**Comparison of Cross Platforms for Mobile Application development**

PANKAJ SINGH**,** *Software Engineering MS student, Arizona State University, Mesa AZ*

**Abstract:** Currently, Field Biologists use a survey mobile application available in both Android and iOS environments to collect data. The source code of the application is developed individually in both the platforms. Two different source codes require 2 different teams with expertise in each platform to maintain and enhance the functionalities of the mobile application. This requires huge amount of effort and expertise and cost. We aim to utilize the capabilities of Cross Platforms for mobile application development to organize and develop a survey mobile application. Multiple such platforms/tools are available in the market, we have considered three most popular ones - PhoneGap, Titanium and Xamarin. We have evaluated these platforms on multiple factors like pricing, licensing, community support, ease of learning and development, and closeness to native mobile features. Every platform is rated against given priority values of each requirement and tabulated. The platform with highest table values is selected for our application development.

**INTRODUCTION**

Mobile Applications are developed and maintained separately by experts in respective platforms. Currently, iOS and Android are the leading smartphone mobile platforms capturing more than 90% of the market. Mobile Applications developed in environment specific technology are generally referred as “Native Applications”.

Having Native applications developed in two separate platforms requires developers to learn two specific languages i.e. Objective C for iOS and java for Android along with their architectures. This requires to develop two different codebases for each platform which ultimately results in huge amounts of effort and cost for development and maintenance. Apart from this, there are some tools/framework/platforms which uses a single technology to develop application/s for multiple environments, and are generally referred as “Cross Platforms”. We tried to explore three most popular cross platforms available in the market i.e. PhoneGap, Titanium and Xamarin. Since our main aim was to utilize one such platform to rewrite survey mobile application, we have evaluated these platforms on our criteria of requirements.

**PROCEDURE OF EVALUATION**

Using each platform, we have developed basic mobile applications using most essential user interface components like Input Text fields, Spinners, Buttons, MySQL (SQLite)

File upload/download, Images, Connection to Wi-Fi. Though these components were not included in every sample project developed but best efforts were made to incorporate most of them. The sample applications(POCs) that was developed or analyzed during the process for each platform can be found at [Github repository](https://github.com/PankajSingh-ASU/HybridPlatformDemos).

Following are the evaluation criteria for each platform in their priority.

1. *Pricing:* Since we do not require to publish our application to any of the app store, we evaluated free version or trial version of the platform only.
2. *Native Mobile Features*: Does the platform provide device hardware support? Can they make use of device hardware and optimize applications using native Android or iOS features? After developing an application how close do they resemble to the same app developed in native platforms.
3. Learning Curve: Availability of the tutorials, documentation of APIs and prerequisite learnings.
4. Community Support: Is there a developer forum to discuss and resolve bugs and errors? Can the errors be easily searched and solved reading other developers post?
5. Core Architecture: Is the platform uses web view functionality of native platforms to perform the applications functions or uses native features to perform native functions. How Is the organization of application directory structure: is it robust or flexible?
6. Developer Community Reviews: What are the views of other developers on this platform. How others evaluators rate this platform.
7. Licensing: We tried to find a platform which is either open source or is available to student/educational institution for free to publish their applications to the marketplaces like iTunes for iOS and PlayStore of Android. Though this is not an immediate requirement but it may be in future.

**PLATFORM 1 - PHONEGAP**

PhoneGap is an open source framework provided by Adobe Systems. It is a cross platform born out of Cordova. “In 2011, the PhoneGap codebase was donated to Apache for incubation, and the project has continued to thrive ever since. Apache Cordova is still the engine that powers PhoneGap, much like WebKit is the engine that powers many modern web browsers” [1]. With PhoneGap apps are built using HTML, CSS and Javascript.

It comes in 2 flavors

1) Build and package locally using the [PhoneGap CLI](http://docs.phonegap.com/references/phonegap-cli/)

2) Use [PhoneGap Build cloud service](http://build.phonegap.com/) for simplifying the build and app packaging process.

