Survey on Native and Hybrid Mobile Application Development Tools

Shruthi Sasidaran

Abstract— **Mobile phones have drastically transformed from being simple devices to being fully – functional pocket – sized computers, equivalent to any desktop or laptop. This typically led to the requirement of mobile application development and mobile – friendly user interface design development. Hence it is important to understand, learn and analyze the different types of mobile applications that are present in the market. This paper conducts a survey on the types of various mobile applications, a brief comparison between them, and the various application development tools available in the market – for both native applications as well as hybrid applications.**

Index Terms—**Native Applications, Hybrid Applications, Xamarin, HTML5, Android, Cordova**

I. INTRODUCTION

Nowadays, every person walks around with at least one smartphone, if not multiple. Along with the ability to make and receive calls and text messages, users pay keen attention to the specification of the smartphone, analyzing the various features and the rich capabilities available, such as gigabytes of memory, multi core processors, high resolution camera and so on. A mobile app, short for mobile application, is a software that is pre – installed or downloadable from an application store, and installed on mobiles and tablets. The count of mobile apps that are being developed and released into the market increases exponentially by the day [4].

Different operating systems run on different smart phones – Android, iOS and Windows being the most widely used. These platforms use different programming environments, each with its own language and APIs, to develop unique and high – end user experience apps. The increased demand of such apps gives rise to the necessity of cross – platform mobile app development frameworks, to ensure that apps gain access to the largest user – base they possibly can, irrespective of the platform they use, resulting a low cost – in terms of money, time and effort [1][5].

Organizations have understood the necessity to effectively use smartphones and mobile apps to reach and attract customers. Many organizations have started encouraging users to utilize their mobile apps and websites for using new and existing operations. The choice between employing mobile apps and websites or both, depends largely on their cost, usability, required features and the audience they target [5].

There are several reasons why mobile apps are preferred over than mobile websites:

1. Better Personalization is offered by mobile apps
2. Flexibility to view, send notifications
3. Making the most of the device's built – in capabilities
4. Ability to work offline
5. Ability to create mobile – friendly user experience design
6. Ability of mobile apps to work faster than websites
7. Provides branding opportunity for apps [6].

Shruthi Sasidaran, Honeywell Technology Solutions, Bangalore, India, 00919535038426

II. DIFFERENT TYPES OF MOBILE APPS

1. Native Apps: These apps are developed exclusively for a platform, and take full advantage of the operating system features and other softwares that are typically installed on the platform [2].

   High performance and great user experience is the main advantage of native apps. Moreover, native apps provide wider access to their APIs which sets no limitation on the app usage. These apps are directly available from their app stores, ensuring a clear reach to the target customers.

   Development of native mobile applications can be tedious, in the sense that, the entire codebase has to be re – written to target a different platform and users, despite having the same business use – case and logic [7].

2. Hybrid Apps: These apps are developed using multi – platform web technologies – like HTML5, CSS3 and Javascript. They are partly native apps that run on the device, inside a native container and device's browser engine to render the HTML and process the JavaScript locally [2].

   Hybrid apps are fast and relatively easy to develop. A single code base for all the platforms ensures low – cost maintenance and smooth updates.

   However, they lack in performance, speed and overall optimization in comparison to the native apps [7].

3. Web Apps: These apps use mobile browser to run and are usually written in HTML5, CSS and JavaScript. They require minimal device memory. Users get access from any device, provided there is an active internet connection, since the databases are saved on the server [2].

One of the main disadvantage is the absence of access to many APIs for developers, with some exceptions such as geolocation [7].

Why Hybrid? Why Native?

Prior to indulging into the pros and cons of both native and hybrid apps, it is important to understand the top most CTQ, or Critical to Quality, of mobile applications. That is, without doubt, the user experience of the app.

The user experience can make or break the success of the app. An app that has a bad user experience will not be entertained by anyone, neither the customers nor the stakeholders. Prior to the development of a mobile application, the main decision that has to be taken is the approach on which the app has to be developed – whether a native or hybrid app.

There are pros and cons for both Native as well as Hybrid apps [8].

The differentiators in case of Native apps are:
1. Best Security
2. Perfect in user experience
3. Great performance
4. Offline mode
5. Native UI helps users to quickly adopt to the functionalities of the app
6. Access to the device hardware / software – GPS, Location, Calendar etc.

