Native Vs Hybrid Apps
Ketan Anant More\textsuperscript{1}, Ms.Priya Chandran\textsuperscript{2}
\textsuperscript{1,2} Bharati Vidyapeeth’s Institute of Management Information Technology, University of Mumbai, Maharashtra, India

Abstract—Objective of this study is to compare Native Apps Vs Hybrid Apps. The comparative study is based on the advantages and disadvantages between Native and Hybrid Apps. We have studied seven popular Hybrid Apps frameworks, Ionic and Titanium while comparing with native frameworks. Based on the comparative study, the developers can choose the best Mobile Application option to build the mobile app.

Keywords—Native App, Hybrid App, Ionic, Titanium, Phone gap, Sencha Touch, Kenda UI, Mobile Angular UI, Intex XDA

I. INTRODUCTION

In Today’s Generation the Mobile apps development are very essential. Mobile apps development are crucial for any business organization. There are Native apps, Hybrid apps and Mobile web view option for mobile development. In this research paper we focus on the Native apps and Hybrid apps. Also we discusses on different frameworks of hybrid apps which is used to develop the mobile apps.

Businesses that need a mobile app have three available options:

1. Native apps can Built using the device’s native programming language, native apps only runs on their designated platforms. For example, Android applications can’t run on iOS and Blackberry applications can’t run on Android, iOS applications can’t run on Windows platform, and so on. Native apps are distributed through their respective platform’s with marketplace/app store and installed on the devices itself.

2. Mobile web apps run in the device’s browsers and operate across all platforms. For example, a mobile web app works equally well on android platform, iOS platform, Blackberry platform, and Windows Phones platform. Unlike native apps, the mobile web apps are distributed via the webs, and are not installed on the devices itself. It’s important to note the differences between a mobile web pages and a mobile web apps. While often confused, mobile web apps and mobile web pages are quite different things. A mobile web page is a static HTML/CSS/JavaScript page formatted for use on a mobile device.

3. Hybrid apps are the cross between native apps and mobile web apps. A hybrid app is a mobile web app wrapped in a platform-specific shells. This platform-specific shell gives the application native qualities, such as full device integration, Portability, native installation, and app store/market distribution.

In Today’s Generation the Mobile apps development are very essential. Mobile apps development are crucial for any business organizations. There are Native apps, Hybrid apps and Mobile web view option for mobile development. In this research paper we focus on the Native apps and Hybrid apps. Also we discusses on different frameworks of hybrid apps which is used to develop the mobile apps.
II. LITERATURE REVIEW

There is several research papers available on Mobile Apps Developments in internet. Lots of Information on Native Apps, Hybrid Apps and Mobile Web Apps. In [15] the authors discusses the comparison of Native app and Mobile web Apps. They show the actual strength of the native and mobile web apps. But they did not study the hybrid apps information. In [10] the authors show the Real Challenges in Mobile Apps Development. They have done the survey on the mobile apps development which gives the actual use and real challenges when develop the mobile apps. In [11] authors provides the information about cross platform development approaches of mobile application. They gives the total information of mobile application which developed in different platforms. In other research papers [18],[19],[20] which gives the information about HTML5 and W3C which in hybrid apps development and Native ,Mobile web and Hybrid apps development information but they not provides comparison between them.

In [23] the cross platforms for mobile apps development is discussed. They shows the different platforms for new mobile apps development. In [21] shows the different stages of mobile app development. In this research paper very essential for basic mobile apps development. But they didn’t show the different frameworks of mobile apps development. also in [6] author shows whole information about Ionic framework. It provides good information to develop the mobile apps using Ionic framework. It focus on whole Ionic framework under the Hybrid Mobile Apps. In [25] authors show the comparison study of cross platform mobile development tools. They provides the different platforms for mobile apps.

This paper focuses on Native Apps, Hybrid Apps and its frameworks, also comparative study between Ionic and Titanium framwork. In this study we comparing the native apps and hybrid apps. We study the comparison between native apps and hybrid apps. Then we also study the 7 top hybrid frameworks which use to develop the mobile apps. Then we detail study on Ionic and Titanium framework which is the best 2 frameworks. Depends upon comparatively study between the Ionic and Titanium we get best frameworks for mobile apps development.

