# A Comprehensive Survey: Techniques for Mobile Development Cross Platform

[1]Pooja Jardosh,[2]Lipi Kothari

[1]Assistant Professor,[2] Assistant Professor
[12]Information Technology,
[12]Silver Oak College of Engineering & Technology , Ahmedabad, India

_____

*Abstract*—**Smart phone is used by most of people/of the group. Over thousands of computer programs are used daily and a new computer program gets launched as per need. In order to do work on phones itself many desktop computer programs are getting converted to mobile version by developers. And it's real challenge to market are these computer programs and reach to the maximum users. From developer point of view for computer program to reach to most of the end users it need to run on max (raised, flat supporting surfaces), this needs (re-building and improving/re-built and improved place) of computer program, we can solve this problem a little bit by developing cross (raised, flat supporting surface) computer program without added/more investment. (more than two, but not a lot of) ways of doing things are available which will help to make it happen. Survey of these ways of doing things could help computer program developers to make a proper choice.**

*IndexTerms*—**Mobile Development, Cross Compiled Approach, Cross Platform Application, Multi-platform, Cross Platform Application Development Tools, Hybrid approach, Interpreted Approach, Web Approach, PhoneGap, Titanium, Android, Operating System, Smartphone.**

_____

## I. INTRODUCTION

Versatile innovation has turned into a vital piece of individuals' life. Today, individuals utilize diverse portable applications ordinary: applications for wellbeing, transportation, shopping and numerous different purposes. Cell phone is as prevalent as consistent portable telephones were five years prior. The quantity of cell phone and tablet clients is expanding, as is the requirement for proficient versatile application advancements [4]. This inclination has constrained organizations to create versatile applications keeping in mind the end goal to take after current market norms and fulfill clients.

Diverse interior architecture of Operating System turns into a purpose behind redevelopment of utilization to make it keep running on each Operating System which thus causes parcel of cost as far as time, cash, and endeavors. It's redundant that everybody is utilizing windows working framework on their Smartphone. Every versatile working framework utilizes diverse programming model, designer would require extra overhead and investigating since engineer may have great hands on specific single stage. Thus, making programming as cross stage application can be a decent choice.

This paper will give an idea about techniques which are used in platform independent application development. The aim is to help designers to assess and comprehend what is the more appropriate tool to use to coordinate their prerequisites.

## II. CROSS PLATFORMS & ITS ADVANTAGES

Cross-development tools have streamlined composition applications for all gadgets and frameworks, which turns out to be progressively essential when clients, representatives and technical support frequently convey their own particular gadgets to work.

**Write-Once, Run Everywhere.** This is where cross-platform development comes in. Developing consumer or business apps for multiple operating systems would be intensely resource-consuming if you had to write, test and support more than one app version for every platform separately. If you were to write natively for Android, iOS and Windows, for example, then your development teams need to know Java, Objective-C and C#, respectively.

Alternatively, cross-platform tools allow, to a varying degree, writing the app once and then compiling the code for each platform the development environment supports.

## III. TOOLS FOR IMPLEMANTAION CROSS PLATFORM APPLICATION

There are number of tools available for cross platform development for example, Titanium, PhoneGap, Xamarin etc.

Figure 1 Cross-Platforms Tools





PhoneGap is an opensource and most straightforward cross-stage structure contrasted with Xamarin and Titanium. It permits making portable applications using Web APIs, i.e. it wraps up web applications in a local application shell and after that actualizes them on local stores for various stages. It utilizes a cloud-based administration called "Work" with which you can accumulate applications for a few working frameworks without the need to introduce SDKs of every stage. Any PhoneGap application is basically a gathering of HTML pages which is rendered as a Web View. To create applications in PhoneGap, you have to utilize HTML5, CSS and JavaScript.

**Advantages**

- Little and straightforward local API sets empower simple porting to various situations.
- High reusability with HTML5, CSS and JavaScript. Anything composed as a site page can be effortlessly wrapped up as a local versatile application.
- Underpins all stages and working frameworks which incorporates iOS, Android, Windows Phone 8, Blackberry, Firefox OS and Ubuntu.
- Engineers who are acclimated to HTML/CSS/JavaScript, think that its simple to begin working with PhoneGap.

**Disadvantages**

- Bring down execution of applications as the first codes of the application remains that of a web application and dispatches by means of a web program. This implies the execution of PhoneGap applications doesn't approach local applications.
- Excessively numerous divided libraries and structures at an exceptionally fundamental level .
- UI of application differs relying upon the nature of Web View rendered.