The differentiators in case of Hybrid Apps are:
1. Portability – one code base, multiple platforms
2. Various hardware / software capabilities access – through plug – ins
3. Lower costs for development
4. Faster speed to market – initially
5. Doesn't have to update each app in the app store to wait for approach.

The following table summarizes the differences between a native app and a hybrid app

	Native	Hybrid
Development Language	Native languages: Java for Android Swift for iOS	Native and web / web only
Device Specific Features	High	Moderate
Code Portability	None	High
UI / UX	High	Moderate
Advanced Graphics	High	Moderate
Application Store	Available	Available
Development Cost	Expensive	Reasonable
Device Access	Complete	Complete
Speed	Fast	Medium
Access to Native APIs	High	Moderate

III. TOOLS FOR MOBILE APPLICATION DEVELOPMENT

1. Xamarin: Xamarin is the most preferred tool for native application development. Xamarin is built on the C# programming language and it runs on the .Net CLI. It is used to build apps for iOS, Android and Windows platforms [14][15]. One of the main advantages of Xamarin is that it reuses the business logic layers and data access layers across various platforms. This feature is beneficial when functions such as large amounts of local data, offline mode and image recognition need to be implemented [1]. It is a mono framework that enables communication with the API of mobile devices. It is very helpful in creating scalable and robust applications [16][17].

2. Appcelerator: Appcelerator speeds up or accelerates the app development process as it allows the developers to create apps with fewer lines of code. It supports iOS, Android, Windows and browser – based HTML5 applications. Appcelerator can be used to develop native apps or cross – platform apps. The Titanium Development Platform by Appcelerator works with web programming languages such as HTML, PHP, Javascript, Ruby and Python. The main advantage of Appcelerator is that it has several platform specific APIs, features, and user interfaces. It ensures enhanced performance since the development is platform targeted. Appcelerator's Titanium used Javascript and XML to build native and hybrid mobile applications [1].

3. PhoneGap: PhoneGap is a cross – platform app development tool. It works on standard web development languages - HTML5, CSS3 and Javascript. PhoneGap allows developers to create mobile applications for Android, Symbian, Blackberry, iOS. The main advantage of applications developed using PhoneGap is that they work uniformly well over multiple platforms without any compromise on the look and feel. It does not require any special skill set or expertise for building apps [18]. The main advantage of PhoneGap is that applications developed work uniformly across multiple platforms, ensuring the same user experience. Also, the powerful backend system increases the development speed of mobile applications. It also taps into the hardware of the device such as camera, geo – location, accelerometer [1].

4. Ionic: Ionic is HTML5 mobile app development framework which is used widely to create hybrid mobile apps. It is an open – source SDK. Hybrid mobile applications are small websites that run in a browser within an app that has access to the native platform layer. Ionic has several default CSS components and Javascript components for building mobile apps. It becomes easier to build applications, as Ionic is built on top of AngularJS framework. As Ionic apps are built in HTML5 framework, they require a native wrapper like PhoneGap or Cordova to deploy as a native app [19].

5. Sencha: Sencha Touch is an MVC – based Javascript framework to build mobile applications. It improves the user experience

through its high level of responsiveness. Sencha Touch is compatible with iOS and Android. For app development using Sencha, code initially written in HTML5 can be translated with the help of another tool like PhoneGap. This ensures easier usage in Android as well as iOS [12][13].

6. RhoMobile: RhoMobile offers Rhodes, an open – source framework based on Ruby. It permits the developer to create native apps for Android, Windows, Symbian, iPhone, RIM. Rhodes framework enables user to write cross platform code by using common web skills. Data synchronization is ensured through the RhoConnect client by RhoMobile. The RhoMobile suite consists of a package of tools for various operations related to app development: Building, Testing, Debugging, Integrating, Deploying and Managing consumer and enterprise apps. This services of this suite are offered in the cloud [9].

