



ISSN: 0975-766X  
CODEN: IJPTFI  
Research Article

Available Online through  
[www.ijptonline.com](http://www.ijptonline.com)

## CROSS-PLATFORM MOBILE APPLICATION DEVELOPMENT USING APACHE CORDOVA

Himanshu Ewangan, Ramya.G

School of Information Technology and Engineering, Institute of Technology, Vellore-Tamilnadu, India.

Email: [himanshu.dewangan2015@vit.ac.in](mailto:himanshu.dewangan2015@vit.ac.in)

Received on 25-10-2016

Accepted on 02-11-2016

### Abstract

In this fast faced world, where technology is changing rapidly in every second, certain technology is considered most which helps to provide better results with fewer resources. When it comes to human most handed tool - Mobile Phone which is run by embedded software running in them. Mobile Apps have gained such an importance that an user can perform all their task in move.

It has to be agreed that the mobile applications allow users to fetch any information in instances. But, Web is the only major computing and Internet service providing platform that has ever existed without having any vendor. Apache Cordova framework is one of the most popular framework which uses Web technology for development of Cross Platform Mobile Apps. This paper describes the approach of creating cross-platform hybrid mobile applications using Apache Cordova.

**Keywords:** Android, Windows, IOS , Cross-platform, Cordova, HTML, CSS3, JavaScript.

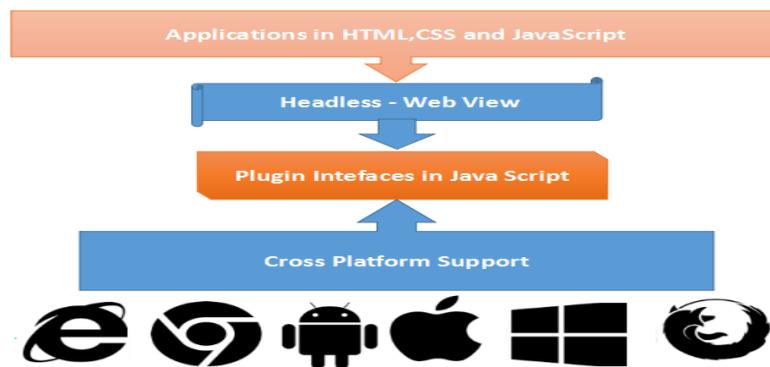
### Introduction

Apache Cordova[1] is a platform for building native cell programs using HTML, CSS and JavaScript. Apache Cordova is a hard and fast of device APIs that allow a cellular app developer to get admission to local tool feature such as the digital camera or acceleration meter from JavaScript. When the use of the Cordova APIs[2], an app can be built without any local code (Java, objective-C, and so on) from the app developer. alternatively, internet technologies are used, and they're hosted within the app itself locally (usually no longer on a far off http server).And due to the fact these JavaScript APIs are steady throughout more than one tool platforms and constructed on net standards, the app have to be portable to different device structures with minimal to no changes. Apps the usage of Cordova are still packaged as apps using the platform SDKs, and may be made available for installation from each tool's app store.

Cordova gives a hard and fast of uniform Java Script libraries that may be invoked, with tool-particular local backing code for the ones JavaScript libraries. Cordova is to be had for the subsequent systems: iOS, Android, Blackberry, windows telephone, Palm internet OS, Bada, and Symbian[3]. After the deployment, all the apps are automated fulfilled with checksum and publishing criteria of different App store. Hence allow us to achieve faster prototyping and publishing of apps.

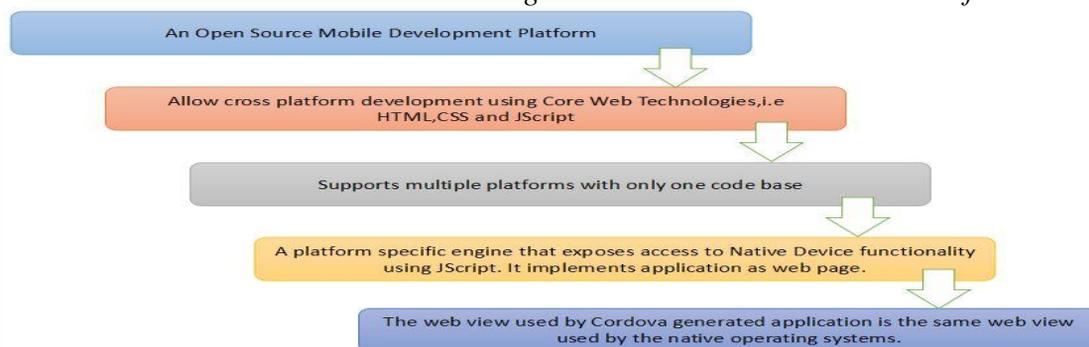
### **Cordova - Under the Hood**

Cordova's user interface is a web view Content. [4]Web view Containers is like a tab in a browser. When Web code is compiled to a Cordova application, it doesn't actually rewrite Web code but automatically converts it into native code, specific to each platform. Cordova acts as a Web container for the app that is developed using web technologies. [5]When the app is compiled, all Web code actually stays intact. The compiler just takes Web code and makes it available to the web view for rendering.[8]This Web View Container is by default in built in every device which supports Internet services.