III. NATIVE APPS

Native apps live on the devices and are accessed through icons on the device home screen. Native apps are installed through an application store which is available on App stores. (such as Google Play or Apple’s App Store). They are developed specifically for one platform, and can take full advantages of all the device features — they can use the cameras, the GPS, the accelerometers, the compass, the list of contacts, Battery icon and so on. They can also incorporate gestures.

IV. HYBRID APPS

Hybrid apps are part of native apps as well as part web apps. It bridges gap between different mobile apps. Like native apps, they live in an app store and can take advantages of the many device features available. Like web apps, they rely on the HTML being rendered in a browser, with the caveat that is the browser is embedded within the apps. Often, companies build hybrid apps as wrappers for an existing web pages; in that way, they hope to get a presence in the app store, without spending significant efforts for developing a different apps. Hybrid apps are also very popular because they are also allow the cross platform development of mobile apps and thus the significantly reduces the development costs of the mobile apps development.
V. NATIVE VS HYBRID APPS

Table I.
An Explanation of How Native And Hybrid Apps Presumably Address Key Characteristics of Apps Usage and Development

<table>
<thead>
<tr>
<th>Skills/tools that needed for cross-platform apps development</th>
<th>Native App</th>
<th>Hybrid App</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Objective-C</td>
<td>• HTML</td>
<td></td>
</tr>
<tr>
<td>• C++</td>
<td>• JavaScript</td>
<td></td>
</tr>
<tr>
<td>• Java</td>
<td>• CSS</td>
<td></td>
</tr>
<tr>
<td>• VB.net</td>
<td>• Web programming languages (i.e., Java etc.)</td>
<td></td>
</tr>
<tr>
<td>• C#</td>
<td>• Mobile development frameworks</td>
<td></td>
</tr>
<tr>
<td>Apps needed to reach all major smartphone platforms</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Installed on device?</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Distribution</td>
<td>App Store/Market</td>
<td>App Store/Market</td>
</tr>
<tr>
<td>Mobile Device integration</td>
<td>Full integration: for eg (camera, microphone, GPS, gyroscope, accelerometer, file upload, contact list)</td>
<td>Full integration: for eg (camera, microphone, GPS, gyroscope, accelerometer, file upload, contact list)</td>
</tr>
<tr>
<td>Best used for</td>
<td>• Highly graphical apps</td>
<td>• Cross-platform apps that need to full devices access.</td>
</tr>
<tr>
<td></td>
<td>• Apps that need to reach the large consumer audience</td>
<td>• Business apps that need to app store distributions</td>
</tr>
</tbody>
</table>

A. ADVANTAGES OF NATIVE APPS

1. Highly graphical
   Built using the main device’s native language, and installed on the devices itself, native apps offer the best graphics and animations. If a business needs a highly graphical applications, such as a game, native apps are their best option. It provide graphical features.

2. App store distribution
   Native apps are distributed through their own platform’s native app stores and marketplace. While not often a requirement for business applications, app store distribution are essential for companies that require wide consumer distribution, or need to sell their applications. It’s important to note that hybrid apps also offer app store distribution. It is satisfied app store distribution.

3. Device integration
   Native apps are provide full access to the device’s hardware, such as its GPS sensor, contact list, camera, microphone, gyroscopes, and accelerometer, bettery. These capabilities are essential for any apps that require device data, like geographical location or device positions.movements. It’s important to note that hybrid apps also offer full device integration, while mobile web apps offer to partial integration. Device Integration very essential for any apps development.

B. DISADVANTAGES OF NATIVE APPS

1. No portability
   Since each native applications only runs on one platform, businesses building native apps must make a choice-builds for one platform or builds for multiple platforms? Unfortunately,
there’s no easy answer. As illustrated by the image below, the mobile platform landscape includes four major smartphone platforms and four major tablet platforms. Native app not contains proper portability. Building an app for just one platform excludes seven platforms, yet building for all platforms requires significant time and resources.

2. **Platform instability**

The mobile platforms landscape is notoriously unstable. A popular platform today may disappear in just a few years. For example, both the Blackberry and the Palm dominated the mobile industry just 5 shorts years ago. Today’s, Blackberry is struggling and Palm doesn’t exist. There fact is, nobody knows what the mobile platform landscape will look like in another 5 years. Companies that are choose the native approach always run the risk of wasting time and money building for a platforms that might not last.