Titanium is a JavaScript-based advancement stage in that, it utilizes JavaScript to compose application codes with local APIs and UI traditions of every stage. This implies, it doesn't attempt to fulfill the idea 'compose once and run anyplace' however it endeavors to compose applications reusing JavaScript with stage particular elements and execution. It is bit more confounded than Xamarin and PhoneGap there is have to take in the UI API of every stage well beyond JavaScript which again is perplexing for building huge applications. Titanium at present backings just Android and iOS.

**Advantages**

- Better execution because of local API utilization, which additionally offers access to components and elements of iOS and Android.
- The look and feel of Titanium applications are superior to anything applications based on different stages as the UI is basically local.

- With Javascript, it guarantees snappy and simple improvement.

**Disadvantages**

- No support for outsider libraries.
- Trouble in creating complex applications.
- Since it doesn't utilize HTML5 and CSS, the livelinesss and DOM components are laggy and less responsive.

Xamarin, initially called MonoTouch is another cross-stage structure that has grabbed the advancement showcase with its own particular IDE. It deals with C# inside .NET system and enables you to make local applications by using local APIs and UIs of every stage. Xamarin accompanies Xamarin.Forms library which enables you to compose local UIs for once and after that offer and change over them to stage particular UIs. Xamarin right now bolsters iOS, Android and Windows stage. It likewise permits creating applications for Blackberry by incorporating Android applications.

**Advantages**

- Xamarin has TestCloud which enables you to test your applications consequently
- Furnishes 100% code reuse with Xamarin.Forms UI improvement utilizing shared code base and rationale. This spares a ton of time and assets
- Bolsters designs like MVC and MVVM
- Xamarin.Android underpins Google Glass gadgets, Android Wear, and Firephone
- Expectation to absorb information is relative. In the event that your group knows C#, it is comparitively simple to begin with Xamarin

**Disadvantages**

- Does not give access to certain Android particular UI controls.
- Impacts stack time as it has its own runtime.
- Does not bolster sharing of codes outside Xamarin condition for local or HTML5 advancement.

## IV. COMPARISON CHART

Give us a chance to burrow further and look at these three one next to the other objectively.

Table 1 Comparison

| | PhoneGap | Titanium | Xamarin |
|---|---|---|---|
| Platform Support | iOS, Android, Windows Phone 7 & 8, Blackberry | Android, iOS & Blackberry | iOS, Android & Windows |
| Language | HTML5, CSS, JavaScript | JavaScript | C# |
| Opensource | Yes | Yes | No |
| UI | Web UI | Native | Native |
| Access to Device API | Limited | Full | Full |
| Web Standard Support | Yes | No | No |
| DOM Support | Yes | No | Yes |
| Native Performance | No | Yes | Yes |
| Used by | IBM, Sony, Mozilla, Intel | Cisco, VMware, Safeguard Properties, Mitsubishi Electric | GitHub, Microsoft, Foursquare, Expensify, Dow Jones |

**REFERENCES**

[1] Apurva P. Pawar, Vandana S. Jagtap, Mamta S. Bhamare,"Survey on Techniques for Cross Platform Mobile Application Development", IJARCET, Volume 3 Issue 10, October 2014.

[2] Manuel Palmieri, Inderjeet Singh, Antonio Cicchetti, "Comparison of Cross-Platform Mobile Development Tools", IEEE, pp. 179-186,2012.

[3] Henning Heitk¨otter, Sebastian Hanschke, and Tim A. Majchrzak, "Evaluating Cross-Platform Development Approaches for Mobile Applications", Springer, Volume 140, pp. 120-138, 2013.

[4] Alireza Pazirandeh , Evelina Vorobyeva ,"Evaluation of cross-platform tools for mobile development", University of Gothenburg.

[5] Isabelle Dalmasso, Soumya Kanti Datta, Christian Bonnet, Navid Nikaein, "Survey, Comparison and Evaluation of Cross Platform Mobile Application Development Tools".

[6] André Ribeiro, Alberto Rodrigues da Silva, "Survey on Cross-Platforms and Languages for Mobile Apps", pp. 256-260.

[7] Andreas Sommer, Stephan Krusche, "Evaluation of cross-platform frameworks for mobile", pp. 366-376.