**Positives**

1. *Licensing*: It’s completely open source framework which generates iOS and Android applications ready to deploy at iTunes and Google PlayStore.
2. *Pricing*: The [Free plan](https://build.phonegap.com/) [2] covers one private app with app size limit of 50 MB. In our scenario, we fall in free plan.
3. *Learning Curve*: Learning is very easy and straight forward. Most of the web developers already know HTML CSS and Javascript. Since there is only one platform i.e. web has to be developed which works as the common code base for conversion to Mobile platforms. Developers are free to use any framework of the web app e.g. JQuery Mobile, Angular, Backbone, React. Plenty of tutorials are available. I followed one given by Christophe Coenrates at [LXJS workshop](https://www.youtube.com/watch?v=9GTqlwsTZ3g) .
4. *Community Support*: As PhoneGap is a wrapper over Cordova so it shares a huge developer community in terms of code sharing and error resolution. Errors and questions can be easily found on stackoverflow.com.

**Negatives**

1. *Native Mobile Features:* The platform provides basic hardware support. For every feature a plugin can be included in the codebase to access hardware/native features. These plugin libraries are documented in the PhoneGap/Cordova plugin library. Though native/hardware features can be used, but it does not provide a close resemblance to look and performance of native applications.
2. *Core Architecture*: The platform uses web view functionality of native platforms to perform the applications functionality. Prior to Android KitKat, the web view feature wasn’t fully functional. So the applications have very low usability. PhoneGap uses xml configuration and pluggable components(plugins) can be integrated for developing applications [3].
3. *Developer Community Reviews:* Developers suggest to use it for cross platform application at the basic level. Applications demanding good native mobile features and performance are not suitable to be developed with PhoneGap [4][5][6][10].

**PLATFORM 2 - APPCELERATOR TITANIUM**

“The Appcelerator Platform helps you develop cross-platform mobile applications using the Titanium SDK, Alloy, Appcelerator Studio and the Appcelerator CLI, then lets you manage the entire lifecycle of the application with debugging, testing, deploying, crash monitoring and analytic data collection” [7]. To start with Appcelerator a sign up is required and platform download and installation can be made next. Installing all the components require up to 17GB of system disk space. It comes with its own IDE and sdks.

**Positives**

1. *Licensing*: To start with the Appcelerator platform a signup with the developer forum is required. Applications developed with Appcelerator can be published with Android or iOS marketplace. Even Appcelerator provides their own marketplace.
2. *Native Mobile Features*: The platform provides basic hardware support. Native UI components can be developed using Titanium App Designer IDE using drag and drop method. APIs for using native mobile features are provided by the platform.
3. *Core Architecture*: The platform compiles whole JavaScript source code almost one to one to native iOS and Android components [8]. It also uses JavaScript, CSS and HTML for the development of applications.
4. *Learning Curve***:** It is very easy for web developers using JavaScript, CSS and HTML. Plenty of tutorials and video lectures are provided by the company. It also provides online courses with the name of Appcelerator University.

**Negatives**

1. *Community Support***:** In January 2016, Appcelerator [embraced Stack Overflow for Appcelerator Community Support](http://www.appcelerator.com/blog/2016/01/embracing-stack-overflow-for-appcelerator-community-support/). Original Q& A’s (2010-2015) are archived and contains outdated information [8].
2. [***Pricing***](http://www.appcelerator.com/pricing/)*:* Appcelerator comes with 30 days of free trial. After that purchase is required for the use of the platform. The pricing is very high. This was one of the showstopper for Appcelerator and further evaluation was dropped.

**PLATFORM 3 - XAMARIN**

“Deliver native Android, iOS, and Windows apps, using existing skills, teams, and code. [9]” The platform uses C# as the programming language. It provides its own IDE called Xamarin Studio or Visual Studio from Microsoft can be used. It also provides access to native mobile features and functions.

**Positives**

1. *Native Mobile Features:* The platform provides maximum mobile hardware support. Native UI components can be developed using Xamarin studio. It provides APIs for using native mobile features are provided by the platform.
2. *Core Architecture*: The platform provides 2 concrete layers in a mobile application. First is the native UI components and hardware access of each platform. They are accessed by C# classes wrapped over native classes. For e.g. IOS storyboard and Androids layout and xml configurations can be accessed. Second, the core business logic and data access logic is common for all the native platforms. It is also developed in C# classes and existing .Net frameworks and libraries can be used.
3. *Pricing*: It provides free platform tools to students and small teams for learning purposes. Students can learn and develop applications mobile applications in Xamarin but for publishing apps in marketplace require purchase of licenses.
4. *Community Support*: Since it uses C# and .Net frameworks, plenty of existing libraries and solutions are available. It is well supported at stackoverflow.com along with Xamarin developer’s forum [11].