3. **Development cost**

While native apps development cost varies depending on the app’s complexity, it’s easily the most expensive and time-consuming approach. Development costs high for native apps. For example, Forrester Research estimates that most native apps require at least of 6 months of full-time work, and cost between $20,000 and $150,000, depending on complexity which in Reference [1]. It’s important to note that those estimates apply to single-platform native apps development. The cost rises exponentially over when developing cross-platform native applications. Based on the development cost estimates listed above, the chart in Reference [1] outlines the cost of developing native applications.

4. **Development time**

As mentioned in Development cost point, Forrester Research estimates that a single native app requires of 6 months of development time. If we try to building native apps for more than one platforms, the time requirements rise depending on the number of developers needed and applications complexity. Development time requires more for native apps. For example, using just one developer for cross-platform smartphone apps development brings the development time up to 2 years (4 applications x 6 months each). However, development time estimates become increasingly complex when using multiple developers.

5. **Maintenance cost**

While all apps requires regular updates and maintenance, native apps require the most future maintenance when compared with the other two mobile apps options. Maintenance cost dependent upon the size of native apps. Beyond regular apps maintenance, native apps must also be updates with every new platform release in market. Additionally, businesses that build the native apps for multiple platforms must maintains multiple applications, duplicating every change or updates across all applications.

6. **Limited control**

When placed in an apps store, a native applications is completely controlled by the app store’s owner (like Apple or Google). For example, if Apple rejects or bans a company’s app from their app store, the company has no recourse. If Apple decides an app doesn’t meet their terms of services, the app is removed. If another company claims copyright over an elements in the app, the app is removed. Or, if Apple decides the app isn’t right for their store, the app is removed. The app store model puts the companies at the mercy of a third party. All of the resources put into their applications are wasted if that app store’s owner decides the app isn’t right for their store in market.
C. ADVANTAGES OF HYBRID APPS

1. Native look-and-feel (without the native cost)

For businesses purpose that prefer the native apps look and feel, without the high development costs, the hybrid apps is the method of choice. Like native apps, hybrid apps are also installed on the device and launched like a typical applications. These native attributes make a hybrid apps virtually indistinguishable from a native apps. In fact, many people even don’t realize that popular “native apps” like LinkedIn, Foursquare, and Twitter are actually hybrid applications. It provides Good look and feel to users.

2. Device integration

As mentioned in above point, mobile web apps offer nearly full device integration, with a couple of exceptions. That is where hybrid apps come in. Hybrid apps offer full device access, including the “native-only” features in mobile apps, like the camera, microphone, and address book.

3. App store distribution

While not usually a requirements for business apps, some companies needed the wide consumer distribution offered by the apps store. For these companies, hybrid apps are the best option. One mobile web app combined with a hybrid mobile framework lets businesses easily create the “native apps” for distribution in every major app store for new generation.

4. Inexpensive cross-platform development

While not as simple as the mobile web apps approach, building a cross-platform hybrid app is considerably cheaper than building a cross-platform native app. Hybrid apps development consists of building a mobile web apps and then the wrapping that apps in platform-specific native wrappers. This process lets businesses develops cross-platform hybrid apps using their current web development skills in market.

VI. FRAMEWORKS OF HYBRID APP

We have studied seven commonly used hybrid framework for comparison with native framework. They are,

1. Ionic

Ionic is one of the most popular and promising HTML 5 mobile application framework. Built using SASS, it provides many UI component to help the develop rich and interactive applications. It uses the JavaScript MVVM framework, AngularJS to power apps. Two-way data binding, interaction with backends services and APIs makes Angular JS a mobile developer’s common choice. With the coming release of Angular JS 2.0, focused on mobile, its sure to gain even more popularity.

The team at Ionic will soon be introduce an easier way to create the Ionic apps with ionic creator. This will be released soon and will have the drag and drop functionality to get started with apps development in minutes.

2. Mobile Angular UI

Mobile Angular UI is an hybrid's HTML 5 framework which uses the bootstrap 3 and Angular JS to create interactive mobile apps.

The main features of Mobile Angular UI includes:
- Bootstrap 3
- Angular JS
- Bootstrap 3 mobile components such as the switches, overlays and sidebars which are missing in normal bootstrap components.
Responsive media queries which are stripped out of bootstrap as separate files, you only need to include what you need. Mobile Angular UI does not have any JQuery dependencies, all you need are some Angular JS directives to create awesome mobile user experiences.