### **Native Device Functionality**

With Cordova, there is access to native device functionality through the use of plugins. Plugins are add-ons that expose a JavaScript API for native components. [6]They allow us to access native device capabilities by making use of the JavaScript API that it exposes. As we might know, plugins are written in native code. Then invocation of the native code through the JavaScript API so that it can be accessed in the web view. There are two type of plugins that we can find in the Cordova plugin ecosystem. [7]First are the official plugins. These are maintained by the Cordova community. Plugin is official if it has the cordova - plugin prefix. The Second type of plugin are custom plugins. These are built by open source community developers. Often times, they are built to solve a very specific problem, such as connecting an app to Facebook or the Ionic Keyboard Plugin to make interacting with the keyboard easier.



## Installation of Cordova

The Work Environment - This working environment of Apache Cordova development is a internet based connected with any Windows or LINUX based Operating System. The steps of installation could change as per your developer perspective respectively. It is very necessary to follow proper documentation of respective environment before proceeding.

In this environment Apache Cordova was installed as a Command Line Interface (CLI) Installation of SDKs - The prerequisite before installing Apache Cordova is to install the appropriate SDKs for running the application, i.e, if we wish to deploy the application onto an Android and an iOS environment, it is mandatory to install the Android SDK and the iOS SDK before proceeding. Installation of Android SDK is as in [3] and installation of iOS SDK is as in [4]

Installation of Apache Cordova First we need to download and install Node.js.

Following installation, we should be able to invoke node and npm on our command line.

Install the Cordova module using npm utility of Node.js by issuing the command

```
$ sudo npm install -g cordova
```

On Windows the same can be called from the cmd line using the command

```
C:\>npm install -g cordova
```

## Creating the App

From the directory where the source code is maintained we issued the command

```
C:\>cordova create hello com.example.hello HelloWorld
```

This may take some time for the command to complete. Running the command with the -d option displays information about its progress.

## Adding Platforms for Cross Platform Mobile Application

All subsequent commands from now on should be run within the project's directory or any any sub directories within Cordova's scope.

```
C:\>cd hello
```

Before we can build the project, we need to specify a set of target platforms. The ability to run these commands depends on whether machine/ the environment supports each SDK and also whether user already installed each SDK.

```
C:\>cordova platform add ios
```

```
C:\>cordova platform add amazon-fireos
```

```
C:\>cordova platform add android
```

```
C:\>cordova platform add blackberry10
```

To check the current set of platforms

```
C:\>cordova platforms ls
```

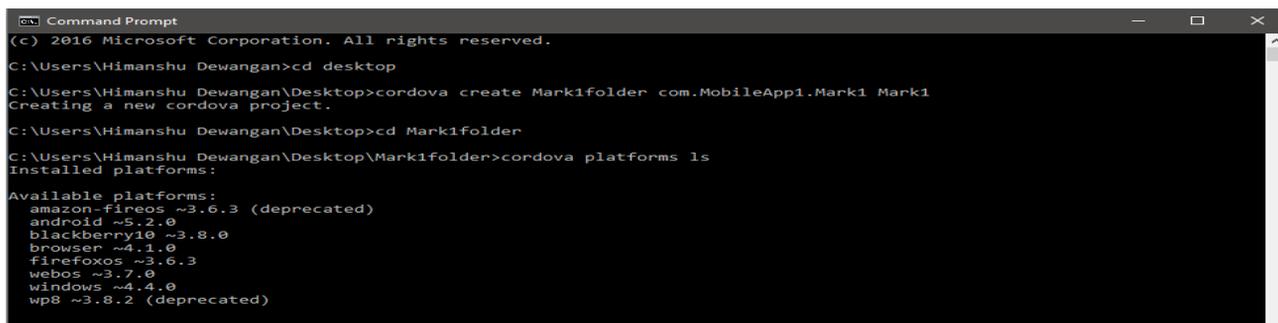
To remove a platform

```
C:\>cordova platform remove android
```

Running commands to add or remove platforms affects the contents of the project's platforms directory, where each specified platform appears as a sub directory. The www source directory is reproduced within each platform's sub directory, appearing for example in platforms/ android/ assets/www. Because the CLI constantly copies over files from the source www folder, only edit these files and not the ones located under the platforms sub directories. If version control software is used, add this source www folder, along with the merges folder, to the version control system.