**Negatives**

1. *Licensing*: To start with the Xamarin platform a signup with the developer forum is required. Xamarin does not allow applications to be published in the marketplaces of mobile platforms without purchasing a valid license.
2. *Learning Curve***:** It is easy for .Net web developers using C#. Since I was dealing with C# for the first time, it requires a lot of effort. As UI features are directly developed following native features it requires understanding of native UI components of every mobile platform. It also provides alternate solution to develop forms in .Net for all the platforms but it does not provide good native UI performances. It also provides online tutorials under the hood of Xamarin University.

**QUANTIFIED EVALUATION**

For comparing the three platforms, I have used priority table. Each feature is given a priority point in the decreasing order of their importance for this project. For an example Pricing i.e. platform being free with no associated cost was the highest priority with maximum points of 10 while Licensing was of low priority with only 4 points. Each platform was given points out of 10 (10 max and 0 min) for each features as evaluated above. If any platform, like Titanium, has a show stopper in any feature it is denoted by X and is not considered for the project. Total value of the platform for a feature is calculated by multiplying priority points with platform points. Summation of these platform values has given it a final score. The platform with the highest final score was selected for the project. In our evaluation Xamarin has been selected with total value of 360 closely followed by PhoneGap with 344 value points and Titanium was not considered for being not free for the development.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Priority Points | PhoneGap | PhoneGap  Value | Titanium | Titanium  Value | Xamarin | Xamarin Value |
| Pricing | 10 | 10 | 100 | X | X | 8 | 80 |
| Native Mobile Features | 9 | 4 | 36 | 7 | 28 | 9 | 81 |
| Learning Curve | 8 | 8 | 64 | 8 | 64 | 5 | 40 |
| Community Support | 7 | 9 | 63 | 4 | 36 | 8 | 56 |
| Core Architecture | 6 | 4 | 24 | 6 | 24 | 8 | 48 |
| Developer Reviews | 5 | 5 | 25 | 6 | 30 | 7 | 35 |
| Licensing | 4 | 8 | 32 | 6 | 48 | 5 | 20 |
|  | Total | 13 | 344 | X | X | 12 | 360 |

**CONCLUSION**

This report focuses on evaluation of platform on the features required for our application development. It recognizes Xamarin being the most suitable platform for our application development. Features of Xamarin matched closely to our application requirements. The most striking feature was in pricing and native mobile features which stands it out of other two. The evaluation criteria may not be suitable in every scenario.

**ACKNOWLEDGEMENT**

I sincerely thank Mr. Richard Whitehouse for mentoring and guiding throughout the evaluation period with his expertise and knowledge.

References:

### [1] “PhoneGap vs. Apache Cordova”: <http://phonegap.com/products/>

### [2] “Choose Your Plan”: <https://build.phonegap.com/>

### [3] “Configuring”: <http://docs.phonegap.com/phonegap-build/configuring/>

### [4] “Pros and Cons of Phone Gap” : <http://mobileappcost.com/what-are-the-pros-and-cons-of-using-phonegap-vs-native-apps/>

# [5] “PhoneGap vs Titanium vs Xamarin”: <http://stackoverflow.com/questions/17249500/xamarin-2-0-vs-appcelerator-titanium-vs-phonegap>

# [6] “PhoneGap or Titanium or Xamarin”: <http://www.cygnet-infotech.com/blog/phonegap-or-titanium-or-xamarin-which-cross-platform-should-you-choose>

# [7] ”Overview”: <http://docs.appcelerator.com/platform/latest/#!/guide/Quick_Start>

# [8] “Q & A archive”: <https://archive.appcelerator.com/>

# [9] “Xamarin”: <https://www.xamarin.com/>

# [10] “What Are Xamarin, PhoneGap, and Titanium?”: <https://www.applicoinc.com/blog/what-are-xamarin-phonegap-and-titanium-and-when-should-they-be-used/>

# [11] “Get help with your mobile apps”: <https://www.xamarin.com/support>