The suite consists of:
Rhodes
RhoElements
RhoStudio
RhoConnect
RhoGallery

7. WidgetPad: WidgetPad is an open – source, collaborative environment for development of mobile applications using standard web technologies – HTML5, CSS# and Javascript. One of the main advantage of WidgetPad is that it eliminates the need to learn platform – specific API sets and tools. [2]

WidgetPad is available through the web as a Software as a Service (SaaS) model. WidgetPad for iPhone enables developers to create fully interactive, media – rich mobile applications. The applications developed using WidgetPad can leverage all of the iPhone's advanced hardware capabilities such as GPS, accelerometer, compass, easily by accessing WidgetPad.com through standard browsers like Safari, Firefox and Internet Explorer [10][11].

Certain key features available for WidgetPad for iPhone are:

- Editing: Source code editing – HTML5, CSS3 and Javascript along with syntax highlighting
- Project management and inventory: Uploading of various resources like images, sounds and other Javascript and CSS files and maintaining them.
- Code Sharing: Sharing public projects automatically to everyone.
- Secure cross – domain scripting: Creating applications easily with existing web services by accessing web – service APIs over the domain boundary in a secured manner.
- Application preview: To be able to run the application on Safari, without the dependency of an iPhone.
- Publishing: Choice to publish the app to AppStore either through a generic WidgetPad client (free) or as the developer's own branded application (free or paid). [3]

IV. CONCLUSION

Hence, in this paper the comparison between native and hybrid mobile applications are discussed. Also, the various tools for mobile app development are summarized. It is clearly evident that narrowing down to the perfect approach – if native or hybrid, as well as the best tool to use for the development of a mobile application purely depends on the business requirement of the app.

REFERENCES

[1] Apurva P. Pawar, Vandana S. Jagtap, Mamta S. Bhamare, "Survey on Techniques for Cross Platform Mobile Application Development", International Journal of Advanced Research in Computer Engineering and Technology, Vol. 3 Issue 10, October 2014.

[2] Pooja Jardosh, Lipi Kothari, "A Comprehensive Survey: Techniques for Mobile Development Cross Platform", IJEDR, Vol. 5 Issue 2, 2017

[3] Mounaim Latif, Younes Lakhrissi, El Habib Nfaoui, "Cross platform approach for mobile application development: A survey", Information Technology for Organizations Development (IT4OD) International Conference, 30 March – 1 April 2016

[4] Pavel Smutny, "Mobile development tools and cross – platform solutions", Carpathian Control Conference (ICCC), 2012

[5] https://ymedialabs.com/hybrid-vs-native-mobile-apps-the-answer-is-clear/ Hybrid vs Native Mobile Apps – The answer is clear

[6] https://www.mobiloud.com/blog/web-hybrid-native-apps/ Native, Web or Hybrid Apps? What's the difference?

[7] http://blog.techmagic.co/native-vs-hybrid-apps/ Native vs Hybrid Apps. What to choose in 2017?

[8] https://impekable.com/native-vs-hybrid-mobile-app-development/ Creating Top Apps: The pros and cons of Native vs Hybrid Mobile App Development

[9] https://en.wikipedia.org/wiki/RhoMobile_Suite RhoMobile Suite

[10] https://readwrite.com/2009/09/21/widgetpad / WidgetPad: Open – source, Web – based Environment for Mobile Developers

[11] http://mrdroidz.blogspot.in/2011/09/widgetpad-introduces-web-based-open.html WidgetPad introduces web - based open –source development environment for iPhone Widgets

[12] https://www.sencha.com/ Sencha

[13] http://senchadevelopers.blogspot.in/2012/10/what-you-can-do-with-sencha.html What is Sencha Framework?

[14] https://www.xamarin.com/ Xamarin

[15] https://msdn.microsoft.com/en-us/library/mt299001.aspx?f=255&MSPPError=-2147217396 Visual Studio and Xamarin

[16] https://msdn.microsoft.com/en-us/magazine/mt790185.aspx?f=255&MSPPError=-2147217396 Xamarin – Xamarin and the Universal Windows Platform

[17] https://msdn.microsoft.com/en-us/library/dn879698.aspx Build apps with native UI using Xamarin in Visual Studio

[18] https://www.phonegap.com Adobe PhoneGap

[19] https://ionicframework.com/ The top open source framework for building amazing mobile apps.

Shruthi Sasidaran has completed Bachelor of Technology degree in Computer Science and Engineering from Amrita University, Coimbatore, India. She is Product Design Engineer at Honeywell Technology Solutions, Bangalore, India. Her field of interest include mobile application development and UX.