3. **Intel XDK**

Intel XDK is the cross platform application tool developed by Intel company. Getting started with Intel XDK is very easy that all you need is to download their application which is free and available for Linux, Windows and Mac operating systems. It provides a number of templates to get started and supports a number of UI frameworks such as the twitter bootstrap, jQuery Mobile and Topcoat. Intel XDK provides a live preview on the connected devices whilst you are developing the alongside many useful tools. On a personal note, I think development using Intel XDK was the very easiest. It uses a drag and drop approach which is very simplest, although it does create a lot of unnecessary codes.

4. **Appcelerator Titanium**

Appcelerator’s Titanium is an one of the open source mobile application framework of hybrid apps that provides an environment to create the native apps for several mobile platforms. Titanium is a complete solution for creating hybrid mobile apps with all you need in one place and develop the mobile apps. To get started with Titanium framework, first download titanium studio. The Titanium SDK which is equipped with a number of mobile platform APIs and Cloud service to use as an app backend for development. It comes with platform independent APIs which makes it easier to access phone hardware. Titanium often uses Alloy, a MVC framework to enable rapid development of mobile applications. Modules created using Alloy are easy to reuse across different applications, hence the significantly reducing the development time of mobile applications and the lines of code. The core component of Titanium is one of the Apache-licensed software development kit, Titanium SDK. Titanium also makes Alloy, an Apache-licensed, Titanium-based model–view–controller framework, and Titanium Studio.

5. **Sencha Touch**

Sencha Touch is an one of the HTML 5 mobile app framework of hybrid apps for creating applications for several platforms including iOS, Android and Blackberry and Windows. It has been in existence for some years now and is popular among hybrid mobile applications developers. Sencha Touch scores highly against its competitors by providing a native look and feel across all of the platforms it supports for development. Getting started with Sencha Touch not that difficult but in order to get the best things out of Sencha Touch which one needs to invest a considerable amount of the time for development of mobile apps.

6. **Kendo UI**

Telerik’s Kendo UI is an one of the HTML 5 framework of hybrid apps which used for creating cross platform mobile application. Kendo UI relies heavily on JQuery and has number of JQuery based widgets. Learning Kendo UI in not difficult, developers familiar with the JQuery will find the Kendo UI easy to learn. Kendo UI has open source most of Kendo UI’s toolset and JavaScript framework feature. However most of the commonly used widgets are still under a commercial license.

7. **Phone Gap**

Phone Gap is the odd one out in this list as it’s not framework for creating an app, but for packaging and releasing an app. Phone gap in based on the Open Source Cordova and is the commercial versioned owned by Adobe. With a dedicated support team, Phone gap is popular amongst many mobile developers.
You can also use any of choice of JavaScript or UI frameworks to get started with the Phone gap. JQuery mobile is along side KnockOut.js or Angular JS is a nice combination. Once you are done with your code, Phone gap takes it from there and wrap it based on the intended platforms. Application built using phone gap use a web view to render their content. Phone gap has minimal set of web APIs to access the phone hardware features and it’s possible to write custom plugin to suit requirement.

8. **Bonus**

Apps.js is JavaScript library for creating mobile web apps. It’s lightweight and unlike other framework, doesn’t use Angular JS. It provides several customs themes and widgets. You can write app using zepto or JQuery.

### D. Disadvantages of hybrid apps

Some of the disadvantages of Hybrid Apps are:

1. **Limited graphics**

   Despite for their “native” appearance, hybrid apps offers the same graphical abilities as mobile web apps. Again, while mobile web apps and hybrid apps offer graphical capabilities suitable for most any business application, highly graphical applications are better suited for the native approach.

2. **Requires familiarity with mobile framework**

   Turning the mobile web apps into a hybrid apps requires familiarity with a mobile frameworks. While still simpler than the native apps development, the hybrid approach adds a level of complexity to the mobile web apps development process, as the developers must familiarize themselves with a hybrid frameworks.