## Building the App across Cross-platforms

Before an application can be built – it is mandate that the platform's SDK path is set as an environment variable. The command `C:\>cordova build` - will iteratively build the project. The output is as shown in the below diagram:



```

Command Prompt
(c) 2016 Microsoft Corporation. All rights reserved.
C:\Users\Himanshu Dewangan>cd desktop
C:\Users\Himanshu Dewangan\Desktop>cordova create Mark1folder com.MobileApp1.Mark1 Mark1
Creating a new cordova project.
C:\Users\Himanshu Dewangan\Desktop>cd Mark1folder
C:\Users\Himanshu Dewangan\Desktop\Mark1folder>cordova platforms ls
Installed platforms:
Available platforms:
amazon-fireos ~3.6.3 (deprecated)
android ~5.2.0
blackberry10 ~3.8.0
browser ~4.1.0
firefoxos ~3.6.3
webos ~3.7.0
windows ~4.4.0
wp8 ~3.8.2 (deprecated)

```

## Building an app for the Android platform

Setting environment variable value – export `ANDROID_HOME=.:/home/user/android-sdk`

C:\>cordova build android – will build the project that could be deployed onto a Android environment.

```
C:\Users\Himanshu Dewangan\Desktop\Mark1folder>cordova platforms add android --save
Adding android project...
Creating Cordova project for the Android platform:
  Path: platforms\android
  Package: com.MobileApp1.Mark1
  Name: Mark1
  Activity: MainActivity
  Android target: android-23
Android project created with cordova-android@5.2.2
Discovered plugin "cordova-plugin-whitelist" in config.xml. Adding it to the project
Fetching plugin "cordova-plugin-whitelist@1" via npm
Installing "cordova-plugin-whitelist" for android

    This plugin is only applicable for versions of cordova-android greater than 4.0. If you have a previous p
platform version, you do *not* need this plugin since the whitelist will be built in.

Saved plugin info for "cordova-plugin-whitelist" to config.xml
--save flag or autosave detected
Saving android@~5.2.2 into config.xml file ...

C:\Users\Himanshu Dewangan\Desktop\Mark1folder>
```

### Building an app for the iOS platform

C:\>cordova build ios – will build the project that can be deployed onto an iOS environment.

The command cordova build is equivalent of 2 commands – cordova prepare and cordova compile.

### Deploying the app on cross platform emulators

Assuming the respective emulators are up and running, for example if the Android emulator is up and running – the app can be tested on its emulator using the command:

C:\>cordova emulate android



Running on Android Emulator AWD



IOS Emulator Intel XDK

### Conclusion

This paper demonstrated the easy usage of deploying a simple application onto various platforms with a small changes in cordova commands.[16] Apache Cordova has certain advantageous in building applications using HTML, JavaScript and CSS3. There are various applications built using Cordova platform.[13] Few of most popular App are

Cordova.

## References

1. Phonegap API.
2. Apache Cordova API
3. <http://www.androidcentral.com/installingandroid-sdk-windows-mac-and-linux-tutorial>
4. <http://www.lynda.com/iOS-tutorials/InstallingXcode-iOS-SDK/159179/159282-4.html>
5. Kerri Shots, Phonegap 3.x Mobile Application Development
6. John M Wargo, Apache Cordova 3 Programming
7. Inputs from [www.tricedesigns.com](http://www.tricedesigns.com)
8. S. Ludwig, "Mobile app usage grows 35%, TV & web not so much", in [venturebeat.com](http://venturebeat.com), (2012).
9. S. Perez, "comScore: In U.S. Mobile Market, Samsung, Android Top the Charts; Apps Overtake Web Browsing", in [techcrunch.com](http://techcrunch.com), (2012).
10. Hitech, "Mobile Application Development Guidelines", (2011).
11. AppBuilder, "What is Hybrid Mobile App?", Telerik Blogs, (2012).
12. S. Higginbotham, "The Global Rise of the Smartphone", Press of GigaOM, (2010).
13. Social Network Service, [http://en.wikipedia.org/wiki/Social\\_networking\\_service](http://en.wikipedia.org/wiki/Social_networking_service)
14. M. Choi and S.-H. Lim, "Development of HTML5 Photo Album Smart Application", *International Journal of Multimedia and Ubiquitous Engineering*, vol. 8. no. 6, (2013), pp 349-360.
15. S.-H. Lim, "Design and Implementation of HTML5 based Hybrid Application for Mobile Social Networking Service", 8th International Conference on Future Generation Communication and Networking, (2014).