Plugins****

Hybrid apps require **plugins** (e.g., Cordova plugins) to access native features. If a required plugin is outdated or unsupported, development becomes challenging.

## ****App Store Rejections****

Some hybrid apps get **rejected** by the **App Store (iOS)** if they don’t meet performance standards or rely too much on WebViews.

Rely on system browser security

1. **Mobile Application Programming Languages: Review and Compare**

Various programming languages dominate the dynamic world of mobile application development, each tailored to specific platforms or offering cross-platform capabilities. From **Java** and **Kotlin** for Android development to **Swift** and **Objective-C** for iOS, developers have an array of options to choose from based on project requirements and platform preferences.

This section discusses and compares the different mobile app programming languages mentioned above in terms of syntax, performance characteristics, optimisations available and benchmarks showing strengths and weaknesses.

* 1. **Java**

Java is a widely used, versatile, object-oriented programming language, often used for developing mobile applications, particularly Android apps, due to its mature ecosystem and powerful framework.

With its object-oriented approach, Java facilitates modular and reusable code, simplifying the development process and enhancing code maintainability

**Features of Java**

**Object Oriented:** In Java everything is an Object. Java Can be easily extended since it is based on an Object Model

**Platform Independent:** Unlike other programming languages, when Java is compiled, it is not compiled into a platform-specific machine, but rather into a platform independent byte code.

**Simple**: Java is designed to be easy to learn. If you understand the basics of OOP, it would be easy to Master

**High Performance**: With the use of Just-in-time compilers, Java enables high performance.

**Secure**: With Java’s secure feature, it enables the development of Virus free, tamper free systems