### Table II.

An Explanation Of Ionic vs Titanium Framework of Hybrid Apps

<table>
<thead>
<tr>
<th><strong>IONIC VS TITANIUM</strong></th>
<th><strong>Type</strong></th>
<th><strong>Ionic (it is the current trend. Everyone is opting it)</strong></th>
<th><strong>Titanium</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Memory Leaks</strong></td>
<td></td>
<td>No memory starvation issues</td>
<td>Memory starvation issues. (Mark this point as the most imp one.)</td>
</tr>
<tr>
<td><strong>Angular is</strong></td>
<td></td>
<td>It come bundled with angular is and Over 500 custom designed font icons MIT licensed. With the coming release of AngularJS 2.0, focused on mobile, it’s sure to gain even more popularity.</td>
<td>It does not support custom JavaScript’s and limited functionality without an Internet connection</td>
</tr>
<tr>
<td><strong>Architecture</strong></td>
<td></td>
<td>As ionic use Angular JS, it provides MV* architecture. Due to this the model and view are separated and with the support of Angular JS, any changes made in model are directly affected to the view.</td>
<td>It doesn't support pure MVC architecture.</td>
</tr>
<tr>
<td><strong>Cross platforms support</strong></td>
<td></td>
<td>Supports multiple platforms like Android/IOS/Firefox/Windows/blackberry complex GUI requires same code for</td>
<td>It only support Android/IOS platforms. Requires different code for different platforms</td>
</tr>
<tr>
<td>Functioning</td>
<td>Accessing Mobile hardware</td>
<td>Speed</td>
<td>GUI</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>---------------------------</td>
<td>--------------------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>It doesn't support 3rd party library</td>
<td>Ionic has a native plugin to directly interact with the hardware. Such as Cordova or phone gap in order to run native app.</td>
<td>It’s fast as native Application, and There is a small delay at app startup due to library loading.</td>
<td>Ionic is HTML5, CSS3 and Angular JS based front end development platform, essentially it’s a browser based framework that helps to build cross platform UI.</td>
</tr>
</tbody>
</table>

While Native Apps offers lots of benefits in graphics, app store distribution, and device integration their lack of portability poses the significant problems for business. Besides the risk of an unstable mobile platform landscape and limited app control, the Native applications required the most significant investment of both time money. Considering that the Hybrid approach also offers apps store distribution and device integration, graphics are the only advantages of unique in to the Native approach. Unless a business required game like graphics in their apps, the Native approach offers few benefits for business. It required more development time and cost. It requires the maintenance cost also.

As also above study we can say that Hybrid apps bridge the gap between Native apps and Mobile web apps. While Mobile web apps still offers the simplest mobile apps development for most of mobile companies, Hybrid apps are the next best option. Hybrid apps are ideal for any company that...
wants to develop simplicity of the mobile web apps and combines with the device integration and app store distribution of Native apps. Ionic framework provides all the functionality that can be found in native mobile apps development SDKs. Users can build their applications, customize them for Android or iOS, and deploy through Cordova. Ionic includes mobile components like typography, interactive paradigms, and an extensible base theme.

V. CONCLUSIONS

Today's Mobile generation, the client architectures used for mobile applications can be broadly classified into the mobile Web Apps approach, Native Apps approach and Hybrid Apps approach. Each of these approaches has its strengths, advantages and the drawbacks. 'One size fits all’ is not applicable. The different choice of the client architecture for any mobile applications depends on the demands of the applications and business considerations for mobile apps. There are distinct difference between the Web Apps approach and the Native apps approach. The two key drawbacks of using the Web Apps approach over the Native apps approach are: Inability of web apps to access the device sensors/other hardware Difficulty in building the unique games like interface with native look and feel for applications. The key benefit of the Web Apps approach is cross-platform support. The Hybrid Apps approach emerged precisely to bridge the gap between the Web Apps and the Native Apps approaches. With the Hybrid Apps approach, one gets cross-platform support without having to forgo access to device capabilities.

The conclusions from comparatively study of Ionic and Titanium frameworks that Ionic is the best framework to develop the Mobile Apps in today’s Environment.

ACKNOWLEDGMENT

Apart from the efforts of us, the success of this research paper depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this paper. We feel motivated and encouraged every time. We are also thankful to the teaching and non-teaching staff of College. Special thanks to people who involve in this survey.

REFERENCES

[11] Henning Heitkötter, Sebastian Hanschke and Tim A. Majchrzak Department of Information Systems, University of Münster, Münster, Germany heitkoetter@wi.uni-muenster.de, sebastianhanschke@gmx.de, tima@ercis.de
[12] Spyros Xanthopoulos Directorate of Technical Services and Computerization Aristotle University of Thessaloniki 54124 Thessaloniki, Greece 0030 2310998828 xant@auth.gr
[13] Stelios Xinogalos Department of Applied Informatics University of Macedonia 54006 Thessaloniki, Greece 0030 2310891820 stelios@uom.gr


